

ZebOS-XP BGP SMI Reference
IP Infusion Inc.

Generated by Doxygen 1.6.1

Wed Dec 16 12:33:34 2015

Contents

1	File Index	1
1.1	File List	1
2	File Documentation	3
2.1	smi_bgp.h File Reference	3
2.1.1	Detailed Description	69
2.1.2	Function Documentation	70
2.1.2.1	smi_bgp4_get_path_attr_aggregator_addr_sdkapi .	70
2.1.2.2	smi_bgp4_get_path_attr_aggregator_as_sdkapi . .	70
2.1.2.3	smi_bgp4_get_path_attr_atomic_aggregate_sdkapi	71
2.1.2.4	smi_bgp4_get_path_attr_best_sdkapi	71
2.1.2.5	smi_bgp4_get_path_attr_calc_local_pref_sdkapi . .	72
2.1.2.6	smi_bgp4_get_path_attr_ip_addr_prefix_len_sdkapi	72
2.1.2.7	smi_bgp4_get_path_attr_ip_addr_prefix_sdkapi . .	73
2.1.2.8	smi_bgp4_get_path_attr_local_pref_sdkapi	73
2.1.2.9	smi_bgp4_get_path_attr_multi_exit_disc_sdkapi . .	74
2.1.2.10	smi_bgp4_get_path_attr_next_hop_sdkapi	74
2.1.2.11	smi_bgp4_get_path_attr_origin_sdkapi	75
2.1.2.12	smi_bgp4_get_path_attr_peer_sdkapi	75
2.1.2.13	smi_bgp_address_family_set	76
2.1.2.14	smi_bgp_af_config_check_sdkapi	76
2.1.2.15	smi_bgp_aggregate_nexthop_check_set	76
2.1.2.16	smi_bgp_aggregate_nexthop_check_set_validate . .	77
2.1.2.17	smi_bgp_aggregate_nexthop_check_unset	77
2.1.2.18	smi_bgp_aggregate_nexthop_check_unset_validate	77
2.1.2.19	smi_bgp_always_compare_med_set	78

2.1.2.20	smi_bgp_always_compare_med_set_validate	78
2.1.2.21	smi_bgp_always_compare_med_unset	78
2.1.2.22	smi_bgp_always_compare_med_unset_validate . .	79
2.1.2.23	smi_bgp_api_address_family_unset	79
2.1.2.24	smi_bgp_aspath_access_list_set_validate	80
2.1.2.25	smi_bgp_aspath_access_list_unset_validate	80
2.1.2.26	smi_bgp_auto_summary_update_set_sdkapi_validate	80
2.1.2.27	smi_bgp_bestpath_aspath_ignore_set	81
2.1.2.28	smi_bgp_bestpath_aspath_ignore_set_validate . . .	81
2.1.2.29	smi_bgp_bestpath_aspath_ignore_unset	82
2.1.2.30	smi_bgp_bestpath_aspath_ignore_unset_validate . .	82
2.1.2.31	smi_bgp_bestpath_compare_confed_aspath_set . .	82
2.1.2.32	smi_bgp_bestpath_compare_confed_aspath_set_- validate	83
2.1.2.33	smi_bgp_bestpath_compare_confed_aspath_unset .	83
2.1.2.34	smi_bgp_bestpath_compare_confed_aspath_- unset_validate	83
2.1.2.35	smi_bgp_bestpath_compare_router_id_set	84
2.1.2.36	smi_bgp_bestpath_compare_router_id_set_validate	84
2.1.2.37	smi_bgp_bestpath_compare_router_id_unset	84
2.1.2.38	smi_bgp_bestpath_compare_router_id_unset_validate	85
2.1.2.39	smi_bgp_bestpath_dont_compare_originator_id_set	85
2.1.2.40	smi_bgp_bestpath_dont_compare_originator_id_- set_validate	85
2.1.2.41	smi_bgp_bestpath_dont_compare_originator_id_unset	86
2.1.2.42	smi_bgp_bestpath_dont_compare_originator_id_- unset_validate	86
2.1.2.43	smi_bgp_bestpath_med_set	86
2.1.2.44	smi_bgp_bestpath_med_set_validate	87
2.1.2.45	smi_bgp_bestpath_med_unset	87
2.1.2.46	smi_bgp_bestpath_med_unset_validate	87
2.1.2.47	smi_bgp_bestpath_tie_break_on_age_set	88
2.1.2.48	smi_bgp_bestpath_tie_break_on_age_set_validate .	88
2.1.2.49	smi_bgp_bestpath_tie_break_on_age_unset	88
2.1.2.50	smi_bgp_bestpath_tie_break_on_age_unset_validate	89

2.1.2.51	smi_bgp_check_instance	89
2.1.2.52	smi_bgp_clear_gen_sdkapi	89
2.1.2.53	smi_bgp_cluster_id_digit_set_sdkapi_validate . . .	90
2.1.2.54	smi_bgp_cluster_id_set_sdkapi_validate	90
2.1.2.55	smi_bgp_cluster_id_unset_sdkapi_validate	91
2.1.2.56	smi_bgp_community_list_entry_unset	91
2.1.2.57	smi_bgp_community_list_set	92
2.1.2.58	smi_bgp_community_list_unset_validate	92
2.1.2.59	smi_bgp_confederation_id_set_sdkapi_validate . .	93
2.1.2.60	smi_bgp_confederation_id_unset_sdkapi_validate .	93
2.1.2.61	smi_bgp_confederation_peer_check_sdkapi	93
2.1.2.62	smi_bgp_confederation_peers_add_sdkapi_validate	94
2.1.2.63	smi_bgp_confederation_peers_remove_sdkapi_- validate	94
2.1.2.64	smi_bgp_create_instance_set_sdkapi_validate . . .	94
2.1.2.65	smi_bgp_debug_validate	95
2.1.2.66	smi_bgp_default_ipv4_unicast_set	96
2.1.2.67	smi_bgp_default_ipv4_unicast_set_validate	96
2.1.2.68	smi_bgp_default_ipv4_unicast_unset	96
2.1.2.69	smi_bgp_default_ipv4_unicast_unset_validate . . .	97
2.1.2.70	smi_bgp_default_local_preference_set_sdkapi_- validate	97
2.1.2.71	smi_bgp_default_local_preference_unset_sdkapi_- validate	97
2.1.2.72	smi_bgp_deterministic_med_set	98
2.1.2.73	smi_bgp_deterministic_med_set_validate	98
2.1.2.74	smi_bgp_deterministic_med_unset	98
2.1.2.75	smi_bgp_deterministic_med_unset_validate	99
2.1.2.76	smi_bgp_disable_adj_out_set	99
2.1.2.77	smi_bgp_disable_adj_out_set_validate	99
2.1.2.78	smi_bgp_disable_adj_out_unset	100
2.1.2.79	smi_bgp_disable_adj_out_unset_validate	100
2.1.2.80	smi_bgp_enforce_first_as_set	100
2.1.2.81	smi_bgp_enforce_first_as_set_validate	101
2.1.2.82	smi_bgp_enforce_first_as_unset	101

2.1.2.83	smi_bgp_enforce_first_as_unset_validate	101
2.1.2.84	smi_bgp_extcommunity_list_entry_unset_validate	102
2.1.2.85	smi_bgp_extcommunity_list_set	102
2.1.2.86	smi_bgp_extcommunity_list_unset	103
2.1.2.87	smi_bgp_fast_external_failover_set	103
2.1.2.88	smi_bgp_fast_external_failover_set_validate	104
2.1.2.89	smi_bgp_fast_external_failover_unset	104
2.1.2.90	smi_bgp_fast_external_failover_unset_validate	104
2.1.2.91	smi_bgp_get_address_family	105
2.1.2.92	smi_bgp_get_grst_restart_time	105
2.1.2.93	smi_bgp_get_grst_stalepath_time	105
2.1.2.94	smi_bgp_get_identifier	106
2.1.2.95	smi_bgp_get_local_as	106
2.1.2.96	smi_bgp_get_nbr_address_family	107
2.1.2.97	smi_bgp_get_peer_admin_status	107
2.1.2.98	smi_bgp_get_peer_connect_retry_interval	108
2.1.2.99	smi_bgp_get_peer_fsm_established_time	108
2.1.2.100	smi_bgp_get_peer_fsm_established_transitions	109
2.1.2.101	smi_bgp_get_peer_hold_time	109
2.1.2.102	smi_bgp_get_peer_hold_time_configured	110
2.1.2.103	smi_bgp_get_peer_identifier	110
2.1.2.104	smi_bgp_get_peer_in_total_messages	111
2.1.2.105	smi_bgp_get_peer_in_update_elapsed_time	111
2.1.2.106	smi_bgp_get_peer_in_updates	112
2.1.2.107	smi_bgp_get_peer_keep_alive	112
2.1.2.108	smi_bgp_get_peer_keep_alive_configured	113
2.1.2.109	smi_bgp_get_peer_last_error	113
2.1.2.110	smi_bgp_get_peer_local_addr	114
2.1.2.111	smi_bgp_get_peer_local_port	114
2.1.2.112	smi_bgp_get_peer_min_as_origination_interval	114
2.1.2.113	smi_bgp_get_peer_min_route_advertisement_interval	115
2.1.2.114	smi_bgp_get_peer_negotiated_version	115
2.1.2.115	smi_bgp_get_peer_out_total_messages	116
2.1.2.116	smi_bgp_get_peer_out_updates	116

2.1.2.117 smi_bgp_get_peer_remote_addr	117
2.1.2.118 smi_bgp_get_peer_remote_as	117
2.1.2.119 smi_bgp_get_peer_remote_port	118
2.1.2.120 smi_bgp_get_peer_state	118
2.1.2.121 smi_bgp_get_peer_timers	118
2.1.2.122 smi_bgp_get_update_delay_val	119
2.1.2.123 smi_bgp_get_version	119
2.1.2.124 smi_bgp_grst_restart_time_set_validate	120
2.1.2.125 smi_bgp_grst_restart_time_unset_validate	120
2.1.2.126 smi_bgp_grst_set_validate	120
2.1.2.127 smi_bgp_grst_stalepath_time_set_validate	121
2.1.2.128 smi_bgp_grst_stalepath_time_unset_validate	121
2.1.2.129 smi_bgp_grst_unset_validate	121
2.1.2.130 smi_bgp_instance_unset_sdkapi_validate	122
2.1.2.131 smi_bgp_maximum_paths_set	122
2.1.2.132 smi_bgp_maximum_paths_set_validate	122
2.1.2.133 smi_bgp_maximum_paths_unset	123
2.1.2.134 smi_bgp_maximum_paths_unset_validate	123
2.1.2.135 smi_bgp_multiple_instance_set	124
2.1.2.136 smi_bgp_multiple_instance_set_validate	124
2.1.2.137 smi_bgp_multiple_instance_unset	124
2.1.2.138 smi_bgp_multiple_instance_unset_validate	125
2.1.2.139 smi_bgp_nbr_address_family_set	125
2.1.2.140 smi_bgp_nbr_address_family_unset	125
2.1.2.141 smi_bgp_network_sync_set_sdkapi_validate	126
2.1.2.142 smi_bgp_network_sync_unset_sdkapi_validate	126
2.1.2.143 smi_bgp_no_debug_validate	127
2.1.2.144 smi_bgp_option_check_sdkapi	127
2.1.2.145 smi_bgp_option_set	128
2.1.2.146 smi_bgp_option_unset_validate	128
2.1.2.147 smi_bgp_peer_group_bind_sdkapi_validate	129
2.1.2.148 smi_bgp_peer_group_delete_unset_sdkapi_validate	129
2.1.2.149 smi_bgp_peer_group_remote_as_delete_unset_- sdkapi_validate	130

2.1.2.150 smi_bgp_peer_group_unbind_sdkapi_validate . . .	130
2.1.2.151 smi_bgp_peer_remote_as_set_sdkapi	131
2.1.2.152 smi_bgp_peer_unset_sdkapi_validate	131
2.1.2.153 smi_bgp_rfc1771_path_select_set	131
2.1.2.154 smi_bgp_rfc1771_path_select_set_validate	132
2.1.2.155 smi_bgp_rfc1771_path_select_unset	132
2.1.2.156 smi_bgp_rfc1771_path_select_unset_validate	132
2.1.2.157 smi_bgp_router_id_set_sdkapi_validate	133
2.1.2.158 smi_bgp_router_id_unset_sdkapi_validate	133
2.1.2.159 smi_bgp_set_peer_admin_status_validate	133
2.1.2.160 smi_bgp_set_peer_connect_retry_interval_validate	134
2.1.2.161 smi_bgp_set_peer_hold_time_configured_validate	134
2.1.2.162 smi_bgp_set_peer_keep_alive_configured_validate	135
2.1.2.163 smi_bgp_set_peer_min_as_origination_interval_- validate	135
2.1.2.164 smi_bgp_set_peer_min_route_advertisement_- interval_validate	136
2.1.2.165 smi_bgp_show_bgp	136
2.1.2.166 smi_bgp_show_bgp_extcommunity_list	137
2.1.2.167 smi_bgp_show_ip_bgp	137
2.1.2.168 smi_bgp_show_ip_bgp_community	138
2.1.2.169 smi_bgp_show_ip_bgp_community_list	138
2.1.2.170 smi_bgp_show_ip_bgp_extcommunity_list_exact_- match	139
2.1.2.171 smi_bgp_show_ip_bgp_extcommunity_list_exact_- match_vrf	139
2.1.2.172 smi_bgp_static_network_set_sdkapi_validate	140
2.1.2.173 smi_bgp_static_network_unset_sdkapi_validate	140
2.1.2.174 smi_bgp_synchronization_set_sdkapi_validate	141
2.1.2.175 smi_bgp_synchronization_unset_sdkapi_validate	141
2.1.2.176 smi_bgp_timers_set_sdkapi	142
2.1.2.177 smi_bgp_timers_unset_sdkapi_validate	142
2.1.2.178 smi_bgp_update_delay_val_set_validate	143
2.1.2.179 smi_bgp_update_delay_val_unset_validate	143
2.1.2.180 smi_bgp_vrf_neighbor_as_override_set	143

2.1.2.181 smi_bgp_vrf_neighbor_as_override_set_validate . .	144
2.1.2.182 smi_bgp_vrf_neighbor_as_override_unset	144
2.1.2.183 smi_bgp_vrf_neighbor_as_override_unset_validate	145
2.1.2.184 smi_filter_list_set_validate	145
2.1.2.185 smi_filter_list_unset_validate	145
2.1.2.186 smi_neighbor_attr_unchanged_as_path_set	146
2.1.2.187 smi_neighbor_attr_unchanged_as_path_unset . . .	146
2.1.2.188 smi_neighbor_attr_unchanged_med_set	147
2.1.2.189 smi_neighbor_attr_unchanged_med_unset	147
2.1.2.190 smi_neighbor_attr_unchanged_nexthop_set	148
2.1.2.191 smi_neighbor_attr_unchanged_nexthop_unset . . .	148
2.1.2.192 smi_neighbor_capability_grst_set	148
2.1.2.193 smi_neighbor_capability_grst_set_validate	149
2.1.2.194 smi_neighbor_capability_grst_unset	149
2.1.2.195 smi_neighbor_capability_grst_unset_validate . . .	150
2.1.2.196 smi_neighbor_capability_orf_prefix_set	150
2.1.2.197 smi_neighbor_capability_orf_prefix_set_validate .	151
2.1.2.198 smi_neighbor_capability_orf_prefix_unset	151
2.1.2.199 smi_neighbor_capability_orf_prefix_unset_validate	151
2.1.2.200 smi_neighbor_capability_route_refresh_set	152
2.1.2.201 smi_neighbor_capability_route_refresh_set_validate	152
2.1.2.202 smi_neighbor_capability_route_refresh_unset . . .	153
2.1.2.203 smi_neighbor_capability_route_refresh_unset_- validate	153
2.1.2.204 smi_neighbor_collide_established_set	153
2.1.2.205 smi_neighbor_collide_established_set_validate . .	154
2.1.2.206 smi_neighbor_collide_established_unset	154
2.1.2.207 smi_neighbor_collide_established_unset_validate .	155
2.1.2.208 smi_neighbor_connection_retry_time_unset_validate	155
2.1.2.209 smi_neighbor_disallow_infinite_timer_set_validate	155
2.1.2.210 smi_neighbor_disallow_infinite_timer_unset_validate	156
2.1.2.211 smi_neighbor_dont_capability_negotiate_unset_- validate	156
2.1.2.212 smi_neighbor_enforce_multihop_set	156

2.1.2.213 smi_neighbor_enforce_multihop_set_validate . . .	157
2.1.2.214 smi_neighbor_enforce_multihop_unset	157
2.1.2.215 smi_neighbor_enforce_multihop_unset_validate . .	158
2.1.2.216 smi_neighbor_filter_list_set_validate	158
2.1.2.217 smi_neighbor_filter_list_unset_validate	158
2.1.2.218 smi_neighbor_g_shut_time_set	159
2.1.2.219 smi_neighbor_g_shut_time_set_validate	159
2.1.2.220 smi_neighbor_g_shut_time_unset	160
2.1.2.221 smi_neighbor_g_shut_time_unset_validate	160
2.1.2.222 smi_neighbor_local_as_set_validate	160
2.1.2.223 smi_neighbor_local_as_unset_validate	161
2.1.2.224 smi_neighbor_override_capability_set	161
2.1.2.225 smi_neighbor_override_capability_set_validate . .	161
2.1.2.226 smi_neighbor_override_capability_unset	162
2.1.2.227 smi_neighbor_override_capability_unset_validate .	162
2.1.2.228 smi_neighbor_remove_private_as_set	163
2.1.2.229 smi_neighbor_remove_private_as_set_validate . . .	163
2.1.2.230 smi_neighbor_remove_private_as_unset	163
2.1.2.231 smi_neighbor_remove_private_as_unset_validate .	164
2.1.2.232 smi_neighbor_route_reflector_client_set_validate .	164
2.1.2.233 smi_neighbor_route_reflector_client_unset_validate	165
2.1.2.234 smi_neighbor_route_server_client_set	165
2.1.2.235 smi_neighbor_route_server_client_set_validate . .	166
2.1.2.236 smi_neighbor_route_server_client_unset	166
2.1.2.237 smi_neighbor_route_server_client_unset_validate .	167
2.1.2.238 smi_neighbor_strict_capability_set	167
2.1.2.239 smi_neighbor_strict_capability_set_validate	167
2.1.2.240 smi_neighbor_strict_capability_unset	168
2.1.2.241 smi_neighbor_strict_capability_unset_validate . . .	168
2.1.2.242 smi_neighbor_transparent_as_set_validate	169
2.1.2.243 smi_neighbor_transparent_nexthop_set_validate . .	169
2.1.2.244 smi_peer_activate_set_sdkapi_validate	169
2.1.2.245 smi_peer_advertise_interval_set_sdkapi_validate .	170
2.1.2.246 smi_peer_advertise_interval_unset_sdkapi_validate	170

2.1.2.247 smi_peer_af_flag_config_check	171
2.1.2.248 smi_peer_af_flag_set_sdkapi_validate	171
2.1.2.249 smi_peer_af_flag_unset_sdkapi_validate	172
2.1.2.250 smi_peer_allowas_in_set_sdkapi_validate	173
2.1.2.251 smi_peer_allowas_in_unset_sdkapi_validate	174
2.1.2.252 smi_peer_aslist_set_sdkapi_validate	174
2.1.2.253 smi_peer_aslist_unset_sdkapi_validate	175
2.1.2.254 smi_peer_asorig_interval_set_sdkapi_validate	175
2.1.2.255 smi_peer_asorig_interval_unset_sdkapi_validate	176
2.1.2.256 smi_peer_deactivate_sdkapi_validate	176
2.1.2.257 smi_peer_default_originate_set_sdkapi_validate	177
2.1.2.258 smi_peer_default_originate_unset_sdkapi	177
2.1.2.259 smi_peer_description_set_sdkapi_validate	178
2.1.2.260 smi_peer_description_unset_sdkapi_validate	178
2.1.2.261 smi_peer_disallow_hold_timer_set_sdkapi	178
2.1.2.262 smi_peer_disallow_hold_timer_unset_sdkapi	179
2.1.2.263 smi_peer_distribute_set_sdkapi_validate	179
2.1.2.264 smi_peer_distribute_unset_sdkapi	180
2.1.2.265 smi_peer_dont_capability_negotiate_set	180
2.1.2.266 smi_peer_dont_capability_negotiate_set_validate	180
2.1.2.267 smi_peer_dont_capability_negotiate_unset	181
2.1.2.268 smi_peer_dynamic_capability_set	181
2.1.2.269 smi_peer_dynamic_capability_set_validate	182
2.1.2.270 smi_peer_dynamic_capability_unset	182
2.1.2.271 smi_peer_dynamic_capability_unset_validate	182
2.1.2.272 smi_peer_ebgp_multihop_set_sdkapi_validate	183
2.1.2.273 smi_peer_ebgp_multihop_unset_sdkapi_validate	183
2.1.2.274 smi_peer_flag_config_check	183
2.1.2.275 smi_peer_flag_set_sdkapi_validate	184
2.1.2.276 smi_peer_flag_unset_sdkapi_validate	185
2.1.2.277 smi_peer_get_advertise_interval	185
2.1.2.278 smi_peer_get_allowas_in	186
2.1.2.279 smi_peer_get_asorig_interval	186
2.1.2.280 smi_peer_get_description	187

2.1.2.281 smi_peer_get_ebgp_multihop	187
2.1.2.282 smi_peer_get_interface	187
2.1.2.283 smi_peer_get_timers	188
2.1.2.284 smi_peer_get_timers_connect	188
2.1.2.285 smi_peer_get_update_source_info	188
2.1.2.286 smi_peer_interface_set_sdkapi_validate	189
2.1.2.287 smi_peer_interface_unset_sdkapi_validate	189
2.1.2.288 smi_peer_maximum_prefix_set_sdkapi_validate	190
2.1.2.289 smi_peer_next_hop_self_set	190
2.1.2.290 smi_peer_next_hop_self_set_validate	191
2.1.2.291 smi_peer_next_hop_self_unset	191
2.1.2.292 smi_peer_next_hop_self_unset_validate	191
2.1.2.293 smi_peer_password_set_validate	192
2.1.2.294 smi_peer_password_unset_sdkapi_validate	192
2.1.2.295 smi_peer_port_set_sdkapi	193
2.1.2.296 smi_peer_port_unset_sdkapi_validate	193
2.1.2.297 smi_peer_prefix_list_set_sdkapi	193
2.1.2.298 smi_peer_prefix_list_unset_sdkapi_validate	194
2.1.2.299 smi_peer_route_map_set_sdkapi_validate	194
2.1.2.300 smi_peer_route_map_unset_sdkapi_validate	195
2.1.2.301 smi_peer_route_reflector_client_set	196
2.1.2.302 smi_peer_route_reflector_client_unset	196
2.1.2.303 smi_peer_shutdown_set	196
2.1.2.304 smi_peer_shutdown_set_validate	197
2.1.2.305 smi_peer_shutdown_unset	197
2.1.2.306 smi_peer_shutdown_unset_validate	198
2.1.2.307 smi_peer_soft_reconfiguration_inbound_set	198
2.1.2.308 smi_peer_soft_reconfiguration_inbound_set_validate	198
2.1.2.309 smi_peer_soft_reconfiguration_inbound_unset	199
2.1.2.310 smi_peer_soft_reconfiguration_inbound_unset_validate	199
2.1.2.311 smi_peer_timers_connect_set_sdkapi_validate	200
2.1.2.312 smi_peer_timers_connect_unset_sdkapi_validate	200
2.1.2.313 smi_peer_timers_set_sdkapi_validate	201

2.1.2.314 smi_peer_timers_unset_sdkapi_validate	201
2.1.2.315 smi_peer_transport_connection_passive_set	201
2.1.2.316 smi_peer_transport_connection_passive_set_validate	202
2.1.2.317 smi_peer_transport_connection_passive_unset	202
2.1.2.318 smi_peer_transport_connection_passive_unset_- validate	203
2.1.2.319 smi_peer_unsuppress_map_set_sdkapi_validate	203
2.1.2.320 smi_peer_unsuppress_map_unset_sdkapi_validate	203
2.1.2.321 smi_peer_update_routing_source_set_sdkapi_validate	204
2.1.2.322 smi_peer_version_set_sdkapi_validate	204
2.1.2.323 smi_peer_version_unset_sdkapi_validate	205
2.1.2.324 smi_peer_weight_set_sdkapi_validate	205
2.1.2.325 smi_peer_weight_unset_sdkapi	206
2.1.2.326 smi_show_bgp_afi_regexp_safi	206
2.1.2.327 smi_show_bgp_afi_route_map_safi	206
2.1.2.328 smi_show_bgp_dampening_parameters	207
2.1.2.329 smi_show_bgp_inconsistent_as	207
2.1.2.330 smi_show_bgp_ip_neighbor_routes	208
2.1.2.331 smi_show_bgp_neighbor_advertised_routes	208
2.1.2.332 smi_show_bgp_neighbor_recieved_routes	209
2.1.2.333 smi_show_bgp_neighbors_recv_prefix_filter	209
2.1.2.334 smi_show_bgp_regexp	210
2.1.2.335 smi_show_bgp_route_map	210
2.1.2.336 smi_show_bgp_sessions	210
2.1.2.337 smi_show_bgp_summary	211
2.1.2.338 smi_show_bgp_V6_neighbors_recv_prefix_filter	211
2.1.2.339 smi_show_ip_bgp	212
2.1.2.340 smi_show_ip_bgp_cidr_only	212
2.1.2.341 smi_show_ip_bgp_community	212
2.1.2.342 smi_show_ip_bgp_dampening_dampend_paths	213
2.1.2.343 smi_show_ip_bgp_dampening_flap_statistics	213
2.1.2.344 smi_show_ip_bgp_filter_list_exact_match	214
2.1.2.345 smi_show_ip_bgp_ipv6_dampening_parameters	214
2.1.2.346 smi_show_ip_bgp_longer_prefixes	215

2.1.2.347	smi_show_ip_bgp_neighbors_HKC	215
2.1.2.348	smi_show_ip_bgp_paths	216
2.1.2.349	smi_show_ip_bgp_prefix_list_exact_match	216
2.1.2.350	smi_show_ip_bgp_quote_regexp	217
2.1.2.351	smi_show_ip_bgp_received_paths	217
2.1.2.352	smi_show_ip_bgp_regexp	218
2.1.2.353	smi_show_ip_bgp_route_map	218
2.1.2.354	smi_show_ip_bgp_safi_regexp	219
2.1.2.355	smi_show_ip_bgp_safi_route_map	219
2.1.2.356	smi_show_ip_bgp_summary	220
2.1.2.357	smi_show_ip_bgp_word_neighbors	220
2.1.2.358	smi_show_ip_bgp_word_peer_neighbors	220
2.1.2.359	smi_show_ip_protocol_all	221
2.1.2.360	smi_transport_connection_passive_set_validate	221
2.1.2.361	smi_transport_connection_passive_unset_validate	222
2.2	smi_bgp_bfd.h File Reference	223
2.2.1	Detailed Description	223
2.2.2	Function Documentation	223
2.2.2.1	smi_bgp_peer_bfd_set	223
2.2.2.2	smi_bgp_peer_bfd_unset	224

Chapter 1

File Index

1.1 File List

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

smi_bgp.h (Provides APIs for managing BGP Protocol)	3
smi_bgp_bfd.h (Provides APIs for managing Bidirectional Forwarding De- tection(BFD) in ZebOS)	223

Chapter 2

File Documentation

2.1 smi_bgp.h File Reference

Provides APIs for managing BGP Protocol. #include "smi_client.h"
#include "smi_bgp_msg.h"

Functions

- s_int32_t [smi_bgp_get_version](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, int *bgpVersion)
This function returns the version of the supported BGP version.
- int [smi_bgp_get_local_as](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, int *bgpAs)
bgp_get function returns the pointer to the specified BGP instance. bgp_get_local_as refer to local autonomous system number, where Autonomous System is a set of routers under a single technical administration.
- int [smi_bgp_get_identifier](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr *routerId)
bgp_get function returns the pointer to the specified BGP instance. The BGP Identifier of the local system.
- int [smi_bgp_get_peer_identifier](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr *peerAddr, struct pal_in4_addr *peerRouterId)
The BGP Identifier of this entry's(Entry containing information about the connection with a BGP peer) BGP peer.This entry MUST be 0.0.0.0 unless the bgpPeerState is in the openconfirm or the established state.
- int [smi_bgp_get_peer_state](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr *peerAddr, int *peerState)

This function returns the pointer to the bgp peer state(BGP instance). If no pointer is returned, it tries to create a new one.

- int [smi_bgp_get_peer_admin_status](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr *peerAddr, int *peerAdminFlag)

The desired state of the BGP connection. A transition from 'stop' to 'start' will cause the BGP Manual Start Event to be generated. A transition from 'start' to 'stop' will cause the BGP Manual Stop Event to be generated. This parameter can be used to restart BGP peer connections. Care should be used in providing write access to this object without adequate authentication.

- int [smi_bgp_set_peer_admin_status_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr peerAddr, s_int32_t peerAdminFlag)

The desired state of the BGP connection. A transition from 'stop' to 'start' will cause the BGP Manual Start Event to be generated. A transition from 'start' to 'stop' will cause the BGP Manual Stop Event to be generated. This parameter can be used to restart BGP peer connections. Care should be used in providing write access to this object without adequate authentication.

- int [smi_bgp_set_peer_admin_status](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr peerAddr, s_int32_t peerAdminFlag)
- int [smi_bgp_get_peer_negotiated_version](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr *peerAddr, int *bgpPeerNegotiatedVersion)

This function gets the negotiated version of BGP running between the two peers. This entry MUST be zero (0) unless the bgpPeerState is in the openconfirm or the established state. Note that legal values for this object are between 0 and 255.

- int [smi_bgp_get_peer_local_addr](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr *peerAddr, struct pal_in4_addr *peerLocalAddr)

This function returns the local address of the peers BGP connection.

- int [smi_bgp_get_peer_local_port](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr *peerAddr, int *peerLocalPort)

This function returns the local port for the TCP connection between the BGP peers.

- int [smi_bgp_get_peer_remote_addr](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr *peerAddr, struct pal_in4_addr *peerRemoteAddr)

The remote IP address of this entry's(Entry containing information about the connection with a BGP peer) BGP peer.

- int [smi_bgp_get_peer_remote_port](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr *peerAddr, int *peerRemotePort)

This function returns the remote port for the TCP connection between the BGP peers.

- int [smi_bgp_get_peer_remote_as](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr *addr, int *peerRemoteAs)
This function returns the pointer to the remote autonomous system number received in the BGP OPEN message.
- int [smi_bgp_get_peer_in_updates](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr *addr, int *peerInUpdates)
This function returns the number of BGP UPDATE messages received on BGP connection.
- int [smi_bgp_get_peer_out_updates](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr *peerAddr, int *peerOutUpdates)
This function returns the number of BGP UPDATE messages transmitted on BGP connection.
- int [smi_bgp_get_peer_in_total_messages](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr *peerAddr, int *peerInTotalMsg)
The total number of messages received from the remote peer on BGP connection.
- int [smi_bgp_get_peer_out_total_messages](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr *peerAddr, int *peerOutTotalMsg)
This function returns the total number of messages transmitted to the remote peer on BGP connection.
- int [smi_bgp_get_peer_last_error](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr *peerAddr, u_int16_t *peerLastError)
The last error code and subcode seen by the peer on BGP connection. If no error has occurred, this field is zero. Otherwise, the first byte of this two byte OCTET STRING contains the error code, and the second byte contains the subcode.
- int [smi_bgp_get_peer_fsm_established_transitions](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr *peerAddr, int *estTxns)
The total number of times the BGP FSM transitioned into the established state for this peer.
- int [smi_bgp_get_peer_fsm_established_time](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr *peerAddr, int *estTime)
This timer indicates how long (in seconds) this peer has been in the established state or how long since this peer was last in the established state. It is set to zero when a new peer is configured or when the router is booted.
- int [smi_bgp_get_peer_connect_retry_interval](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr *peerAddr, int *connRetryInterval)
Time interval (in seconds) for the ConnectRetry timer. The suggested value for this timer is 120 seconds.

- int [smi_bgp_set_peer_connect_retry_interval_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr peerAddr, int connRetryInterval)

Time interval (in seconds) for the ConnectRetry timer. The suggested value for this timer is 120 seconds.

- int [smi_bgp_set_peer_connect_retry_interval](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr peerAddr, int connRetryInterval)
- int [smi_bgp_get_peer_hold_time](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr *peerAddr, int *holdTime)

This function returns the time interval in seconds that the Hold timer has been established with the BGP peer. The value must be at least 3 seconds or zero (0), which means the Hold timer has not been established with the peer.

- int [smi_bgp_get_peer_keep_alive](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr *peerAddr, int *keepAlive)

Time interval (in seconds) for the KeepAlive timer established with the peer. The value of this object is calculated by this BGP speaker such that, when compared with bgpPeerHoldTime, it has the same proportion that bgpPeerKeepAliveConfigured has, compared with bgpPeerHoldTimeConfigured.

- int [smi_bgp_get_peer_hold_time_configured](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr *peerAddr, int *holdTimeConf)

Time interval (in seconds) for the Hold Time configured for this BGP speaker with this peer. This value is placed in an OPEN message sent to this peer by this BGP speaker, and is compared with the Hold Time field in an OPEN message received from the peer when determining the Hold Time (bgpPeerHoldTime) with the peer. This value must not be less than three seconds if it is not zero (0). If it is zero (0), the Hold Time is NOT to be established with the peer. The suggested value for this timer is 90 seconds.

- int [smi_bgp_set_peer_hold_time_configured_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr addr, int holdTimeConf)

This function modifies the time interval in seconds for the hold time configured for this BGP speaker with the peer. The value must be at least 3 seconds or 0 (zero), which means the Hold timer has not been established with the peer. The suggested value for this timer is 90 seconds.

- int [smi_bgp_set_peer_hold_time_configured](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr peerAddr, int holdTimeConf)

- int [smi_bgp_get_peer_keep_alive_configured](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr *peerAddr, int *keepAliveConf)

Time interval (in seconds) for the KeepAlive timer established with the peer. The value of this object is calculated by this BGP speaker such that, when compared with

bgpPeerHoldTime, it has the same proportion that bgpPeerKeepAliveConfigured has, compared with bgpPeerHoldTimeConfigured.

- int [smi_bgp_set_peer_keep_alive_configured_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr peerAddr, int keepAliveConf)

This function modifies the time interval in seconds for the KeepAlive timer configured for this BGP speaker with the peer. If the value of this object is zero, no periodical KEEPALIVE messages are sent to the peer after the BGP connection has been established. The suggested value for this timer is 30 seconds.

- int [smi_bgp_set_peer_keep_alive_configured](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr peerAddr, int keepAliveConf)

- int [smi_bgp_get_peer_min_as_origination_interval](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpAs, struct pal_in4_addr *peerAddr, int *minAsOrigInterval)

Time interval (in seconds) for the MinASOriginationInterval timer. The suggested value for this timer is 15 seconds.

- int [smi_bgp_set_peer_min_as_origination_interval_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpAs, struct pal_in4_addr peerAddr, int minAsOrigInterval)

Time interval (in seconds) for the MinASOriginationInterval timer. The suggested value for this timer is 15 seconds.

- int [smi_bgp_set_peer_min_as_origination_interval](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpAs, struct pal_in4_addr peerAddr, int minAsOrigInterval)

- int [smi_bgp_set_peer_min_as_origination_interval_wrap_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpAs, char *peerAddr, int minAsOrigInterval)

- int [smi_bgp_set_peer_min_as_origination_interval_wrap](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpAs, char *peerAddr, int minAsOrigInterval)

- int [smi_bgp_get_peer_min_route_advertisement_interval](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr *peerAddr, int *minRouteAdInterval)

Time interval (in seconds) for the MinRouteAdvertisementInterval timer. The suggested value for this timer is 30 seconds for EBGp connections and 5 seconds for IBGP connections.

- int [smi_bgp_set_peer_min_route_advertisement_interval_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr peerAddr, int minRouteAdInterval)

This function modifies the time interval in seconds for the MinRouteAdvertisementInterval timer. The suggested value for this timer is 30 seconds.

- `int smi_bgp_set_peer_min_route_advertisement_interval` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `int bgpProcId`, `struct pal_in4_addr peerAddr`, `int minRouteAdInterval`)
- `int smi_bgp_get_peer_in_update_elapsed_time` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `int bgpProcId`, `struct pal_in4_addr *peerAddr`, `int *inUpdateElaps`)

Elapsed time (in seconds) since the last BGP UPDATE message was received from the peer. Each time `bgpPeerInUpdates` is incremented, the value of this object is set to zero (0).

- `int smi_bgp4_get_path_attr_peer_sdkapi` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `int bgpProcId`, `struct prefix_ipv4 *routeAddr`, `struct pal_in4_addr *peerAddr`, `struct pal_in4_addr *pathAttrPeerAddr`)

The IP address of the peer where the path information was learned.

- `int smi_bgp4_get_path_attr_ip_addr_prefix_len_sdkapi` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `int bgpProcId`, `struct prefix_ipv4 *routeAddr`, `struct pal_in4_addr *peerAddr`, `int *pathAttrPeerLen`)

Length in bits of the IP address prefix in the Network Layer Reachability Information field.

- `int smi_bgp4_get_path_attr_ip_addr_prefix_sdkapi` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `int bgpProcId`, `struct prefix_ipv4 *routeAddr`, `struct pal_in4_addr *peerAddr`, `struct pal_in4_addr *pathAttrIpAddrPrefix`)

An IP address prefix in the Network Layer Reachability Information field. This object is an IP address containing the prefix with length specified by `bgp4PathAttrIpAddrPrefixLen`. Any bits beyond the length specified by `bgp4PathAttrIpAddrPrefixLen` are zeroed.

- `int smi_bgp4_get_path_attr_origin_sdkapi` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `int bgpProcId`, `struct prefix_ipv4 *routeAddr`, `struct pal_in4_addr *peerAddr`, `int *origin`)

The ultimate origin of the path information.

- `int smi_bgp4_get_path_attr_next_hop_sdkapi` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `int bgpProcId`, `struct prefix_ipv4 *routeAddr`, `struct pal_in4_addr *peerAddr`, `struct pal_in4_addr *pathAttrNextHop`)

The address of the border router that should be used for the destination network. This address is the NEXT_HOP address received in the UPDATE packet.

- `int smi_bgp4_get_path_attr_multi_exit_disc_sdkapi` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `int bgpProcId`, `struct prefix_ipv4 *routeAddr`, `struct pal_in4_addr *peerAddr`, `int *med`)

This metric is used to discriminate between multiple exit points to an adjacent autonomous system. A value of -1 indicates the absence of this attribute.

- `int smi_bgp4_get_path_attr_local_pref_sdkapi` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `int bgpProcId`, `struct prefix_ipv4 *routeAddr`, `struct pal_in4_addr *peerAddr`, `int *localPref`)

The originating BGP4 speaker's degree of preference for an advertised route. A value of -1 indicates the absence of this attribute.

- int [smi_bgp4_get_path_attr_atomic_aggregate_sdkapi](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct prefix_ipv4 *routeAddr, struct pal_in4_addr *peerAddr, int *atomic)

This function returns the pointer to the specified BGP instance. If no pointer is returned, it tries to create a new one. ATOMIC_AGGREGATE is a primarily informational attribute.

- int [smi_bgp4_get_path_attr_aggregator_as_sdkapi](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct prefix_ipv4 *routeAddr, struct pal_in4_addr *peerAddr, int *pathAttrAggregatorAs)

The AS number of the last BGP4 speaker that performed route aggregation. A value of zero (0) indicates the absence of this attribute.

- int [smi_bgp4_get_path_attr_aggregator_addr_sdkapi](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct prefix_ipv4 *routeAddr, struct pal_in4_addr *peerAddr, struct pal_in4_addr *pathAttrAggregatorAddr)

The IP address of the last BGP4 speaker that performed route aggregation. A value of 0.0.0.0 indicates the absence of this attribute.

- int [smi_bgp4_get_path_attr_calc_local_pref_sdkapi](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct prefix_ipv4 *routeAddr, struct pal_in4_addr *peerAddr, int *localPref)

The degree of preference calculated by the receiving BGP4 speaker for an advertised route. A value of -1 indicates the absence of this attribute.

- int [smi_bgp4_get_path_attr_best_sdkapi](#) (struct smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct prefix_ipv4 *routeAddr, struct pal_in4_addr *peerAddr, int *pathAttrBest)

An indication of whether this route was chosen as the best BGP4 route for this destination.

- s_int32_t [smi_bgp_option_set](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, u_int32_t optFlag)

This function sets the BGP option. The BGP option is a system-wide pre-configurable setting, and is usually not accessible to the end user.

- s_int32_t [smi_bgp_option_set_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t optFlag)

- s_int32_t [smi_bgp_option_unset_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t optFlag)

This function unsets the BGP option. The BGP option is a system-wide pre-configurable setting, and is usually not accessible to the end user.

- s_int32_t [smi_bgp_option_unset](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, u_int32_t optFlag)

- int [smi_bgp_clear_gen_sdkapi](#) (struct smiclient_globals *azg, u_int32_t vrId, char *name, int afi, int safi, int sort, s_int32_t stype, char *clearString)

Clear BGP connections.

- int [smi_bgp_clear_gen_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, char *name, int afi, int safi, int sort, s_int32_t stype, char *clearString)
- int [smi_bgp_clear_wrap](#) (struct smiclient_globals *azg, u_int32_t vrId)
- int [smi_bgp_community_list_set](#) (struct smiclient_globals *azg, u_int32_t vrId, char *commListName, char *commListValue, int nameType, int action, int entryType)

Configure BGP community filtering.

- int [smi_bgp_community_list_set_sdkapi](#) (struct smiclient_globals *azg, u_int32_t vrId, char *commListName, char *commListValue, int action, int entryType)
- int [smi_bgp_community_list_set_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, char *commListName, char *commListValue, int action, int entryType)
- int [smi_bgp_community_list_unset_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, char *commListName)

Unconfigure BGP community filtering.

- int [smi_bgp_community_list_unset](#) (struct smiclient_globals *azg, u_int32_t vrId, char *commListName)
- int [smi_bgp_community_list_entry_unset](#) (struct smiclient_globals *azg, u_int32_t vrId, char *commListName, char *commListValue, int nameType, int action, int entryType)

Unconfigure BGP community filtering.

- int [smi_bgp_community_list_entry_unset_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, char *commListName, char *commListValue, int nameType, int action, int entryType)
- int [smi_bgp_extcommunity_list_set](#) (struct smiclient_globals *azg, u_int32_t vrId, char *commListName, char *commListValue, int nameType, int action, int entryType)

Configure BGP extended community filtering.

- int [smi_bgp_extcommunity_list_set_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, char *commListName, char *commListValue, int nameType, int action, int entryType)
- int [smi_bgp_extcommunity_list_unset](#) (struct smiclient_globals *azg, u_int32_t vrId, char *commListName)

Unconfigure BGP extended community filtering.

- int [smi_bgp_extcommunity_list_unset_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, char *commListName)

- int [smi_bgp_extcommunity_list_entry_unset_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, char *commListName, char *commListValue, int nameType, int action, int entryType)

Unconfigure BGP extended community filtering.

- int **smi_bgp_extcommunity_list_entry_unset** (struct smiclient_globals *azg, u_int32_t vrId, char *commListName, char *commListValue, int nameType, int action, int entryType)
- int [smi_bgp_aspath_access_list_set_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, char *accessListName, char *bgpRegExp, int action)

Unconfigure BGP Autonomous System path filtering defined by the regular expression.

- int **smi_bgp_aspath_access_list_set** (struct smiclient_globals *azg, u_int32_t vrId, char *accessListName, char *bgpRegExp, int action)
- int [smi_bgp_aspath_access_list_unset_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, char *accessListName)

Unconfigure BGP Autonomous System path filtering defined by the regular expression.

- int **smi_bgp_aspath_access_list_unset** (struct smiclient_globals *azg, u_int32_t vrId, char *accessListName)
- s_int32_t [smi_bgp_peer_remote_as_set_sdkapi](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int peerAs)

Sets the remote Autonomous System number of this entry's BGP peer group.

- s_int32_t **smi_bgp_peer_remote_as_set_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int peerAs)
- s_int32_t [smi_bgp_peer_group_remote_as_delete_unset_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerGroup)

Removes the remote Autonomous System number of this entry's BGP peer group.

- s_int32_t **smi_bgp_peer_group_remote_as_delete_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerGroup)
- s_int32_t [smi_bgp_create_instance_set_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs)

Gets the BGP instance of given Autonomous System number if already exists or Creates new instance.

- s_int32_t **smi_bgp_create_instance_set_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs)
- s_int32_t [smi_bgp_instance_unset_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs)

Deletes the specified BGP instance.

- s_int32_t **smi_bgp_instance_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs)
- int [smi_bgp_router_id_set_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *routerIpAddr)

Configure the BGP router ID.

- int **smi_bgp_router_id_set_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *routerIpAddr)
- int [smi_bgp_router_id_unset_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *routerIpAddr)

Deletes the BGP router ID.

- int **smi_bgp_router_id_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *routerIpAddr)
- int [smi_bgp_cluster_id_set_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *clusterId)

Sets the BGP Route-Reflector Cluster-id as in IP address format.

- int **smi_bgp_cluster_id_set_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *clusterId)
- int **smi_bgp_cluster_id_set_sdkapi_wrap** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, char *clusterId)
- int **smi_bgp_cluster_id_set_sdkapi_wrap_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, char *clusterId)
- int [smi_bgp_cluster_id_unset_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs)

Deletes the BGP Route-Reflector Cluster-id.

- int **smi_bgp_cluster_id_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs)
- int [smi_bgp_cluster_id_digit_set_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, u_int32_t clusterIdDigit)

Sets the BGP Route-Reflector Cluster-id as in 32 bit quantity.

- int **smi_bgp_cluster_id_digit_set_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, u_int32_t clusterIdDigit)
- int [smi_bgp_confederation_id_set_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int confedId)

Sets the AS Confederation identifier of BGP confederations. BGP Confederations is used to create a confederation of autonomous systems that is represented as a single autonomous system to BGP peers external to the confederation, thereby removing the "full mesh" requirement.

- int **smi_bgp_confederation_id_set_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int confedId)
- int [smi_bgp_confederation_id_unset_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs)

Deletes the AS Confederation identifier of BGP confederations .

- int **smi_bgp_confederation_id_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs)

- int [smi_bgp_confederation_peers_add_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int confedId)
Adds a Peer Member-AS Number of BGP confederation.
- int [smi_bgp_confederation_peers_add_sdkapi](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int confedId)
- int [smi_bgp_confederation_peers_remove_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int confedId)
Deletes a Peer Member-AS Number of BGP confederation.
- int [smi_bgp_confederation_peers_remove_sdkapi](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int confedId)
- int [smi_bgp_timers_set_sdkapi](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, u_int16_t keepAlive, u_int16_t holdTime)
Sets the time intervals in seconds for BGP's Hold Timer and KeepAlive Timer.
- int [smi_bgp_timers_set_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, u_int16_t keepAlive, u_int16_t holdTime)
- int [smi_bgp_timers_unset_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs)
Unsets the BGP's Hold Timer and KeepAlive Timer.
- int [smi_bgp_timers_unset_sdkapi](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs)
- int [smi_bgp_default_local_preference_set_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, u_int32_t localPref)
Sets the default LOCAL_PREF attribute of a BGP speaker. A BGP speaker uses it to inform its other internal peers of the advertising speaker's degree of preference for an advertised route.
- int [smi_bgp_default_local_preference_set_sdkapi](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, u_int32_t localPref)
- int [smi_bgp_default_local_preference_unset_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs)
Unsets the default LOCAL_PREF attribute of a BGP speaker.
- int [smi_bgp_default_local_preference_unset_sdkapi](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs)
- s_int32_t [smi_bgp_auto_summary_update_set_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, bool_t autoSummary)
Enables automatic network number summarization.
- s_int32_t [smi_bgp_auto_summary_update_set_sdkapi](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, bool_t autoSummary)
- s_int32_t [smi_bgp_address_family_auto_summary_update_set_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int afi, int safi, bool_t autoSummary)

- s_int32_t **smi_bgp_address_family_auto_summary_update_set_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int afi, int safi, bool_t autoSummary)
- s_int32_t **smi_bgp_synchronization_set_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int afi, int safi)

Enables IGP synchronization of BGP routes.

- s_int32_t **smi_bgp_synchronization_set_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int afi, int safi)
- s_int32_t **smi_bgp_synchronization_unset_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int afi, int safi)

Disables IGP synchronization of BGP routes.

- s_int32_t **smi_bgp_synchronization_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int afi, int safi)
- s_int32_t **smi_bgp_network_sync_set_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int afi, int safi)

Sets to perform IGP synchronization of network routes to announce via BGP.

- s_int32_t **smi_bgp_network_sync_set_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int afi, int safi)
- s_int32_t **smi_bgp_network_sync_unset_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int afi, int safi)

Unsets to perform IGP synchronization of network routes to announce via BGP.

- s_int32_t **smi_bgp_network_sync_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int afi, int safi)
- s_int32_t **smi_bgp_peer_unset_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr)

Deletes the specified peer from the peer-group.

- s_int32_t **smi_bgp_peer_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr)
- s_int32_t **smi_bgp_peer_group_delete_unset_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *pgTag)

Deletes the specified peer-group.

- s_int32_t **smi_bgp_peer_group_delete_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *pgTag)
- int **smi_peer_activate_set_wrap_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, bool_t peerActivate)
- int **smi_peer_activate_set_wrap_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, bool_t peerActivate)
- int **smi_peer_addr_family_activate_set_wrap_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *vrf_name, int peerAfi, int peerSafi, char *peerAddr, bool_t peerActivateAf)

- int **smi_peer_addr_family_activate_set_wrap_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *vrf_name, int peerAfi, int peerSafi, char *peerAddr, bool_t peerActivateAf)
- int **smi_peer_activate_set_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr, int afi, int safi)

Activate the Address Family for this Neighbor.

- int **smi_peer_activate_set_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr, int afi, int safi)
- int **smi_peer_deactivate_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, char *peerAddr)

Deactivate the Address Family for this Neighbor.

- int **smi_peer_deactivate_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, char *peerAddr)
- int **smi_peer_addr_family_deactivate_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, char *peerAddr, int afi, int safi)
- int **smi_peer_addr_family_deactivate_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, char *peerAddr, int afi, int safi)
- s_int32_t **smi_bgp_peer_group_bind_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, char *pgName)

Binds a peer to specified peer-group. When a peer does not exist, it creates a new peer.

- s_int32_t **smi_bgp_peer_group_bind_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, char *pgName)
- s_int32_t **smi_bgp_addr_family_peer_group_bind_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int afi, int safi, char *pgName)
- s_int32_t **smi_bgp_addr_family_peer_group_bind_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int afi, int safi, char *pgName)
- s_int32_t **smi_bgp_peer_group_unbind_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, char *pgName)

Unbinds a peer from a specified peer-group.

- s_int32_t **smi_bgp_peer_group_unbind_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, char *pgName)
- s_int32_t **smi_bgp_addr_family_peer_group_unbind_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int afi, int safi, char *pgName)
- s_int32_t **smi_bgp_addr_family_peer_group_unbind_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int afi, int safi, char *pgName)
- int **smi_peer_flag_set_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr, u_int32_t peerFlag)

Sets the peer configuration flag.

- int **smi_peer_flag_set_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr, u_int32_t peerFlag)
- int [smi_peer_flag_unset_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr, u_int32_t peerFlag)

Unsets the peer configuration flag.

- int **smi_peer_flag_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr, u_int32_t peerFlag)
- int [smi_peer_af_flag_set_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr, int afi, int safi, u_int32_t peerAfFlag)

Sets the peer's address family only configuration flag.

- int **smi_peer_af_flag_set_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr, int afi, int safi, u_int32_t peerAfFlag)
- int [smi_peer_af_flag_unset_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr, int afi, int safi, u_int32_t peerAfFlag)

Unsets the peer's address family only configuration flag.

- int **smi_peer_af_flag_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr, int afi, int safi, u_int32_t peerAfFlag)
- int [smi_peer_ebgp_multihop_set_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, u_int8_t timeToLive)

Sets TTL to EBGp neighbors that are not on directly connected networks.

- int **smi_peer_ebgp_multihop_set_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, u_int8_t timeToLive)
- int [smi_peer_ebgp_multihop_unset_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr)

Unsets TTL to EBGp neighbors that are not on directly connected networks.

- int **smi_peer_ebgp_multihop_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr)
- int [smi_peer_description_set_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, char *peerDesc)

Sets the BGP Neighbor's description.

- int **smi_peer_description_set_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, char *peerDesc)
- int [smi_peer_description_unset_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr)

Unsets the BGP Neighbor's description.

- int **smi_peer_description_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr)

- int [smi_peer_update_routing_source_set_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, char *sourceId)

Sets the source for routing updates.

- int **smi_peer_update_routing_source_set_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, char *sourceId)
- int [smi_peer_default_originate_set_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, char *rmapName)

Sets the source for originate default route to this neighbor, using route-map or without using.

- int **smi_peer_default_originate_set_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, char *rmapName)
- int **smi_peer_addr_family_default_originate_set_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int afi, int safi, char *rmapName)
- int **smi_peer_addr_family_default_originate_set_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int afi, int safi, char *rmapName)
- int [smi_peer_default_originate_unset_sdkapi](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr)

Unsets the source for originate default route to this neighbor.

- int **smi_peer_default_originate_unset_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr)
- int **smi_peer_addr_family_default_originate_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int afi, int safi)
- int **smi_peer_addr_family_default_originate_unset_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int afi, int safi)
- int [smi_peer_port_set_sdkapi](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, u_int16_t bgpPort)

Sets neighbor's BGP port number.

- int **smi_peer_port_set_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, u_int16_t bgpPort)
- int [smi_peer_port_unset_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, char *bgpPort)

Unsets neighbor's BGP port number.

- int **smi_peer_port_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, char *bgpPort)
- int [smi_peer_weight_set_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, u_int16_t weight)

Sets the default weight for routes from this port of neighbors.

- **int smi_peer_weight_set_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, u_int16_t weight)
- **int smi_peer_addr_family_weight_set_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, u_int16_t weight, int afi, int safi)
- **int smi_peer_addr_family_weight_set_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, u_int16_t weight, int afi, int safi)
- **int smi_peer_weight_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr)

Unsets the default weight for routes from this port of neighbors.

- **int smi_peer_weight_unset_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr)
- **int smi_peer_addr_family_weight_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int afi, int safi)
- **int smi_peer_addr_family_weight_unset_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int afi, int safi)
- **int smi_peer_timers_set_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, u_int16_t keepAlive, u_int16_t holdTime)

Sets the time intervals in seconds for peer's Hold Timer and KeepAlive Timer.

- **int smi_peer_timers_set_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, u_int16_t keepAlive, u_int16_t holdTime)
- **int smi_peer_timers_unset_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr)

Sets the time intervals to default in seconds for peer's Hold Timer and KeepAlive Timer.

- **int smi_peer_timers_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr)
- **int smi_peer_timers_connect_set_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, u_int32_t peerConnectInterval)

Sets time interval (in seconds) for the ConnectRetry timer.

- **int smi_peer_timers_connect_set_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, u_int32_t peerConnectInterval)
- **int smi_peer_timers_connect_unset_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr)

Unsets time interval (in seconds) for the ConnectRetry timer.

- **int smi_peer_timers_connect_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr)

- int [smi_peer_asorig_interval_set_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, u_int32_t peerAsorigInterval)
Sets time interval (in seconds) between sending AS-origination routing updates.
- int **smi_peer_asorig_interval_set_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, u_int32_t peerAsorigInterval)
- int [smi_peer_asorig_interval_unset_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr)
Unsets time interval (in seconds) between sending AS-origination routing updates.
- int **smi_peer_asorig_interval_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr)
- int [smi_peer_advertise_interval_set_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, u_int32_t peerRaInterval)
Sets time interval (in seconds) interval between sending BGP routing updates.
- int **smi_peer_advertise_interval_set_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, u_int32_t peerRaInterval)
- int [smi_peer_advertise_interval_unset_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr)
Sets time interval (in seconds) interval between sending BGP routing updates to default value.
- int **smi_peer_advertise_interval_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr)
- int [smi_peer_version_set_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, u_int32_t bgpVersion)
Sets the Neighbor's BGP version.
- int **smi_peer_version_set_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, u_int32_t bgpVersion)
- int [smi_peer_version_unset_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr)
This function unsets the BGP version.
- int **smi_peer_version_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr)
- int [smi_peer_interface_set_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, char *ifName)
Sets the peer's interface local IP address.
- int **smi_peer_interface_set_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, char *ifName)
- int [smi_peer_interface_unset_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, char *ifName)

Unsets the peer's interface local IP address.

- **int smi_peer_interface_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, char *ifName)
- **int smi_peer_allowas_in_set_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, u_int32_t allowAsNum)

Enables to accept AS path with my AS present in it for MPLS VPN/BGP environment.

- **int smi_peer_allowas_in_set_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, u_int32_t allowAsNum)
- **int smi_peer_addr_family_allowas_in_set_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int afi, int safi, u_int32_t allowAsNum)
- **int smi_peer_addr_family_allowas_in_set_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int afi, int safi, u_int32_t allowAsNum)
- **int smi_peer_allowas_in_unset_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr)

Disables the AS path loop check for MPLS VPN/BGP environment.

- **int smi_peer_allowas_in_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr)
- **int smi_peer_addr_family_allowas_in_unset_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int afi, int safi)
- **int smi_peer_addr_family_allowas_in_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int afi, int safi)
- **int smi_peer_distribute_set_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, u_int32_t direction, char *aclInfo)

Sets to filter UPDATEs to/from this neighbor.

- **int smi_peer_distribute_set_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, u_int32_t direction, char *aclInfo)
- **int smi_peer_addr_family_distribute_set_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int afi, int safi, u_int32_t direction, char *aclInfo)
- **int smi_peer_addr_family_distribute_set_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int afi, int safi, u_int32_t direction, char *aclInfo)
- **int smi_peer_distribute_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, u_int32_t direction)

Unsets to filter UPDATEs to/from this neighbor.

- **int smi_peer_distribute_unset_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, u_int32_t direction)

- int **smi_peer_addr_family_distribute_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int afi, int safi, u_int32_t direction)
- int **smi_peer_addr_family_distribute_unset_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int afi, int safi, u_int32_t direction)
- int **smi_peer_prefix_list_set_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, u_int32_t direction, char *aclInfo)

Sets to filter address prefixes to/from this neighbor.

- int **smi_peer_prefix_list_set_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, u_int32_t direction, char *aclInfo)
- int **smi_peer_addr_family_prefix_list_set_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int afi, int safi, u_int32_t direction, char *aclInfo)
- int **smi_peer_addr_family_prefix_list_set_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int afi, int safi, u_int32_t direction, char *aclInfo)
- int **smi_peer_prefix_list_unset_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, u_int32_t direction)

Unsets to filter address prefixes to/from this neighbor.

- int **smi_peer_prefix_list_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, u_int32_t direction)
- int **smi_peer_addr_family_prefix_list_unset_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int afi, int safi, u_int32_t direction)
- int **smi_peer_addr_family_prefix_list_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int afi, int safi, u_int32_t direction)
- int **smi_peer_aslist_set_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t bgpAs, char *peerAddr, u_int32_t direction, char *aclInfo, u_int32_t vrId)

Sets to filter AS Path segments to/from this neighbor.

- int **smi_peer_aslist_set_sdkapi** (struct smiclient_globals *azg, u_int32_t bgpAs, char *peerAddr, u_int32_t direction, char *aclInfo, u_int32_t vrId)
- int **smi_peer_addr_family_aslist_set_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t bgpAs, char *peerAddr, int afi, int safi, u_int32_t direction, char *aclInfo, u_int32_t vrId)
- int **smi_peer_addr_family_aslist_set_sdkapi** (struct smiclient_globals *azg, u_int32_t bgpAs, char *peerAddr, int afi, int safi, u_int32_t direction, char *aclInfo, u_int32_t vrId)
- int **smi_peer_aslist_unset_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t bgpAs, char *peerAddr, u_int32_t direction, u_int32_t vrId)

Unsets to filter AS Path segments to/from this neighbor.

- **int smi_peer_aslist_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t bgpAs, char *peerAddr, u_int32_t direction, u_int32_t vrId)
- **int smi_peer_addr_family_aslist_unset_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t bgpAs, char *peerAddr, int afi, int safi, u_int32_t direction, u_int32_t vrId)
- **int smi_peer_addr_family_aslist_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t bgpAs, char *peerAddr, int afi, int safi, u_int32_t direction, u_int32_t vrId)
- **int smi_peer_route_map_set_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t bgpAs, char *peerAddr, u_int32_t direction, char *aclInfo, u_int32_t vrId)

Sets to filter Route-Map segments to/from this neighbor.

- **int smi_peer_route_map_set_sdkapi** (struct smiclient_globals *azg, u_int32_t bgpAs, char *peerAddr, u_int32_t direction, char *aclInfo, u_int32_t vrId)
- **int smi_peer_addr_family_route_map_set_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t bgpAs, char *peerAddr, int afi, int safi, u_int32_t direction, char *aclInfo, u_int32_t vrId)
- **int smi_peer_addr_family_route_map_set_sdkapi** (struct smiclient_globals *azg, u_int32_t bgpAs, char *peerAddr, int afi, int safi, u_int32_t direction, char *aclInfo, u_int32_t vrId)
- **int smi_peer_route_map_unset_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t bgpAs, char *peerAddr, u_int32_t direction, u_int32_t vrId)

Sets to filter Route-Map segments to/from this neighbor.

- **int smi_peer_route_map_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t bgpAs, char *peerAddr, u_int32_t direction, u_int32_t vrId)
- **int smi_peer_addr_family_route_map_unset_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t bgpAs, char *peerAddr, int afi, int safi, u_int32_t direction, u_int32_t vrId)
- **int smi_peer_addr_family_route_map_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t bgpAs, char *peerAddr, int afi, int safi, u_int32_t direction, u_int32_t vrId)
- **int smi_peer_unsuppress_map_set_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t bgpAs, char *peerAddr, char *unSuppressAclInfo, u_int32_t vrId)

Sets the Route-Map to selectively unsuppress suppressed routes.

- **int smi_peer_unsuppress_map_set_sdkapi** (struct smiclient_globals *azg, u_int32_t bgpAs, char *peerAddr, char *unsuppressAclInfo, u_int32_t vrId)
- **int smi_peer_addr_family_unsuppress_map_set_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t bgpAs, char *peerAddr, int afi, int safi, char *unSuppressAclInfo, u_int32_t vrId)
- **int smi_peer_addr_family_unsuppress_map_set_sdkapi** (struct smiclient_globals *azg, u_int32_t bgpAs, char *peerAddr, int afi, int safi, char *unSuppressAclInfo, u_int32_t vrId)
- **int smi_peer_unsuppress_map_unset_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t bgpAs, char *peerAddr, u_int32_t vrId)

Unsets the Route-Map to selectively unsuppress suppressed routes.

- int **smi_peer_unsuppress_map_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t bgpAs, char *peerAddr, u_int32_t vrfId)
- int **smi_peer_addr_family_unsuppress_map_unset_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t bgpAs, char *peerAddr, int afi, int safi, u_int32_t vrfId)
- int **smi_peer_addr_family_unsuppress_map_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t bgpAs, char *peerAddr, int afi, int safi, u_int32_t vrfId)
- int **smi_peer_maximum_prefix_set_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t bgpAs, char *peerAddr, u_int32_t maxPrefixes, u_int32_t threshold, bool_t warning, u_int32_t vrfId)

Sets the maximum number of prefixes accepted from this peer.

- int **smi_peer_maximum_prefix_set_sdkapi** (struct smiclient_globals *azg, u_int32_t bgpAs, char *peerAddr, u_int32_t maxPrefixes, u_int32_t threshold, bool_t warning, u_int32_t vrfId)
- int **smi_peer_addr_family_maximum_prefix_set_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t bgpAs, char *peerAddr, int afi, int safi, u_int32_t maxPrefixes, u_int32_t threshold, bool_t warning, u_int32_t vrfId)
- int **smi_peer_addr_family_maximum_prefix_set_sdkapi** (struct smiclient_globals *azg, u_int32_t bgpAs, char *peerAddr, int afi, int safi, u_int32_t maxPrefixes, u_int32_t threshold, bool_t warning, u_int32_t vrfId)
- int **smi_peer_addr_family_maximum_prefix_unset_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t bgpAs, char *peerAddr, int afi, int safi, u_int32_t vrfId)
- int **smi_peer_addr_family_maximum_prefix_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t bgpAs, char *peerAddr, int afi, int safi, u_int32_t vrfId)
- int **smi_peer_password_set_validate** (struct smiclient_globals *azg, u_int32_t bgpAs, char *peerAddr, char *password, u_int32_t vrfId)

Sets Password to the neighbour.

- int **smi_peer_password_set** (struct smiclient_globals *azg, u_int32_t bgpAs, char *peerAddr, char *password, u_int32_t vrfId)
- int **smi_peer_password_set_sdkapi** (struct smiclient_globals *azg, u_int32_t bgpAs, char *peerAddr, int type, char *password, u_int32_t vrfId)
- int **smi_peer_password_unset_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t bgpAs, char *peerAddr, u_int32_t vrfId)

Unsets Password to the neighbour.

- int **smi_peer_password_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t as, char *peerAddr, u_int32_t vrfId)
- s_int32_t **smi_bgp_static_network_set_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t bgpAs, char *localAddr, u_int32_t backdoor, char *rmapName, u_int32_t vrfId)

Specifies a network to announce via BGP.

- `s_int32_t smi_bgp_static_network_set_sdkapi` (struct `smiclient_globals *azg`, `u_int32_t bgpAs`, `char *localAddr`, `u_int32_t backdoor`, `char *rmapName`, `u_int32_t vrId`)
- `s_int32_t smi_bgp_addr_family_static_network_set_sdkapi_validate` (struct `smiclient_globals *azg`, `u_int32_t bgpAs`, `char *localAddrAf`, `char *localAddrMaskAf`, `int afi`, `int safi`, `u_int32_t backdoorAf`, `char *networkRmapNameAf`, `u_int32_t vrId`)
- `s_int32_t smi_bgp_addr_family_static_network_set_sdkapi` (struct `smiclient_globals *azg`, `u_int32_t bgpAs`, `char *localAddrAf`, `char *localAddrMaskAf`, `int afi`, `int safi`, `u_int32_t backdoorAf`, `char *networkRmapNameAf`, `u_int32_t vrId`)
- `s_int32_t smi_bgp_static_network_unset_sdkapi_validate` (struct `smiclient_globals *azg`, `u_int32_t bgpAs`, `char *localAddr`, `u_int32_t vrId`)

Unspecifies a network to announce via BGP.

- `s_int32_t smi_bgp_static_network_unset_sdkapi` (struct `smiclient_globals *azg`, `u_int32_t bgpAs`, `char *localAddr`, `u_int32_t vrId`)
- `s_int32_t smi_bgp_addr_family_static_network_unset_sdkapi_validate` (struct `smiclient_globals *azg`, `u_int32_t bgpAs`, `char *localAddr`, `char *mask`, `int afi`, `int safi`, `u_int32_t vrId`)
- `s_int32_t smi_bgp_addr_family_static_network_unset_sdkapi` (struct `smiclient_globals *azg`, `u_int32_t bgpAs`, `char *localAddr`, `char *mask`, `int afi`, `int safi`, `u_int32_t vrId`)
- `int smi_bgp_debug_validate` (struct `smiclient_globals *azg`, `int debugFlag`, `u_int32_t vrId`)

Use this function to enable all BGP troubleshooting functions.

- `int smi_bgp_debug` (struct `smiclient_globals *azg`, `int debugFlag`, `u_int32_t vrId`)
- `int smi_bgp_no_debug_validate` (struct `smiclient_globals *azg`, `int debugFlag`, `u_int32_t vrId`)

Use this function to disable all BGP troubleshooting functions.

- `int smi_bgp_no_debug` (struct `smiclient_globals *azg`, `int debugFlag`, `u_int32_t vrId`)
- `int smi_bgp_disable_adj_out_set` (struct `smiclient_globals *azg`, `u_int32_t vrId`)

Sets the bgp disable adjacent.

- `int smi_bgp_disable_adj_out_unset` (struct `smiclient_globals *azg`, `u_int32_t vrId`)

unsets the bgp disable adjacent

- `int smi_bgp_maximum_paths_set` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `int bgpType`, `int multipathsNum`)

Sets bgp maximum paths.

- int [smi_bgp_maximum_paths_unset](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int bgpType)
Unsets bgp maximum paths.
- int [smi_bgp_aggregate nexthop_check_set](#) (struct smiclient_globals *azg, u_int32_t vrId)
Sets the check for bgp aggregate nexthop.
- int [smi_bgp_aggregate nexthop_check_unset](#) (struct smiclient_globals *azg, u_int32_t vrId)
Unsets the check for bgp aggregate nexthop.
- int [smi_bgp_fast_external_failover_set](#) (struct smiclient_globals *azg, u_int32_t vrId)
Sets the bgp fast external failover.
- int [smi_bgp_fast_external_failover_unset](#) (struct smiclient_globals *azg, u_int32_t vrId)
Unsets the bgp fast external failover.
- int [smi_bgp_rfc1771_path_select_set](#) (struct smiclient_globals *azg, u_int32_t vrId)
Sets the bgp rfc1771 path select.
- int [smi_bgp_rfc1771_path_select_unset](#) (struct smiclient_globals *azg, u_int32_t vrId)
Unsets the bgp rfc1771 path select.
- s_int32_t [smi_bgp_always_compare_med_set](#) (struct smiclient_globals *azg, u_int32_t vrId)
Sets the bgp always compare.
- s_int32_t [smi_bgp_always_compare_med_unset](#) (struct smiclient_globals *azg, u_int32_t vrId)
Unsets the bgp always compare.
- s_int32_t [smi_bgp_bestpath_aspath_ignore_set](#) (struct smiclient_globals *azg, u_int32_t vrId)
Sets the bgp bestpath as path ignore.
- s_int32_t [smi_bgp_bestpath_aspath_ignore_unset](#) (struct smiclient_globals *azg, u_int32_t vrId)
Unsets the bgp bestpath as path ignore.
- s_int32_t [smi_bgp_bestpath_compare_confed_aspath_set](#) (struct smiclient_globals *azg, u_int32_t vrId)
Sets the bgp bestpath compare confed as path.

- s_int32_t [smi_bgp_bestpath_compare_confed_aspath_unset](#) (struct smiclient_globals *azg, u_int32_t vrId)
Sets the bgp bestpath compare confed as path.
- s_int32_t [smi_bgp_bestpath_compare_router_id_set](#) (struct smiclient_globals *azg, u_int32_t vrId)
Sets the router-id for bgp bestpath compare.
- s_int32_t [smi_bgp_bestpath_compare_router_id_unset](#) (struct smiclient_globals *azg, u_int32_t vrId)
Unsets the router-id for bgp bestpath compare.
- s_int32_t [smi_bgp_bestpath_dont_compare_originator_id_set](#) (struct smiclient_globals *azg, u_int32_t vrId)
Sets the bestpath dont compare originator.
- s_int32_t [smi_bgp_bestpath_dont_compare_originator_id_unset](#) (struct smiclient_globals *azg, u_int32_t vrId)
Unsets the bestpath dont compare originator.
- s_int32_t [smi_bgp_bestpath_tie_break_on_age_set](#) (struct smiclient_globals *azg, u_int32_t vrId)
Sets the bestpath tie break on age.
- s_int32_t [smi_bgp_bestpath_tie_break_on_age_unset](#) (struct smiclient_globals *azg, u_int32_t vrId)
Unsets the bestpath tie break on age.
- s_int32_t [smi_bgp_default_ipv4_unicast_set](#) (struct smiclient_globals *azg, u_int32_t vrId)
Sets the bgp default ipv4 unicast.
- s_int32_t [smi_bgp_default_ipv4_unicast_unset](#) (struct smiclient_globals *azg, u_int32_t vrId)
Unsets the bgp default ipv4 unicast.
- s_int32_t [smi_bgp_deterministic_med_set](#) (struct smiclient_globals *azg, u_int32_t vrId)
Sets the bgp deterministic med.
- s_int32_t [smi_bgp_deterministic_med_unset](#) (struct smiclient_globals *azg, u_int32_t vrId)
Unsets the bgp deterministic med.
- s_int32_t [smi_bgp_enforce_first_as_set](#) (struct smiclient_globals *azg, u_int32_t vrId)

Sets the bgp enforce as first.

- s_int32_t [smi_bgp_enforce_first_as_unset](#) (struct smiclient_globals *azg, u_int32_t vrId)

Unsets the bgp enforce as first.

- s_int32_t [smi_peer_dynamic_capability_set](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr)

Sets the neighbor capability dynamic.

- s_int32_t [smi_peer_dynamic_capability_unset](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr)

Unsets the neighbor capability dynamic.

- s_int32_t [smi_neighbor_capability_route_refresh_set](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr)

Sets the neighbor capability route refresh.

- s_int32_t [smi_neighbor_capability_route_refresh_unset](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr)

Unsets the neighbor capability route refresh.

- s_int32_t [smi_neighbor_collide_established_set](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr)

Sets the neighbor collide established.

- s_int32_t [smi_neighbor_collide_established_unset](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr)

Unsets the neighbor collide established.

- s_int32_t [smi_bgp_vrf_neighbor_as_override_set](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr, int afi, int safi)

Sets vrf neighbor as override.

- s_int32_t [smi_bgp_vrf_neighbor_as_override_unset](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr, int afi, int safi)

Unsets vrf neighbor as override.

- s_int32_t [smi_neighbor_capability_grst_set](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr, int afi, int safi)

Sets the neighbor capability graceful.

- s_int32_t [smi_neighbor_capability_grst_unset](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr, int afi, int safi)

Unsets the neighbor capability graceful.

- s_int32_t [smi_neighbor_remove_private_as_set](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr, int afi, int safi)
Sets the neighbor remove private as.
- s_int32_t [smi_neighbor_remove_private_as_unset](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr, int afi, int safi)
Unsets the neighbor remove private as.
- s_int32_t [smi_neighbor_attr_unchanged_as_path_set](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr, int afi, int safi)
sets the neighbor transparent as
- s_int32_t [smi_neighbor_attr_unchanged_as_path_unset](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr, int afi, int safi)
unsets the neighbor transparent as
- s_int32_t [smi_neighbor_attr_unchanged_nexthop_set](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr, int afi, int safi)
sets the neighbor transparent nexthop
- s_int32_t [smi_neighbor_attr_unchanged_nexthop_unset](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr, int afi, int safi)
sets the neighbor transparent nexthop
- s_int32_t [smi_neighbor_attr_unchanged_med_set](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr, int afi, int safi)
sets the neighbor transparent med
- s_int32_t [smi_neighbor_attr_unchanged_med_unset](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr, int afi, int safi)
sets the neighbor transparent med
- s_int32_t [smi_peer_route_reflector_client_set](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr, int afi, int safi)
sets the neighbor route reflector client
- s_int32_t [smi_peer_route_reflector_client_unset](#) (struct smiclient_globals *azg, u_int32_t vrId, char *peerAddr, int afi, int safi)
unsets the neighbor route reflector client
- s_int32_t [smi_neighbor_route_server_client_set](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr, int afi, int safi)
sets the neighbor route server client

- s_int32_t [smi_neighbor_route_server_client_unset](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr, int afi, int safi)
unsets the neighbor route server client
- s_int32_t [smi_neighbor_enforce_multihop_set](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr)
sets the neighbor enforce multihop
- s_int32_t [smi_neighbor_enforce_multihop_unset](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr)
unsets the neighbor enforce multihop
- s_int32_t [smi_neighbor_override_capability_set](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr)
sets the neighbor override capability
- s_int32_t [smi_neighbor_override_capability_unset](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr)
unsets the neighbor override capability
- s_int32_t [smi_neighbor_strict_capability_set](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr)
set the neighbor strict capability
- s_int32_t [smi_neighbor_strict_capability_unset](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr)
unset the neighbor strict capability
- s_int32_t [smi_peer_disallow_hold_timer_set_sdkapi](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr)
set the neighbor disallow infinite timer
- s_int32_t [smi_peer_disallow_hold_timer_set_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vr_id, char *vrfName, char *peer_id)
- s_int32_t [smi_peer_disallow_hold_timer_unset_sdkapi](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr)
unset the neighbor disallow infinite timer
- s_int32_t [smi_peer_disallow_hold_timer_unset_sdkapi_validate](#) (struct smiclient_globals *azg, u_int32_t vr_id, char *vrfName, char *peer_id)
- s_int32_t [smi_peer_dont_capability_negotiate_set](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr)
sets the neighbor dont capability negotiate
- s_int32_t [smi_peer_dont_capability_negotiate_unset](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr)
unsets the neighbor dont capability negotiate

- s_int32_t [smi_bgp_multiple_instance_set](#) (struct smiclient_globals *azg, u_int32_t vrId)
sets the multiple instance
- s_int32_t [smi_bgp_multiple_instance_unset](#) (struct smiclient_globals *azg, u_int32_t vrId)
unsets the multiple instance
- s_int32_t [smi_neighbor_g_shut_time_set](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, u_int32_t shut_time)
sets neighbor graceful shut time
- s_int32_t [smi_neighbor_g_shut_time_unset](#) (struct smiclient_globals *azg, u_int32_t vrId, char *peerAddr)
unsets neighbor graceful shut time
- s_int32_t [smi_peer_transport_connection_passive_set](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr)
sets neighbor passive
- s_int32_t [smi_peer_transport_connection_passive_unset](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr)
unsets neighbor passive
- s_int32_t [smi_peer_shutdown_set](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr)
sets neighbor shutdown
- s_int32_t [smi_peer_shutdown_unset](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr)
unsets neighbor shutdown
- s_int32_t [smi_peer_soft_reconfiguration_inbound_set](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr, int afi, int safi)
sets the neighbor soft reconfiguration
- s_int32_t [smi_peer_soft_reconfiguration_inbound_unset](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr, int afi, int safi)
unsets the neighbor soft reconfiguration
- s_int32_t [smi_peer_next_hop_self_set](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr, int afi, int safi)
sets the neighbor nexthop self
- s_int32_t [smi_peer_next_hop_self_unset](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr, int afi, int safi)

unsets the neighbor nexthop self

- s_int32_t [smi_neighbor_capability_orf_prefix_set](#) (struct smiclient_globals *azg, char *peerAddr, char *orfPrefixOpt, int afi, int safi, u_int32_t vrId, u_int32_t bgpAs)

sets neighbor capability orf prefix

- s_int32_t [smi_neighbor_capability_orf_prefix_unset](#) (struct smiclient_globals *azg, char *peerAddr, char *orfPrefixOpt, int afi, int safi, u_int32_t vrId, u_int32_t bgpAs)

unsets neighbor capability orf prefix

- s_int32_t [smi_bgp_bestpath_med_set](#) (struct smiclient_globals *azg, u_int32_t vrId, char *medType)

sets bestpath med

- s_int32_t [smi_bgp_bestpath_med_unset](#) (struct smiclient_globals *azg, u_int32_t vrId, char *medType)

unsets bestpath med

- int [smi_bgp_disable_adj_out_set_validate](#) (struct smiclient_globals *azg, u_int32_t vrId)

Sets the bgp disable adjacent.

- int [smi_bgp_disable_adj_out_unset_validate](#) (struct smiclient_globals *azg, u_int32_t vrId)

unsets the bgp disable adjacent

- int [smi_bgp_maximum_paths_set_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int bgpType, int multipathsNum)

Sets bgp maximum paths.

- int [smi_bgp_maximum_paths_unset_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int bgp_type)

Unsets bgp maximum paths.

- int [smi_bgp_aggregate_nexthop_check_set_validate](#) (struct smiclient_globals *azg, u_int32_t vrId)

Sets the check for bgp aggregate_nexthop.

- int [smi_bgp_aggregate_nexthop_check_unset_validate](#) (struct smiclient_globals *azg, u_int32_t vrId)

Unsets the check for bgp aggregate_nexthop.

- int [smi_bgp_fast_external_failover_set_validate](#) (struct smiclient_globals *azg, u_int32_t vrId)

Sets the bgp fast external failover.

- `int smi_bgp_fast_external_failover_unset_validate` (struct smiclient_globals *azg, u_int32_t vrId)
Unsets the bgp fast external failover.
- `int smi_bgp_rfc1771_path_select_set_validate` (struct smiclient_globals *azg, u_int32_t vrId)
Sets the bgp rfc1771 path select.
- `int smi_bgp_rfc1771_path_select_unset_validate` (struct smiclient_globals *azg, u_int32_t vrId)
Unsets the bgp rfc1771 path select.
- `s_int32_t smi_bgp_always_compare_med_set_validate` (struct smiclient_globals *azg, u_int32_t vrId)
Sets the bgp always compare.
- `s_int32_t smi_bgp_always_compare_med_unset_validate` (struct smiclient_globals *azg, u_int32_t vrId)
Unsets the bgp always compare.
- `s_int32_t smi_bgp_bestpath_aspath_ignore_set_validate` (struct smiclient_globals *azg, u_int32_t vrId)
Sets the bgp bestpath as path ignore.
- `s_int32_t smi_bgp_bestpath_aspath_ignore_unset_validate` (struct smiclient_globals *azg, u_int32_t vrId)
Unsets the bgp bestpath as path ignore.
- `s_int32_t smi_bgp_bestpath_compare_confed_aspath_set_validate` (struct smiclient_globals *azg, u_int32_t vrId)
Sets the bgp bestpath compare confed as path.
- `s_int32_t smi_bgp_bestpath_compare_confed_aspath_unset_validate` (struct smiclient_globals *azg, u_int32_t vrId)
Sets the bgp bestpath compare confed as path.
- `s_int32_t smi_bgp_bestpath_compare_router_id_set_validate` (struct smiclient_globals *azg, u_int32_t vrId)
Sets the router-id for bgp bestpath compare.
- `s_int32_t smi_bgp_bestpath_compare_router_id_unset_validate` (struct smiclient_globals *azg, u_int32_t vrId)
Unsets the router-id for bgp bestpath compare.
- `s_int32_t smi_bgp_bestpath_dont_compare_originator_id_set_validate` (struct smiclient_globals *azg, u_int32_t vrId)

Sets the bestpath dont compare originator.

- s_int32_t [smi_bgp_bestpath_dont_compare_originator_id_unset_validate](#) (struct smiclient_globals *azg, u_int32_t vrId)

Unsets the bestpath dont compare originator.

- s_int32_t [smi_bgp_bestpath_tie_break_on_age_set_validate](#) (struct smiclient_globals *azg, u_int32_t vrId)

Sets the bestpath tie break on age.

- s_int32_t [smi_bgp_bestpath_tie_break_on_age_unset_validate](#) (struct smiclient_globals *azg, u_int32_t vrId)

Unsets the bestpath tie break on age.

- s_int32_t [smi_bgp_default_ipv4_unicast_set_validate](#) (struct smiclient_globals *azg, u_int32_t vrId)

Sets the bgp default ipv4 unicast.

- s_int32_t [smi_bgp_default_ipv4_unicast_unset_validate](#) (struct smiclient_globals *azg, u_int32_t vrId)

Unsets the bgp default ipv4 unicast.

- s_int32_t [smi_bgp_deterministic_med_set_validate](#) (struct smiclient_globals *azg, u_int32_t vrId)

Sets the bgp deterministic med.

- s_int32_t [smi_bgp_deterministic_med_unset_validate](#) (struct smiclient_globals *azg, u_int32_t vrId)

Unsets the bgp deterministic med.

- s_int32_t [smi_bgp_enforce_first_as_set_validate](#) (struct smiclient_globals *azg, u_int32_t vrId)

Sets the bgp enforce as first.

- s_int32_t [smi_bgp_enforce_first_as_unset_validate](#) (struct smiclient_globals *azg, u_int32_t vrId)

Unsets the bgp enforce as first.

- s_int32_t [smi_bgp_grst_set_validate](#) (struct smiclient_globals *azg, u_int32_t vrId)

Sets the bgp graceful.

- s_int32_t [smi_bgp_grst_unset_validate](#) (struct smiclient_globals *azg, u_int32_t vrId)

Unsets the bgp graceful.

- s_int32_t [smi_peer_dynamic_capability_set_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, char *peerAddr)
Sets the neighbor capability dynamic.
- s_int32_t [smi_peer_dynamic_capability_unset_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, char *peerAddr)
Unsets the neighbor capability dynamic.
- s_int32_t [smi_neighbor_capability_route_refresh_set_validate](#) (struct smiclient_globals *azg, char *peerAddr, u_int32_t vrId)
Sets the neighbor capability route refresh.
- s_int32_t [smi_neighbor_capability_route_refresh_unset_validate](#) (struct smiclient_globals *azg, char *peerAddr, u_int32_t vrId)
Unsets the neighbor capability route refresh.
- s_int32_t [smi_neighbor_collide_established_set_validate](#) (struct smiclient_globals *azg, char *peerAddr, u_int32_t vrId)
Sets the neighbor collide established.
- s_int32_t [smi_neighbor_collide_established_unset_validate](#) (struct smiclient_globals *azg, char *peerAddr, u_int32_t vrId)
Unsets the neighbor collide established.
- s_int32_t [smi_bgp_vrf_neighbor_as_override_set_validate](#) (struct smiclient_globals *azg, char *peerAddr, int afi, int safi, u_int32_t vrId)
Sets vrf neighbor as override.
- s_int32_t [smi_bgp_vrf_neighbor_as_override_unset_validate](#) (struct smiclient_globals *azg, char *peerAddr, int afi, int safi, u_int32_t vrId)
Unsets vrf neighbor as override.
- s_int32_t [smi_neighbor_capability_grst_set_validate](#) (struct smiclient_globals *azg, char *peerAddr, int afi, int safi, u_int32_t vrId)
Sets the neighbor capability graceful.
- s_int32_t [smi_neighbor_capability_grst_unset_validate](#) (struct smiclient_globals *azg, char *peerAddr, int afi, int safi, u_int32_t vrId)
Unsets the neighbor capability graceful.
- s_int32_t [smi_neighbor_remove_private_as_set_validate](#) (struct smiclient_globals *azg, char *peerAddr, int afi, int safi, u_int32_t vrId)
Sets the neighbor remove private as.
- s_int32_t [smi_neighbor_remove_private_as_unset_validate](#) (struct smiclient_globals *azg, char *peerAddr, int afi, int safi, u_int32_t vrId)
Unsets the neighbor remove private as.

- `s_int32_t smi_neighbor_filter_list_set_validate` (struct smiclient_globals *azg, char *peerAddr, int afi, int safi, u_int32_t direction, char *aclInfo, u_int32_t vrId)
Sets the neighbor filter list.
- `s_int32_t smi_neighbor_filter_list_unset_validate` (struct smiclient_globals *azg, char *peerAddr, int afi, int safi, u_int32_t direction, u_int32_t vrId)
unsets the neighbor filter list
- `s_int32_t smi_filter_list_set_validate` (struct smiclient_globals *azg, char *peerAddr, int afi, int safi, u_int32_t direction, char *aclInfo, u_int32_t vrId)
Sets the filter list.
- `s_int32_t smi_filter_list_unset_validate` (struct smiclient_globals *azg, char *peerAddr, int afi, int safi, u_int32_t direction, u_int32_t vrId)
unsets the filter list
- `s_int32_t smi_neighbor_local_as_set_validate` (struct smiclient_globals *azg, char *peerAddr, u_int32_t vrId)
sets the neighbor local as
- `s_int32_t smi_neighbor_local_as_unset_validate` (struct smiclient_globals *azg, char *peerAddr, u_int32_t vrId)
unsets the neighbor local as
- `s_int32_t smi_neighbor_transparent_as_set_validate` (struct smiclient_globals *azg, char *peerAddr, int afi, int safi, u_int32_t vrId)
sets the neighbor transparent as
- `s_int32_t smi_neighbor_transparent_nexthop_set_validate` (struct smiclient_globals *azg, char *peerAddr, int afi, int safi, u_int32_t vrId)
sets the neighbor transparent nexthop
- `s_int32_t smi_neighbor_route_reflector_client_set_validate` (struct smiclient_globals *azg, char *peerAddr, int afi, int safi, u_int32_t vrId)
sets the neighbor route reflector client
- `s_int32_t smi_neighbor_route_reflector_client_unset_validate` (struct smiclient_globals *azg, char *peerAddr, int afi, int safi, u_int32_t vrId)
unsets the neighbor route reflector client
- `s_int32_t smi_neighbor_route_server_client_set_validate` (struct smiclient_globals *azg, char *peerAddr, int afi, int safi, u_int32_t vrId)
sets the neighbor route server client

- s_int32_t [smi_neighbor_route_server_client_unset_validate](#) (struct smiclient_globals *azg, char *peerAddr, int afi, int safi, u_int32_t vrId)
unsets the neighbor route server client
- s_int32_t [smi_neighbor_enforce_multihop_set_validate](#) (struct smiclient_globals *azg, char *peerAddr, u_int32_t vrId)
sets the neighbor enforce multihop
- s_int32_t [smi_neighbor_enforce_multihop_unset_validate](#) (struct smiclient_globals *azg, char *peerAddr, u_int32_t vrId)
unsets the neighbor enforce multihop
- s_int32_t [smi_neighbor_override_capability_set_validate](#) (struct smiclient_globals *azg, char *peerAddr, u_int32_t vrId)
sets the neighbor override capability
- s_int32_t [smi_neighbor_override_capability_unset_validate](#) (struct smiclient_globals *azg, char *peerAddr, u_int32_t vrId)
unsets the neighbor override capability
- s_int32_t [smi_neighbor_strict_capability_set_validate](#) (struct smiclient_globals *azg, char *peerAddr, u_int32_t vrId)
set the neighbor strict capability
- s_int32_t [smi_neighbor_strict_capability_unset_validate](#) (struct smiclient_globals *azg, char *peerAddr, u_int32_t vrId)
unset the neighbor strict capability
- s_int32_t [smi_neighbor_connection_retry_time_unset_validate](#) (struct smiclient_globals *azg, char *peerAddr, u_int32_t vrId)
unset the neighbor connection retry time
- s_int32_t [smi_neighbor_disallow_infinite_timer_set_validate](#) (struct smiclient_globals *azg, char *peerAddr, u_int32_t vrId)
set the neighbor disallow infinite timer
- s_int32_t [smi_neighbor_disallow_infinite_timer_unset_validate](#) (struct smiclient_globals *azg, char *peerAddr, u_int32_t vrId)
unset the neighbor disallow infinite timer
- s_int32_t [smi_peer_dont_capability_negotiate_set_validate](#) (struct smiclient_globals *azg, char *peerAddr, u_int32_t vrId)
sets the neighbor dont capability negotiate
- s_int32_t [smi_neighbor_dont_capability_negotiate_unset_validate](#) (struct smiclient_globals *azg, char *peerAddr, u_int32_t vrId)
unsets the neighbor dont capability negotiate

- s_int32_t [smi_bgp_multiple_instance_set_validate](#) (struct smiclient_globals *azg, u_int32_t vrId)
sets the multiple instance
- s_int32_t [smi_bgp_multiple_instance_unset_validate](#) (struct smiclient_globals *azg, u_int32_t vrId)
unsets the multiple instance
- s_int32_t [smi_bgp_graceful_restart_set_wrap_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, bool_t grstSet)
- s_int32_t [smi_bgp_graceful_restart_set_wrap](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, bool_t grstSet)
- s_int32_t [smi_bgp_graceful_restart_set](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs)
- s_int32_t [smi_bgp_graceful_restart_unset](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs)
- s_int32_t [smi_bgp_grst_restart_time_set_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, u_int32_t restartTime)
sets graceful restart time
- s_int32_t [smi_bgp_grst_restart_time_unset_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, u_int32_t restartTime)
unsets graceful restart time
- s_int32_t [smi_bgp_grst_stalepath_time_set_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, u_int32_t stalepathTime)
sets graceful stalepath time
- s_int32_t [smi_bgp_grst_stalepath_time_unset_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, u_int32_t stalepathTime)
unsets graceful stalepath time
- s_int32_t [smi_bgp_update_delay_val_set_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, u_int32_t deferTime)
sets update delay value
- s_int32_t [smi_bgp_update_delay_val_unset_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs)
unsets update delay vlaue
- s_int32_t [smi_neighbor_g_shut_time_set_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, u_int32_t shut_time)
sets neighbor graceful shut time
- s_int32_t [smi_neighbor_g_shut_time_unset_validate](#) (struct smiclient_globals *azg, char *peerAddr, u_int32_t shutTime, u_int32_t vrId)

unsets neighbor graceful shut time

- s_int32_t [smi_peer_transport_connection_passive_set_validate](#) (struct smiclient_globals *azg, char *peerAddr, u_int32_t vrId)

sets neighbor passive

- s_int32_t [smi_peer_transport_connection_passive_unset_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, char *peerAddr)

unsets neighbor passive

- s_int32_t [smi_peer_shutdown_set_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrf_name, char *peerAddr)

sets neighbor shutdown

- s_int32_t [smi_peer_shutdown_unset_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrf_name, char *peerAddr)

unsets neighbor shutdown

- s_int32_t [smi_peer_soft_reconfiguration_inbound_set_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr, int afi, int safi)

sets the neighbor soft reconfiguration

- s_int32_t [smi_peer_soft_reconfiguration_inbound_unset_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr, int afi, int safi)

unsets the neighbor soft reconfiguration

- s_int32_t [smi_peer_next_hop_self_set_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr, int afi, int safi)

sets the neighbor nexthop self

- s_int32_t [smi_peer_next_hop_self_unset_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr, int afi, int safi)

unsets the neighbor nexthop self

- s_int32_t [smi_transport_connection_passive_set_validate](#) (struct smiclient_globals *azg, char *peerAddr)

sets the transport connection passive

- s_int32_t [smi_transport_connection_passive_unset_validate](#) (struct smiclient_globals *azg, char *peerAddr)

unsets the transport connection passive

- s_int32_t [smi_neighbor_capability_orf_prefix_set_validate](#) (struct smiclient_globals *azg, char *peerAddr, char *nbrOrfPrefixOpt, int afi, int safi)

sets neighbor capability orf prefix

- s_int32_t [smi_neighbor_capability_orf_prefix_unset_validate](#) (struct smiclient_globals *azg, char *peerAddr, char *nbrOrfPrefixOpt, int afi, int safi)
unsets neighbor capability orf prefix
- s_int32_t [smi_bgp_bestpath_med_set_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, char *medType)
sets bestpath med
- s_int32_t [smi_bgp_bestpath_med_unset_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, char *medType)
unsets bestpath med
- int [smi_nbr_unset_shut_tm_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, u_int32_t shutTime)
- int [smi_nbr_set_shut_tm_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, u_int32_t shutTime)
- int [smi_nbr_set_shut_tm](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, u_int32_t shutTime)
- int [smi_nbr_unset_shut_tm](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, u_int32_t shutTime)
- int [smi_nbr_graceful_shut_set](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int afi, int safi)
- int [smi_nbr_graceful_shut_set_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int afi, int safi)
- int [smi_nbr_graceful_shut_unset](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr)
- int [smi_nbr_graceful_shut_set_wrap_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int afi, int safi, bool_t setGracefulShut)
- int [smi_nbr_graceful_shut_set_wrap](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int afi, int safi, bool_t setGracefulShut)
- int [smi_nbr_graceful_shut_unset_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr)
- s_int32_t [smi_bgp_dampening_set](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t as, int afi, int safi)
- s_int32_t [smi_bgp_dampening_set_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t as, int afi, int safi)
- s_int32_t [smi_bgp_dampening_half_life_set](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t as, int afi, int safi, u_int32_t halfLife)
- s_int32_t [smi_bgp_dampening_half_life_set_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t as, int afi, int safi, u_int32_t halfLife)
- s_int32_t [smi_bgp_dampening_half_life_reuse_suppress_maxsuppress_set_validate](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t as, int afi, int safi, u_int32_t halfLife, u_int32_t reusePenalty, u_int32_t suppressPenalty, u_int32_t maxSuppress)

- `s_int32_t smi_bgp_dampening_half_life_reuse_supress_maxsuppress_set` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t as`, `int afi`, `int safi`, `u_int32_t halfLife`, `u_int32_t reusePenalty`, `u_int32_t suppressPenalty`, `u_int32_t maxSuppress`)
- `s_int32_t smi_bgp_dampening_half_life_reuse_supress_maxsuppress_unreachability_set_validate` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t as`, `int afi`, `int safi`, `u_int32_t halfLife`, `u_int32_t reusePenalty`, `u_int32_t suppressPenalty`, `u_int32_t maxSuppress`, `u_int32_t unreachHalfLife`)
- `s_int32_t smi_bgp_dampening_half_life_reuse_supress_maxsuppress_unreachability_set` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t as`, `int afi`, `int safi`, `u_int32_t halfLife`, `u_int32_t reusePenalty`, `u_int32_t suppressPenalty`, `u_int32_t maxSuppress`, `u_int32_t unreachHalfLife`)
- `s_int32_t smi_bgp_dampening_routemap_name_set` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t as`, `int afi`, `int safi`, `char *rmapName`)
- `s_int32_t smi_bgp_dampening_routemap_name_set_validate` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t as`, `int afi`, `int safi`, `char *rmapName`)
- `s_int32_t smi_bgp_dampening_unset` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t as`, `int afi`, `int safi`)
- `s_int32_t smi_bgp_dampening_unset_validate` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t as`, `int afi`, `int safi`)
- `s_int32_t smi_bgp_dampening_half_life_unset` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t as`, `int afi`, `int safi`, `u_int32_t halfLife`)
- `s_int32_t smi_bgp_dampening_half_life_unset_validate` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t as`, `int afi`, `int safi`, `u_int32_t halfLife`)
- `s_int32_t smi_bgp_dampening_half_life_reuse_supress_maxsuppress_unset_validate` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t as`, `int afi`, `int safi`, `u_int32_t halfLife`, `u_int32_t reusePenalty`, `u_int32_t suppressPenalty`, `u_int32_t maxSuppress`)
- `s_int32_t smi_bgp_dampening_half_life_reuse_supress_maxsuppress_unset` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t as`, `int afi`, `int safi`, `u_int32_t halfLife`, `u_int32_t reusePenalty`, `u_int32_t suppressPenalty`, `u_int32_t maxSuppress`)
- `s_int32_t smi_bgp_dampening_half_life_reuse_supress_maxsuppress_unreachability_unset_validate` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t as`, `int afi`, `int safi`, `u_int32_t halfLife`, `u_int32_t reusePenalty`, `u_int32_t suppressPenalty`, `u_int32_t maxSuppress`, `u_int32_t unreachHalfLife`)
- `s_int32_t smi_bgp_dampening_half_life_reuse_supress_maxsuppress_unreachability_unset` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t as`, `int afi`, `int safi`, `u_int32_t halfLife`, `u_int32_t reusePenalty`, `u_int32_t suppressPenalty`, `u_int32_t maxSuppress`, `u_int32_t unreachHalfLife`)
- `s_int32_t smi_bgp_dampening_routemap_name_unset_validate` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t as`, `int afi`, `int safi`, `char *rmapName`)

- `s_int32_t smi_bgp_dampening_routemap_name_unset` (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t as, int afi, int safi, char *rmapName)
- `s_int32_t smi_bgp_network_set_wrap` (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *localAddr, u_int32_t backdoor)
- `s_int32_t smi_bgp_network_set_wrap_validate` (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *localAddr, u_int32_t backdoor)
- `s_int32_t smi_bgp_network_set_cmd` (struct smiclient_globals *azg, u_int32_t vrId, char *ip_str, int afi, int safi, u_int32_t backdoor)
- `s_int32_t smi_bgp_network_set_cmd_validate` (struct smiclient_globals *azg, u_int32_t vrId, char *localAddr, int afi, int safi, u_int32_t backdoor)
- `s_int32_t smi_bgp_api_distance_config_set_validate` (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int afi, int safi, u_int8_t distanceEbgp, u_int8_t distanceIbgp, u_int8_t distanceLocal)
- `s_int32_t smi_bgp_api_distance_config_set` (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int afi, int safi, u_int8_t distanceEbgp, u_int8_t distanceIbgp, u_int8_t distanceLocal)
- `s_int32_t smi_bgp_api_distance_config_unset_validate` (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int afi, int safi, u_int8_t distanceEbgp, u_int8_t distanceIbgp, u_int8_t distanceLocal)
- `s_int32_t smi_bgp_api_distance_config_unset` (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int afi, int safi, u_int8_t distanceEbgp, u_int8_t distanceIbgp, u_int8_t distanceLocal)
- `int smi_bgp_af_redistribute_set_wrap_validate` (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, int afi, int safi, int redist_type, char *vrfName)
- `int smi_bgp_af_redistribute_set_wrap` (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, int afi, int safi, int redist_type, char *vrfName)
- `int smi_bgp_af_redistribute_unset_wrap_validate` (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, int afi, int safi, int redist_type, char *vrfName)
- `int smi_bgp_af_redistribute_unset_wrap` (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, int afi, int safi, int redist_type, char *vrfName)
- `s_int32_t smi_bgp_redistribute_set_sdkapi_validate` (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int redistType)
- `s_int32_t smi_bgp_redistribute_set_sdkapi` (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int redistType)
- `s_int32_t smi_bgp_redistribute_routemap_set_sdkapi_validate` (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int redistType, char *rmapName)
- `s_int32_t smi_bgp_redistribute_routemap_set_sdkapi` (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int redistType, char *rmapName)
- `s_int32_t smi_bgp_redistribute_unset_sdkapi` (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int redistType)
- `s_int32_t smi_bgp_redistribute_unset_sdkapi_validate` (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int redistType)

- `s_int32_t smi_bgp_redistribute_routemap_unset_sdkapi_validate` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `int redistType`)
 - `s_int32_t smi_bgp_redistribute_routemap_unset_sdkapi` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `int redistType`)
 - `s_int32_t smi_bgp_addr_family_redistribute_set_sdkapi_validate` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `int afi`, `int safi`, `int redistType`)
 - `s_int32_t smi_bgp_addr_family_redistribute_set_sdkapi` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `int afi`, `int safi`, `int redistType`)
 - `s_int32_t smi_bgp_addr_family_redistribute_routemap_set_sdkapi_validate` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `int afi`, `int safi`, `int redistType`, `char *rmapName`)
 - `s_int32_t smi_bgp_addr_family_redistribute_routemap_set_sdkapi` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `int afi`, `int safi`, `int redistType`, `char *rmapName`)
 - `s_int32_t smi_bgp_addr_family_redistribute_unset_sdkapi` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `int afi`, `int safi`, `int redistType`)
 - `s_int32_t smi_bgp_addr_family_redistribute_unset_sdkapi_validate` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `int afi`, `int safi`, `int redistType`)
 - `s_int32_t smi_bgp_addr_family_redistribute_routemap_unset_sdkapi_validate` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `int afi`, `int safi`, `int redistType`)
 - `s_int32_t smi_bgp_addr_family_redistribute_routemap_unset_sdkapi` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `int afi`, `int safi`, `int redistType`)
 - `int smi_peer_update_routing_source_unset_sdkapi_validate` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `char *peerAddr`)
 - `int smi_peer_update_routing_source_unset_sdkapi` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `char *peerAddr`)
 - `s_int32_t smi_bgp_api_soo_set_validate` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `char *peerAddr`, `int afi`, `int safi`, `char *siteOriginId`)
 - `s_int32_t smi_bgp_api_soo_unset_validate` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `char *peerAddr`, `int afi`, `int safi`)
 - `s_int32_t smi_bgp_api_soo_unset` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `char *peerAddr`, `int afi`, `int safi`)
 - `s_int32_t smi_bgp_api_soo_set` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `char *peerAddr`, `int afi`, `int safi`, `char *siteOriginId`)
 - `int smi_bgp_check_instance` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t *as`, `char *bgpName`)
- This API checks if the BGP instance is enabled.*
- `int smi_bgp_confederation_peer_check_sdkapi` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `int as`)

This API checks if the BGP peer confederation information is configured.

- int [smi_bgp_get_peer_timers](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int16_t *keepAlive, u_int16_t *holdTime)

This API get the configured BGP keepalive and holdtime timer.

- int [smi_peer_get_ebgp_multihop](#) (struct smiclient_globals *azg, u_int32_t vrId, char *peerAddr, u_int8_t *ttl)

This API get the configured BGP multihop.

- int [smi_peer_get_description](#) (struct smiclient_globals *azg, u_int32_t vrId, char *peerAddr, char *peerDesc)

This API get the configured BGP peer description.

- int [smi_peer_get_update_source_info](#) (struct smiclient_globals *azg, u_int32_t vrId, char *peerAddr, char *updateIf, char *updateSource)

This API get the configured BGP peer routing update source information.

- int [smi_peer_get_timers](#) (struct smiclient_globals *azg, u_int32_t vrId, char *peerAddr, u_int32_t *keepAlive, u_int32_t *holdTime)

This API get the configured BGP peer keepalive and holdtime.

- int [smi_peer_get_timers_connect](#) (struct smiclient_globals *azg, u_int32_t vrId, char *peerAddr, u_int32_t *peerConnectInterval)

This API get the configured BGP peer connect timer.

- int [smi_peer_get_asorig_interval](#) (struct smiclient_globals *azg, u_int32_t vrId, char *peerAddr, u_int32_t *peerAsorigInterval)

This API get the configured BGP peer asorig interval.

- int [smi_peer_get_advertise_interval](#) (struct smiclient_globals *azg, u_int32_t vrId, char *peerAddr, u_int32_t *ra_interval)

This API get the configured BGP peer advertise interval.

- int [smi_peer_get_interface](#) (struct smiclient_globals *azg, u_int32_t vrId, char *peerAddr, char *ifName)

This API get the configured interface for BGP peer.

- int [smi_peer_get_allowas_in](#) (struct smiclient_globals *azg, u_int32_t vrId, char *peerAddr, int af_id, int subaf_id, u_int32_t *allowAsNum)

This API get the configured BGP peer allow-as.

- int [smi_peer_af_flag_config_check](#) (struct smiclient_globals *azg, u_int32_t vrId, char *peerAddr, int af_id, int subaf_id, u_int32_t peerAfFlag)

This API checks if the BGP peer address-family flag is configured.

- int [smi_bgp_option_check_sdkapi](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t optFlag)

This API checks if the BGP flag is configured.

- int [smi_peer_flag_config_check](#) (struct smiclient_globals *azg, u_int32_t vrId, char *peerAddr, u_int32_t peerFlag)

This API checks if the BGP peer flag is configured.

- int [smi_bgp_af_config_check_sdkapi](#) (struct smiclient_globals *azg, u_int32_t vrId, int af_id, int subaf_id, u_int32_t afFlag)

This API checks if the BGP address-family flag is configured.

- int [smi_bgp_get_grst_restart_time](#) (struct smiclient_globals *azg, u_int32_t vrId, char *peerAddr, u_int32_t *restartTime)

This API gets the configured BGP graceful restart time.

- int [smi_bgp_get_grst_stalepath_time](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t *stalepathTime)

This API gets the configured BGP graceful restart time.

- int [smi_bgp_get_update_delay_val](#) (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t *deferTime)

This API gets the configured BGP graceful restart time.

- int [smi_show_bgp_dampening_parameters](#) (struct smiclient_globals *azg, int afi, int safi, struct rfdConfigData *rfdOutInfo, struct list *bgpDampeningParaOutList, u_int32_t(*callbackFunc)(struct list *bgpDampeningParaOutList))

show bgp info about dampening,

- int [smi_show_ip_bgp_ipv6_dampening_parameters](#) (struct smiclient_globals *azg, char *vrfName, int afi, int safi, struct rfdConfigData *rfdOutInfo, struct list *bgpDampeningParaOutList, u_int32_t(*callbackFunc)(struct list *bgpDampeningParaOutList))

show bgp info about dampening parameters info

- int [smi_show_ip_bgp_summary](#) (struct smiclient_globals *azg, char *vrfName, int afi, int safi, struct list *bgpSummaryList, u_int32_t(*callbackFunc)(struct list *bgpSummaryList))

show bgp summary info of neighbor status

- int [smi_show_bgp_summary](#) (struct smiclient_globals *azg, char *vrfName, u_int8_t safi, struct list *bgpSummaryList, u_int32_t(*callbackFunc)(struct list *bgpSummaryList))

show bgp summary info of neighbor status for IPv4 environment

- int [smi_show_ip_bgp_longer_prefixes](#) (struct smiclient_globals *azg, char *prefix, char *vrfName, u_int8_t afi, struct list *bgpPrefixList, u_int32_t(*callbackFunc)(struct list *bgpPrefixList))
show bgp network information from the mask information
- int [smi_show_ip_bgp_received_paths](#) (struct smiclient_globals *azg, char *vrfName, struct list *bgpReceivedPathList, u_int32_t(*callbackFunc)(struct list *bgpReceivedPathList))
show bgp neighbor information for the received paths
- int [smi_show_ip_bgp_dampening_dampend_paths](#) (struct smiclient_globals *azg, char *vrfName, struct list *bgpDampendPathList, u_int32_t(*callbackFunc)(struct list *bgpDampendPathList))
show bgp information about dampening dampened parameters info
- int [smi_show_ip_bgp_dampening_flap_statistics](#) (struct smiclient_globals *azg, char *vrfName, int vrfOption, struct list *bgpDampingFlapList, u_int32_t(*callbackFunc)(struct list *bgpDampingFlapList))
show bgp information about dampening flapping statistics
- int [_merge_bgp_dampening_param_list](#) (struct list *listDest, struct list *listSrc)
- int [_merge_bgp_summary_info_list](#) (struct list *listDest, struct list *listSrc)
- int [_merge_bgp_longer_prefix_list](#) (struct list *listDest, struct list *listSrc)
- int [_merge_bgp_received_paths_list](#) (struct list *listDest, struct list *listSrc)
- int [_merge_bgp_dampening_dampened_paths_list](#) (struct list *listDest, struct list *listSrc)
- int [_merge_bgp_dampening_flap_stat_list](#) (struct list *listDest, struct list *listSrc)
- int [smi_show_ip_protocol_all](#) (struct smiclient_globals *azg, char *af, struct list *showList, int(*callback)(struct list *showlist))
show bgp information about protocols
- int [merge_list_session_info](#) (struct list *listDst, struct list *listSrc)
- int [merge_list_info](#) (struct list *listDst, struct list *listSrc)
- int [smi_show_ip_bgp](#) (struct smiclient_globals *azg, char *vrfName, char *af, char *saf, struct list *showList, int(*callback)(struct list *showlist))
show bgp info about neighbor instance of the given instance
- int [smi_show_ip_bgp_cidr_only](#) (struct smiclient_globals *azg, char *af, char *saf, struct list *showList, int(*callback)(struct list *showlist))
show bgp routes info with non-natural network mask
- int [smi_show_ip_bgp_community](#) (struct smiclient_globals *azg, char *vrfName, char *af, char *saf, struct list *showList, int(*callback)(struct list *showlist))
show bgp info about neighbor instance of the given instance

- `int smi_show_ip_bgp_filter_list_exact_match` (struct smiclient_globals *azg, char *filterList_name, char *vrfName, char *af, char *saf, int type, struct list *showList, int(*callback)(struct list *showlist))
show bgp multicast/unicast filter list for IPv4/IPv6 environment
- `int smi_show_bgp_inconsistent_as` (struct smiclient_globals *azg, char *af, char *saf, struct list *showList, int(*callback)(struct list *showlist))
show bgp multicast/unicast route with inconsistent AS path for IPv4/IPv6 environment
- `int smi_show_bgp_sessions` (struct smiclient_globals *azg, char *name, struct list *showList, int(*callback)(struct list *showlist))
show bgp established session info
- `int smi_show_ip_bgp_prefix_list_exact_match` (struct smiclient_globals *azg, char *prefixList_name, char *vrfName, char *afi, char *saf, int type, struct list *showList, int(*callback)(struct list *showlist))
show bgp routes matching the prefix list name otherwise show all the prefix list
- `int smi_show_ip_bgp_word_neighbors` (struct smiclient_globals *azg, char *name, char *ipAddr, struct list *bgpShowAllList, int(*callback)(struct list *showlist))
show neighbors matching the neighbor name or else show for aall neighbors
- `int smi_show_ip_bgp_word_peer_neighbors` (struct smiclient_globals *azg, char *name, char *ipAddr, struct list *bgpShowPeerList, int(*callback)(struct list *showlist))
show neighbors matching the neighbor ipaddress or else show for all neighbors
- `int smi_show_ip_bgp_neighbors_HKC` (struct smiclient_globals *azg, char *name, char *ipAddr, char *type, struct list *bgpShowHKCList, int(*callback)(struct list *showlist))
show hold-time | keepalive-interval| connection-retry for all neighbors
- `int smi_show_bgp_neighbor_advertised_routes` (struct smiclient_globals *azg, char *ipAddr, char *name, char *af, char *saf, struct list *bgpShowList, int(*callback)(struct list *showlist))
show advertised routes for all neighbors
- `int smi_show_bgp_neighbor_recieved_routes` (struct smiclient_globals *azg, char *ipAddr, char *name, char *af, char *saf, struct list *bgpShowList, int(*callback)(struct list *showlist))
show recieved routes for all neighbors
- `int smi_show_bgp_neighbors_rcv_prefix_filter` (struct smiclient_globals *azg, char *ipAddr, char *af, char *saf, struct list *bgpShowList, int(*callback)(struct list *showlist))

neighbors matching the given prefix filter

- int [smi_show_bgp_V6_neighbors_rcv_prefix_filter](#) (struct smiclient_globals *azg, char *ipAddr, char *af, char *saf, struct list *bgpShowList, int(*callback)(struct list *showlist))

neighbors matching the given prefix filter fro ipv6 address

- int [_merge_bgp_neighbors_rcv_prefix_filter](#) (struct list *listDest, struct list *listSrc)
- int [_merge_bgp_word_neighbors](#) (struct list *listDest, struct list *listSrc)
- int [_merge_bgp_neighbors_HKC](#) (struct list *listDest, struct list *listSrc)
- int [_merge_bgp_neighbor_advertised_routes](#) (struct list *listDest, struct list *listSrc)
- int [smi_bgp_show_ip_bgp_community_list](#) (struct smiclient_globals *azg, char *commListName, char *af, char *saf, char *vrfOption, int exact, struct list *showList, int(*callback)(struct list *showlist))

show ip bgp community list displays routes matching the community-list

- int [smi_bgp_show_ip_bgp_community](#) (struct smiclient_globals *azg, char *commListName, char *af, char *saf, int exact, struct list *showList, int(*callback)(struct list *showlist))

show ip bgp community displays routes matching the communities

- int [smi_bgp_show_ip_bgp](#) (struct smiclient_globals *azg, char *name, char *ipAddr, char *af, char *saf, enum smi_show_type vrfOption, char *vrfName, int prefixCheck, struct list *showList, int(*callback)(struct list *showlist))

show ip bgp displays routes matching the given view name

- int [smi_bgp_show_bgp](#) (struct smiclient_globals *azg, char *ip_addr, char *af, char *saf, char *vrfName, int prefixCheck, struct list *showList, int(*callback)(struct list *showlist))

show bgp displays bgp routes

- int [smi_show_ip_bgp_paths](#) (struct smiclient_globals *azg, struct list *showList, int(*callback)(struct list *showlist))

show ip bgp paths displays the path information

- int [smi_show_ip_bgp_regexp](#) (struct smiclient_globals *azg, char *vrfName, char *bgpRegExp, char *af, char *saf, struct list *showList, int(*callback)(struct list *showlist))

show ip bgp paths displays the routes matching the AS path regular expression

- int [smi_show_ip_bgp_safi_regexp](#) (struct smiclient_globals *azg, char *vrfName, char *bgpRegExp, char *af, char *saf, struct list *showList, int(*callback)(struct list *showlist))

show ip bgp paths displays the ipv4 routes matching the AS path regular expression

- int [smi_show_bgp_regexp](#) (struct smiclient_globals *azg, char *bgpRegExp, struct list *showList, int(*callback)(struct list *showlist))

show ip bgp paths displays the routes matching the AS path regular expression

- int [smi_show_bgp_afi_regexp_safi](#) (struct smiclient_globals *azg, char *vrfName, char *bgpRegExp, char *af, char *saf, struct list *showList, int(*callback)(struct list *showlist))

show ip bgp paths displays the ipv4/ipv6 routes matching the AS path regular expression

- int [smi_show_ip_bgp_quote_regexp](#) (struct smiclient_globals *azg, char *vrfName, char *bgpRegExp, char *af, char *saf, struct list *showList, int(*callback)(struct list *showlist))

show ip bgp paths displays the ipv4/ipv6 routes matching the AS path quote-regular expression word

- int [smi_show_bgp_ip_neighbor_routes](#) (struct smiclient_globals *azg, char *vrfName, char *peerAddr, char *af, char *saf, struct list *showList, int(*callback)(struct list *showlist))

show ip bgp paths displays the information on TCP and BGP ipv4/ipv6 neighbor connections

- int [smi_show_ip_bgp_route_map](#) (struct smiclient_globals *azg, char *vrfName, char *bgpRegExp, char *af, char *saf, struct list *showList, int(*callback)(struct list *showlist))

show ip bgp paths displays the routes matching the route-map

- int [smi_show_ip_bgp_safi_route_map](#) (struct smiclient_globals *azg, char *vrfName, char *bgpRegExp, char *af, char *saf, struct list *showList, int(*callback)(struct list *showlist))

show ip bgp paths displays the ipv4 routes matching the route-map

- int [smi_show_bgp_route_map](#) (struct smiclient_globals *azg, char *bgpRegExp, struct list *showList, int(*callback)(struct list *showlist))

show ip bgp paths displays the routes matching the route-map

- int [smi_show_bgp_afi_route_map_safi](#) (struct smiclient_globals *azg, char *vrfName, char *bgpRegExp, char *af, char *saf, struct list *showList, int(*callback)(struct list *showlist))

show ip bgp paths displays the ipv4/ipv6 routes matching the route-map

- int [smi_bgp_show_bgp_extcommunity_list](#) (struct smiclient_globals *azg, char *commListame, struct list *bgpExtCommList, int startIndex, int endIndex, u_int32_t(*callbackFunc)(struct list *bgpExtCommList))

displays the configured extcommunity-list

- int [smi_bgp_show_ip_bgp_extcommunity_list_exact_match_vrf](#) (struct smiclient_globals *azg, char *commListame, char *af, char *saf, struct list *bgpExtCommList, char *vrfName, int exactMatchFlag, u_int32_t(*callbackFunc)(struct list *bgpExtCommList))
displays the routes of configured extcommunity-list
- int [smi_bgp_show_ip_bgp_extcommunity_list_exact_match](#) (struct smiclient_globals *azg, char *commListame, char *af, char *saf, struct list *bgpExtCommList, int exactMatchFlag, u_int32_t(*callbackFunc)(struct list *bgpExtCommList))
displays the routes of configured extcommunity-list
- int **smi_show_ip_bgp_statistics_info** (struct smiclient_globals *azg, struct bgpStatsInfo *bgpstats)
- int **smi_show_ip_bgp_attribute_info** (struct smiclient_globals *azg, struct list *showList, int(*callback)(struct list *showlist))
- int **smi_show_bgp_process_info** (struct smiclient_globals *azg, struct list *showList, int(*callback)(struct list *showlist))
- int [smi_bgp_address_family_set](#) (struct smiclient_globals *azg, u_int32_t vrId, int afi, int safi)
Sets the BGP af_flag.
- int **smi_bgp_address_family_set_validate** (struct smiclient_globals *azg, u_int32_t vrId, int afi, int safi)
- int **smi_bgp_set_address_family_wrap** (struct smiclient_globals *azg, u_int32_t vr_id, int bgpAs, int afi)
- int **smi_bgp_set_address_family_wrap_validate** (struct smiclient_globals *azg, u_int32_t vr_id, int bgpAs, int afi)
- int **smi_bgp_set_subaddress_family_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, int bgpAs, int afi, int safi, char *vrfName)
- int **smi_bgp_set_subaddress_family_wrap** (struct smiclient_globals *azg, u_int32_t vr_id, int bgpAs, int afi, int safi, char *vrfName)
- int [smi_bgp_api_address_family_unset](#) (struct smiclient_globals *azg, u_int32_t vrId, int afi, int safi)
Unsets the BGP af_flag.
- int **smi_bgp_api_address_family_unset_validate** (struct smiclient_globals *azg, u_int32_t vrId, int afi, int safi)
- int [smi_bgp_nbr_address_family_set](#) (struct smiclient_globals *azg, u_int32_t vrId, char *peerAddr, int afi, int safi)
Sets the BGP af_flag for neighbor.
- int **smi_bgp_nbr_address_family_set_validate** (struct smiclient_globals *azg, u_int32_t vrId, char *peerAddr, int afi, int safi)
- int [smi_bgp_nbr_address_family_unset](#) (struct smiclient_globals *azg, u_int32_t vrId, int afi, int safi, char *peerAddr)
Unsets the BGP af_flag for neighbor.

- int **smi_bgp_nbr_address_family_unset_validate** (struct smiclient_globals *azg, u_int32_t vrId, int afi, int safi, char *peerAddr)
- int **smi_bgp_get_address_family** (struct smiclient_globals *azg, u_int32_t vrId, int afi, int safi, enum address_family_flag *addressFamilyFlag)

Returns a enum value corresponding to the router address family flag configured.

- int **smi_bgp_get_nbr_address_family** (struct smiclient_globals *azg, u_int32_t vrId, char *peerAddr, int afi, int safi, enum nbr_addr_family *addressFamilyFlag)

Returns a enum value corresponding to the neighbor address family flag configured.

- int **smi_bgp_vrf_bind_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName)
- s_int32_t **smi_bgp_peer_group_add_set_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, char *peerGroupTag)
- s_int32_t **smi_bgp_peer_group_add_set_sdkapi** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, char *peerGroupTag)
- s_int32_t **smi_bgp_aggregate_addr_set_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, char *aggregateAddr, int aggregateType)
- s_int32_t **smi_bgp_addr_family_aggregate_addr_set_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, char *aggregateAddr, int afi, int safi, int aggregateType)
- s_int32_t **smi_bgp_aggregate_addr_set_sdkapi** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, char *aggregateAddr, int aggregateType)
- s_int32_t **smi_bgp_addr_family_aggregate_addr_set_sdkapi** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, char *aggregateAddr, int afi, int safi, int aggregateType)
- s_int32_t **smi_bgp_aggregate_addr_unset_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, char *aggregateAddr, int aggregateType)
- s_int32_t **smi_bgp_addr_family_aggregate_addr_unset_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, char *aggregateAddr, int afi, int safi)
- s_int32_t **smi_bgp_aggregate_addr_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, char *aggregateAddr, int aggregateType)
- s_int32_t **smi_bgp_addr_family_aggregate_addr_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, char *aggregateAddr, int afi, int safi)
- s_int32_t **smi_bgp_conf_ext_asn_cap_set_validate** (struct smiclient_globals *azg, u_int32_t vrId, bool_t setExtAsnCap)
- s_int32_t **smi_bgp_conf_ext_asn_cap_set** (struct smiclient_globals *azg, u_int32_t vrId, bool_t setExtAsnCap)
- s_int32_t **smi_bgp_config_nexthop_tracking_validate** (struct smiclient_globals *azg, u_int32_t vrId, bool_t setNexthopTriggerEnable)

- s_int32_t **smi_bgp_config_nexthop_tracking** (struct smiclient_globals *azg, u_int32_t vrId, bool_t setNexthopTriggerEnable)
- s_int32_t **smi_bgp_config_nht_delay_interval_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int16_t delayInterval)
- s_int32_t **smi_bgp_config_nht_delay_interval** (struct smiclient_globals *azg, u_int32_t vrId, u_int16_t delayInterval)
- int **smi_bgp_rfc1771_path_strict_set** (struct smiclient_globals *azg, int vrId)
- int **smi_bgp_rfc1771_path_strict_unset** (struct smiclient_globals *azg, int vrId)
- int **smi_bgp_unset_local_as_count_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int localAsCount)
- int **smi_bgp_set_local_as_count_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int localAsCount)
- s_int32_t **smi_bgp_set_client_to_client_reflected_routes** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs)
- s_int32_t **smi_bgp_unset_client_to_client_reflected_routes** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs)
- s_int32_t **smi_bgp_set_inbound_route_filter** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs)
- s_int32_t **smi_bgp_unset_inbound_route_filter** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs)
- s_int32_t **smi_bgp_set_log_neighbor_changes** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs)
- s_int32_t **smi_bgp_unset_log_neighbor_changes** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs)
- s_int32_t **smi_bgp_scan_time_set_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, u_int32_t scanInterval)
- s_int32_t **smi_bgp_scan_time_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs)
- int **smi_bgp_distance_set_wrap** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, u_char distance, char *distanceSrcIp)
- int **smi_bgp_distance_set_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, u_char distance, char *distanceSrcIp)
- int **smi_bgp_distance_unset_wrap** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *distanceSrcIp)
- int **smi_bgp_distance_unset_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *distanceSrcIp)
- s_int32_t **smi_bgp_distance_set_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, u_char distance, char *distanceSrcIp, char *distanceACLName)
- s_int32_t **smi_bgp_distance_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *distanceSrcIp, char *distanceACLName)
- s_int32_t **smi_bgp_mpls_resolution_set_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs)
- s_int32_t **smi_bgp_mpls_resolution_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs)

- `s_int32_t smi_bgp_timers_disallow_hold_timer_set_sdkapi` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`)
- `s_int32_t smi_bgp_timers_disallow_hold_timer_unset_sdkapi` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`)
- `int smi_bgp_client_create` (struct `smiclient_globals *azg`)
- `int smi_bgp_client_set_service` (struct `smi_client *ac`, `int service`, `int module`)
- `struct smi_client_handler * smi_bgp_client_handler_create` (struct `smi_client *ac`, `int type`, `int module`)
- `int smi_bgp_afi_peerAddr` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `char *peerAddr`, `int peerAfi`)
- `int smi_bgp_afi_peerAddr_validate` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `char *peerAddr`, `int peerAfi`)
- `int smi_bgp_safi_peerAddr_validate` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `char *peerAddr`, `int peerAfi`, `int peerSafi`)
- `int smi_bgp_safi_peerAddr` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `char *peerAddr`, `int peerAfi`, `int peerSafi`)
- `int smi_bgp_nbr_address_family_wrap_validate` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `char *peerAddr`, `int afi`, `int safi`, `bool_t setAfFlag`)
- `int smi_bgp_nbr_address_family_wrap` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `char *peerAddr`, `int afi`, `int safi`, `bool_t setAfFlag`)
- `int smi_bgp_dampening_routemap_name_wrap_validate` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `char *rmapName`, `bool_t setDrFlag`)
- `int smi_bgp_dampening_routemap_name_wrap` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `char *rmapName`, `bool_t setDrFlag`)
- `int smi_bgp_dampening_wrap_validate` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `bool_t setDmFlag`)
- `int smi_bgp_dampening_wrap` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `bool_t setDmFlag`)
- `int smi_bgp_dampening_half_life_wrap_validate` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `u_int32_t halfLife`, `bool_t setDhlFlag`)
- `int smi_bgp_dampening_half_life_wrap` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `u_int32_t halfLife`, `bool_t setDhlFlag`)
- `int smi_bgp_dampening_half_life_reuse_suppress_maxsuppress_wrap_validate` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `u_int32_t halfLife`, `u_int32_t reusePenalty`, `u_int32_t suppressPenalty`, `u_int32_t maxSuppress`, `bool_t setDsmFlag`)
- `int smi_bgp_dampening_half_life_reuse_suppress_maxsuppress_wrap` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `u_int32_t halfLife`, `u_int32_t reusePenalty`, `u_int32_t suppressPenalty`, `u_int32_t maxSuppress`, `bool_t setDsmFlag`)
- `int smi_bgp_dampening_half_life_reuse_suppress_maxsuppress_unreachability_wrap_validate` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `u_int32_t halfLife`, `u_int32_t reusePenalty`, `u_int32_t suppressPenalty`, `u_int32_t maxSuppress`, `u_int32_t unreachHalfLife`, `bool_t setDsmuFlag`)

- int **smi_bgp_dampening_half_life_reuse_supress_maxsupress_unreachability_wrap** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, u_int32_t halfLife, u_int32_t reusePenalty, u_int32_t suppressPenalty, u_int32_t maxSuppress, u_int32_t unreachHalfLife, bool_t setDsmuFlag)
- int **smi_bgp_addr_family_dampening_routemap_name_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int afi, int safi, char *rmapName, bool_t setDrFlag)
- int **smi_bgp_addr_family_dampening_routemap_name_wrap** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int afi, int safi, char *rmapName, bool_t setDrFlag)
- int **smi_bgp_addr_family_dampening_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int afi, int safi, bool_t setDmFlag)
- int **smi_bgp_addr_family_dampening_wrap** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int afi, int safi, bool_t setDmFlag)
- int **smi_bgp_addr_family_dampening_half_life_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int afi, int safi, u_int32_t halfLife, bool_t setDhlFlag)
- int **smi_bgp_addr_family_dampening_half_life_wrap** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int afi, int safi, u_int32_t halfLife, bool_t setDhlFlag)
- int **smi_bgp_addr_family_dampening_half_life_reuse_supress_maxsuppress_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int afi, int safi, u_int32_t halfLife, u_int32_t reusePenalty, u_int32_t suppressPenalty, u_int32_t maxSuppress, bool_t setDsmFlag)
- int **smi_bgp_addr_family_dampening_half_life_reuse_supress_maxsuppress_wrap** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int afi, int safi, u_int32_t halfLife, u_int32_t reusePenalty, u_int32_t suppressPenalty, u_int32_t maxSuppress, bool_t setDsmFlag)
- int **smi_bgp_addr_family_dampening_half_life_reuse_supress_maxsupress_unreachability_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int afi, int safi, u_int32_t halfLife, u_int32_t reusePenalty, u_int32_t suppressPenalty, u_int32_t maxSuppress, u_int32_t unreachHalfLife, bool_t setDsmuFlag)
- int **smi_bgp_addr_family_dampening_half_life_reuse_supress_maxsupress_unreachability_wrap** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int afi, int safi, u_int32_t halfLife, u_int32_t reusePenalty, u_int32_t suppressPenalty, u_int32_t maxSuppress, u_int32_t unreachHalfLife, bool_t setDsmuFlag)
- int **smi_bgp_api_distance_config_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, int afi, int safi, u_int8_t distanceEbgp, u_int8_t distanceIbgp, u_int8_t distanceLocal, bool_t setDcFlag)
- int **smi_bgp_redistribute_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, int afi, int redistType, bool_t setRdFlag)
- int **smi_bgp_redistribute_wrap** (struct smiclient_globals *azg, u_int32_t vrId, int afi, int redistType, bool_t setRdFlag)
- int **smi_bgp_bestpath_med_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *medType, bool_t setBpmFlag)

- **int smi_bgp_bestpath_med_wrap** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *medType, bool_t setBpmFlag)
- **int smi_peer_transport_connection_passive_wrap** (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr, bool_t setTcpFlag)
- **int smi_bgp_grst_stalepath_time_wrap** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t stalepath_time, bool_t setNsFlag)
- **int smi_neighbor_disallow_infinite_timer_wrap** (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr, bool_t setNcFlag)
- **int smi_bgp_config_set_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t configType, bool_t setConfigFlag)
- **int smi_bgp_config_set** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t configType, bool_t setConfigFlag)
- **int smi_bgp_synchronization_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, bool_t setSyncFlag)
- **int smi_bgp_address_family_synchronization_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int afi, int safi, bool_t setSyncFlag)
- **int smi_bgp_synchronization_wrap** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, bool_t setSyncFlag)
- **int smi_bgp_address_family_synchronization_wrap** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int afi, int safi, bool_t setSyncFlag)
- **int smi_bgp_network_sync_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, bool_t setNwSyncFlag)
- **int smi_bgp_address_family_network_sync_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int afi, int safi, bool_t setNwSyncFlag)
- **int smi_bgp_network_sync_wrap** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, bool_t setNwSyncFlag)
- **int smi_bgp_address_family_network_sync_wrap** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int afi, int safi, bool_t setNwSyncFlag)
- **int smi_bgp_confederation_peers_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int confedId, bool_t setConfedPeerFlag)
- **int smi_bgp_confederation_peers_wrap** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int confedId, bool_t setConfedPeerFlag)
- **s_int32_t smi_peer_dynamic_capability_set_wrap** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, bool_t dynamicCapability)
- **s_int32_t smi_peer_dynamic_capability_set_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, bool_t dynamicCapability)
- **s_int32_t smi_neighbor_capability_route_refresh_set_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, bool_t capabilityRouteRefresh)
- **s_int32_t smi_neighbor_capability_route_refresh_set_wrap** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, bool_t capabilityRouteRefresh)

- `s_int32_t smi_neighbor_collide_established_set_wrap_validate` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `char *peerAddr`, `bool_t collideEstablished`)
- `s_int32_t smi_neighbor_collide_established_set_wrap` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `char *peerAddr`, `bool_t collideEstablished`)
- `s_int32_t smi_bgp_vrf_neighbor_as_override_set_wrap_validate` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `char *peerAddr`, `int afi`, `int safi`, `bool_t neighborAsOverride`)
- `s_int32_t smi_bgp_vrf_neighbor_as_override_set_wrap` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `char *peerAddr`, `int afi`, `int safi`, `bool_t neighborAsOverride`)
- `s_int32_t smi_neighbor_capability_grst_set_wrap_validate` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `char *peerAddr`, `int afi`, `int safi`, `bool_t neighborCapabilityGrst`)
- `s_int32_t smi_neighbor_capability_grst_set_wrap` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `char *peerAddr`, `int afi`, `int safi`, `bool_t neighborCapabilityGrst`)
- `s_int32_t smi_neighbor_af_remove_private_as_set_wrap_validate` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `char *peerAddr`, `int afi`, `int safi`, `bool_t neighborRemovePvtAs`)
- `s_int32_t smi_neighbor_af_remove_private_as_set_wrap` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `char *peerAddr`, `int afi`, `int safi`, `bool_t neighborRemovePvtAs`)
- `s_int32_t smi_neighbor_remove_private_as_set_wrap_validate` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `char *peerAddr`, `bool_t neighborRemovePvtAs`)
- `s_int32_t smi_neighbor_remove_private_as_set_wrap` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `char *peerAddr`, `bool_t neighborRemovePvtAs`)
- `s_int32_t smi_neighbor_attr_unchanged_as_path_set_wrap` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `char *peerAddr`, `bool_t neighborAttrUnchangedAsPath`)
- `s_int32_t smi_neighbor_attr_unchanged_as_path_set_wrap_validate` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `char *peerAddr`, `bool_t neighborAttrUnchangedAsPath`)
- `s_int32_t smi_neighbor_attr_unchanged_nexthop_set_wrap` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `char *peerAddr`, `int afi`, `int safi`, `bool_t neighborAttrUnchangedNexthop`)
- `s_int32_t smi_neighbor_attr_unchanged_nexthop_set_wrap_validate` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `char *peerAddr`, `int afi`, `int safi`, `bool_t neighborAttrUnchangedNexthop`)
- `s_int32_t smi_neighbor_attr_unchanged_med_set_wrap` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `char *peerAddr`, `int afi`, `int safi`, `bool_t neighborAttrUnchangedMed`)
- `s_int32_t smi_neighbor_attr_unchanged_med_set_wrap_validate` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `char *peerAddr`, `int afi`, `int safi`, `bool_t neighborAttrUnchangedMed`)

- `s_int32_t smi_peer_route_reflector_client_set_wrap` (struct `smiclient_globals` *azg, `u_int32_t` vrId, `u_int32_t` bgpAs, `char` *peerAddr, `int` afi, `int` safi, `bool_t` peerRouteReflector)
- `s_int32_t smi_peer_route_reflector_client_set_wrap_validate` (struct `smiclient_globals` *azg, `u_int32_t` vrId, `u_int32_t` bgpAs, `char` *peerAddr, `int` afi, `int` safi, `bool_t` peerRouteReflector)
- `s_int32_t smi_neighbor_route_server_client_set_wrap` (struct `smiclient_globals` *azg, `u_int32_t` vrId, `u_int32_t` bgpAs, `char` *peerAddr, `int` afi, `int` safi, `bool_t` neighborRouteServerClient)
- `s_int32_t smi_neighbor_route_server_client_set_wrap_validate` (struct `smiclient_globals` *azg, `u_int32_t` vrId, `u_int32_t` bgpAs, `char` *peerAddr, `int` afi, `int` safi, `bool_t` neighborRouteServerClient)
- `s_int32_t smi_neighbor_enforce_multihop_set_wrap` (struct `smiclient_globals` *azg, `u_int32_t` vrId, `u_int32_t` bgpAs, `char` *peerAddr, `bool_t` enforceMultihop)
- `s_int32_t smi_neighbor_enforce_multihop_set_wrap_validate` (struct `smiclient_globals` *azg, `u_int32_t` vrId, `u_int32_t` bgpAs, `char` *peerAddr, `bool_t` enforceMultihop)
- `s_int32_t smi_neighbor_override_capability_set_wrap` (struct `smiclient_globals` *azg, `u_int32_t` vrId, `u_int32_t` bgpAs, `char` *peerAddr, `bool_t` neighborOverrideCapability)
- `s_int32_t smi_neighbor_override_capability_set_wrap_validate` (struct `smiclient_globals` *azg, `u_int32_t` vrId, `u_int32_t` bgpAs, `char` *peerAddr, `bool_t` neighborOverrideCapability)
- `s_int32_t smi_neighbor_strict_capability_set_wrap` (struct `smiclient_globals` *azg, `u_int32_t` vrId, `u_int32_t` bgpAs, `char` *peerAddr, `bool_t` neighborStrictCapability)
- `s_int32_t smi_neighbor_strict_capability_set_wrap_validate` (struct `smiclient_globals` *azg, `u_int32_t` vrId, `u_int32_t` bgpAs, `char` *peerAddr, `bool_t` neighborStrictCapability)
- `s_int32_t smi_peer_disallow_hold_timer_set_wrap` (struct `smiclient_globals` *azg, `u_int32_t` vrId, `u_int32_t` bgpAs, `char` *peerAddr, `bool_t` disallowHoldtimer)
- `s_int32_t smi_peer_disallow_hold_timer_set_wrap_validate` (struct `smiclient_globals` *azg, `u_int32_t` vrId, `u_int32_t` bgpAs, `char` *peerAddr, `bool_t` disallowHoldtimer)
- `s_int32_t smi_peer_dont_capability_negotiate_set_wrap` (struct `smiclient_globals` *azg, `u_int32_t` vrId, `u_int32_t` bgpAs, `char` *peerAddr, `bool_t` dontCapabilityNegotiate)
- `s_int32_t smi_peer_dont_capability_negotiate_set_wrap_validate` (struct `smiclient_globals` *azg, `u_int32_t` vrId, `u_int32_t` bgpAs, `char` *peerAddr, `bool_t` dontCapabilityNegotiate)
- `s_int32_t smi_bgp_multiple_instance_set_wrap` (struct `smiclient_globals` *azg, `u_int32_t` vrId, `bool_t` multiInstance)
- `s_int32_t smi_bgp_multiple_instance_set_wrap_validate` (struct `smiclient_globals` *azg, `u_int32_t` vrId, `bool_t` multiInstance)
- `s_int32_t smi_peer_transport_connection_passive_set_wrap` (struct `smiclient_globals` *azg, `u_int32_t` vrId, `u_int32_t` bgpAs, `char` *peerAddr, `bool_t` transportConnectionPassive)

- s_int32_t **smi_peer_transport_connection_passive_set_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, bool_t transportConnectionPassive)
- s_int32_t **smi_peer_shutdown_set_wrap** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, bool_t peerShutdown)
- s_int32_t **smi_peer_shutdown_set_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, bool_t peerShutdown)
- s_int32_t **smi_peer_soft_reconfiguration_inbound_set_wrap** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int afi, int safi, bool_t softReconfigInbound)
- s_int32_t **smi_peer_soft_reconfiguration_inbound_set_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int afi, int safi, bool_t softReconfigInbound)
- s_int32_t **smi_peer_send_community_set_wrap** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int afi, int safi, bool_t sendCommunity)
- s_int32_t **smi_peer_send_community_set_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int afi, int safi, bool_t sendCommunity)
- s_int32_t **smi_peer_next_hop_self_set_wrap** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int afi, int safi, bool_t nextHopSelf)
- s_int32_t **smi_peer_next_hop_self_set_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int afi, int safi, bool_t nextHopSelf)
- s_int32_t **smi_bgp_set_address_family_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int afi, int safi)
- s_int32_t **smi_bgp_set_address_family** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int afi, int safi)
- s_int32_t **smi_peer_set_address_family_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int afi, int safi)
- s_int32_t **smi_peer_set_address_family** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int afi, int safi)
- s_int32_t **smi_bgp_unset_address_family_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int afi, int safi)
- s_int32_t **smi_bgp_unset_address_family** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int afi, int safi)
- s_int32_t **smi_peer_unset_address_family_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int afi, int safi)
- s_int32_t **smi_peer_unset_address_family** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, int afi, int safi)
- int **smi_bgp_aggregate_nexthop_check_set_wrap** (struct smiclient_globals *azg, u_int32_t vr_id, int aggNexthop)
- int **smi_bgp_aggregate_nexthop_check_set_wrap_validate** (struct smiclient_globals *azg, u_int32_t vr_id, int aggNexthop)
- int **smi_bgp_fast_external_failover_set_wrap** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, int externalFailover)
- int **smi_bgp_fast_external_failover_set_wrap_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, int externalFailover)

- int **smi_bgp_rfc1771_path_select_set_wrap** (struct smiclient_globals *azg, u_int32_t vr_id, int pathSelect)
- int **smi_bgp_rfc1771_path_select_set_wrap_validate** (struct smiclient_globals *azg, u_int32_t vr_id, int pathSelect)
- int **smi_bgp_always_compare_med_set_wrap** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, int compareMed)
- int **smi_bgp_always_compare_med_set_wrap_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, int compareMed)
- int **smi_bgp_bestpath_aspath_ignore_set_wrap** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int bestpathAspath)
- int **smi_bgp_bestpath_aspath_ignore_set_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int bestpathAspath)
- int **smi_bgp_bestpath_compare_confed_aspath_set_wrap** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, int bestpathCompareConfed)
- int **smi_bgp_bestpath_compare_confed_aspath_set_wrap_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, int bestpathCompareConfed)
- int **smi_bgp_bestpath_compare_router_id_set_wrap** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int bestpathCompareRouterId)
- int **smi_bgp_bestpath_compare_router_id_set_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int bestpathCompareRouterId)
- int **smi_bgp_bestpath_dont_compare_originator_id_set_wrap** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int bestpathDontCompareOriginator)
- int **smi_bgp_bestpath_dont_compare_originator_id_set_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int bestpathDontCompareOriginator)
- int **smi_bgp_bestpath_tie_break_on_age_set_wrap** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int bestpathTieBreak)
- int **smi_bgp_bestpath_tie_break_on_age_set_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int bestpathTieBreak)
- int **smi_bgp_default_ipv4_unicast_set_wrap** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int ipv4Unicast)
- int **smi_bgp_default_ipv4_unicast_set_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int ipv4Unicast)
- int **smi_bgp_deterministic_med_set_wrap** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int deterministicMed)
- int **smi_bgp_deterministic_med_set_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int deterministicMed)
- int **smi_bgp_enforce_first_as_set_wrap** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, bool_t enforceFirst)
- int **smi_bgp_enforce_first_as_set_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, bool_t enforceFirst)
- int **smi_bgp_grst_set_wrap** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, bool_t grst)

- int **smi_bgp_grst_set_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, bool_t grst)
- int **smi_bgp_disable_adj_out_set_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, bool_t setDisableAdjOut)
- int **smi_bgp_disable_adj_out_set_wrap** (struct smiclient_globals *azg, u_int32_t vrId, bool_t setDisableAdjOut)
- int **smi_bgp_rfc1771_path_strict_set_wrap_validate** (struct smiclient_globals *azg, int vrId, bool_t rfc1771StrictSet)
- int **smi_bgp_rfc1771_path_strict_set_wrap** (struct smiclient_globals *azg, int vrId, bool_t rfc1771StrictSet)
- int **smi_bgp_set_local_as_count_sdkapi_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int localAsCount, bool_t setLocalAsCount)
- int **smi_bgp_set_local_as_count_sdkapi_wrap** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, int localAsCount, bool_t setLocalAsCount)
- s_int32_t **smi_bgp_set_client_to_client_reflected_routes_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, bool_t setAfClientReflect)
- s_int32_t **smi_bgp_set_client_to_client_reflected_routes_wrap** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, bool_t setAfClientReflect)
- s_int32_t **smi_bgp_set_inbound_route_filter_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, bool_t setInboundRouteFilter)
- s_int32_t **smi_bgp_set_inbound_route_filter_wrap** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, bool_t setInboundRouteFilter)
- s_int32_t **smi_bgp_set_log_neighbor_changes_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, bool_t setLogNbrChanges)
- s_int32_t **smi_bgp_set_log_neighbor_changes_wrap** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, bool_t setLogNbrChanges)
- s_int32_t **smi_bgp_scan_time_set_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, u_int32_t scanInterval)
- s_int32_t **smi_bgp_scan_time_unset_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs)
- s_int32_t **smi_bgp_distance_set_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, u_char distance, char *distanceSrcIp, char *distanceACLName)
- s_int32_t **smi_bgp_distance_unset_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, u_char distance, char *distanceSrcIp, char *distanceACLName)
- s_int32_t **smi_bgp_mpls_resolution_set_sdkapi_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, bool_t setMplsResolution)
- s_int32_t **smi_bgp_mpls_resolution_set_sdkapi_wrap** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, bool_t setMplsResolution)
- s_int32_t **smi_bgp_set_gshut_capable_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, bool_t setGshutCapable)

- `s_int32_t smi_bgp_set_gshut_capable_wrap` (struct `smiclient_globals *azg`, `u_int32_t vrId`, `u_int32_t bgpAs`, `bool_t setGshutCapable`)
- `int smi_neighbor_capability_orf_prefix_wrap_validate` (struct `smiclient_globals *azg`, `char *peerAddr`, `char *orfPrefixOpt`, `bool_t setNcpFlag`, `u_int32_t vr_id`, `u_int32_t bgpAs`)
- `int smi_neighbor_capability_orf_prefix_wrap` (struct `smiclient_globals *azg`, `char *peerAddr`, `char *orfPrefixOpt`, `bool_t setNcpFlag`, `u_int32_t vr_id`, `u_int32_t bgpAs`)
- `int smi_neighbor_addr_family_capability_orf_prefix_wrap_validate` (struct `smiclient_globals *azg`, `char *peerAddr`, `char *orfPrefixOpt`, `int afi`, `int safi`, `bool_t setNcpFlag`, `u_int32_t vr_id`, `u_int32_t bgpAs`)
- `int smi_neighbor_addr_family_capability_orf_prefix_wrap` (struct `smiclient_globals *azg`, `char *peerAddr`, `char *orfPrefixOpt`, `int afi`, `int safi`, `bool_t setNcpFlag`, `u_int32_t vr_id`, `u_int32_t bgpAs`)
- `s_int32_t smi_bgp_static_network_addr_nomask_unset_sdkapi_validate` (struct `smiclient_globals *azg`, `u_int32_t as`, `char *ip_str`, `u_int32_t vr_id`)
- `s_int32_t smi_bgp_static_network_addr_nomask_unset_sdkapi` (struct `smiclient_globals *azg`, `u_int32_t as`, `char *localAddr`, `u_int32_t vrId`)
- `s_int32_t smi_bgp_static_network_addr_mask_unset_sdkapi_validate` (struct `smiclient_globals *azg`, `u_int32_t as`, `char *ip_str`, `char *mask`, `u_int32_t vr_id`)
- `s_int32_t smi_bgp_static_network_addr_mask_unset_sdkapi` (struct `smiclient_globals *azg`, `u_int32_t as`, `char *ip_str`, `char *mask`, `u_int32_t vr_id`)
- `s_int32_t smi_bgp_addr_family_static_network_addr_nomask_unset_sdkapi_validate` (struct `smiclient_globals *azg`, `u_int32_t as`, `char *ip_str`, `int afi`, `int safi`, `u_int32_t vr_id`)
- `s_int32_t smi_bgp_addr_family_static_network_addr_nomask_unset_sdkapi` (struct `smiclient_globals *azg`, `u_int32_t as`, `char *ip_str`, `int afi`, `int safi`, `u_int32_t vr_id`)
- `s_int32_t smi_bgp_addr_family_static_network_addr_mask_unset_sdkapi_validate` (struct `smiclient_globals *azg`, `u_int32_t as`, `char *ip_str`, `char *mask`, `int afi`, `int safi`, `u_int32_t vr_id`)
- `s_int32_t smi_bgp_addr_family_static_network_addr_mask_unset_sdkapi` (struct `smiclient_globals *azg`, `u_int32_t as`, `char *ip_str`, `char *mask`, `int afi`, `int safi`, `u_int32_t vr_id`)
- `s_int32_t smi_bgp_static_network_backdoor_nomask_unset_sdkapi` (struct `smiclient_globals *azg`, `u_int32_t as`, `char *ip_str`, `u_int32_t backdoor`, `u_int32_t vr_id`)
- `s_int32_t smi_bgp_static_network_backdoor_nomask_set_sdkapi` (struct `smiclient_globals *azg`, `u_int32_t as`, `char *ip_str`, `u_int32_t backdoor`, `u_int32_t vr_id`)
- `s_int32_t smi_bgp_static_network_backdoor_nomask_set_sdkapi_validate` (struct `smiclient_globals *azg`, `u_int32_t as`, `char *ip_str`, `u_int32_t backdoor`, `u_int32_t vr_id`)
- `s_int32_t smi_bgp_static_network_backdoor_mask_unset_sdkapi` (struct `smiclient_globals *azg`, `u_int32_t as`, `char *ip_str`, `char *mask`, `u_int32_t backdoor`, `u_int32_t vr_id`)

- s_int32_t **smi_bgp_static_network_backdoor_mask_unset_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t as, char *ip_str, char *mask, u_int32_t backdoor, u_int32_t vr_id)
- s_int32_t **smi_bgp_addr_family_static_network_backdoor_nomask_unset_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t as, char *ip_str, u_int32_t backdoor, int afi, int safi, u_int32_t vr_id)
- s_int32_t **smi_bgp_addr_family_static_network_backdoor_nomask_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t as, char *ip_str, u_int32_t backdoor, int afi, int safi, u_int32_t vr_id)
- s_int32_t **smi_bgp_addr_family_static_network_backdoor_mask_unset_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t as, char *ip_str, char *mask, u_int32_t backdoor, int afi, int safi, u_int32_t vr_id)
- s_int32_t **smi_bgp_addr_family_static_network_backdoor_mask_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t as, char *ip_str, char *mask, u_int32_t backdoor, int afi, int safi, u_int32_t vr_id)
- s_int32_t **smi_bgp_static_network_rmap_nomask_unset_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t as, char *ip_str, char *networkRmapName, u_int32_t vr_id)
- s_int32_t **smi_bgp_static_network_rmap_nomask_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t as, char *ip_str, char *networkRmapName, u_int32_t vr_id)
- s_int32_t **smi_bgp_static_network_rmap_mask_unset_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t as, char *ip_str, char *mask, char *networkRmapName, u_int32_t vr_id)
- s_int32_t **smi_bgp_static_network_rmap_mask_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t as, char *ip_str, char *mask, char *networkRmapName, u_int32_t vr_id)
- s_int32_t **smi_bgp_addr_family_static_network_rmap_nomask_unset_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t as, char *ip_str, char *rmap_name, int afi, int safi, u_int32_t vr_id)
- s_int32_t **smi_bgp_addr_family_static_network_rmap_nomask_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t as, char *ip_str, char *rmap_name, int afi, int safi, u_int32_t vr_id)
- s_int32_t **smi_bgp_addr_family_static_network_rmap_mask_unset_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t as, char *ip_str, char *mask, char *networkRmapNameAf, int afi, int safi, u_int32_t vr_id)
- s_int32_t **smi_bgp_addr_family_static_network_rmap_mask_unset_sdkapi** (struct smiclient_globals *azg, u_int32_t as, char *ip_str, char *mask, char *networkRmapNameAf, int afi, int safi, u_int32_t vr_id)
- s_int32_t **smi_bgp_static_network_addr_nomask_set_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t bgpAs, char *localAddr, u_int32_t vrId)
- s_int32_t **smi_bgp_static_network_addr_nomask_set_sdkapi** (struct smiclient_globals *azg, u_int32_t bgpAs, char *localAddr, u_int32_t vrId)
- s_int32_t **smi_bgp_static_network_addr_mask_set_sdkapi_validate** (struct smiclient_globals *azg, u_int32_t as, char *ip_str, char *mask, u_int32_t vr_id)

- `s_int32_t smi_bgp_static_network_addr_mask_set_sdkapi` (struct `smiclient_globals *azg`, `u_int32_t as`, `char *ip_str`, `char *mask`, `u_int32_t vr_id`)
- `s_int32_t smi_bgp_addr_family_static_network_addr_nomask_set_sdkapi_validate` (struct `smiclient_globals *azg`, `u_int32_t as`, `char *ip_str`, `int afi`, `int safi`, `u_int32_t vr_id`)
- `s_int32_t smi_bgp_addr_family_static_network_addr_nomask_set_sdkapi` (struct `smiclient_globals *azg`, `u_int32_t as`, `char *ip_str`, `int afi`, `int safi`, `u_int32_t vr_id`)
- `s_int32_t smi_bgp_addr_family_static_network_addr_mask_set_sdkapi_validate` (struct `smiclient_globals *azg`, `u_int32_t as`, `char *ip_str`, `char *mask`, `int afi`, `int safi`, `u_int32_t vr_id`)
- `s_int32_t smi_bgp_addr_family_static_network_addr_mask_set_sdkapi` (struct `smiclient_globals *azg`, `u_int32_t as`, `char *ip_str`, `char *mask`, `int afi`, `int safi`, `u_int32_t vr_id`)
- `s_int32_t smi_bgp_static_network_backdoor_mask_set_sdkapi` (struct `smiclient_globals *azg`, `u_int32_t as`, `char *ip_str`, `char *mask`, `u_int32_t backdoor`, `u_int32_t vr_id`)
- `s_int32_t smi_bgp_static_network_backdoor_mask_set_sdkapi_validate` (struct `smiclient_globals *azg`, `u_int32_t as`, `char *ip_str`, `char *mask`, `u_int32_t backdoor`, `u_int32_t vr_id`)
- `s_int32_t smi_bgp_addr_family_static_network_backdoor_nomask_set_sdkapi_validate` (struct `smiclient_globals *azg`, `u_int32_t as`, `char *ip_str`, `u_int32_t backdoor`, `int afi`, `int safi`, `u_int32_t vr_id`)
- `s_int32_t smi_bgp_addr_family_static_network_backdoor_nomask_set_sdkapi` (struct `smiclient_globals *azg`, `u_int32_t as`, `char *ip_str`, `u_int32_t backdoor`, `int afi`, `int safi`, `u_int32_t vr_id`)
- `s_int32_t smi_bgp_addr_family_static_network_backdoor_mask_set_sdkapi_validate` (struct `smiclient_globals *azg`, `u_int32_t as`, `char *ip_str`, `char *mask`, `u_int32_t backdoor`, `int afi`, `int safi`, `u_int32_t vr_id`)
- `s_int32_t smi_bgp_addr_family_static_network_backdoor_mask_set_sdkapi` (struct `smiclient_globals *azg`, `u_int32_t as`, `char *ip_str`, `char *mask`, `u_int32_t backdoor`, `int afi`, `int safi`, `u_int32_t vr_id`)
- `s_int32_t smi_bgp_static_network_rmap_nomask_set_sdkapi_validate` (struct `smiclient_globals *azg`, `u_int32_t as`, `char *ip_str`, `char *networkRmapName`, `u_int32_t vr_id`)
- `s_int32_t smi_bgp_static_network_rmap_nomask_set_sdkapi` (struct `smiclient_globals *azg`, `u_int32_t as`, `char *ip_str`, `char *rmap_name`, `u_int32_t vr_id`)
- `s_int32_t smi_bgp_static_network_rmap_mask_set_sdkapi_validate` (struct `smiclient_globals *azg`, `u_int32_t as`, `char *ip_str`, `char *mask`, `char *networkRmapName`, `u_int32_t vr_id`)
- `s_int32_t smi_bgp_static_network_rmap_mask_set_sdkapi` (struct `smiclient_globals *azg`, `u_int32_t as`, `char *ip_str`, `char *mask`, `char *networkRmapName`, `u_int32_t vr_id`)
- `s_int32_t smi_bgp_addr_family_static_network_rmap_nomask_set_sdkapi_validate` (struct `smiclient_globals *azg`, `u_int32_t as`, `char *ip_str`, `char *rmap_name`, `int afi`, `int safi`, `u_int32_t vr_id`)

- `s_int32_t smi_bgp_addr_family_static_network_rmap_nomask_set_sdkapi` (struct smiclient_globals *azg, u_int32_t as, char *ip_str, char *rmap_name, int afi, int safi, u_int32_t vr_id)
- `s_int32_t smi_bgp_addr_family_static_network_rmap_mask_set_sdkapi_validate` (struct smiclient_globals *azg, u_int32_t as, char *ip_str, char *mask, char *networkRmapNameAf, int afi, int safi, u_int32_t vr_id)
- `s_int32_t smi_bgp_addr_family_static_network_rmap_mask_set_sdkapi` (struct smiclient_globals *azg, u_int32_t as, char *ip_str, char *mask, char *networkRmapNameAf, int afi, int safi, u_int32_t vr_id)
- `int smi_bgp_fetch_attribute` (struct smiclient_globals *azg, int max_entries, struct list **smi_obj_list, struct list *attrIdList, struct list *runtime_key_list, int have_more_entries)
- `s_int32_t smi_bgp_addr_family_redistribute_vrf_set_sdkapi_validate` (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, int afi, int safi, int redistrib_type, char *vrfName)
- `s_int32_t smi_bgp_addr_family_redistribute_vrf_set_sdkapi` (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, int afi, int safi, int redistrib_type, char *vrfName)
- `s_int32_t smi_bgp_addr_family_redistribute_vrf_unset_sdkapi_validate` (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, int afi, int safi, int redistrib_type, char *vrfName)
- `s_int32_t smi_bgp_addr_family_redistribute_vrf_unset_sdkapi` (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, int afi, int safi, int redistrib_type, char *vrfName)
- `int smi_bgp_vrf_rt_set_validate` (struct smiclient_globals *azg, u_int32_t vr_id, char *vrfName, char *rd_str, int direct)
- `int smi_bgp_vrf_rt_set` (struct smiclient_globals *azg, u_int32_t vr_id, char *vrfName, char *rd_str, int direct)
- `int smi_bgp_vrf_rt_unset_validate` (struct smiclient_globals *azg, u_int32_t vr_id, char *vrfName, char *rd_str, int direct)
- `int smi_bgp_vrf_rt_unset` (struct smiclient_globals *azg, u_int32_t vr_id, char *vrfName, char *rd_str, int direct)
- `int smi_bgp_vrf_rd_set_validate` (struct smiclient_globals *azg, u_int32_t vr_id, char *vrfName, char *rd_str)
- `int smi_bgp_vrf_rd_set` (struct smiclient_globals *azg, u_int32_t vr_id, char *vrfName, char *rd_str)
- `s_int32_t smi_bgp_peer_remote_as_vrf_set_sdkapi_validate` (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, char *vrf_name, int afi, int safi, char *peer_addr, int as)
- `s_int32_t smi_bgp_peer_remote_as_vrf_set_sdkapi` (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, char *vrf_name, int afi, int safi, char *peer_addr, int as)
- `s_int32_t smi_bgp_peer_remote_as_vrf_unset_sdkapi_validate` (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, char *vrf_name, int afi, int safi, char *peer_addr, int as)
- `s_int32_t smi_bgp_peer_remote_as_vrf_unset_sdkapi` (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, char *vrf_name, int afi, int safi, char *peer_addr, int as)

- `s_int32_t smi_bgp_vrf_address_family_set_validate` (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, int afi, int safi, char *vrfName)
- `s_int32_t smi_bgp_vrf_address_family_set` (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, int afi, int safi, char *vrfName)
- `s_int32_t smi_bgp_vrf_address_family_unset_validate` (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, int afi, int safi, char *vrfName)
- `s_int32_t smi_bgp_vrf_address_family_unset` (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, int afi, int safi, char *vrfName)
- `int smi_bgp_set_gshut_wrap_validate` (struct smiclient_globals *azg, u_int32_t vr_id, int bgpAs, bool_t setGshut)
- `int smi_bgp_set_gshut_wrap` (struct smiclient_globals *azg, u_int32_t vr_id, int bgpAs, bool_t setGshut)
- `int smi_bgp_set_gshut_unset_validate` (struct smiclient_globals *azg, u_int32_t vr_id)
- `int smi_bgp_set_gshut_set_validate` (struct smiclient_globals *azg, u_int32_t vr_id)
- `int smi_bgp_set_gshut_unset` (struct smiclient_globals *azg, u_int32_t vr_id)
- `int smi_bgp_set_gshut_set` (struct smiclient_globals *azg, u_int32_t vr_id)
- `s_int32_t smi_bgp_aggregate_addr_set_validate` (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, char *aggregateAddr)
- `s_int32_t smi_bgp_aggregate_addr_set` (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, char *aggregateAddr)
- `s_int32_t smi_bgp_addr_family_aggregate_addr_set_validate` (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, int afi, int safi, char *aggregateAddrAf)
- `s_int32_t smi_bgp_addr_family_aggregate_addr_set` (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, int afi, int safi, char *aggregateAddrAf)
- `int smi_peer_originate_set_sdkapi_validate` (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, char *peer_id, u_int8_t rmap)
- `int smi_peer_originate_set_sdkapi` (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, char *peer_id, u_int8_t rmap)
- `int smi_peer_addr_family_default_rmap_originate_set_sdkapi_validate` (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, char *peer_id, int afi, int safi, u_int8_t rmap)
- `int smi_peer_addr_family_default_rmap_originate_set_sdkapi` (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, char *peer_id, int afi, int safi, u_int8_t rmap)
- `int smi_peer_originate_unset_sdkapi_validate` (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, char *peer_id)
- `int smi_peer_originate_unset_sdkapi` (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, char *peer_id)
- `int smi_peer_timers_unset_wrap_validate` (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, char *peer_id, u_int32_t peerKeepAlive, u_int32_t peerHoldTime)
- `int smi_peer_timers_unset_wrap` (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, char *peer_id, u_int32_t peerKeepAlive, u_int32_t peerHoldTime)

- int **smi_peer_maximum_prefix_set_wrap_validate** (struct smiclient_globals *azg, u_int32_t as, char *peer_id, u_int32_t max, u_int32_t vr_id)
- int **smi_peer_maximum_prefix_set_wrap** (struct smiclient_globals *azg, u_int32_t as, char *peer_id, u_int32_t max, u_int32_t vr_id)
- s_int32_t **smi_neighbor_attr_unchanged_as_path_set_validate** (struct smiclient_globals *azg, u_int32_t vr_id, char *vrf_name, char *peer_str, int afi, int safi)
- s_int32_t **smi_neighbor_attr_unchanged_as_path_unset_validate** (struct smiclient_globals *azg, u_int32_t vr_id, char *vrf_name, char *peer_str, int afi, int safi)
- int **smi_bgp_api_distance_config_set_wrap_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, u_int8_t distance_ebgp, u_int8_t distance_ibgp, u_int8_t distance_local)
- int **smi_bgp_api_distance_config_set_wrap** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, u_int8_t distance_ebgp, u_int8_t distance_ibgp, u_int8_t distance_local)
- int **smi_bgp_api_distance_config_unset_wrap_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, u_int8_t distance_ebgp, u_int8_t distance_ibgp, u_int8_t distance_local)
- int **smi_bgp_api_distance_config_unset_wrap** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, u_int8_t distance_ebgp, u_int8_t distance_ibgp, u_int8_t distance_local)
- s_int32_t **smi_neighbor_attr_unchanged_as_path_af_set_wrap_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, char *peer_str, int afi, int safi, bool_t neighborAttrUnchangedAsPath)
- s_int32_t **smi_neighbor_attr_unchanged_as_path_af_set_wrap** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, char *peer_str, int afi, int safi, bool_t neighborAttrUnchangedAsPath)
- int **smi_bgp_timers_unset_sdkapi_wrap_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, u_int16_t keepalive, u_int16_t holdtime)
- int **smi_bgp_timers_unset_sdkapi_wrap** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, u_int16_t keepalive, u_int16_t holdtime)
- int **smi_peer_maximum_prefix_unset_wrap_validate** (struct smiclient_globals *azg, u_int32_t as, char *peer_id, u_int32_t max, u_int32_t vr_id)
- int **smi_peer_maximum_prefix_unset_wrap** (struct smiclient_globals *azg, u_int32_t as, char *peer_id, u_int32_t max, u_int32_t vr_id)
- int **smi_peer_maximum_prefix_threshold_set_validate** (struct smiclient_globals *azg, u_int32_t as, char *peer_id, u_int32_t max, u_int32_t threshold, u_int32_t vr_id)
- int **smi_peer_maximum_prefix_threshold_set** (struct smiclient_globals *azg, u_int32_t as, char *peer_id, u_int32_t max, u_int32_t threshold, u_int32_t vr_id)
- int **smi_peer_maximum_prefix_threshold_unset_validate** (struct smiclient_globals *azg, u_int32_t as, char *peer_id, u_int32_t max, u_int32_t threshold, u_int32_t vr_id)

- int **smi_peer_maximum_prefix_threshold_unset** (struct smiclient_globals *azg, u_int32_t as, char *peer_id, u_int32_t max, u_int32_t threshold, u_int32_t vr_id)
- s_int32_t **smi_bgp_addr_family_aggregate_addr_unset_wrap** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, char *aggregateAddr, int afi, int safi, int aggregateTypeAf)
- s_int32_t **smi_bgp_addr_family_aggregate_addr_unset_wrap_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, char *aggregateAddr, int afi, int safi, int aggregateTypeAf)
- s_int32_t **smi_bgp_aggregate_addr_unset_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, char *aggregateAddr)
- s_int32_t **smi_bgp_aggregate_addr_unset** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, char *aggregateAddr)
- s_int32_t **smi_bgp_aggregate_addr_set_wrap_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, char *aggregateAddr, bool_t aggregateAsSet, bool_t aggregateSummOnly)
- s_int32_t **smi_bgp_aggregate_addr_set_wrap** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, char *aggregateAddr, bool_t aggregateAsSet, bool_t aggregateSummOnly)
- s_int32_t **smi_bgp_addr_family_aggregate_addr_set_wrap_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, char *aggregateAddrAf, int afi, int safi, bool_t aggregateAsSetAf, bool_t aggregateSummOnlyAf)
- s_int32_t **smi_bgp_addr_family_aggregate_addr_set_wrap** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, char *aggregateAddrAf, int afi, int safi, bool_t aggregateAsSetAf, bool_t aggregateSummOnlyAf)
- s_int32_t **smi_bgp_addr_family_aggregate_addr_unset_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, int afi, int safi, char *aggregateAddrAf)
- s_int32_t **smi_bgp_addr_family_aggregate_addr_unset** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, int afi, int safi, char *aggregateAddrAf)
- int **smi_bgp_aspath_access_list_unset_wrap_validate** (struct smiclient_globals *azg, u_int32_t vr_id, char *list_name, char *bgpRegExp, int accessListDirection)
- int **smi_bgp_aspath_access_list_unset_wrap** (struct smiclient_globals *azg, u_int32_t vr_id, char *list_name, char *bgpRegExp, int accessListDirection)
- int **smi_bgp_static_network_unset_wrap_validate** (struct smiclient_globals *azg, u_int32_t as, char *ip_str, u_int32_t backdoor, char *rmap_name, u_int32_t vr_id)
- int **smi_bgp_static_network_unset_wrap** (struct smiclient_globals *azg, u_int32_t as, char *ip_str, u_int32_t backdoor, char *rmap_name, u_int32_t vr_id)
- int **smi_bgp_addr_family_dampening_half_life_unset_wrap_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, int afi, int safi, u_int32_t halfLife)
- int **smi_bgp_addr_family_dampening_half_life_unset_wrap** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, int afi, int safi, u_int32_t halfLife)

- int **smi_bgp_addr_family_dampening_half_life_set_wrap_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, int afi, int safi, u_int32_t halfLife)
- int **smi_bgp_addr_family_dampening_half_life_set_wrap** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, int afi, int safi, u_int32_t halfLife)
- int **smi_bgp_addr_family_dampening_half_life_reuse_supress_maxsuppress_unset_wrap_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, int afi, int safi, u_int32_t halfLife, u_int32_t reusePenalty, u_int32_t suppressPenalty, u_int32_t maxSuppress)
- int **smi_bgp_addr_family_dampening_half_life_reuse_supress_maxsuppress_unset_wrap** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, int afi, int safi, u_int32_t halfLife, u_int32_t reusePenalty, u_int32_t suppressPenalty, u_int32_t maxSuppress)
- int **smi_bgp_addr_family_dampening_half_life_reuse_supress_maxsuppress_set_wrap_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, int afi, int safi, u_int32_t halfLife, u_int32_t reusePenalty, u_int32_t suppressPenalty, u_int32_t maxSuppress)
- int **smi_bgp_addr_family_dampening_half_life_reuse_supress_maxsuppress_set_wrap** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, int afi, int safi, u_int32_t halfLife, u_int32_t reusePenalty, u_int32_t suppressPenalty, u_int32_t maxSuppress)
- int **smi_bgp_addr_family_dampening_half_life_reuse_supress_maxsuppress_unreachability_unset_wrap_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, int afi, int safi, u_int32_t halfLife, u_int32_t reusePenalty, u_int32_t suppressPenalty, u_int32_t maxSuppress, u_int32_t unreachHalfLife)
- int **smi_bgp_addr_family_dampening_half_life_reuse_supress_maxsuppress_unreachability_unset_wrap** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, int afi, int safi, u_int32_t halfLife, u_int32_t reusePenalty, u_int32_t suppressPenalty, u_int32_t maxSuppress, u_int32_t unreachHalfLife)
- int **smi_bgp_addr_family_dampening_half_life_reuse_supress_maxsuppress_unreachability_set_wrap_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, int afi, int safi, u_int32_t halfLife, u_int32_t reusePenalty, u_int32_t suppressPenalty, u_int32_t maxSuppress, u_int32_t unreachHalfLife)
- int **smi_bgp_addr_family_dampening_half_life_reuse_supress_maxsuppress_unreachability_set_wrap** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, int afi, int safi, u_int32_t halfLife, u_int32_t reusePenalty, u_int32_t suppressPenalty, u_int32_t maxSuppress, u_int32_t unreachHalfLife)
- int **smi_bgp_addr_family_dampening_routemap_name_unset_wrap_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, int afi, int safi, char *rmapName)
- int **smi_bgp_addr_family_dampening_routemap_name_unset_wrap** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, int afi, int safi, char *rmapName)

- int **smi_bgp_addr_family_dampening_routemap_name_set_wrap_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, int afi, int safi, char *rmapName)
- int **smi_bgp_addr_family_dampening_routemap_name_set_wrap** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, int afi, int safi, char *rmapName)
- int **smi_bgp_dampening_half_life_unset_wrap_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, u_int32_t halfLife)
- int **smi_bgp_dampening_half_life_unset_wrap** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, u_int32_t halfLife)
- int **smi_bgp_dampening_half_life_set_wrap_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, u_int32_t halfLife)
- int **smi_bgp_dampening_half_life_set_wrap** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, u_int32_t halfLife)
- int **smi_bgp_dampening_half_life_reuse_suppress_maxsuppress_unset_wrap_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, u_int32_t halfLife, u_int32_t reusePenalty, u_int32_t suppressPenalty, u_int32_t maxSuppress)
- int **smi_bgp_dampening_half_life_reuse_suppress_maxsuppress_unset_wrap** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, u_int32_t halfLife, u_int32_t reusePenalty, u_int32_t suppressPenalty, u_int32_t maxSuppress)
- int **smi_bgp_dampening_half_life_reuse_suppress_maxsuppress_set_wrap_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, u_int32_t halfLife, u_int32_t reusePenalty, u_int32_t suppressPenalty, u_int32_t maxSuppress)
- int **smi_bgp_dampening_half_life_reuse_suppress_maxsuppress_set_wrap** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, u_int32_t halfLife, u_int32_t reusePenalty, u_int32_t suppressPenalty, u_int32_t maxSuppress)
- int **smi_bgp_dampening_half_life_reuse_suppress_maxsuppress_unreachability_unset_wrap_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, u_int32_t halfLife, u_int32_t reusePenalty, u_int32_t suppressPenalty, u_int32_t maxSuppress, u_int32_t unreachHalfLife)
- int **smi_bgp_dampening_half_life_reuse_suppress_maxsuppress_unreachability_unset_wrap** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, u_int32_t halfLife, u_int32_t reusePenalty, u_int32_t suppressPenalty, u_int32_t maxSuppress, u_int32_t unreachHalfLife)
- int **smi_bgp_dampening_half_life_reuse_suppress_maxsuppress_unreachability_set_wrap_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, u_int32_t halfLife, u_int32_t reusePenalty, u_int32_t suppressPenalty, u_int32_t maxSuppress, u_int32_t unreachHalfLife)
- int **smi_bgp_dampening_half_life_reuse_suppress_maxsuppress_unreachability_set_wrap** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, u_int32_t halfLife, u_int32_t reusePenalty, u_int32_t suppressPenalty, u_int32_t maxSuppress, u_int32_t unreachHalfLife)
- int **smi_bgp_dampening_routemap_name_unset_wrap_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, char *rmapName)
- int **smi_bgp_dampening_routemap_name_unset_wrap** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, char *rmapName)

- int **smi_bgp_dampening_routemap_name_set_wrap_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, char *rmapName)
- int **smi_bgp_dampening_routemap_name_set_wrap** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t as, char *rmapName)
- int **smi_bgp_bestpath_as_path_multipath_relax_wrap_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, u_int8_t multipathRelax)
- int **smi_bgp_bestpath_as_path_multipath_relax_wrap** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, u_int8_t multipathRelax)
- int **smi_bgp_bestpath_as_path_multipath_relax_set** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs)
- int **smi_bgp_bestpath_as_path_multipath_relax_unset** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs)
- s_int32_t **smi_bgp_vrf_address_family_set_wrap_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, int afi, int safi, char *vrfName)
- int **smi_bgp_vrf_address_family_set_wrap** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, int afi, int safi, char *vrfName)
- s_int32_t **smi_bgp_vrf_address_family_unset_wrap_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, int afi, int safi, char *vrfName)
- int **smi_bgp_vrf_address_family_unset_wrap** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, int afi, int safi, char *vrfName)
- s_int32_t **smi_bgp_address_family_unset_wrap_validate** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, int afi, int safi, char *vrfName)
- int **smi_bgp_address_family_unset_wrap** (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, int afi, int safi, char *vrfName)
- int **smi_bgp_maximum_paths_unset_wrap_validate** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t as, int type, int multipathsNum)
- int **smi_bgp_maximum_paths_unset_wrap** (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t as, int type, int multipathsNum)

2.1.1 Detailed Description

Provides APIs for managing BGP Protocol. The Border Gateway Protocol (BGP) is primarily used as the routing protocol for the Internet. BGP shares routing information between various autonomous systems (ASs). BGP is typically an inter-AS routing protocol. The intra-AS routing is usually handled by the IGP (for example, OSPF, RIP). BGP requires the routes to be reachable before it can advertise it to another AS. The routes are advertised to the other ASs through network layer reachability information.

2.1.2 Function Documentation

2.1.2.1 `int smi_bgp4_get_path_attr_aggregator_addr_sdkapi (struct smiclient_globals * azg, u_int32_t vrId, int bgpProcId, struct prefix_ipv4 * routeAddr, struct pal_in4_addr * peerAddr, struct pal_in4_addr * pathAttrAggregatorAddr)`

The IP address of the last BGP4 speaker that performed route aggregation. A value of 0.0.0.0 indicates the absence of this attribute. `smi_bgp4_get_path_attr_aggregator_addr_sdkapi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *routeAddr* The route address
- ← *peerAddr* The peer address The peer IP address
- *pathAttrAggregatorAddr* The IP address of the router

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.2 `int smi_bgp4_get_path_attr_aggregator_as_sdkapi (struct smiclient_globals * azg, u_int32_t vrId, int bgpProcId, struct prefix_ipv4 * routeAddr, struct pal_in4_addr * peerAddr, int * pathAttrAggregatorAs)`

The AS number of the last BGP4 speaker that performed route aggregation. A value of zero (0) indicates the absence of this attribute. `smi_bgp4_get_path_attr_aggregator_as_sdkapi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *routeAddr* The route address
- ← *peerAddr* The peer address The peer IP address
- *pathAttrAggregatorAs* AS number

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.3 `int smi_bgp4_get_path_attr_atomic_aggregate_sdkapi (struct smiclient_globals * azg, u_int32_t vrId, int bgpProcId, struct prefix_ipv4 * routeAddr, struct pal_in4_addr * peerAddr, int * atomic)`

This function returns the pointer to the specified BGP instance. If no pointer is returned, it tries to create a new one. ATOMIC_AGGREGATE is a primarily informational attribute. `smi_bgp4_get_path_attr_atomic_aggregate_sdkapi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *routeAddr* The address of the routing process
- ← *peerAddr* The peer address The peer IP address
- *atomic* The value

Returns:

BGP_API_GET_SUCCESS on success

2.1.2.4 `int smi_bgp4_get_path_attr_best_sdkapi (struct smiclient_globals * azg, u_int32_t vrId, int bgpProcId, struct prefix_ipv4 * routeAddr, struct pal_in4_addr * peerAddr, int * pathAttrBest)`

An indication of whether this route was chosen as the best BGP4 route for this destination. `smi_bgp4_get_path_attr_best_sdkapi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *routeAddr* The address of the routing process
- ← *peerAddr* The peer address The peer IP address
- *best* The boolean value

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.5 `int smi_bgp4_get_path_attr_calc_local_pref_sdkapi (struct smiclient_globals * azg, u_int32_t vrId, int bgpProcId, struct prefix_ipv4 * routeAddr, struct pal_in4_addr * peerAddr, int * localPref)`

The degree of preference calculated by the receiving BGP4 speaker for an advertised route. A value of -1 indicates the absence of this attribute. `smi_bgp4_get_path_attr_calc_local_pref_sdkapi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *routeAddr* The route address
- ← *peerAddr* The peer address The peer IP address
- *localPref* The degree of local preference

Returns:

- BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
- BGP_API_GET_ERROR

2.1.2.6 `int smi_bgp4_get_path_attr_ip_addr_prefix_len_sdkapi (struct smiclient_globals * azg, u_int32_t vrId, int bgpProcId, struct prefix_ipv4 * routeAddr, struct pal_in4_addr * peerAddr, int * pathAttrPeerLen)`

Length in bits of the IP address prefix in the Network Layer Reachability Information field. `smi_bgp4_get_path_attr_ip_addr_prefix_len_sdkapi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *routeAddr* Route address
- ← *peerAddr* The peer address The peer address
- *pathAttrPeerLen* prefix length of peer address

Returns:

- BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
- BGP_API_GET_ERROR

2.1.2.7 `int smi_bgp4_get_path_attr_ip_addr_prefix_sdkapi (struct smiclient_globals * azg, u_int32_t vrId, int bgpProcId, struct prefix_ipv4 * routeAddr, struct pal_in4_addr * peerAddr, struct pal_in4_addr * pathAttrIpAddrPrefix)`

An IP address prefix in the Network Layer Reachability Information field. This object is an IP address containing the prefix with length specified by `bgp4PathAttrIpAddrPrefixLen`. Any bits beyond the length specified by `bgp4PathAttrIpAddrPrefixLen` are zeroed. `smi_bgp4_get_path_attr_ip_addr_prefix_sdkapi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *routeAddr* The route address
- ← *peerAddr* The peer address The peer address
- *pathAttrIpAddrPrefix* The IP address prefix

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.8 `int smi_bgp4_get_path_attr_local_pref_sdkapi (struct smiclient_globals * azg, u_int32_t vrId, int bgpProcId, struct prefix_ipv4 * routeAddr, struct pal_in4_addr * peerAddr, int * localPref)`

The originating BGP4 speaker's degree of preference for an advertised route. A value of -1 indicates the absence of this attribute. `smi_bgp4_get_path_attr_local_pref_sdkapi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *routeAddr* The address of the routing process
- ← *peerAddr* The peer address The peer IP address
- *pref* The originating BGP4 speaker's degree of preference.

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.9 `int smi_bgp4_get_path_attr_multi_exit_disc_sdkapi (struct smiclient_globals * azg, u_int32_t vrId, int bgpProcId, struct prefix_ipv4 * routeAddr, struct pal_in4_addr * peerAddr, int * med)`

This metric is used to discriminate between multiple exit points to an adjacent autonomous system. A value of -1 indicates the absence of this attribute. `smi_bgp4_get_path_attr_multi_exit_disc_sdkapi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *routeAddr* The route address
- ← *peerAddr* The peer address The peer IP address
- *med* The metric

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.10 `int smi_bgp4_get_path_attr_next_hop_sdkapi (struct smiclient_globals * azg, u_int32_t vrId, int bgpProcId, struct prefix_ipv4 * routeAddr, struct pal_in4_addr * peerAddr, struct pal_in4_addr * pathAttrNextHop)`

The address of the border router that should be used for the destination network. This address is the NEXT_HOP address received in the UPDATE packet. `smi_bgp4_get_path_attr_next_hop_sdkapi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *routeAddr* The route address
- ← *peerAddr* The peer address The peer IP address
- *pathAttrNextHop* The address of the border router

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.11 `int smi_bgp4_get_path_attr_origin_sdkapi (struct smiclient_globals * azg, u_int32_t vrId, int bgpProcId, struct prefix_ipv4 * routeAddr, struct pal_in4_addr * peerAddr, int * origin)`

The ultimate origin of the path information. `smi_bgp4_get_path_attr_origin_sdkapi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *routeAddr* The route address
- ← *peerAddr* The peer address The peer IP address
- *origin* The path information origin

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.12 `int smi_bgp4_get_path_attr_peer_sdkapi (struct smiclient_globals * azg, u_int32_t vrId, int bgpProcId, struct prefix_ipv4 * routeAddr, struct pal_in4_addr * peerAddr, struct pal_in4_addr * pathAttrPeerAddr)`

The IP address of the peer where the path information was learned. `smi_bgp4_get_path_attr_peer_sdkapi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *routeAddr* The route address
- ← *peerAddr* The peer address The peer address
- *pathAttrPeer* The local address of the peers BGP

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.13 `int smi_bgp_address_family_set (struct smiclient_globals * azg, u_int32_t vrId, int afi, int safi)`

Sets the BGP af_flag. smi_bgp_address_family_set

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *afi* Address family identifier Address family identifier (1-ipv4/2-ipv6/3-vpnv4/4-vpnv6)
- ← *safi* Sub-address family identifier Sub-Address family identifier (1-unicast/2-multiicast)
- ← *vrId* Virtual router-id

Returns:

BGP_API_SET_SUCCESS on success, otherwise error codes: SMI_ERROR
 BGP_API_SET_ERR_INVALID_AF
 BGP_ERR_INVALID_SAFI
 BGP_API_ERR_BGP_DEFAULT_LOOKUP_FAIL

2.1.2.14 `int smi_bgp_af_config_check_sdkapi (struct smiclient_globals * azg, u_int32_t vrId, int af_id, int subaf_id, u_int32_t afFlag)`

This API checks if the BGP address-family flag is configured. smi_bgp_af_config_check_sdkapi

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual router id
- ← *af_id* Address family <1-IPv4/2-IPv6>
- ← *subaf_id* Sub-address family <1-Unicast/2-Multicast>
- ← *afFlag* BGP address-family flag

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes

2.1.2.15 `int smi_bgp_aggregate_nexthop_check_set (struct smiclient_globals * azg, u_int32_t vrId)`

Sets the check for bgp aggregate_nexthop. smi_bgp_aggregate_nexthop_check_set

Parameters:

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

← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.16 int smi_bgp_aggregate_nexthop_check_set_validate (struct smiclient_globals * *azg*, u_int32_t *vrId*)

Sets the check for bgp aggregate_nexthop. smi_bgp_aggregate_nexthop_check_set_validate

Parameters:

← *azg* Pointer to the SMI client global structure
← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.17 int smi_bgp_aggregate_nexthop_check_unset (struct smiclient_globals * *azg*, u_int32_t *vrId*)

Unsets the check for bgp aggregate_nexthop. smi_bgp_aggregate_nexthop_check_unset

Parameters:

← *azg* Pointer to the SMI client global structure
← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.18 int smi_bgp_aggregate_nexthop_check_unset_validate (struct smiclient_globals * *azg*, u_int32_t *vrId*)

Unsets the check for bgp aggregate_nexthop. smi_bgp_aggregate_nexthop_check_unset_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.19 `s_int32_t smi_bgp_always_compare_med_set (struct smiclient_globals * azg, u_int32_t vrId)`

Sets the bgp always compare. `smi_bgp_always_compare_med_set`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.20 `s_int32_t smi_bgp_always_compare_med_set_validate (struct smiclient_globals * azg, u_int32_t vrId)`

Sets the bgp always compare. `smi_bgp_always_compare_med_set_validate`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.21 `s_int32_t smi_bgp_always_compare_med_unset (struct smiclient_globals * azg, u_int32_t vrId)`

Unsets the bgp always compare. `smi_bgp_always_compare_med_unset`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.22 s_int32_t smi_bgp_always_compare_med_unset_validate (struct smiclient_globals * *azg*, u_int32_t *vrId*)

Unsets the bgp always compare. smi_bgp_always_compare_med_unset_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.23 int smi_bgp_api_address_family_unset (struct smiclient_globals * *azg*, u_int32_t *vrId*, int *afi*, int *safi*)

Unsets the BGP af_flag. smi_bgp_address_family_unset

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual router-id
- ← *afi* Address family identifier Address family identifier (1-ipv4/2-ipv6/3-vpnv4/4-vpnv6)
- ← *safi* Sub-address family identifier Sub-Address family identifier (1-unicast/2-multicast)

Returns:

BGP_API_SET_SUCCESS on success, otherwise error codes: SMI_ERROR
 BGP_API_SET_ERR_INVALID_AF
 BGP_ERR_INVALID_SAFI
 BGP_API_ERR_BGP_DEFAULT_LOOKUP_FAIL

2.1.2.24 `int smi_bgp_aspath_access_list_set_validate (struct smiclient_globals * azg, u_int32_t vrId, char * accessListName, char * bgpRegExp, int action)`

Unconfigure BGP Autonomous System path filtering defined by the regular expression.
smi_bgp_aspath_access_list_set

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *accessListName* AS path access list name
- ← *regExp* Regular expression to match BGP AS paths
- ← *action* Action type <0-1> 0 - Deny
1 - Permit
- ← *vrId* Virtual router id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
BGP_API_SET_ERR_INVALID_VALUE
BGP_API_SET_ERR_REGEX_COMPILE_FAIL

2.1.2.25 `int smi_bgp_aspath_access_list_unset_validate (struct smiclient_globals * azg, u_int32_t vrId, char * accessListName)`

Unconfigure BGP Autonomous System path filtering defined by the regular expression.
smi_bgp_aspath_access_list_unset

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *accessListName* AS path access list name
- ← *vrId* Virtual router id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
BGP_API_SET_ERR_INVALID_VALUE
BGP_API_SET_ERR_REGEX_COMPILE_FAIL

2.1.2.26 `s_int32_t smi_bgp_auto_summary_update_set_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, bool_t autoSummary)`

Enables automatic network number summarization. smi_bgp_auto_summary_update_set_sdkapi

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *afi* Address family identifier Address family identifier <1-2> (1=IP | 2=IP6)
- ← *safi* Sub-address family identifier Sub-Address family identifier <1-4> (UC|MC|UC_MC|MPLS)
- ← *autoSummary* Yes/No flag
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/VRF name)

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_INVALID_ROUTE_NODE
 BGP_API_SET_ERR_AUTO_SUMMARY_ENABLED
 BGP_API_SET_ERR_AUTO_SUMMARY_DISABLED

2.1.2.27 s_int32_t smi_bgp_bestpath_aspath_ignore_set (struct smiclient_globals * azg, u_int32_t vrId)

Sets the bgp bestpath as path ignore. smi_bgp_bestpath_aspath_ignore_set

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.28 s_int32_t smi_bgp_bestpath_aspath_ignore_set_validate (struct smiclient_globals * azg, u_int32_t vrId)

Sets the bgp bestpath as path ignore. smi_bgp_bestpath_aspath_ignore_set_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.29 `s_int32_t smi_bgp_bestpath_aspath_ignore_unset (struct smiclient_globals * azg, u_int32_t vrId)`

Unsets the bgp bestpath as path ignore. `smi_bgp_bestpath_aspath_ignore_unset`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.30 `s_int32_t smi_bgp_bestpath_aspath_ignore_unset_validate (struct smiclient_globals * azg, u_int32_t vrId)`

Unsets the bgp bestpath as path ignore. `smi_bgp_bestpath_aspath_ignore_unset_validate`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.31 `s_int32_t smi_bgp_bestpath_compare_confed_aspath_set (struct smiclient_globals * azg, u_int32_t vrId)`

Sets the bgp bestpath compare confed as path. `smi_bgp_bestpath_compare_confed_aspath_set`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.32 `s_int32_t smi_bgp_bestpath_compare_confed_aspath_set_validate` (`struct smiclient_globals * azg, u_int32_t vrId`)

Sets the bgp bestpath compare confed as path. `smi_bgp_bestpath_compare_confed_aspath_set_validate`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.33 `s_int32_t smi_bgp_bestpath_compare_confed_aspath_unset` (`struct smiclient_globals * azg, u_int32_t vrId`)

Sets the bgp bestpath compare confed as path. `smi_bgp_bestpath_compare_confed_aspath_unset`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.34 `s_int32_t smi_bgp_bestpath_compare_confed_aspath_unset_validate` (`struct smiclient_globals * azg, u_int32_t vrId`)

Sets the bgp bestpath compare confed as path. `smi_bgp_bestpath_compare_confed_aspath_unset_validate`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.35 `s_int32_t smi_bgp_bestpath_compare_router_id_set (struct smiclient_globals * azg, u_int32_t vrId)`

Sets the router-id for bgp bestpath compare. `smi_bgp_bestpath_compare_router_id_set`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.36 `s_int32_t smi_bgp_bestpath_compare_router_id_set_validate (struct smiclient_globals * azg, u_int32_t vrId)`

Sets the router-id for bgp bestpath compare. `smi_bgp_bestpath_compare_router_id_set_validate`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.37 `s_int32_t smi_bgp_bestpath_compare_router_id_unset (struct smiclient_globals * azg, u_int32_t vrId)`

Unsets the router-id for bgp bestpath compare. `smi_bgp_bestpath_compare_router_id_unset`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.38 s_int32_t smi_bgp_bestpath_compare_router_id_unset_validate (struct smiclient_globals * azg, u_int32_t vrId)

Unsets the router-id for bgp bestpath compare. smi_bgp_bestpath_compare_router_id_unset_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.39 s_int32_t smi_bgp_bestpath_dont_compare_originator_id_set (struct smiclient_globals * azg, u_int32_t vrId)

Sets the bestpath dont compare originator. smi_bgp_bestpath_dont_compare_originator_id_set

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.40 s_int32_t smi_bgp_bestpath_dont_compare_originator_id_set_validate (struct smiclient_globals * azg, u_int32_t vrId)

Sets the bestpath dont compare originator. smi_bgp_bestpath_dont_compare_originator_id_set_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.41 `s_int32_t smi_bgp_bestpath_dont_compare_originator_id_unset` (`struct smiclient_globals * azg, u_int32_t vrId`)

Unsets the bestpath dont compare originator. `smi_bgp_bestpath_dont_compare_originator_id_unset`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.42 `s_int32_t smi_bgp_bestpath_dont_compare_originator_id_unset_validate` (`struct smiclient_globals * azg, u_int32_t vrId`)

Unsets the bestpath dont compare originator. `smi_bgp_bestpath_dont_compare_originator_id_unset_validate`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.43 `s_int32_t smi_bgp_bestpath_med_set` (`struct smiclient_globals * azg, u_int32_t vrId, char * medType`)

sets bestpath med `smi_bgp_bestpath_med_set`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual router-id (Default-0)
- ← *medType* Bestpath med option (remove-recv-med|remove-send-med|confed)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.44 s_int32_t smi_bgp_bestpath_med_set_validate (struct smiclient_globals * *azg*, u_int32_t *vrId*, char * *medType*)

sets bestpath med smi_bgp_bestpath_med_set_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual router-id (Default-0)
- ← *medType* Bestpath med option (remove-recv-med|remove-send-med|confed)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.45 s_int32_t smi_bgp_bestpath_med_unset (struct smiclient_globals * *azg*, u_int32_t *vrId*, char * *medType*)

unsets bestpath med smi_bgp_bestpath_med_unset

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual router-id (Default-0)
- ← *medType* Bestpath med option (remove-recv-med|remove-send-med|confed)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.46 s_int32_t smi_bgp_bestpath_med_unset_validate (struct smiclient_globals * *azg*, u_int32_t *vrId*, char * *medType*)

unsets bestpath med smi_bgp_bestpath_med_unset_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual router-id (Default-0)
- ← *medType* Bestpath med option (remove-recv-med|remove-send-med|confed)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.47 s_int32_t smi_bgp_bestpath_tie_break_on_age_set (struct smiclient_globals * azg, u_int32_t vrId)

Sets the bestpath tie break on age. smi_bgp_bestpath_tie_break_on_age_set

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.48 s_int32_t smi_bgp_bestpath_tie_break_on_age_set_validate (struct smiclient_globals * azg, u_int32_t vrId)

Sets the bestpath tie break on age. smi_bgp_bestpath_tie_break_on_age_set_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.49 s_int32_t smi_bgp_bestpath_tie_break_on_age_unset (struct smiclient_globals * azg, u_int32_t vrId)

Unsets the bestpath tie break on age. smi_bgp_bestpath_tie_break_on_age_unset

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.50 `s_int32_t smi_bgp_bestpath_tie_break_on_age_unset_validate (struct smiclient_globals * azg, u_int32_t vrId)`

Unsets the bestpath tie break on age. `smi_bgp_bestpath_tie_break_on_age_unset_validate`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.51 `int smi_bgp_check_instance (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t * as, char * bgpName)`

This API checks if the BGP instance is enabled. `smi_bgp_check_instance`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual router id
- ← *as* Autonomous System number
- ← *bgpName* BGP view name

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes

2.1.2.52 `int smi_bgp_clear_gen_sdkapi (struct smiclient_globals * azg, u_int32_t vrId, char * name, int afi, int safi, int sort, s_int32_t stype, char * clearString)`

Clear BGP connections. `smi_bgp_clear_gen_sdkapi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *name* BGP instance name: optional, usually NULL
- ← *afi* Address family identifier Address family identifier <1-2> (1=IP | 2=IP6)
- ← *safi* Sub-address family identifier Sub-Address family identifier <1-4> (UC|MC|UC_MC|MPLS)

- ← **sort** Clear command type <0-5> 0 - clear_all Clear all neighbors 1 - clear_peer Clear given neighbors 2 - clear_group Clear all peer-group members 3 - clear_external Clear external BGP connections 4 - clear_as Clear BGP connections of AS 5 - clear_rfd
- ← **stype** Clear flag <1-5> (None|Out|In|Both|InPerf)
- ← **clearString** Sting contains AS number <1-65535> or Peer address or Group address
- ← **vrId** Virtual router id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_ERR_BGP_NAME_LOOKUP_FAIL
 BGP_API_ERR_BGP_DEFAULT_LOOKUP_FAIL
 BGP_API_SET_ERROR
 BGP_API_SET_ERR_UNKNOWN_OBJECT
 BGP_API_SET_ERR_AF_UNCONFIGURED
 BGP_API_SET_ERR_INVALID_VALUE
 BGP_API_SET_ERR_SOFT_RECONFIG_UNCONFIGURED

2.1.2.53 int smi_bgp_cluster_id_digit_set_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, u_int32_t clusterIdDigit)

Sets the BGP Route-Reflector Cluster-id as in 32 bit quantity. smi_bgp_cluster_id_digit_set_sdkapi

Parameters:

- ← **azg** Pointer to the SMI client global structure
- ← **clusterId** BGP route reflector cluster ID <1-4294967295>
- ← **vrId** Virtual router id

Returns:

BGP_API_SET_SUCCESS on success.

2.1.2.54 int smi_bgp_cluster_id_set_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, char * clusterId)

Sets the BGP Route-Reflector Cluster-id as in IP address format. smi_bgp_cluster_id_set_sdkapi

Parameters:

- ← **azg** Pointer to the SMI client global structure
- ← **clusterId** BGP route reflector cluster ID

← *vrId* Virtual router id

Returns:

BGP_API_SET_SUCCESS on success.

2.1.2.55 int smi_bgp_cluster_id_unset_sdkapi_validate (struct smiclient_globals * *azg*, u_int32_t *vrId*, u_int32_t *bgpAs*)

Deletes the BGP Route-Reflector Cluster-id. smi_bgp_cluster_id_unset_sdkapi

Parameters:

← *azg* Pointer to the SMI client global structure

← *vrId* Virtual router id

Returns:

BGP_API_SET_SUCCESS on success.

2.1.2.56 int smi_bgp_community_list_entry_unset (struct smiclient_globals * *azg*, u_int32_t *vrId*, char * *commListName*, char * *commListValue*, int *nameType*, int *action*, int *entryType*)

Unconfigure BGP community filtering. smi_bgp_community_list_entry_unset

Parameters:

← *azg* Pointer to the SMI client global structure

← *commListName* Community list name

← *commListValue* Community value string

← *nameType* List name type <0-1> 0 - COMMUNITY_LIST_STRING
1 - COMMUNITY_LIST_NUMBER

← *action* Action type <0-1> 0 - Deny
1 - Permit

← *entryType* List entry type <0-2> 0 - COMMUNITY_LIST_STANDARD
1 - COMMUNITY_LIST_EXPANDED
2 - COMMUNITY_LIST_AUTO

← *vrId* Virtual router id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
BGP_API_SET_ERR_INVALID_VALUE
BGP_API_SET_ERROR
BGP_API_SET_ERR_MALFORMED_ARG

2.1.2.57 `int smi_bgp_community_list_set (struct smiclient_globals * azg,
u_int32_t vrId, char * commListName, char * commListValue, int
nameType, int action, int entryType)`

Configure BGP community filtering. `smi_bgp_community_list_set`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *commListName* Community list name
- ← *commListValue* Community value string
- ← *nameType* List name type <0-1> 0 - COMMUNITY_LIST_STRING
1 - COMMUNITY_LIST_NUMBER
- ← *action* Action type <0-1> 0 - Deny
1 - Permit
- ← *entryType* List entry type <0-2> 0 - COMMUNITY_LIST_STANDARD
1 - COMMUNITY_LIST_EXPANDED
2 - COMMUNITY_LIST_AUTO
- ← *vrId* Virtual router id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
BGP_API_SET_ERR_INVALID_VALUE
BGP_API_SET_ERROR
BGP_API_SET_ERR_CLIST_DEFINE_CONFLICT
BGP_API_SET_ERR_MALFORMED_ARG

2.1.2.58 `int smi_bgp_community_list_unset_validate (struct smiclient_globals *
azg, u_int32_t vrId, char * commListName)`

Unconfigure BGP community filtering. `smi_bgp_community_list_unset`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *commListName* Community list name
- ← *vrId* Virtual router id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
BGP_API_SET_ERR_UNKNOWN_OBJECT
BGP_API_SET_ERR_MALFORMED_ARG

2.1.2.59 int smi_bgp_confederation_id_set_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, int confedId)

Sets the AS Confederation identifier of BGP confederations. BGP Confederations is used to create a confederation of autonomous systems that is represented as a single autonomous system to BGP peers external to the confederation, thereby removing the "full mesh" requirement. smi_bgp_confederation_id_set_sdkapi

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *confedId* Confederation ID <1-65535>
- ← *vrId* Virtual router id
- ← *vrName* VRF name (default/VRF name)

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
BGP_API_SET_ERR_INVALID_AS

2.1.2.60 int smi_bgp_confederation_id_unset_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs)

Deletes the AS Confederation identifier of BGP confederations . smi_bgp_confederation_id_unset_sdkapi

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual router id
- ← *vrName* VRF name (default/VRF name)

Returns:

BGP_API_SET_SUCCESS on success.

2.1.2.61 int smi_bgp_confederation_peer_check_sdkapi (struct smiclient_globals * azg, u_int32_t vrId, int as)

This API checks if the BGP peer confederation information is configured. smi_bgp_confederation_peer_check_sdkapi

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual router id
- ← *as* Autonomous System number

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes

2.1.2.62 **int smi_bgp_confederation_peers_add_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, int confedId)**

Adds a Peer Member-AS Number of BGP confederation. smi_bgp_confederation_peers_add_sdkapi

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *confedId* Confederation ID <1-65535>
- ← *vrId* Virtual router id
- ← *vrName* VRF name (default/VRF name)

Returns:

BGP_API_SET_SUCCESS
 BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_INVALID_BGP
 BGP_API_SET_ERR_INVALID_AS

2.1.2.63 **int smi_bgp_confederation_peers_remove_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, int confedId)**

Deletes a Peer Member-AS Number of BGP confederation. smi_bgp_confederation_peers_remove_sdkapi

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *confedId* Confederation ID <1-65535>
- ← *vrId* Virtual router id
- ← *vrName* VRF name (default/VRF name)

Returns:

BGP_API_SET_SUCCESS on success.

2.1.2.64 **s_int32_t smi_bgp_create_instance_set_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs)**

Gets the BGP instance of given Autonomous System number if already exists or Creates new instance. smi_bgp_create_instance_set_sdkapi

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *as* AS number
- ← *bgpName* BGP instance name: optional, usually NULL
- ← *vrId* Virtual router id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_MULTIPLE_INSTANCE_NOT_SET
 BGP_API_SET_ERR_AS_MISMATCH
 BGP_API_SET_ERR_INSTANCE_MISMATCH
 BGP_API_NHT_NOT_ENABLED_SET_ERR
 BGP_API_SET_ERR_DEFAULTINS_FOR_SAMEPEER BGP_API_SET_-
 ERROR

2.1.2.65 int smi_bgp_debug_validate (struct smiclient_globals * azg, int debugFlag, u_int32_t vrId)

Use this function to enable all BGP troubleshooting functions. smi_bgp_debug

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *debugFlag* Pass debug flag as following:
 - SMI_BGP_DBG_ALL - For all debugging for BGP
 - SMI_BGP_DBG_NHT - Specify debugging for BGP NHT
 - SMI_BGP_DBG_NSM - Specify debugging for NSM messages
 - SMI_BGP_DBG_FSM - Specify debugging for BGP Finite State Machine (FSM)
 - SMI_BGP_DBG_EVENTS - Specify debugging for BGP events
 - SMI_BGP_DBG_FILTER - Specify debugging for BGP filters
 - SMI_BGP_DBG_KEEPA_LIVE - Specify debugging for BGP keepalives
 - SMI_BGP_DBG_UPDATE - Updates (in|out) Specify debugging for BGP updates
 - SMI_BGP_DBG_UPDATE_IN - Inbound updates
 - SMI_BGP_DBG_UPDATE_OUT - Outbound updates.
 - SMI_BGP_DBG_RFD - Specify debugging for BGP dampening
 - SMI_BGP_DBG_BFD - Specified debugging for BGP Bidirectional Forwarding Detection
 - SMI_BGP_DBG_MPLS - Specify debugging for BGP Multiprotocol Label Switching
 - SMI_BGP_DBG_VPLS - Specify debugging for BGP VPLS
- ← *vrId* Virtual Router Id

Returns:

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

2.1.2.66 `s_int32_t smi_bgp_default_ipv4_unicast_set (struct smiclient_globals * azg, u_int32_t vrId)`

Sets the bgp default ipv4 unicast. `smi_bgp_default_ipv4_unicast_set`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.67 `s_int32_t smi_bgp_default_ipv4_unicast_set_validate (struct smiclient_globals * azg, u_int32_t vrId)`

Sets the bgp default ipv4 unicast. `smi_bgp_default_ipv4_unicast_set_validate`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.68 `s_int32_t smi_bgp_default_ipv4_unicast_unset (struct smiclient_globals * azg, u_int32_t vrId)`

Unsets the bgp default ipv4 unicast. `smi_bgp_default_ipv4_unicast_unset`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.69 s_int32_t smi_bgp_default_ipv4_unicast_unset_validate (struct smiclient_globals * azg, u_int32_t vrId)

Unsets the bgp default ipv4 unicast. smi_bgp_default_ipv4_unicast_unset_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.70 int smi_bgp_default_local_preference_set_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, u_int32_t localPref)

Sets the default LOCAL_PREF attribute of a BGP speaker. A BGP speaker uses it to inform its other internal peers of the advertising speaker's degree of preference for an advertised route. smi_bgp_default_local_preference_set_sdkapi

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *localPref* BGP default local preference <0-4294967295>; default value is 100
- ← *vrId* Virtual router id

Returns:

BGP_API_SET_SUCCESS on success.

2.1.2.71 int smi_bgp_default_local_preference_unset_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs)

Unsets the default LOCAL_PREF attribute of a BGP speaker. smi_bgp_default_local_preference_unset_sdkapi

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual router id

Returns:

BGP_API_SET_SUCCESS on success.

2.1.2.72 `s_int32_t smi_bgp_deterministic_med_set (struct smiclient_globals * azg, u_int32_t vrId)`

Sets the bgp deterministic med. `smi_bgp_deterministic_med_set`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.73 `s_int32_t smi_bgp_deterministic_med_set_validate (struct smiclient_globals * azg, u_int32_t vrId)`

Sets the bgp deterministic med. `smi_bgp_deterministic_med_set_validate`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.74 `s_int32_t smi_bgp_deterministic_med_unset (struct smiclient_globals * azg, u_int32_t vrId)`

Unsets the bgp deterministic med. `smi_bgp_deterministic_med_unset`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.75 s_int32_t smi_bgp_deterministic_med_unset_validate (struct smiclient_globals * azg, u_int32_t vrId)

Unsets the bgp deterministic med. smi_bgp_deterministic_med_unset_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.76 int smi_bgp_disable_adj_out_set (struct smiclient_globals * azg, u_int32_t vrId)

Sets the bgp disable adjacent. smi_bgp_disable_adj_out_set

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.77 int smi_bgp_disable_adj_out_set_validate (struct smiclient_globals * azg, u_int32_t vrId)

Sets the bgp disable adjacent. smi_bgp_disable_adj_out_set_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

**2.1.2.78 int smi_bgp_disable_adj_out_unset (struct smiclient_globals * *azg*,
u_int32_t *vrId*)**

unsets the bgp disable adjacent smi_bgp_disable_adj_out_unset

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

**2.1.2.79 int smi_bgp_disable_adj_out_unset_validate (struct smiclient_globals
* *azg*, u_int32_t *vrId*)**

unsets the bgp disable adjacent smi_bgp_disable_adj_out_unset_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

**2.1.2.80 s_int32_t smi_bgp_enforce_first_as_set (struct smiclient_globals * *azg*,
u_int32_t *vrId*)**

Sets the bgp enforce as first. smi_bgp_enforce_first_as_set

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.81 s_int32_t smi_bgp_enforce_first_as_set_validate (struct smiclient_globals * *azg*, u_int32_t *vrId*)

Sets the bgp enforce as first. smi_bgp_enforce_first_as_set_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.82 s_int32_t smi_bgp_enforce_first_as_unset (struct smiclient_globals * *azg*, u_int32_t *vrId*)

Unsets the bgp enforce as first. smi_bgp_enforce_first_as_unset

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.83 s_int32_t smi_bgp_enforce_first_as_unset_validate (struct smiclient_globals * *azg*, u_int32_t *vrId*)

Unsets the bgp enforce as first. smi_bgp_enforce_first_as_unset_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.84 `int smi_bgp_extcommunity_list_entry_unset_validate (struct smiclient_globals * azg, u_int32_t vrId, char * commListName, char * commListValue, int nameType, int action, int entryType)`

Unconfigure BGP extended community filtering. `smi_bgp_extcommunity_list_entry_unset`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *commListName* Community list name
- ← *commListValue* Community value string
- ← *nameType* List name type <0-1> 0 - COMMUNITY_LIST_STRING
1 - COMMUNITY_LIST_NUMBER
- ← *action* Action type <0-1> 0 - Deny
1 - Permit
- ← *entryType* List entry type <3-5> 3 -EXTCOMMUNITY_LIST_STANDARD
4 -EXTCOMMUNITY_LIST_EXPANDED
5 -EXTCOMMUNITY_LIST_AUTO
- ← *vrId* Virtual router id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_INVALID_VALUE
 BGP_API_SET_ERROR
 BGP_API_SET_ERR_CLIST_DEFINE_CONFLICT
 BGP_API_SET_ERR_MALFORMED_ARG

2.1.2.85 `int smi_bgp_extcommunity_list_set (struct smiclient_globals * azg, u_int32_t vrId, char * commListName, char * commListValue, int nameType, int action, int entryType)`

Configure BGP extended community filtering. `smi_bgp_extcommunity_list_set`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *commListName* Community list name
- ← *commListValue* Community value string
- ← *nameType* List name type <0-1> 0 - COMMUNITY_LIST_STRING
1 - COMMUNITY_LIST_NUMBER
- ← *action* Action type <0-1> 0 - Deny
1 - Permit

← **entryType** List entry type <3-5> 3 -EXTCOMMUNITY_LIST_STANDARD
 4 -EXTCOMMUNITY_LIST_EXPANDED
 5 -EXTCOMMUNITY_LIST_AUTO

← **vrId** Virtual router id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_INVALID_VALUE
 BGP_API_SET_ERROR
 BGP_API_SET_ERR_CLIST_DEFINE_CONFLICT
 BGP_API_SET_ERR_MALFORMED_ARG

2.1.2.86 int smi_bgp_extcommunity_list_unset (struct smiclient_globals * *azg*, u_int32_t *vrId*, char * *commListName*)

Unconfigure BGP extended community filtering. smi_bgp_extcommunity_list_unset

Parameters:

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

← **commListName** ACL number

← **vrId** Virtual router id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_UNKNOWN_OBJECT
 BGP_API_SET_ERR_MALFORMED_ARG

2.1.2.87 int smi_bgp_fast_external_failover_set (struct smiclient_globals * *azg*, u_int32_t *vrId*)

Sets the bgp fast external failover. smi_bgp_fast_external_failover_set

Parameters:

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

← **vrId** Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR BGP_API_SET_ERR_INVALID_BGP

2.1.2.88 `int smi_bgp_fast_external_failover_set_validate (struct smiclient_globals * azg, u_int32_t vrId)`

Sets the bgp fast external failover. `smi_bgp_fast_external_failover_set_validate`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR BGP_API_SET_ERR_INVALID_BGP

2.1.2.89 `int smi_bgp_fast_external_failover_unset (struct smiclient_globals * azg, u_int32_t vrId)`

Unsets the bgp fast external failover. `smi_bgp_fast_external_failover_unset`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR BGP_API_SET_ERR_INVALID_BGP

2.1.2.90 `int smi_bgp_fast_external_failover_unset_validate (struct smiclient_globals * azg, u_int32_t vrId)`

Unsets the bgp fast external failover. `smi_bgp_fast_external_failover_unset_validate`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR BGP_API_SET_ERR_INVALID_BGP

2.1.2.91 `int smi_bgp_get_address_family (struct smiclient_globals * azg,
u_int32_t vrId, int afi, int safi, enum address_family_flag *
addressFamilyFlag)`

Returns a enum value corresponding to the router address family flag configured. smi_bgp_get_address_family

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *afi* Address family identifier Address family identifier (1-ipv4/2-ipv6/3-vpnv4/4-vpnv6)
- ← *safi* Sub-address family identifier Sub-Address family identifier (1-unicast/2-multicast)
- ← *addressFamilyFlag* Type of address family flag hose returned values are ADDR_FAMILY_IPV4_UNICAST = 1, ADDR_FAMILY_IPV4_MULTICAST = 2, ADDR_FAMILY_IPV6_UNICAST = 3, ADDR_FAMILY_VPNV4_UNICAST = 4, ADDR_FAMILY_VPNV6_UNICAST = 5,
- ← *vrId* Virtual router-id

Returns:

BGP_API_SET_SUCCESS on success, otherwise error codes: SMI_ERROR
BGP_API_ERR_BGP_DEFAULT_LOOKUP_FAIL

2.1.2.92 `int smi_bgp_get_grst_restart_time (struct smiclient_globals * azg,
u_int32_t vrId, char * peerAddr, u_int32_t * restartTime)`

This API gets the configured BGP graceful restart time. smi_bgp_get_grst_restart_time

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual router id
- ← *peerAddr* The peer address Peerd IP address
- *restartTime* Restart time

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes

2.1.2.93 `int smi_bgp_get_grst_stalepath_time (struct smiclient_globals * azg,
u_int32_t vrId, u_int32_t * stalepathTime)`

This API gets the configured BGP graceful restart time. smi_bgp_get_grst_stalepath_time

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual router id
- *stalepathTime* Stalepath time

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes

2.1.2.94 **int smi_bgp_get_identifier (struct smiclient_globals * *azg*, u_int32_t *vrId*, int *bgpProcId*, struct pal_in4_addr * *routerId*)**

bgp_get function returns the pointer to the specified BGP instance. The BGP Identifier of the local system. smi_bgp_get_identifier

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- *routerId* BGP identifier

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.95 **int smi_bgp_get_local_as (struct smiclient_globals * *azg*, u_int32_t *vrId*, int *bgpProcId*, int * *bgpAs*)**

bgp_get function returns the pointer to the specified BGP instance. bgp_get_local_-as refer to local autonomous system number, where Autonomous System is a set of routers under a single technical administration. smi_bgp_get_local_as

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- *as* AS number

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.96 `int smi_bgp_get_nbr_address_family (struct smiclient_globals *
azg, u_int32_t vrId, char * peerAddr, int afi, int safi, enum
nbr_addr_family * addressFamilyFlag)`

Returns a enum value corresponding to the neighbor address family flag configured.
smi_bgp_get_nbr_address_family

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address Neighbor id address
- ← *afi* Address family identifier Address family identifier (1-ipv4/2-ipv6)
- ← *safi* Sub-address family identifier Sub-Address family identifier (1-unicast/2-multicast)
- *addressFamilyFlag* Type of address family flag hose returned values are
NBR_ADDR_FAMILY_IPV4_UNICAST = 1, NBR_ADDR_FAMILY_IPV4_MULTICAST = 2, NBR_ADDR_FAMILY_IPV6_UNICAST = 3,
- ← *vrId* Virtual router-id

Returns:

BGP_API_SET_SUCCESS on success, otherwise error codes: SMI_ERROR
BGP_API_ERR_BGP_DEFAULT_LOOKUP_FAIL

2.1.2.97 `int smi_bgp_get_peer_admin_status (struct smiclient_globals * azg,
u_int32_t vrId, int bgpProcId, struct pal_in4_addr * peerAddr, int *
peerAdminFlag)`

The desired state of the BGP connection. A transition from 'stop' to 'start' will cause the BGP Manual Start Event to be generated. A transition from 'start' to 'stop' will cause the BGP Manual Stop Event to be generated. This parameter can be used to restart BGP peer connections. Care should be used in providing write access to this object without adequate authentication. smi_bgp_get_peer_admin_status

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *peerpeerAddr* The address of the peer
- *peerAdminFlag* The administration status. The status is BGP_PeerAdmin_stop if there is a peer shutdown flag; otherwise it is BGP_PeerAdmin_start

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.98 `int smi_bgp_get_peer_connect_retry_interval (struct smiclient_globals * azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr * peerAddr, int * connRetryInterval)`

Time interval (in seconds) for the ConnectRetry timer. The suggested value for this timer is 120 seconds. `smi_bgp_get_peer_connect_retry_interval`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *peerAddr* The peer address The address of the peer
- *connRetryInterval* time interval (in seconds)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.99 `int smi_bgp_get_peer_fsm_established_time (struct smiclient_globals * azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr * peerAddr, int * estTime)`

This timer indicates how long (in seconds) this peer has been in the established state or how long since this peer was last in the established state. It is set to zero when a new peer is configured or when the router is booted. `smi_bgp_get_peer_fsm_established_time`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *peerAddr* The peer address The address of the peer
- *estTime* time interval (in seconds)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.100 `int smi_bgp_get_peer_fsm_established_transitions (struct smiclient_globals * azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr * peerAddr, int * estTxns)`

The total number of times the BGP FSM transitioned into the established state for this peer. `smi_bgp_get_peer_fsm_established_transitions`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *peerAddr* The peer address The address of the peer
- *estTxns* The number of established transitions

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.101 `int smi_bgp_get_peer_hold_time (struct smiclient_globals * azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr * peerAddr, int * holdTime)`

This function returns the time interval in seconds that the Hold timer has been established with the BGP peer. The value must be at least 3 seconds or zero (0), which means the Hold timer has not been established with the peer. `smi_bgp_get_peer_hold_time`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *peerAddr* The peer address The address of the peer
- *holdTime* The hold time interval (in seconds)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.102 `int smi_bgp_get_peer_hold_time_configured (struct smiclient_globals * azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr * peerAddr, int * holdTimeConf)`

Time interval (in seconds) for the Hold Time configured for this BGP speaker with this peer. This value is placed in an OPEN message sent to this peer by this BGP speaker, and is compared with the Hold Time field in an OPEN message received from the peer when determining the Hold Time (`bgpPeerHoldTime`) with the peer. This value must not be less than three seconds if it is not zero (0). If it is zero (0), the Hold Time is NOT to be established with the peer. The suggested value for this timer is 90 seconds. `smi_bgp_get_peer_hold_time_configured`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *peerAddr* The peer address The address of the peer
- *holdTimeConf* Time interval (in seconds)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.103 `int smi_bgp_get_peer_identifier (struct smiclient_globals * azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr * peerAddr, struct pal_in4_addr * peerRouterId)`

The BGP Identifier of this entry's (Entry containing information about the connection with a BGP peer) BGP peer. This entry MUST be 0.0.0.0 unless the `bgpPeerState` is in the openconfirm or the established state. `smi_bgp_get_peer_identifier`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *peerAddr* The address of the peer
- *peerRouterId* BGP identifier of the peer

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.104 `int smi_bgp_get_peer_in_total_messages (struct smiclient_globals *
azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr *peerAddr,
int *peerInTotalMsg)`

The total number of messages received from the remote peer on BGP connection. smi_bgp_get_peer_in_total_messages

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *peerAddr* The peer address The address of the peer
- *peerInTotalMsg* The total number of messages

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.105 `int smi_bgp_get_peer_in_update_elapsed_time (struct
smiclient_globals *azg, u_int32_t vrId, int bgpProcId, struct
pal_in4_addr *peerAddr, int *inUpdateElaps)`

Elapsed time (in seconds) since the last BGP UPDATE message was received from the peer. Each time bgpPeerInUpdates is incremented, the value of this object is set to zero (0). smi_bgp_get_peer_in_update_elapsed_time

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *peerAddr* The peer address The address of the peer
- *inUpdateElaps* time interval (in seconds)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.106 `int smi_bgp_get_peer_in_updates (struct smiclient_globals * azg,
u_int32_t vrId, int bgpProcId, struct pal_in4_addr * addr, int *
peerInUpdates)`

This function returns the number of BGP UPDATE messages received on BGP connection. `smi_bgp_get_peer_in_updates`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *peerAddr* The peer address The address of the peer
- *peerInUpdates* The number of BGP UPDATE messages received

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.107 `int smi_bgp_get_peer_keep_alive (struct smiclient_globals * azg,
u_int32_t vrId, int bgpProcId, struct pal_in4_addr * peerAddr, int *
keepAlive)`

Time interval (in seconds) for the KeepAlive timer established with the peer. The value of this object is calculated by this BGP speaker such that, when compared with `bgpPeerHoldTime`, it has the same proportion that `bgpPeerKeepAliveConfigured` has, compared with `bgpPeerHoldTimeConfigured`. `smi_bgp_get_peer_keep_alive`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *peerAddr* The peer address The address of the peer
- *keepAlive* The KeepAlive time interval (in seconds)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.108 `int smi_bgp_get_peer_keep_alive_configured (struct smiclient_globals * azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr * peerAddr, int * keepAliveConf)`

Time interval (in seconds) for the KeepAlive timer established with the peer. The value of this object is calculated by this BGP speaker such that, when compared with `bgpPeerHoldTime`, it has the same proportion that `bgpPeerKeepAliveConfigured` has, compared with `bgpPeerHoldTimeConfigured`. `smi_bgp_get_peer_keep_alive_configured`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *peerAddr* The peer address The address of the peer
- *keepAliveConf* time interval (in seconds)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.109 `int smi_bgp_get_peer_last_error (struct smiclient_globals * azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr * peerAddr, u_int16_t * peerLastError)`

The last error code and subcode seen by the peer on BGP connection. If no error has occurred, this field is zero. Otherwise, the first byte of this two byte OCTET STRING contains the error code, and the second byte contains the subcode. `smi_bgp_get_peer_last_error`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *peerAddr* The peer address The address of the peer
- *peerLastError* Error seen by the peer on BGP connection

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.110 `int smi_bgp_get_peer_local_addr (struct smiclient_globals * azg,
u_int32_t vrId, int bgpProcId, struct pal_in4_addr * peerAddr, struct
pal_in4_addr * peerLocalAddr)`

This function returns the local address of the peers BGP connection. `smi_bgp_get_peer_local_addr`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *peerAddr* The address of the peer
- *peerLocalAddr* The local address of the peers BGP connection

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.111 `int smi_bgp_get_peer_local_port (struct smiclient_globals * azg,
u_int32_t vrId, int bgpProcId, struct pal_in4_addr * peerAddr, int *
peerLocalPort)`

This function returns the local port for the TCP connection between the BGP peers. `smi_bgp_get_peer_local_port`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *peerAddr* The peer address The address of the peer
- *peerLocalPort* The local port

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.112 `int smi_bgp_get_peer_min_as_origination_interval (struct
smiclient_globals * azg, u_int32_t vrId, int bgpAs, struct
pal_in4_addr * peerAddr, int * minAsOrigInterval)`

Time interval (in seconds) for the MinASOriginationInterval timer. The suggested value for this timer is 15 seconds. `smi_bgp_get_peer_min_as_origination_interval`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *peerAddr* The peer address The address of the peer
- *minAsOrigInterval* time interval (in seconds)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.113 int smi_bgp_get_peer_min_route_advertisement_interval (struct smiclient_globals * *azg*, u_int32_t *vrId*, int *bgpProcId*, struct pal_in4_addr * *peerAddr*, int * *minRouteAdInterval*)

Time interval (in seconds) for the MinRouteAdvertisementInterval timer. The suggested value for this timer is 30 seconds for EBGp connections and 5 seconds for IBGP connections. smi_bgp_get_peer_min_route_advertisement_interval

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *peerAddr* The peer address The address of the peer
- *minRouteAdInterval* time interval (in seconds)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.114 int smi_bgp_get_peer_negotiated_version (struct smiclient_globals * *azg*, u_int32_t *vrId*, int *bgpProcId*, struct pal_in4_addr * *peerAddr*, int * *bgpPeerNegotiatedVersion*)

This function gets the negotiated version of BGP running between the two peers. This entry MUST be zero (0) unless the bgpPeerState is in the openconfirm or the established state. Note that legal values for this object are between 0 and 255. smi_bgp_get_peer_negotiated_version

Parameters:

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

- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *peerpeerAddr* The address of the peer
- *bgpPeerNegotiatedVersion* The negotiated BGP version

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes

BGP_API_GET_ERROR

2.1.2.115 `int smi_bgp_get_peer_out_total_messages (struct smiclient_globals * azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr * peerAddr, int * peerOutTotalMsg)`

This function returns the total number of messages transmitted to the remote peer on BGP connection. `smi_bgp_get_peer_out_total_messages`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *peerAddr* The peer address The address of the peer
- *peerOutTotalMsg* The total number of transmitted messages

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes

BGP_API_GET_ERROR

2.1.2.116 `int smi_bgp_get_peer_out_updates (struct smiclient_globals * azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr * peerAddr, int * peerOutUpdates)`

This function returns the number of BGP UPDATE messages transmitted on BGP connection. `smi_bgp_get_peer_out_updates`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *peerAddr* The peer address The address of the peer

→ *peerOutUpdates* The total number of transmitted BGP UPDATE messages

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.117 `int smi_bgp_get_peer_remote_addr (struct smiclient_globals * azg,
u_int32_t vrId, int bgpProcId, struct pal_in4_addr * peerAddr, struct
pal_in4_addr * peerRemoteAddr)`

The remote IP address of this entry's (Entry containing information about the connection with a BGP peer) BGP peer. smi_bgp_get_peer_remote_addr

Parameters:

← *azg* Pointer to the SMI client global structure
← *vrId* Virtual Router Id
← *bgpProcId* BGP process Id
← *peerAddr* The peer address The address of the peer
→ *peerRemoteAddr* The remote address of the peer's BGP connection

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.118 `int smi_bgp_get_peer_remote_as (struct smiclient_globals * azg,
u_int32_t vrId, int bgpProcId, struct pal_in4_addr * addr, int *
peerRemoteAs)`

This function returns the pointer to the remote autonomous system number received in the BGP OPEN message. smi_bgp_get_peer_remote_as

Parameters:

← *azg* Pointer to the SMI client global structure
← *vrId* Virtual Router Id
← *bgpProcId* BGP process Id
← *peerAddr* The peer address The address of the peer
→ *peerRemoteAs* The remote AS number

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.119 `int smi_bgp_get_peer_remote_port (struct smiclient_globals * azg,
u_int32_t vrId, int bgpProcId, struct pal_in4_addr * peerAddr, int *
peerRemotePort)`

This function returns the remote port for the TCP connection between the BGP peers.
smi_bgp_get_peer_remote_port

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *peerAddr* The peer address The address of the peer
- *peerRemotePort* The remote port

Returns:

- BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
- BGP_API_GET_ERROR

2.1.2.120 `int smi_bgp_get_peer_state (struct smiclient_globals * azg, u_int32_t
vrId, int bgpProcId, struct pal_in4_addr * peerAddr, int * peerState)`

This function returns the pointer to the bgp peer state(BGP instance). If no pointer is returned, it tries to create a new one. smi_bgp_get_peer_state

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *peerpeerAddr* The address of the peer
- *state* The connection state

Returns:

- BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
- BGP_API_GET_ERROR

2.1.2.121 `int smi_bgp_get_peer_timers (struct smiclient_globals * azg,
u_int32_t vrId, u_int16_t * keepAlive, u_int16_t * holdTime)`

This API get the configured BGP keepalive and holdtime timer. smi_bgp_get_peer_timers

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual router id
- *keepAlive* Keepalive time
- *holdTime* Holdtime

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes

2.1.2.122 `int smi_bgp_get_update_delay_val (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t * deferTime)`

This API gets the configured BGP graceful restart time. `smi_bgp_get_update_delay_val`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual router id
- *deferTime* Delay time

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes

2.1.2.123 `s_int32_t smi_bgp_get_version (struct smiclient_globals * azg, u_int32_t vrId, int bgpProcId, int * bgpVersion)`

This function returns the version of the supported BGP version. `smi_bgp_get_version`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- *bgpVersion* The supported BGP version

Returns:

BGP_API_GET_SUCCESS on success

2.1.2.124 `s_int32_t smi_bgp_grst_restart_time_set_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, u_int32_t restartTime)`

sets graceful restart time smi_bgp_grst_restart_time_set_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *restartTime* Graceful restart time
- ← *vrId* Virtual router-id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.125 `s_int32_t smi_bgp_grst_restart_time_unset_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, u_int32_t restartTime)`

unsets graceful restart time smi_bgp_grst_restart_time_unset_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *restartTime* Graceful restart time
- ← *vrId* Virtual router-id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.126 `s_int32_t smi_bgp_grst_set_validate (struct smiclient_globals * azg, u_int32_t vrId)`

Sets the bgp graceful. smi_bgp_grst_set_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.127 `s_int32_t smi_bgp_grst_stalepath_time_set_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, u_int32_t stalepathTime)`

sets graceful stalepath time smi_bgp_grst_stalepath_time_set_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *stalepathTime* Graceful stalepath time
- ← *vrId* Virtual router-id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.128 `s_int32_t smi_bgp_grst_stalepath_time_unset_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, u_int32_t stalepathTime)`

unsets graceful stalepath time smi_bgp_grst_stalepath_time_unset_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *stalepathTime* Graceful stalepath time
- ← *vrId* Virtual router-id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.129 `s_int32_t smi_bgp_grst_unset_validate (struct smiclient_globals * azg, u_int32_t vrId)`

Unsets the bgp graceful. smi_bgp_grst_unset_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.130 `s_int32_t smi_bgp_instance_unset_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs)`

Deletes the specified BGP instance. `smi_bgp_instance_unset_sdkapi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *as* AS number
- ← *bgpName* BGP instance name: optional, usually NULL
- ← *vrId* Virtual router id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_UNKNOWN_OBJECT
 BGP_API_SET_ERR_INSTANCE_MISMATCH
 BGP_API_SET_ERR_MULTIPLE_INSTANCE_NOT_SET
 BGP_API_SET_ERROR
 BGP_API_SET_ERR_AS_MISMATCH

2.1.2.131 `int smi_bgp_maximum_paths_set (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, int bgpType, int multipathsNum)`

Sets bgp maximum paths. `smi_bgp_maximum_paths_set`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *bgpAflag* BGP Address family flag
- ← *bgp_type* Type of BGP session (ebgp/ibgp)
- ← *multipathsNum* Supported multipaths number <2-64>
- ← *vrId* Virtual router id
- ← *vrName* VRF name (default/VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.132 `int smi_bgp_maximum_paths_set_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, int bgpType, int multipathsNum)`

Sets bgp maximum paths. `smi_bgp_maximum_paths_set_validate`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *bgpAfFlag*
- ← *type*
- ← *multipaths*
- ← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.133 `int smi_bgp_maximum_paths_unset (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, int bgpType)`

Unsets bgp maximum paths. smi_bgp_maximum_paths_unset

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *bgp_type* Type of BGP session (ebgp/ibgp)
- ← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.134 `int smi_bgp_maximum_paths_unset_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, int bgp_type)`

Unsets bgp maximum paths. smi_bgp_maximum_paths_unset_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *flag*
- ← *type*
- ← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.135 `s_int32_t smi_bgp_multiple_instance_set (struct smiclient_globals * azg, u_int32_t vrId)`

sets the multiple instance smi_bgp_multiple_instance_set

Parameters:

← *azg* Pointer to the SMI client global structure

← *vrId*

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes

BGP_API_GET_ERROR

2.1.2.136 `s_int32_t smi_bgp_multiple_instance_set_validate (struct smiclient_globals * azg, u_int32_t vrId)`

sets the multiple instance smi_bgp_multiple_instance_set_validate

Parameters:

← *azg* Pointer to the SMI client global structure

← *vrId*

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes

BGP_API_GET_ERROR

2.1.2.137 `s_int32_t smi_bgp_multiple_instance_unset (struct smiclient_globals * azg, u_int32_t vrId)`

unsets the multiple instance smi_bgp_multiple_instance_unset

Parameters:

← *azg* Pointer to the SMI client global structure

← *vrId*

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes

BGP_API_GET_ERROR

2.1.2.138 `s_int32_t smi_bgp_multiple_instance_unset_validate (struct smiclient_globals * azg, u_int32_t vrId)`

unsets the multiple instance smi_bgp_multiple_instance_unset_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual router-id(Default-0)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.139 `int smi_bgp_nbr_address_family_set (struct smiclient_globals * azg, u_int32_t vrId, char * peerAddr, int afi, int safi)`

Sets the BGP af_flag for neighbor. smi_bgp_nbr_address_family_set

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address Neighbor IP address
- ← *afi* Address family identifier Address family identifier (1-ipv4/2-ipv6)
- ← *safi* Sub-address family identifier Sub-Address family identifier (1-unicast/2-multicast)
- ← *vrId* Virtual router-id

Returns:

BGP_API_SET_SUCCESS on success, otherwise error codes: SMI_ERROR
BGP_API_SET_ERR_INVALID_AF
BGP_ERR_INVALID_SAFI
BGP_API_ERR_BGP_DEFAULT_LOOKUP_FAIL

2.1.2.140 `int smi_bgp_nbr_address_family_unset (struct smiclient_globals * azg, u_int32_t vrId, int afi, int safi, char * peerAddr)`

Unsets the BGP af_flag for neighbor. smi_bgp_nbr_address_family_unset

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *afi* Address family identifier Address family identifier(1-ipv4/2-ipv6)
- ← *safi* Sub-address family identifier Sub-Address family identifier (1-unicast/2-multicast)

← *peerAddr* The peer address Peer address

← *vrId* Virtual router-id

Returns:

BGP_API_SET_SUCCESS on success, otherwise error codes: SMI_ERROR

BGP_API_SET_ERR_INVALID_AF

BGP_ERR_INVALID_SAFI

BGP_API_ERR_BGP_DEFAULT_LOOKUP_FAIL

2.1.2.141 s_int32_t smi_bgp_network_sync_set_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, int afi, int safi)

Sets to perform IGP synchronization of network routes to announce via BGP. smi_bgp_network_sync_set_sdkapi

Parameters:

← *azg* Pointer to the SMI client global structure

← *afi* Address family identifier Address family identifier <1-2> (1=IP | 2=IP6)

← *safi* Sub-address family identifier Sub-Address family identifier <1-4> (UC|MC|UC_MC|MPLS)

← *vrId* Virtual router id

← *vrfName* VRF name (default/VRF name)

Returns:

BGP_API_SET_SUCCESS on success

2.1.2.142 s_int32_t smi_bgp_network_sync_unset_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, int afi, int safi)

Unsets to perform IGP synchronization of network routes to announce via BGP. smi_bgp_network_sync_unset_sdkapi

Parameters:

← *azg* Pointer to the SMI client global structure

← *afi* Address family identifier Address family identifier <1-2> (1=IP | 2=IP6)

← *safi* Sub-address family identifier Sub-Address family identifier <1-4> (UC|MC|UC_MC|MPLS)

← *vrId* Virtual router id

← *vrfName* VRF name (default/VRF name)

Returns:

BGP_API_SET_SUCCESS on success

2.1.2.143 `int smi_bgp_no_debug_validate (struct smiclient_globals * azg, int debugFlag, u_int32_t vrId)`

Use this function to disable all BGP troubleshooting functions. `smi_bgp_no_debug`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *debugFlag* Pass debug flag as following:
 - SMI_BGP_DBG_ALL - For all debugging for BGP
 - SMI_BGP_DBG_NHT - Specify debugging for BGP NHT
 - SMI_BGP_DBG_NSM - Specify debugging for NSM messages
 - SMI_BGP_DBG_FSM - Specify debugging for BGP Finite State Machine (FSM)
 - SMI_BGP_DBG_EVENTS - Specify debugging for BGP events
 - SMI_BGP_DBG_FILTER - Specify debugging for BGP filters
 - SMI_BGP_DBG_KEEPALIVE - Specify debugging for BGP keepalives
 - SMI_BGP_DBG_UPDATE - Updates (in|out) Specify debugging for BGP updates
 - SMI_BGP_DBG_RFD - Specify debugging for BGP dampening
 - SMI_BGP_DBG_BFD - Specified debugging for BGP Bidirectional Forwarding Detection
 - SMI_BGP_DBG_MPLS - Specify debugging for BGP Multiprotocol Label Switching
 - SMI_BGP_DBG_VPLS - Specify debugging for BGP VPLS
- ← *vrId* Virtual Router Id

Returns:

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

2.1.2.144 `int smi_bgp_option_check_sdkapi (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t optFlag)`

This API checks if the BGP flag is configured. `smi_bgp_option_check_sdkapi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual router id
- ← *optFlag* Config flag

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes

2.1.2.145 `s_int32_t smi_bgp_option_set (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, u_int32_t optFlag)`

This function sets the BGP option. The BGP option is a system-wide pre-configurable setting, and is usually not accessible to the end user. `smi_bgp_option_set`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *optFlag* BGP options:
 - BGP_OPT_NO_FIB.do not install BGP route into FIB
 - BGP_OPT_MULTIPLE_INSTANCE.multiple instance
 - BGP_OPT_CONFIG_STANDARD.industry-standard oriented configuration
 - BGP_OPT_RFC1771_PATH_SELECT.RFC1771 style path selection
 - BGP_OPT_RFC1771_STRICT.strictly follow RFC1771 description
 - BGP_OPT_AGGREGATE_NEXTHOP_CHECK.aggregate route only when next hop is same
 - BGP_OPT_ANVL_DAMPENING_CONFIG.ANVL style dampening config parse
 - BGP_OPT_EXTENDED_ASN_CAP.extended ASN capability
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/VRF name)

Returns:

- BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
- BGP_API_SET_ERR_MULT_INST_DEL_CONFIG
- BGP_API_SET_ERR_ADJ_OUT_DYNAMIC

2.1.2.146 `s_int32_t smi_bgp_option_unset_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t optFlag)`

This function unsets the BGP option. The BGP option is a system-wide pre-configurable setting, and is usually not accessible to the end user. `smi_bgp_option_unset`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *optFlag* BGP options:
 - BGP_OPT_NO_FIB.do not install BGP route into FIB
 - BGP_OPT_MULTIPLE_INSTANCE.multiple instance
 - BGP_OPT_CONFIG_STANDARD.industry-standard oriented configuration
 - BGP_OPT_RFC1771_PATH_SELECT.RFC1771 style path selection

BGP_OPT_RFC1771_STRICT:strictly follow RFC1771 description
 BGP_OPT_AGGREGATE_NEXTHOP_CHECK:aggregate route only when next hop is same
 BGP_OPT_ANVL_DAMPENING_CONFIG:ANVL style dampening config parse
 BGP_OPT_EXTENDED_ASN_CAP:extended ASN capability

← *vrId* Virtual router id

← *vrfName* VRF name (default/VRF name)

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_ADJ_OUT_DYNAMIC
 BGP_API_SET_ERR_MULT_INST_DEL_CONFIG

2.1.2.147 `s_int32_t smi_bgp_peer_group_bind_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, char * peerAddr, char * pgName)`

Binds a peer to specified peer-group. When a peer does not exist, it creates a new peer.
 smi_bgp_peer_group_bind_sdkapi

Parameters:

← *azg* Pointer to the SMI client global structure

← *peerAddr* The peer address BGP peer IP address or Tag

← *pgName* BGP peer group name

← *afi* Address family identifier Address family identifier <1-2> (1=IP | 2=IP6)

← *safi* Sub-address family identifier Sub-Address family identifier <1-4> (UC|MC|UC_MC|MPLS)

→ *as* AS number when error occurs

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_PEER_GROUP_UNCONFIGURED
 BGP_API_SET_ERR_PEER_GROUP_NO_REMOTE_AS
 BGP_API_SET_ERR_PEER_GROUP_CANT_CHANGE
 BGP_API_SET_ERR_PEER_GROUP_MISMATCH
 BGP_API_SET_ERR_PEER_GROUP_PEER_TYPE_DIFFERENT

2.1.2.148 `s_int32_t smi_bgp_peer_group_delete_unset_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, char * pgTag)`

Deletes the specified peer-group. smi_bgp_peer_group_delete_unset_sdkapi

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *pgTag* BGP neighbor router tag
- ← *vrId* Virtual router id

Returns:

BGP_API_SET_SUCCESS on success.

2.1.2.149 `s_int32_t smi_bgp_peer_group_remote_as_delete_unset_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, char * peerGroup)`

Removes the remote Autonomous System number of this entry's BGP peer group. smi_bgp_peer_group_remote_as_delete_unset_sdkapi

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerGroup* IP address of this entry's BGP peer group in the format: A.B.C.D.
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/VRF name)

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes

2.1.2.150 `s_int32_t smi_bgp_peer_group_unbind_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, char * peerAddr, char * pgName)`

Unbinds a peer from a specified peer-group. smi_bgp_peer_group_unbind_sdkapi

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address BGP peer IP address or Tag
- ← *pgName* BGP peer group name
- ← *afi* Address family identifier Address family identifier <1-2> (1=IP | 2=IP6)
- ← *safi* Sub-address family identifier Sub-Address family identifier <1-4> (UC|MC|UC_MC|MPLS)
- ← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
BGP_API_SET_ERR_PEER_GROUP_MISMATCH

2.1.2.151 `s_int32_t smi_bgp_peer_remote_as_set_sdkapi (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, char * peerAddr, int peerAs)`

Sets the remote Autonomous System number of this entry's BGP peer group. `smi_bgp_peer_remote_as_set_sdkapi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer addressess The peer address
- ← *peerAs* Remote AS number <0-65535>
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/VRF name)

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERROR
 BGP_API_SET_ERR_AS_MISMATCH

2.1.2.152 `s_int32_t smi_bgp_peer_unset_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, char * peerAddr)`

Deletes the specified peer from the peer-group. `smi_bgp_peer_unset_sdkapi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address BGP peer address in the format A.B.C.D
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/VRF name)

Returns:

BGP_API_SET_SUCCESS on success.

2.1.2.153 `int smi_bgp_rfc1771_path_select_set (struct smiclient_globals * azg, u_int32_t vrId)`

Sets the bgp rfc1771 path select. `smi_bgp_rfc1771_path_select_set`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.154 int smi_bgp_rfc1771_path_select_set_validate (struct smiclient_globals * *azg*, u_int32_t *vrId*)

Sets the bgp rfc1771 path select. smi_bgp_rfc1771_path_select_set_validate

Parameters:

← *azg* Pointer to the SMI client global structure
← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.155 int smi_bgp_rfc1771_path_select_unset (struct smiclient_globals * *azg*, u_int32_t *vrId*)

Unsets the bgp rfc1771 path select. smi_bgp_rfc1771_path_select_unset

Parameters:

← *azg* Pointer to the SMI client global structure
← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.156 int smi_bgp_rfc1771_path_select_unset_validate (struct smiclient_globals * *azg*, u_int32_t *vrId*)

Unsets the bgp rfc1771 path select. smi_bgp_rfc1771_path_select_unset_validate

Parameters:

← *azg* Pointer to the SMI client global structure
← *vrId* Virtual Router Id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.157 int smi_bgp_router_id_set_sdkapi_validate (struct smiclient_globals * *azg*, u_int32_t *vrId*, u_int32_t *bgpAs*, char * *routerIpAddr*)

Configure the BGP router ID. smi_bgp_router_id_set_sdkapi

Parameters:

← *azg* Pointer to the SMI client global structure
 ← *routerIpAddr* Router-id IP address
 ← *vrId* Virtual router id

Returns:

BGP_API_SET_SUCCESS on success.

2.1.2.158 int smi_bgp_router_id_unset_sdkapi_validate (struct smiclient_globals * *azg*, u_int32_t *vrId*, u_int32_t *bgpAs*, char * *routerIpAddr*)

Deletes the BGP router ID. smi_bgp_router_id_unset_sdkapi

Parameters:

← *azg* Pointer to the SMI client global structure
 ← *vrId* Virtual router id

Returns:

BGP_API_SET_SUCCESS on success.

2.1.2.159 int smi_bgp_set_peer_admin_status_validate (struct smiclient_globals * *azg*, u_int32_t *vrId*, int *bgpProcId*, struct pal_in4_addr *peerAddr*, s_int32_t *peerAdminFlag*)

The desired state of the BGP connection. A transition from 'stop' to 'start' will cause the BGP Manual Start Event to be generated. A transition from 'start' to 'stop' will cause the BGP Manual Stop Event to be generated. This parameter can be used to restart BGP peer connections. Care should be used in providing write access to this object without adequate authentication. smi_bgp_set_peer_admin_status

Parameters:

← *azg* Pointer to the SMI client global structure

- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *peerpeerAddr* The address of the peer
- ← *peerAdminFlag* The current status of the peer

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.160 `int smi_bgp_set_peer_connect_retry_interval_validate (struct smiclient_globals * azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr peerAddr, int connRetryInterval)`

Time interval (in seconds) for the ConnectRetry timer. The suggested value for this timer is 120 seconds. `smi_bgp_set_peer_connect_retry_interval`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *peerAddr* The peer address The address of the peer
- ← *connRetryInterval* The new time interval (in seconds)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.161 `int smi_bgp_set_peer_hold_time_configured_validate (struct smiclient_globals * azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr addr, int holdTimeConf)`

This function modifies the time interval in seconds for the hold time configured for this BGP speaker with the peer. The value must be at least 3 seconds or 0 (zero), which means the Hold timer has not been established with the peer. The suggested value for this timer is 90 seconds. `smi_bgp_set_peer_hold_time_configured`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id

- ← *bgpProcId* BGP process Id
- ← *peerAddr* The peer address The address of the peer
- ← *holdTimeConf* time interval (in seconds)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.162 int smi_bgp_set_peer_keep_alive_configured_validate (struct smiclient_globals * *azg*, u_int32_t *vrId*, int *bgpProcId*, struct pal_in4_addr *peerAddr*, int *keepAliveConf*)

This function modifies the time interval in seconds for the KeepAlive timer configured for this BGP speaker with the peer. If the value of this object is zero, no periodical KEEPALIVE messages are sent to the peer after the BGP connection has been established. The suggested value for this timer is 30 seconds. smi_bgp_set_peer_keep_alive_configured

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *peerAddr* The peer address The address of the peer
- ← *keepAliveConf* time interval (in seconds)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.163 int smi_bgp_set_peer_min_as_origination_interval_validate (struct smiclient_globals * *azg*, u_int32_t *vrId*, int *bgpAs*, struct pal_in4_addr *peerAddr*, int *minAsOrigInterval*)

Time interval (in seconds) for the MinASOriginationInterval timer. The suggested value for this timer is 15 seconds. smi_bgp_set_peer_min_as_origination_interval

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id

- ← *peerAddr* The peer address The address of the peer
- ← *minAsOrigInterval* time interval (in seconds)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.164 int smi_bgp_set_peer_min_route_advertisement_interval_validate (struct smiclient_globals * azg, u_int32_t vrId, int bgpProcId, struct pal_in4_addr peerAddr, int minRouteAdInterval)

This function modifies the time interval in seconds for the MinRouteAdvertisementInterval timer. The suggested value for this timer is 30 seconds. smi_bgp_set_peer_min_route_advertisement_interval

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual Router Id
- ← *bgpProcId* BGP process Id
- ← *peerAddr* The peer address The address of the peer
- ← *minRouteAdInterval* time interval (in seconds)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.165 int smi_bgp_show_bgp (struct smiclient_globals * azg, char * ip_addr, char * af, char * saf, char * vrfName, int prefixCheck, struct list * showList, int(*) (struct list * showlist) callback)

show bgp displays bgp routes smi_bgp_show_bgp

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *ip_addr* IPV4/IPV6 address with or without network mask
- ← *af* Address family identifier (ipv4/ipv6)
- ← *saf* Sub-Address family identifier (multicast/unicast)
- ← *vrfOption* vrf_option can be passed as default, all or VRF name
- ← *prefixCheck* set one if you are giving a network with mask else 0

- *showList,Pointer* to the linked list of structure bgpRouteInfo
- ← *callbackFunc* Callback func pointer

Returns:

BGP_API_SHOW_SUCCESS on success, otherwise error codes: SMI_ERROR

2.1.2.166 `int smi_bgp_show_bgp_extcommunity_list (struct smiclient_globals * azg, char * commListame, struct list * bgpExtCommList, int startIndex, int endIndex, u_int32_t(*)(struct list *bgpExtCommList) callbackFunc)`

displays the configured extcommunity-list smi_bgp_show_bgp_extcommunity_list

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *commListame* List name.
- ← *startIndex* Starting index number of the list, used to customised display of the list
- ← *endIndex* Ending index number of the list, used to customised display of the list
- *bgpExtCommList,Pointer* to the linked list of structure smiBgpExtCommunity
- ← *callbackFunc* Callback func pointer

Returns:

BGP_API_SHOW_SUCCESS on success, otherwise error codes: SMI_ERROR

2.1.2.167 `int smi_bgp_show_ip_bgp (struct smiclient_globals * azg, char * name, char * ipAddr, char * af, char * saf, enum smi_show_type vrfOption, char * vrfName, int prefixCheck, struct list * showList, int(*)(struct list *showlist) callback)`

show ip bgp displays routes matching the given view name smi_bgp_show_ip_bgp

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *name* view name for which routes needs to be displayed
- ← *ipAddr* IPV4/IPV6 address with or without network mask
- ← *af* Address family identifier (ipv4/ipv6)
- ← *saf* Sub-Address family identifier (multicast/unicast)
- ← *vrfOption* can hold any of the enum values of smi_show_type 1 for default and 2 for all

- ← *vrfName* can have VRF name
- ← *prefixCheck* set one if you are giving a network with mask else 0
- *showList,Pointer* to the linked list of structure bgpRouteInfo
- ← *callbackFunc* Callback func pointer

Returns:

BGP_API_SHOW_SUCCESS on success, otherwise error codes: SMI_ERROR

2.1.2.168 `int smi_bgp_show_ip_bgp_community (struct smiclient_globals * azg, char * commListName, char * af, char * saf, int exact, struct list * showList, int(*) (struct list * showlist) callback)`

show ip bgp community displays routes matching the communities smi_bgp_show_ip_bgp_community

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *commListName* community name for which routes needs to be displayed
- ← *af* Address family identifier (ipv4/ipv6)
- ← *saf* Sub-Address family identifier (multicast/unicast)
- ← *exact* variable is for choosing the type of output 0 = To print community list for the exact value 1 = To print all community list
- *showList,Pointer* to the linked list of structure bgpSummaryList
- ← *callbackFunc* Callback func pointer

Returns:

BGP_API_SHOW_SUCCESS on success, otherwise error codes: SMI_ERROR

2.1.2.169 `int smi_bgp_show_ip_bgp_community_list (struct smiclient_globals * azg, char * commListName, char * af, char * saf, char * vrfOption, int exact, struct list * showList, int(*) (struct list * showlist) callback)`

show ip bgp community list displays routes matching the community-list smi_bgp_show_ip_bgp_community_list

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *commListName* community-list name for which routes needs to be displayed
- ← *af* Address family identifier (ipv4/ipv6)
- ← *saf* Sub-Address family identifier (multicast/unicast)
- ← *vrfOption* vrf_option can be passed as default, all or VRF name

- ← *exact* variable is for choosing the type of output 0 = To print community list for the exact value 1 = To print all community list
- *showList,Pointer* to the linked list of structure bgpSummaryList
- ← *callbackFunc* Callback func pointer

Returns:

BGP_API_SHOW_SUCCESS on success, otherwise error codes: SMI_ERROR

2.1.2.170 `int smi_bgp_show_ip_bgp_extcommunity_list_exact_match
(struct smiclient_globals * azg, char * commListame, char * af,
char * saf, struct list * bgpExtCommList, int exactMatchFlag,
u_int32_t(*) (struct list * bgpExtCommList) callbackFunc)`

displays the routes of configured extcommunity-list smi_show_ip_bgp_extcommunity_list_exact_match_vrf

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *commListame* List name.
- ← *af* Address family identifier (ipv4/ipv6)
- ← *saf* Sub-Address family identifier (multicast/unicast)
- ← *exactMatchFlag* Display exact-match when value is 1. Value 0 will show the normal ext-community list routes
- *bgpExtCommList,Pointer* to the linked list of structure smiBgpExtCommunity
- ← *callbackFunc* Callback func pointer

Returns:

BGP_API_SHOW_SUCCESS on success, otherwise error codes: SMI_ERROR

2.1.2.171 `int smi_bgp_show_ip_bgp_extcommunity_list_exact_match_vrf
(struct smiclient_globals * azg, char * commListame, char * af,
char * saf, struct list * bgpExtCommList, char * vrfName, int
exactMatchFlag, u_int32_t(*) (struct list * bgpExtCommList)
callbackFunc)`

displays the routes of configured extcommunity-list smi_show_ip_bgp_extcommunity_list_exact_match_vrf

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *commListame* List name.

- ← *af* Address family identifier (ipv4/ipv6)
- ← *saf* Sub-Address family identifier (multicast/unicast)
- ← *vrfName* VRF name.
- ← *exactMatchFlag* Display exact-match when value is 1. Value 0 will show the normal ext-community list routes
- *bgpExtCommList, Pointer* to the linked list of structure smiBgpExtCommunity
- ← *callbackFunc* Callback func pointer

Returns:

BGP_API_SHOW_SUCCESS on success, otherwise error codes: SMI_ERROR

2.1.2.172 s_int32_t smi_bgp_static_network_set_sdkapi_validate (struct smiclient_globals * azg, u_int32_t bgpAs, char * localAddr, u_int32_t backdoor, char * rmapName, u_int32_t vrId)

Specifies a network to announce via BGP. smi_bgp_static_network_set_sdkapi

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *localAddr* IP Prefix
- ← *afi* Address family identifier Address family identifier <1-2> (1=IP | 2=IP6)
- ← *safi* Sub-address family identifier Sub-Address family identifier <1-4> (UC|MC|UC_MC|MPLS)
- ← *backdoor* BGP backdoor route Yes/No flag
- ← *rmapName* Name of the Route-Map
- ← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_INVALID_NETWORK
 BGP_API_SET_ERROR
 BGP_API_SET_ERR_OBJECT_ALREADY_EXIST

2.1.2.173 s_int32_t smi_bgp_static_network_unset_sdkapi_validate (struct smiclient_globals * azg, u_int32_t bgpAs, char * localAddr, u_int32_t vrId)

Unspecifies a network to announce via BGP. smi_bgp_static_network_unset_sdkapi

Parameters:

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

- ← *localAddr* IP Prefix
- ← *afi* Address family identifier Address family identifier <1-2> (1=IP | 2=IP6)
- ← *safi* Sub-address family identifier Sub-Address family identifier <1-4> (UC|MC|UC_MC|MPLS)
- ← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_INVALID_NETWORK
 BGP_API_SET_ERROR
 BGP_API_SET_ERR_UNKNOWN_OBJECT

2.1.2.174 s_int32_t smi_bgp_synchronization_set_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, int afi, int safi)

Enables IGP synchronization of BGP routes. smi_bgp_synchronization_set_sdkapi

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *afi* Address family identifier Address family identifier <1-2> (1=IP | 2=IP6)
- ← *safi* Sub-address family identifier Sub-Address family identifier <1-4> (UC|MC|UC_MC|MPLS)
- ← *vrId* Virtual router id
- ← *vrName* VRF name (default/VRF name)

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_INVALID_VALUE

2.1.2.175 s_int32_t smi_bgp_synchronization_unset_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, int afi, int safi)

Disables IGP synchronization of BGP routes. smi_bgp_synchronization_unset_sdkapi

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *afi* Address family identifier Address family identifier <1-2> (1=IP | 2=IP6)
- ← *safi* Sub-address family identifier Sub-Address family identifier <1-4> (UC|MC|UC_MC|MPLS)
- ← *vrId* Virtual router id

← *vrfName* VRF name (default/VRF name)

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
BGP_API_SET_ERR_INVALID_VALUE

2.1.2.176 `int smi_bgp_timers_set_sdkapi (struct smiclient_globals * azg,
u_int32_t vrId, u_int32_t bgpAs, u_int16_t keepAlive, u_int16_t
holdTime)`

Sets the time intervals in seconds for BGP's Hold Timer and KeepAlive Timer. smi_bgp_timers_set_sdkapi

Parameters:

← *azg* Pointer to the SMI client global structure
← *keepAlive* BGP keepalive time <0|3-65535>
← *holdTime* BGP holdtime <0|1-21845>
← *vrId* Virtual router id
← *vrfName* VRF name (default/VRF name)

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
BGP_API_SET_ERR_INVALID_VALUE
BGP_API_SET_ERR_INFINITE_HOLD_TIME_VALUE
BGP_API_SET_WARN_HOLD_AND_KEEPALIVE_INVALID
BGP_API_SET_ERR_INVALID_HOLD_TIME
BGP_API_SET_ERR_HOLD_LESS_EQUAL_KEEPALIVE

2.1.2.177 `int smi_bgp_timers_unset_sdkapi_validate (struct smiclient_globals *
azg, u_int32_t vrId, u_int32_t bgpAs)`

Unsets the BGP's Hold Timer and KeepAlive Timer. smi_bgp_timers_unset_sdkapi

Parameters:

← *azg* Pointer to the SMI client global structure
← *vrId* Virtual router id
← *vrfName* VRF name (default/VRF name)

Returns:

BGP_API_SET_SUCCESS on success.

2.1.2.178 `s_int32_t smi_bgp_update_delay_val_set_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, u_int32_t deferTime)`

sets update delay value smi_bgp_update_delay_val_set_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *deferTime* Update delay value
- ← *vrId* Virtual router-id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.179 `s_int32_t smi_bgp_update_delay_val_unset_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs)`

unsets update delay vlaue smi_bgp_update_delay_val_unset_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual router-id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.180 `s_int32_t smi_bgp_vrf_neighbor_as_override_set (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr, int afi, int safi)`

Sets vrf neighbor as override. smi_bgp_vrf_neighbor_as_override_set

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *afi* Address family identifier
- ← *safi* Sub-address family identifier
- ← *vrId* Virtual router id

← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.181 s_int32_t smi_bgp_vrf_neighbor_as_override_set_validate (struct smiclient_globals * azg, char * peerAddr, int afi, int safi, u_int32_t vrId)

Sets vrf neighbor as override. smi_bgp_vrf_neighbor_as_override_set_validate

Parameters:

← *azg* Pointer to the SMI client global structure
← *peerAddr* The peer address
← *afi* Address family identifier
← *safi* Sub-address family identifier
← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.182 s_int32_t smi_bgp_vrf_neighbor_as_override_unset (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr, int afi, int safi)

Unsets vrf neighbor as override. smi_bgp_vrf_neighbor_as_override_unset

Parameters:

← *azg* Pointer to the SMI client global structure
← *peerAddr* The peer address
← *afi* Address family identifier
← *safi* Sub-address family identifier
← *vrId* Virtual router id
← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.183 `s_int32_t smi_bgp_vrf_neighbor_as_override_unset_validate (struct smiclient_globals * azg, char * peerAddr, int afi, int safi, u_int32_t vrId)`

Unsets vrf neighbor as override. `smi_bgp_vrf_neighbor_as_override_unset_validate`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *afi* Address family identifier
- ← *safi* Sub-address family identifier
- ← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.184 `s_int32_t smi_filter_list_set_validate (struct smiclient_globals * azg, char * peerAddr, int afi, int safi, u_int32_t direction, char * aclInfo, u_int32_t vrId)`

Sets the filter list. `smi_filter_list_set_validate`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *afi* Address family identifier
- ← *safi* Sub-address family identifier
- ← *direction* Filter direction (in-1|out-0)
- ← *aclInfo* Access-list info
- ← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.185 `s_int32_t smi_filter_list_unset_validate (struct smiclient_globals * azg, char * peerAddr, int afi, int safi, u_int32_t direction, u_int32_t vrId)`

unsets the filter list `smi_filter_list_unset_validate`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *afi* Address family identifier
- ← *safi* Sub-address family identifier
- ← *direction* Filter direction (in-1|out-0)
- ← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.186 `s_int32_t smi_neighbor_attr_unchanged_as_path_set (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr, int afi, int safi)`

sets the neighbor transparent as smi_neighbor_transparent_as_set

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *afi* Address family identifier
- ← *safi* Sub-address family identifier
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.187 `s_int32_t smi_neighbor_attr_unchanged_as_path_unset (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr, int afi, int safi)`

unsets the neighbor transparent as smi_neighbor_transparent_as_unset

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address

- ← *afi* Address family identifier
- ← *safi* Sub-address family identifier
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.188 `s_int32_t smi_neighbor_attr_unchanged_med_set (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr, int afi, int safi)`

sets the neighbor transparent med smi_neighbor_attr_unchanged_med_set

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *localAddr*
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.189 `s_int32_t smi_neighbor_attr_unchanged_med_unset (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr, int afi, int safi)`

sets the neighbor transparent med smi_neighbor_attr_unchanged_med_unset

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *localAddr*
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.190 `s_int32_t smi_neighbor_attr_unchanged_nexthop_set (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr, int afi, int safi)`

sets the neighbor transparent nexthop `smi_neighbor_attr_unchanged_nexthop_set`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *localAddr*
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.191 `s_int32_t smi_neighbor_attr_unchanged_nexthop_unset (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr, int afi, int safi)`

sets the neighbor transparent nexthop `smi_neighbor_attr_unchanged_nexthop_unset`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *localAddr*
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.192 `s_int32_t smi_neighbor_capability_grst_set (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr, int afi, int safi)`

Sets the neighbor capability graceful. `smi_neighbor_capability_grst_set`

Parameters:

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

- ← *peerAddr* The peer address
- ← *afi* Address family identifier
- ← *safi* Sub-address family identifier
- ← *vrId* Virtual router id
- ← *vrName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.193 s_int32_t smi_neighbor_capability_grst_set_validate (struct smiclient_globals * azg, char * peerAddr, int afi, int safi, u_int32_t vrId)

Sets the neighbor capability graceful. smi_neighbor_capability_grst_set_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *afi* Address family identifier
- ← *safi* Sub-address family identifier
- ← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.194 s_int32_t smi_neighbor_capability_grst_unset (struct smiclient_globals * azg, u_int32_t vrId, char * vrName, char * peerAddr, int afi, int safi)

Unsets the neighbor capability graceful. smi_neighbor_capability_grst_unset

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *afi* Address family identifier
- ← *safi* Sub-address family identifier
- ← *vrId* Virtual router id

← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.195 s_int32_t smi_neighbor_capability_grst_unset_validate (struct smiclient_globals * azg, char * peerAddr, int afi, int safi, u_int32_t vrid)

Unsets the neighbor capability graceful. smi_neighbor_capability_grst_unset_validate

Parameters:

← *azg* Pointer to the SMI client global structure
← *peerAddr* The peer address
← *afi* Address family identifier
← *safi* Sub-address family identifier
← *vrid* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.196 s_int32_t smi_neighbor_capability_orf_prefix_set (struct smiclient_globals * azg, char * peerAddr, char * orfPrefixOpt, int afi, int safi, u_int32_t vrid, u_int32_t bgpAs)

sets neighbor capability orf prefix smi_neighbor_capability_orf_prefix_set

Parameters:

← *azg* Pointer to the SMI client global structure
← *peerAddr* The peer address
← *orfPrefixOpt* Neighbor capability ORF prefix option(both|receive|send)
← *afi* Address family identifier
← *safi* Sub-address family identifier
← *vrid* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.197 `s_int32_t smi_neighbor_capability_orf_prefix_set_validate (struct smiclient_globals * azg, char * peerAddr, char * nbrOrfPrefixOpt, int afi, int safi)`

sets neighbor capability orf prefix smi_neighbor_capability_orf_prefix_set_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *nbrOrfPrefixOpt* Neighbor capability ORF prefix option(both|receive|send)
- ← *afi* Address family identifier
- ← *safi* Sub-address family identifier
- ← *vrId* Virtual router-id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.198 `s_int32_t smi_neighbor_capability_orf_prefix_unset (struct smiclient_globals * azg, char * peerAddr, char * orfPrefixOpt, int afi, int safi, u_int32_t vrId, u_int32_t bgpAs)`

unsets neighbor capability orf prefix smi_neighbor_capability_orf_prefix_unset

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *orfPrefixOpt* Neighbor capability ORF prefix option(both|receive|send)
- ← *afi* Address family identifier
- ← *safi* Sub-address family identifier
- ← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.199 `s_int32_t smi_neighbor_capability_orf_prefix_unset_validate (struct smiclient_globals * azg, char * peerAddr, char * nbrOrfPrefixOpt, int afi, int safi)`

unsets neighbor capability orf prefix smi_neighbor_capability_orf_prefix_unset_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *nbrOrfPrefixOpt* Neighbor capability ORF prefix option(both|receive|send)
- ← *afi* Address family identifier
- ← *safi* Sub-address family identifier

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.200 `s_int32_t smi_neighbor_capability_route_refresh_set (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr)`

Sets the neighbor capability route refresh. `smi_neighbor_capability_route_refresh_set`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.201 `s_int32_t smi_neighbor_capability_route_refresh_set_validate (struct smiclient_globals * azg, char * peerAddr, u_int32_t vrId)`

Sets the neighbor capability route refresh. `smi_neighbor_capability_route_refresh_set_validate`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.202 `s_int32_t smi_neighbor_capability_route_refresh_unset (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr)`

Unsets the neighbor capability route refresh. `smi_neighbor_capability_route_refresh_unset`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.203 `s_int32_t smi_neighbor_capability_route_refresh_unset_validate (struct smiclient_globals * azg, char * peerAddr, u_int32_t vrId)`

Unsets the neighbor capability route refresh. `smi_neighbor_capability_route_refresh_unset_validate`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.204 `s_int32_t smi_neighbor_collide_established_set (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr)`

Sets the neighbor collide established. `smi_neighbor_collide_established_set`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address

- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.205 `s_int32_t smi_neighbor_collide_established_set_validate (struct smiclient_globals * azg, char * peerAddr, u_int32_t vrId)`

Sets the neighbor collide established. smi_neighbor_collide_established_set_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.206 `s_int32_t smi_neighbor_collide_established_unset (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr)`

Unsets the neighbor collide established. smi_neighbor_collide_established_unset

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.207 `s_int32_t smi_neighbor_collide_established_unset_validate (struct smiclient_globals * azg, char * peerAddr, u_int32_t vrId)`

Unsets the neighbor collide established. `smi_neighbor_collide_established_unset_validate`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.208 `s_int32_t smi_neighbor_connection_retry_time_unset_validate (struct smiclient_globals * azg, char * peerAddr, u_int32_t vrId)`

unset the neighbor connection retry time `smi_neighbor_connection_retry_time_unset_validate`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.209 `s_int32_t smi_neighbor_disallow_infinite_timer_set_validate (struct smiclient_globals * azg, char * peerAddr, u_int32_t vrId)`

set the neighbor disallow infinite timer `smi_neighbor_disallow_infinite_timer_set_validate`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.210 s_int32_t smi_neighbor_disallow_infinite_timer_unset_validate
 (struct smiclient_globals * *azg*, char * *peerAddr*, u_int32_t *vrId*)

unset the neighbor disallow infinite timer smi_neighbor_disallow_infinite_timer_unset_validate

Parameters:

← *azg* Pointer to the SMI client global structure
 ← *peerAddr* The peer address
 ← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.211 s_int32_t smi_neighbor_dont_capability_negotiate_unset_validate
 (struct smiclient_globals * *azg*, char * *peerAddr*, u_int32_t *vrId*)

unsets the neighbor dont capability negotiate smi_neighbor_dont_capability_negotiate_unset_validate

Parameters:

← *azg* Pointer to the SMI client global structure
 ← *peerAddr* The peer address
 ← *vrId* Virtual router-id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.212 s_int32_t smi_neighbor_enforce_multihop_set (struct
 smiclient_globals * *azg*, u_int32_t *vrId*, char * *vrName*, char *
peerAddr)

sets the neighbor enforce multihop smi_neighbor_enforce_multihop_set

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.213 s_int32_t smi_neighbor_enforce_multihop_set_validate (struct smiclient_globals * *azg*, char * *peerAddr*, u_int32_t *vrId*)

sets the neighbor enforce multihop smi_neighbor_enforce_multihop_set_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.214 s_int32_t smi_neighbor_enforce_multihop_unset (struct smiclient_globals * *azg*, u_int32_t *vrId*, char * *vrfName*, char * *peerAddr*)

unsets the neighbor enforce multihop smi_neighbor_enforce_multihop_unset

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.215 `s_int32_t smi_neighbor_enforce_multihop_unset_validate (struct smiclient_globals * azg, char * peerAddr, u_int32_t vrId)`

unsets the neighbor enforce multihop `smi_neighbor_enforce_multihop_unset_validate`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.216 `s_int32_t smi_neighbor_filter_list_set_validate (struct smiclient_globals * azg, char * peerAddr, int afi, int safi, u_int32_t direction, char * aclInfo, u_int32_t vrId)`

Sets the neighbor filter list. `smi_neighbor_filter_list_set_validate`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *afi* Address family identifier
- ← *safi* Sub-address family identifier
- ← *direction* Filter direction (in-1|out-0)
- ← *aclInfo* Access-list info Access-list info
- ← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.217 `s_int32_t smi_neighbor_filter_list_unset_validate (struct smiclient_globals * azg, char * peerAddr, int afi, int safi, u_int32_t direction, u_int32_t vrId)`

unsets the neighbor filter list `smi_neighbor_filter_list_unset_validate`

Parameters:

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

- ← *peerAddr* The peer address
- ← *afi* Address family identifier
- ← *safi* Sub-address family identifier
- ← *direction* Filter direction (in-1|out-0)
- ← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.218 `s_int32_t smi_neighbor_g_shut_time_set (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, char * peerAddr, u_int32_t shut_time)`

sets neighbor graceful shut time smi_neighbor_g_shut_time_set

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *shut_time*
- ← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.219 `s_int32_t smi_neighbor_g_shut_time_set_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, char * peerAddr, u_int32_t shut_time)`

sets neighbor graceful shut time smi_neighbor_g_shut_time_set_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *shutTime* Graceful shut time
- ← *vrId* Virtual router-id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.220 `s_int32_t smi_neighbor_g_shut_time_unset (struct smiclient_globals * azg, u_int32_t vrId, char * peerAddr)`

unsets neighbor graceful shut time `smi_neighbor_g_shut_time_unset`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *shut_time*
- ← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.221 `s_int32_t smi_neighbor_g_shut_time_unset_validate (struct smiclient_globals * azg, char * peerAddr, u_int32_t shutTime, u_int32_t vrId)`

unsets neighbor graceful shut time `smi_neighbor_g_shut_time_unset_validate`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *shutTime* Graceful shut time
- ← *vrId* Virtual router-id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.222 `s_int32_t smi_neighbor_local_as_set_validate (struct smiclient_globals * azg, char * peerAddr, u_int32_t vrId)`

sets the neighbor local as `smi_neighbor_local_as_set_validate`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.223 s_int32_t smi_neighbor_local_as_unset_validate (struct smiclient_globals * *azg*, char * *peerAddr*, u_int32_t *vrId*)

unsets the neighbor local as smi_neighbor_local_as_unset_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.224 s_int32_t smi_neighbor_override_capability_set (struct smiclient_globals * *azg*, u_int32_t *vrId*, char * *vrfName*, char * *peerAddr*)

sets the neighbor override capability smi_neighbor_override_capability_set

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.225 s_int32_t smi_neighbor_override_capability_set_validate (struct smiclient_globals * *azg*, char * *peerAddr*, u_int32_t *vrId*)

sets the neighbor override capability smi_neighbor_override_capability_set_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.226 `s_int32_t smi_neighbor_override_capability_unset (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr)`

unsets the neighbor override capability `smi_neighbor_override_capability_unset`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.227 `s_int32_t smi_neighbor_override_capability_unset_validate (struct smiclient_globals * azg, char * peerAddr, u_int32_t vrId)`

unsets the neighbor override capability `smi_neighbor_override_capability_unset_validate`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.228 `s_int32_t smi_neighbor_remove_private_as_set (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr, int afi, int safi)`

Sets the neighbor remove private as. `smi_neighbor_remove_private_as_set`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *afi* Address family identifier
- ← *safi* Sub-address family identifier
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.229 `s_int32_t smi_neighbor_remove_private_as_set_validate (struct smiclient_globals * azg, char * peerAddr, int afi, int safi, u_int32_t vrId)`

Sets the neighbor remove private as. `smi_neighbor_remove_private_as_set_validate`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *afi* Address family identifier
- ← *safi* Sub-address family identifier
- ← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.230 `s_int32_t smi_neighbor_remove_private_as_unset (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr, int afi, int safi)`

Unsets the neighbor remove private as. `smi_neighbor_remove_private_as_unset`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *afi* Address family identifier
- ← *safi* Sub-address family identifier
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.231 `s_int32_t smi_neighbor_remove_private_as_unset_validate (struct smiclient_globals * azg, char * peerAddr, int afi, int safi, u_int32_t vrId)`

Unsets the neighbor remove private as. `smi_neighbor_remove_private_as_unset_validate`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *afi* Address family identifier
- ← *safi* Sub-address family identifier
- ← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.232 `s_int32_t smi_neighbor_route_reflector_client_set_validate (struct smiclient_globals * azg, char * peerAddr, int afi, int safi, u_int32_t vrId)`

sets the neighbor route reflector client `smi_neighbor_route_reflector_client_set_validate`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address

- ← *afi* Address family identifier
- ← *safi* Sub-address family identifier
- ← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.233 s_int32_t smi_neighbor_route_reflector_client_unset_validate (struct smiclient_globals * azg, char * peerAddr, int afi, int safi, u_int32_t vrId)

unsets the neighbor route reflector client smi_neighbor_route_reflector_client_unset_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *afi* Address family identifier
- ← *safi* Sub-address family identifier
- ← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.234 s_int32_t smi_neighbor_route_server_client_set (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr, int afi, int safi)

sets the neighbor route server client smi_neighbor_route_server_client_set

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *afi* Address family identifier
- ← *safi* Sub-address family identifier
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.235 s_int32_t smi_neighbor_route_server_client_set_validate (struct smiclient_globals * azg, char * peerAddr, int afi, int safi, u_int32_t vrId)

sets the neighbor route server client smi_neighbor_route_server_client_set_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *afi* Address family identifier
- ← *safi* Sub-address family identifier
- ← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.236 s_int32_t smi_neighbor_route_server_client_unset (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr, int afi, int safi)

unsets the neighbor route server client smi_neighbor_route_server_client_unset

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *afi* Address family identifier
- ← *safi* Sub-address family identifier
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.237 `s_int32_t smi_neighbor_route_server_client_unset_validate (struct smiclient_globals * azg, char * peerAddr, int afi, int safi, u_int32_t vrId)`

unsets the neighbor route server client smi_neighbor_route_server_client_unset_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *afi* Address family identifier
- ← *safi* Sub-address family identifier
- ← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.238 `s_int32_t smi_neighbor_strict_capability_set (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr)`

set the neighbor strict capability smi_neighbor_strict_capability_set

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.239 `s_int32_t smi_neighbor_strict_capability_set_validate (struct smiclient_globals * azg, char * peerAddr, u_int32_t vrId)`

set the neighbor strict capability smi_neighbor_strict_capability_set_validate

Parameters:

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

← *peerAddr* The peer address

← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes

BGP_API_GET_ERROR

2.1.2.240 `s_int32_t smi_neighbor_strict_capability_unset (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr)`

unset the neighbor strict capability smi_neighbor_strict_capability_unset

Parameters:

← *azg* Pointer to the SMI client global structure

← *peerAddr* The peer address

← *vrId* Virtual router id

← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes

BGP_API_GET_ERROR

2.1.2.241 `s_int32_t smi_neighbor_strict_capability_unset_validate (struct smiclient_globals * azg, char * peerAddr, u_int32_t vrId)`

unset the neighbor strict capability smi_neighbor_strict_capability_unset_validate

Parameters:

← *azg* Pointer to the SMI client global structure

← *peerAddr* The peer address

← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes

BGP_API_GET_ERROR

2.1.2.242 `s_int32_t smi_neighbor_transparent_as_set_validate (struct smiclient_globals * azg, char * peerAddr, int afi, int safi, u_int32_t vrId)`

sets the neighbor transparent as smi_neighbor_transparent_as_set_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *afi* Address family identifier
- ← *safi* Sub-address family identifier
- ← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.243 `s_int32_t smi_neighbor_transparent_nexthop_set_validate (struct smiclient_globals * azg, char * peerAddr, int afi, int safi, u_int32_t vrId)`

sets the neighbor transparent nexthop smi_neighbor_transparent_nexthop_set_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *localAddr*
- ← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.244 `int smi_peer_activate_set_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr, int afi, int safi)`

Activate the Address Family for this Neighbor. smi_peer_activate_set_sdkapi

Parameters:

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

- ← *peerAddr* The peer address BGP peer IP address or Tag
- ← *afi* Address family identifier Address family identifier <1-2> (1=IP | 2=IP6)
- ← *safi* Sub-address family identifier Sub-Address family identifier <1-4> (UC|MC|UC_MC|MPLS)
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/VRF name)

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERROR
 BGP_API_SET_ERR_PEER_GROUP__INVALID
 BGP_API_SET_ERR_UNSUP_VPNVF_CONF

2.1.2.245 int smi_peer_advertise_interval_set_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, char * peerAddr, u_int32_t peerRaInterval)

Sets time interval (in seconds) interval between sending BGP routing updates. smi_peer_advertise_interval_set_sdkapi

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address BGP peer IP address or Tag
- ← *ra_interval* Minimum route advertisement interval <0-65535>
- ← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_INVALID_VALUE
 BGP_API_SET_ERR_INVALID_FOR_PEER_GROUP_MEMBER
 BGP_API_SET_ERR_ALREADY_SET

2.1.2.246 int smi_peer_advertise_interval_unset_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, char * peerAddr)

Sets time interval (in seconds) interval between sending BGP routing updates to default value. smi_peer_advertise_interval_unset_sdkapi

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address BGP peer IP address or Tag

← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
BGP_API_SET_ERR_INVALID_FOR_PEER_GROUP_MEMBER

2.1.2.247 `int smi_peer_af_flag_config_check (struct smiclient_globals * azg, u_int32_t vrId, char * peerAddr, int af_id, int subaf_id, u_int32_t peerAfFlag)`

This API checks if the BGP peer address-family flag is configured. `smi_peer_af_flag_config_check`

Parameters:

← *azg* Pointer to the SMI client global structure
← *vrId* Virtual router id
← *peerAddr* The peer address Peer IP address
← *af_id* Address family <1-IPv4/2-IPv6>
← *subaf_id* Sub-address family <1-Unicast/2-Multicast>
← *peerAfFlag* Peer address-family flag

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes

2.1.2.248 `int smi_peer_af_flag_set_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr, int afi, int safi, u_int32_t peerAfFlag)`

Sets the peer's address family only configuration flag. `smi_peer_af_flag_set_sdkapi`

Parameters:

← *azg* Pointer to the SMI client global structure
← *peerAddr* The peer address BGP peer IP address or Tag
← *afi* Address family identifier Address family identifier <1-2> (1=IP | 2=IP6)
← *safi* Sub-address family identifier Sub-Address family identifier <1-4> (UC|MC|UC_MC|MPLS)
← *peerAfFlag* Peer address family only configuration flag:
 PEER_FLAG_SEND_COMMUNITY.set neighbor send-community flag
 PEER_FLAG_SEND_EXT_COMMUNITY.set neighbor send-community extended flag
 PEER_FLAG_NEXTHOP_SELF.set neighbor next-hop-self flag

PEER_FLAG_REFLECTOR_CLIENT.set neighbor route-reflector-client flag
 PEER_FLAG_RSERVER_CLIENT.set neighbor route-server-client flag
 PEER_FLAG_SOFT_RECONFIG.set neighbor soft-reconfiguration inbound flag
 PEER_FLAG_AS_PATH_UNCHANGED.set neighbor attribute-unchanged as-path flag
 PEER_FLAG_NEXTHOP_UNCHANGED.set neighbor attribute-unchanged next-hop flag
 PEER_FLAG_MED_UNCHANGED.set neighbor attribute-unchanged med flag
 PEER_FLAG_REMOVE_PRIVATE_AS.set neighbor remove-private-AS flag
 PEER_FLAG_AS_OVERRIDE.set neighbor as-override flag
 PEER_FLAG_GRST_CAPABILITY.set neighbor capability graceful-restart flag

← *vrId* Virtual Router Id

← *vrfName* VRF name (default/VRF name)

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_INVALID_FLAG
 BGP_API_SET_ERR_PEER_INACTIVE
 GP_API_SET_ERR_INVALID_FOR_PEER_GROUP_MEMBER
 BGP_API_SET_ERR_PEER_GROUP_HAS_THE_FLAG
 BGP_API_SET_ERR_NOT_INTERNAL_PEER
 BGP_API_SET_ERR_REMOVE_PRIVATE_AS

2.1.2.249 `int smi_peer_af_flag_unset_sdkapi_validate (struct smiclient_globals *azg, u_int32_t vrId, char *vrfName, char *peerAddr, int afi, int safi, u_int32_t peerAfFlag)`

Unsets the peer's address family only configuration flag. `smi_peer_af_flag_unset_sdkapi`

Parameters:

← *azg* Pointer to the SMI client global structure

← *peerAddr* The peer address BGP peer IP address or Tag

← *afi* Address family identifier Address family identifier <1-2> (1=IP | 2=IP6)

← *safi* Sub-address family identifier Sub-Address family identifier <1-4> (UC|MC|UC_MC|MPLS)

← *peerAfFlag* flag Peer address family only configuration flag:

PEER_FLAG_SEND_COMMUNITY.set neighbor send-community flag

PEER_FLAG_SEND_EXT_COMMUNITY.set neighbor send-community
 extended flag
 PEER_FLAG_NEXTHOP_SELF.set neighbor next-hop-self flag
 PEER_FLAG_REFLECTOR_CLIENT.set neighbor route-reflector-client
 flag
 PEER_FLAG_RSERVER_CLIENT.set neighbor route-server-client flag
 PEER_FLAG_SOFT_RECONFIG.set neighbor soft-reconfiguration in-
 bound flag
 PEER_FLAG_AS_PATH_UNCHANGED.set neighbor attribute-unchanged
 as-path flag
 PEER_FLAG_NEXTHOP_UNCHANGED.set neighbor attribute-
 unchanged next-hop flag
 PEER_FLAG_MED_UNCHANGED.set neighbor attribute-unchanged med
 flag
 PEER_FLAG_REMOVE_PRIVATE_AS.set neighbor remove-private-AS
 flag
 PEER_FLAG_AS_OVERRIDE.set neighbor as-override flag
 PEER_FLAG_GRST_CAPABILITY.set neighbor capability graceful-restart
 flag

← *vrId* Virtual Router Id

← *vrfName* VRF name (default/VRF name)

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_INVALID_FLAG
 BGP_API_SET_ERR_PEER_INACTIVE
 GP_API_SET_ERR_INVALID_FOR_PEER_GROUP_MEMBER
 BGP_API_SET_ERR_PEER_GROUP_HAS_THE_FLAG
 BGP_API_SET_ERR_NOT_INTERNAL_PEER
 BGP_API_SET_ERR_REMOVE_PRIVATE_AS

2.1.2.250 int smi_peer_allowas_in_set_sdkapi_validate (struct smiclient_globals *azg, u_int32_t vrId, u_int32_t bgpAs, char *peerAddr, u_int32_t allowAsNum)

Enables to accept AS path with my AS present in it for MPLS VPN/BGP environment.
 smi_peer_allowas_in_set_sdkapi

Parameters:

← *azg* Pointer to the SMI client global structure

← *vrId* Virtual Router Id

← *peerAddr* The peer address BGP peer IP address or Tag

← *afi* Address family identifier Address family identifier <1-2> (1=IP | 2=IP6)

- ← *safi* Sub-address family identifier Sub-Address family identifier <1-4>
(UC|MC|UC_MC|MPLS)
- ← *allowAsNum* Allow AS number <1-10>

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_INVALID_VALUE
 BGP_API_SET_ERR_PEER_GROUP_HAS_THE_FLAG

2.1.2.251 `int smi_peer_allowas_in_unset_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, char * peerAddr)`

Disables the AS path loop check for MPLS VPN/BGP environment. `smi_peer_allowas_in_unset_sdkapi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address BGP peer IP address or Tag
- ← *afi* Address family identifier Address family identifier <1-2> (1=IP | 2=IP6)
- ← *safi* Sub-address family identifier Sub-Address family identifier <1-4>
(UC|MC|UC_MC|MPLS)
- ← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_INVALID_FOR_PEER_GROUP_MEMBER
 BGP_API_SET_ERR_PEER_GROUP_HAS_THE_FLAG

2.1.2.252 `int smi_peer_aslist_set_sdkapi_validate (struct smiclient_globals * azg, u_int32_t bgpAs, char * peerAddr, u_int32_t direction, char * aclInfo, u_int32_t vrId)`

Sets to filter AS Path segments to/from this neighbor. `smi_peer_aslist_set_sdkapi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address BGP peer IP address or Tag
- ← *afi* Address family identifier Address family identifier <1-2> (1=IP | 2=IP6)
- ← *safi* Sub-address family identifier Sub-Address family identifier <1-4>
(UC|MC|UC_MC|MPLS)
- ← *direction* Direction of the filter (0=FILTER_IN | 1=FILTER_OUT)

← *aclInfo* Access-list info Access-list number

← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_INVALID_VALUE
 BGP_API_SET_ERR_PEER_INACTIVE
 BGP_API_SET_ERR_INVALID_FOR_PEER_GROUP_MEMBER

2.1.2.253 `int smi_peer_aslist_unset_sdkapi_validate (struct smiclient_globals * azg, u_int32_t bgpAs, char * peerAddr, u_int32_t direction, u_int32_t vrId)`

Unsets to filter AS Path segments to/from this neighbor. smi_peer_aslist_unset_sdkapi

Parameters:

← *azg* Pointer to the SMI client global structure

← *peerAddr* The peer address BGP peer IP address or Tag

← *afi* Address family identifier Address family identifier <1-2> (1=IP | 2=IP6)

← *safi* Sub-address family identifier Sub-Address family identifier <1-4> (UC|MC|UC_MC|MPLS)

← *direction* Direction of the filter (0=FILTER_IN | 1=FILTER_OUT)

← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_INVALID_VALUE
 BGP_API_SET_ERR_PEER_INACTIVE
 BGP_API_SET_ERR_INVALID_FOR_PEER_GROUP_MEMBER
 BGP_API_SET_ERR_PEER_GROUP_HAS_THE_FLAG

2.1.2.254 `int smi_peer_asorig_interval_set_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, char * peerAddr, u_int32_t peerAsorigInterval)`

Sets time interval (in seconds) between sending AS-origination routing updates. smi_peer_asorig_interval_set_sdkapi

Parameters:

← *azg* Pointer to the SMI client global structure

← *peerAddr* The peer address BGP peer IP address or Tag

← *peerAsorigInterval* Time interval <1-65535>

← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success.

2.1.2.255 `int smi_peer_asorig_interval_unset_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, char * peerAddr)`

Unsets time interval (in seconds) between sending AS-origination routing updates.
smi_peer_asorig_interval_unset_sdkapi

Parameters:

← *azg* Pointer to the SMI client global structure
 ← *peerAddr* The peer address BGP peer IP address or Tag
 ← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success.

2.1.2.256 `int smi_peer_deactivate_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, char * peerAddr)`

Deactivate the Address Family for this Neighbor. smi_peer_deactivate_sdkapi

Parameters:

← *azg* Pointer to the SMI client global structure
 ← *peerAddr* The peer address BGP peer IP address or Tag
 ← *afi* Address family identifier Address family identifier <1-2> (1=IP | 2=IP6)
 ← *saafi* Sub-address family identifier Sub-Address family identifier <1-4> (UC|MC|UC_MC|MPLS)
 ← *vrId* Virtual router id
 ← *vrfName* VRF name (default/VRF name)

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERROR
 BGP_API_SET_ERR_PEER_GROUP_INVALID
 BGP_API_SET_ERR_UNSUP_VPNVF_CONF

2.1.2.257 `int smi_peer_default_originate_set_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, char * peerAddr, char * rmapName)`

Sets the source for originate default route to this neighbor, using route-map or without using. `smi_peer_default_originate_set_sdkapi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address BGP peer IP address or Tag
- ← *afi* Address family identifier Address family identifier <1-2> (1=IP | 2=IP6)
- ← *safi* Sub-address family identifier Sub-Address family identifier <1-4> (UC|MC|UC_MC|MPLS)
- ← *rmapName* Route-map name
- ← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success.

2.1.2.258 `int smi_peer_default_originate_unset_sdkapi (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, char * peerAddr)`

Unsets the source for originate default route to this neighbor. `smi_peer_default_originate_unset_sdkapi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address BGP peer IP address or Tag
- ← *afi* Address family identifier Address family identifier <1-2> (1=IP | 2=IP6)
- ← *safi* Sub-address family identifier Sub-Address family identifier <1-4> (UC|MC|UC_MC|MPLS)
- ← *peerRemoveFlag* Yes/No flag
- ← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_INVALID
 BGP_API_SET_ERR_PEER_INACTIVE
 BGP_API_SET_ERR_PEER_GROUP_HAS_THE_FLAG

2.1.2.259 `int smi_peer_description_set_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, char * peerAddr, char * peerDesc)`

Sets the BGP Neighbor's description. `smi_peer_description_set_sdkapi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address BGP peer IP address or Tag
- ← *peerDesc* Peer's description string
- ← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success

2.1.2.260 `int smi_peer_description_unset_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, char * peerAddr)`

Unsets the BGP Neighbor's description. `smi_peer_description_unset_sdkapi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address BGP peer IP address or Tag
- ← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success

2.1.2.261 `s_int32_t smi_peer_disallow_hold_timer_set_sdkapi (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr)`

set the neighbor disallow infinite timer `smi_peer_disallow_hold_timer_set_sdkapi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.262 `s_int32_t smi_peer_disallow_hold_timer_unset_sdkapi (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr)`

unset the neighbor disallow infinite timer `smi_peer_disallow_hold_timer_unset_sdkapi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.263 `int smi_peer_distribute_set_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, char * peerAddr, u_int32_t direction, char * aclInfo)`

Sets to filter UPDATES to/from this neighbor. `smi_peer_distribute_set_sdkapi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address BGP peer IP address or Tag
- ← *afi* Address family identifier Address family identifier <1-2> (1=IP | 2=IP6)
- ← *safi* Sub-address family identifier Sub-Address family identifier <1-4> (UC|MC|UC_MC|MPLS)
- ← *direction* Direction of the filter (0=FILTER_IN | 1=FILTER_OUT)
- ← *aclInfo* Access-list info Access-list number
- ← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
BGP_API_SET_ERR_PEER_INACTIVE
BGP_API_SET_ERR_INVALID_VALUE
BGP_API_SET_ERR_INVALID_FOR_PEER_GROUP_MEMBER
BGP_API_SET_ERR_PEER_GROUP_HAS_THE_FLAG
BGP_API_SET_ERR_PEER_FILTER_CONFLICT

2.1.2.264 `int smi_peer_distribute_unset_sdkapi (struct smiclient_globals *
azg, u_int32_t vrId, u_int32_t bgpAs, char * peerAddr, u_int32_t
direction)`

Unsets to filter UPDATES to/from this neighbor. smi_peer_distribute_unset_sdkapi

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address BGP peer IP address or Tag
- ← *afi* Address family identifier Address family identifier <1-2> (1=IP | 2=IP6)
- ← *safi* Sub-address family identifier Sub-Address family identifier <1-4> (UC|MC|UC_MC|MPLS)
- ← *direction* Direction of the filter (0=FILTER_IN | 1=FILTER_OUT)
- ← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
BGP_API_SET_ERR_PEER_INACTIVE
BGP_API_SET_ERR_INVALID_VALUE
BGP_API_SET_ERR_INVALID_FOR_PEER_GROUP_MEMBER

2.1.2.265 `s_int32_t smi_peer_dont_capability_negotiate_set (struct
smiclient_globals * azg, u_int32_t vrId, char * vrfName, char *
peerAddr)`

sets the neighbor dont capability negotiate smi_peer_dont_capability_negotiate_set

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.266 `s_int32_t smi_peer_dont_capability_negotiate_set_validate (struct
smiclient_globals * azg, char * peerAddr, u_int32_t vrId)`

sets the neighbor dont capability negotiate smi_peer_dont_capability_negotiate_set_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *vrId* Virtual router-id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.267 `s_int32_t smi_peer_dont_capability_negotiate_unset (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr)`

unsets the neighbor dont capability negotiate smi_peer_dont_capability_negotiate_unset

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.268 `s_int32_t smi_peer_dynamic_capability_set (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr)`

Sets the neighbor capability dynamic. smi_peer_dynamic_capability_set

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.269 `s_int32_t smi_peer_dynamic_capability_set_validate (struct smiclient_globals * azg, u_int32_t vrId, char * peerAddr)`

Sets the neighbor capability dynamic. `smi_peer_dynamic_capability_set_validate`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.270 `s_int32_t smi_peer_dynamic_capability_unset (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr)`

Unsets the neighbor capability dynamic. `smi_peer_dynamic_capability_unset`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.271 `s_int32_t smi_peer_dynamic_capability_unset_validate (struct smiclient_globals * azg, u_int32_t vrId, char * peerAddr)`

Unsets the neighbor capability dynamic. `smi_peer_dynamic_capability_unset_validate`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.272 `int smi_peer_ebgp_multihop_set_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, char * peerAddr, u_int8_t timeToLive)`

Sets TTL to EBGp neighbors that are not on directly connected networks. `smi_peer_ebgp_multihop_set_sdkapi`

Parameters:

← *azg* Pointer to the SMI client global structure
 ← *peerAddr* The peer address BGP peer IP address or Tag
 ← *tll* eBGp multihop TTL value <0-255>
 ← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_PEER_GROUP_HAS_THE_FLAG

2.1.2.273 `int smi_peer_ebgp_multihop_unset_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, char * peerAddr)`

Unets TTL to EBGp neighbors that are not on directly connected networks. `smi_peer_ebgp_multihop_unset_sdkapi`

Parameters:

← *azg* Pointer to the SMI client global structure
 ← *peerAddr* The peer address BGP peer IP address or Tag
 ← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_PEER_GROUP_HAS_THE_FLAG

2.1.2.274 `int smi_peer_flag_config_check (struct smiclient_globals * azg, u_int32_t vrId, char * peerAddr, u_int32_t peerFlag)`

This API checks if the BGP peer flag is configured. `smi_peer_flag_config_check`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual router id
- ← *peerAddr* The peer address Peer IP address
- ← *peerFlag* Peer address-family flag

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes

2.1.2.275 `int smi_peer_flag_set_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr, u_int32_t peerFlag)`

Sets the peer configuration flag. smi_peer_flag_set_sdkapi

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address BGP peer IP address or Tag
- ← *peerFlag* Peer's configuration flag for all address family
 - PEER_FLAG_PASSIVE.set neighbor passive flag
 - PEER_FLAG_SHUTDOWN.set neighbor shutdown flag
 - PEER_FLAG_DONT_CAPABILITY.set neighbor dont-capability-negotiate flag
 - PEER_FLAG_OVERRIDE_CAPABILITY.set neighbor override-capability flag
 - PEER_FLAG_STRICT_CAP_MATCH.set neighbor strict-capability-match flag
 - PEER_FLAG_NO_ROUTE_REFRESH_CAP.set no neighbor capability route-refresh flag
 - PEER_FLAG_DYNAMIC_CAPABILITY.set neighbor capability dynamic flag
 - PEER_FLAG_ENFORCE_MULTIHOP.set neighbor enforce-multihop flag
 - PEER_FLAG_COLLIDE_ESTABLISHED.set neighbor collide establishment flag
- ← *vrId* Virtual Router Id
- ← *vrfName* VRF name (default/VRF name)

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_INVALID_FLAG
 BGP_API_SET_ERR_INVALID_FOR_PEER_GROUP_MEMBER
 BGP_API_SET_ERR_PEER_FLAG_CONFLICT
 PEER_FLAG_STRICT_CAP_MATCH
 PEER_FLAG_OVERRIDE_CAPABILITY

2.1.2.276 `int smi_peer_flag_unset_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr, u_int32_t peerFlag)`

Unsets the peer configuration flag. smi_peer_flag_unset_sdkapi

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address BGP peer IP address or Tag
- ← *peerFlag* Peer's configuration flag for all address family
 - PEER_FLAG_PASSIVE.set neighbor passive flag
 - PEER_FLAG_SHUTDOWN.set neighbor shutdown flag
 - PEER_FLAG_DONT_CAPABILITY.set neighbor dont-capability-negotiate flag
 - PEER_FLAG_OVERRIDE_CAPABILITY.set neighbor override-capability flag
 - PEER_FLAG_STRICT_CAP_MATCH.set neighbor strict-capability-match flag
 - PEER_FLAG_NO_ROUTE_REFRESH_CAP.set no neighbor capability route-refresh flag
 - PEER_FLAG_DYNAMIC_CAPABILITY.set neighbor capability dynamic flag
 - PEER_FLAG_ENFORCE_MULTIHOP.set neighbor enforce-multihop flag
 - PEER_FLAG_COLLIDE_ESTABLISHED.neighbor collide establishment flag
- ← *vrId* Virtual Router Id
- ← *vrfName* VRF name (default/VRF name)

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_INVALID_FLAG
 BGP_API_SET_ERR_INVALID_FOR_PEER_GROUP_MEMBER
 BGP_API_SET_ERR_PEER_GROUP_HAS_THE_FLAG
 BGP_API_SET_ERR_PEER_FLAG_CONFLICT
 PEER_FLAG_STRICT_CAP_MATCH
 PEER_FLAG_OVERRIDE_CAPABILITY
 BGP_API_SET_ERR_PEER_GROUP_SHUTDOWN
 BGP_API_SET_ERR_NO_GRST_SUPPORT

2.1.2.277 `int smi_peer_get_advertise_interval (struct smiclient_globals * azg, u_int32_t vrId, char * peerAddr, u_int32_t * ra_interval)`

This API get the configured BGP peer advertise interval. smi_peer_get_advertise_interval

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual router id
- ← *peerAddr* The peer address Peerd IP address
- *ra_interval* Advertise interval

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes

2.1.2.278 `int smi_peer_get_allowas_in (struct smiclient_globals * azg, u_int32_t vrId, char * peerAddr, int af_id, int subaf_id, u_int32_t * allowAsNum)`

This API get the configured BGP peer allow-as. `smi_peer_get_allowas_in`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual router id
- ← *peerAddr* The peer address Peerd IP address
- ← *af_id* Address family <1-IPv4/2-IPv6>
- ← *subaf_id* Sub-address family <1-Unicast/2-Multicast>
- *allowAsNum* Allow AS number

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes

2.1.2.279 `int smi_peer_get_asorig_interval (struct smiclient_globals * azg, u_int32_t vrId, char * peerAddr, u_int32_t * peerAsorigInterval)`

This API get the configured BGP peer asorig interval. `smi_peer_get_asorig_interval`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual router id
- ← *peerAddr* The peer address Peerd IP address
- *peerAsorigInterval* AS Orig interval

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes

2.1.2.280 `int smi_peer_get_description (struct smiclient_globals * azg,
u_int32_t vrId, char * peerAddr, char * peerDesc)`

This API get the configured BGP peer description. `smi_peer_get_description`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual router id
- ← *peerAddr* The peer address Peerd IP address
- *peerDesc* Peer description

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes

2.1.2.281 `int smi_peer_get_ebgp_multihop (struct smiclient_globals * azg,
u_int32_t vrId, char * peerAddr, u_int8_t * ttl)`

This API get the configured BGP multihop. `smi_peer_get_ebgp_multihop`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual router id
- ← *peerAddr* The peer address Peerd IP address
- *ttl* Multihop count

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes

2.1.2.282 `int smi_peer_get_interface (struct smiclient_globals * azg, u_int32_t
vrId, char * peerAddr, char * ifName)`

This API get the configured interface for BGP peer. `smi_peer_get_interface`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual router id
- ← *peerAddr* The peer address Peerd IP address
- *ifname* Interface name

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes

2.1.2.283 `int smi_peer_get_timers (struct smiclient_globals * azg, u_int32_t vrId, char * peerAddr, u_int32_t * keepAlive, u_int32_t * holdTime)`

This API get the configured BGP peer keepalive and holdtime. `smi_peer_get_timers`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual router id
- ← *peerAddr* The peer address Peer IP address
- *keepAlive* Keepalive timer
- *holdTime* Holdtime timer

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes

2.1.2.284 `int smi_peer_get_timers_connect (struct smiclient_globals * azg, u_int32_t vrId, char * peerAddr, u_int32_t * peerConnectInterval)`

This API get the configured BGP peer connect timer. `smi_peer_get_timers_connect`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual router id
- ← *peerAddr* The peer address Peer IP address
- *peerConnectInterval* Connect timer

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes

2.1.2.285 `int smi_peer_get_update_source_info (struct smiclient_globals * azg, u_int32_t vrId, char * peerAddr, char * updateIf, char * updateSource)`

This API get the configured BGP peer routing update source information. `smi_peer_get_update_source_info`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrId* Virtual router id
- ← *peerAddr* The peer address Peer IP address
- *updateIf* Interface name

→ *updateSource* Update source

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes

2.1.2.286 `int smi_peer_interface_set_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, char * peerAddr, char * ifName)`

Sets the peer's interface local IP address. `smi_peer_interface_set_sdkapi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address BGP peer IP address or Tag
- ← *ifName* Interface name
- ← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
BGP_API_SET_ERR_INVALID_VALUE
BGP_API_IP_NOT_IN_SAME_SUBNET

2.1.2.287 `int smi_peer_interface_unset_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, char * peerAddr, char * ifName)`

Unsets the peer's interface local IP address. `smi_peer_interface_unset_sdkapi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address BGP peer IP address or Tag
- ← *ifName* Interface name
- ← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
BGP_API_SET_ERROR
BGP_API_INVALID_INTERFACE_NAME

2.1.2.288 `int smi_peer_maximum_prefix_set_sdkapi_validate (struct smiclient_globals * azg, u_int32_t bgpAs, char * peerAddr, u_int32_t maxPrefixes, u_int32_t threshold, bool_t warning, u_int32_t vrId)`

Sets the maximum number of prefixes accepted from this peer. smi_peer_maximum_prefix_set_sdkapi

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address BGP peer IP address or Tag
- ← *afi* Address family identifier Address family identifier <1-2> (1=IP | 2=IP6)
- ← *safi* Sub-address family identifier Sub-Address family identifier <1-4> (UC|MC|UC_MC|MPLS)
- ← *maxPrefixes* Maximum number of prefixes <1-4294967295>
- ← *threshold* Threshold-value in percent <1-100>
- ← *warning* Throw warning if exceeds threshold-value Yes/No flag
- ← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_INVALID_FOR_PEER_GROUP_MEMBER
 BGP_API_SET_ERR_PEER_GROUP_HAS_THE_FLAG
 BGP_API_SET_ERR_PEER_INACTIVE

2.1.2.289 `s_int32_t smi_peer_next_hop_self_set (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr, int afi, int safi)`

sets the neighbor nexthop self smi_peer_next_hop_self_set

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *afi* Address family identifier
- ← *safi* Sub-address family identifier
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.290 `s_int32_t smi_peer_next_hop_self_set_validate (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr, int afi, int safi)`

sets the neighbor nexthop self smi_peer_next_hop_self_set_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *afi* Address family identifier
- ← *safi* Sub-address family identifier

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.291 `s_int32_t smi_peer_next_hop_self_unset (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr, int afi, int safi)`

unsets the neighbor nexthop self smi_peer_next_hop_self_unset

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *afi* Address family identifier
- ← *safi* Sub-address family identifier
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.292 `s_int32_t smi_peer_next_hop_self_unset_validate (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr, int afi, int safi)`

unsets the neighbor nexthop self smi_peer_next_hop_self_unset_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *afi* Address family identifier
- ← *safi* Sub-address family identifier

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.293 `int smi_peer_password_set_validate (struct smiclient_globals * azg,
 u_int32_t bgpAs, char * peerAddr, char * password, u_int32_t vrId)`

Sets Password to the neighbour. smi_peer_password_set_sdkapi

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address BGP peer IP address or Tag
- ← *type* Password type <0>
- ← *password* Password
- ← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_PEER_GROUP_HAS_THE_FLAG

2.1.2.294 `int smi_peer_password_unset_sdkapi_validate (struct
 smiclient_globals * azg, u_int32_t bgpAs, char * peerAddr, u_int32_t
vrId)`

Unsets Password to the neighbour. smi_peer_password_unset_sdkapi

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address BGP peer IP address or Tag
- ← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success.

2.1.2.295 `int smi_peer_port_set_sdkapi (struct smiclient_globals * azg,
u_int32_t vrId, u_int32_t bgpAs, char * peerAddr, u_int16_t bgpPort)`

Sets neighbor's BGP port number. `smi_peer_port_set_sdkapi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address BGP peer IP address or Tag
- ← *bgpPort* BGP port number <0-65535>
- ← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_INVALID_FOR_PEER_GROUP_MEMBER
 BGP_API_SET_ERR_PEER_GROUP_HAS_THE_FLAG

2.1.2.296 `int smi_peer_port_unset_sdkapi_validate (struct smiclient_globals
* azg, u_int32_t vrId, u_int32_t bgpAs, char * peerAddr, char *
bgpPort)`

Unsets neighbor's BGP port number. `smi_peer_port_unset_sdkapi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address BGP peer IP address or Tag
- ← *bgpPort* NULL value
- ← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_PEER_GROUP_HAS_THE_FLAG

2.1.2.297 `int smi_peer_prefix_list_set_sdkapi (struct smiclient_globals * azg,
u_int32_t vrId, u_int32_t bgpAs, char * peerAddr, u_int32_t
direction, char * aclInfo)`

Sets to filter address prefixes to/from this neighbor. `smi_peer_prefix_list_set_sdkapi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address BGP peer IP address or Tag
- ← *afi* Address family identifier Address family identifier <1-2> (1=IP | 2=IP6)

- ← *safi* Sub-address family identifier Sub-Address family identifier <1-4> (UC|MC|UC_MC|MPLS)
- ← *direction* Direction of the filter (0=FILTER_IN | 1=FILTER_OUT)
- ← *aclInfo* Access-list info Access-list number
- ← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_PEER_INACTIVE
 BGP_API_SET_ERR_INVALID_VALUE
 BGP_API_SET_ERR_INVALID_FOR_PEER_GROUP_MEMBER
 BGP_API_SET_ERR_PEER_FILTER_CONFLICT
 BGP_API_SET_ERR_PEER_GROUP_HAS_THE_FLAG

2.1.2.298 `int smi_peer_prefix_list_unset_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, char * peerAddr, u_int32_t direction)`

Unsets to filter address prefixes to/from this neighbor. smi_peer_prefix_list_unset_sdkapi

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address BGP peer IP address or Tag
- ← *afi* Address family identifier Address family identifier <1-2> (1=IP | 2=IP6)
- ← *safi* Sub-address family identifier Sub-Address family identifier <1-4> (UC|MC|UC_MC|MPLS)
- ← *direction* Direction of the filter (0=FILTER_IN | 1=FILTER_OUT)
- ← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_PEER_INACTIVE
 BGP_API_SET_ERR_INVALID_VALUE
 BGP_API_SET_ERR_INVALID_FOR_PEER_GROUP_MEMBER
 BGP_API_SET_ERR_PEER_GROUP_HAS_THE_FLAG

2.1.2.299 `int smi_peer_route_map_set_sdkapi_validate (struct smiclient_globals * azg, u_int32_t bgpAs, char * peerAddr, u_int32_t direction, char * aclInfo, u_int32_t vrId)`

Sets to filter Route-Map segments to/from this neighbor. smi_peer_route_map_set_sdkapi

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address BGP peer IP address or Tag
- ← *afi* Address family identifier Address family identifier <1-2> (1=IP | 2=IP6)
- ← *safi* Sub-address family identifier Sub-Address family identifier <1-4> (UC|MC|UC_MC|MPLS)
- ← *direction* Direction of the filter (0=FILTER_IN | 1=FILTER_OUT)
- ← *aclInfo* Access-list info Access-list number
- ← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_INVALID_VALUE
 BGP_API_SET_ERR_INVALID_FOR_PEER_GROUP_MEMBER
 BGP_API_SET_ERR_PEER_GROUP_HAS_THE_FLAG
 BGP_API_SET_ERR_PEER_INACTIVE

2.1.2.300 `int smi_peer_route_map_unset_sdkapi_validate (struct smiclient_globals * azg, u_int32_t bgpAs, char * peerAddr, u_int32_t direction, u_int32_t vrId)`

Sets to filter Route-Map segments to/from this neighbor. smi_peer_route_map_unset_sdkapi

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address BGP peer IP address or Tag
- ← *afi* Address family identifier Address family identifier <1-2> (1=IP | 2=IP6)
- ← *safi* Sub-address family identifier Sub-Address family identifier <1-4> (UC|MC|UC_MC|MPLS)
- ← *direction* Direction of the filter (0=FILTER_IN | 1=FILTER_OUT)
- ← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_INVALID_VALUE
 BGP_API_SET_ERR_INVALID_FOR_PEER_GROUP_MEMBER
 BGP_API_SET_ERR_PEER_GROUP_HAS_THE_FLAG
 BGP_API_SET_ERR_PEER_INACTIVE

2.1.2.301 `s_int32_t smi_peer_route_reflector_client_set (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr, int afi, int safi)`

sets the neighbor route reflector client `smi_peer_route_reflector_client_set`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *afi* Address family identifier
- ← *safi* Sub-address family identifier
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.302 `s_int32_t smi_peer_route_reflector_client_unset (struct smiclient_globals * azg, u_int32_t vrId, char * peerAddr, int afi, int safi)`

unsets the neighbor route reflector client `smi_neighbor_route_reflector_client_unset`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *afi* Address family identifier
- ← *safi* Sub-address family identifier
- ← *vrId* Virtual router id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.303 `s_int32_t smi_peer_shutdown_set (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr)`

sets neighbor shutdown `smi_peer_shutdown_set`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.304 s_int32_t smi_peer_shutdown_set_validate (struct smiclient_globals * azg, u_int32_t vrId, char * vrf_name, char * peerAddr)

sets neighbor shutdown smi_peer_shutdown_set_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *vrId* Virtual router-id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.305 s_int32_t smi_peer_shutdown_unset (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr)

unsets neighbor shutdown smi_peer_shutdown_unset

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.306 `s_int32_t smi_peer_shutdown_unset_validate (struct smiclient_globals * azg, u_int32_t vrId, char * vrf_name, char * peerAddr)`

unsets neighbor shutdown smi_peer_shutdown_unset_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *vrId* Virtual router-id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.307 `s_int32_t smi_peer_soft_reconfiguration_inbound_set (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr, int afi, int safi)`

sets the neighbor soft reconfiguration smi_peer_soft_reconfiguration_inbound_set

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *afi* Address family identifier
- ← *safi* Sub-address family identifier
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.308 `s_int32_t smi_peer_soft_reconfiguration_inbound_set_validate (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr, int afi, int safi)`

sets the neighbor soft reconfiguration smi_peer_soft_reconfiguration_inbound_set_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *afi* Address family identifier
- ← *safi* Sub-address family identifier
- ← *vrId* Virtual router-id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.309 `s_int32_t smi_peer_soft_reconfiguration_inbound_unset (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr, int afi, int safi)`

unsets the neighbor soft reconfiguration `smi_peer_soft_reconfiguration_inbound_unset`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *afi* Address family identifier
- ← *safi* Sub-address family identifier
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.310 `s_int32_t smi_peer_soft_reconfiguration_inbound_unset_validate (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr, int afi, int safi)`

unsets the neighbor soft reconfiguration `smi_peer_soft_reconfiguration_inbound_unset_validate`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address

- ← *afi* Address family identifier
- ← *safi* Sub-address family identifier
- ← *vrId* Virtual router-id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.311 `int smi_peer_timers_connect_set_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, char * peerAddr, u_int32_t peerConnectInterval)`

Sets time interval (in seconds) for the ConnectRetry timer. `smi_peer_timers_connect_set_sdkapi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address BGP peer IP address or Tag
- ← *peerConnectInterval* Connect timer value <1-65535>
- ← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_INVALID_VALUE
 BGP_API_SET_ERR_INVALID_FOR_PEER_GROUP_MEMBER
 BGP_API_SET_ERR_PEER_GROUP_HAS_THE_FLAG

2.1.2.312 `int smi_peer_timers_connect_unset_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, char * peerAddr)`

Unsets time interval (in seconds) for the ConnectRetry timer. `smi_peer_timers_connect_unset_sdkapi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address BGP peer IP address or Tag
- ← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_INVALID_FOR_PEER_GROUP_MEMBER
 BGP_API_SET_ERR_PEER_GROUP_HAS_THE_FLAG

2.1.2.313 `int smi_peer_timers_set_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, char * peerAddr, u_int16_t keepAlive, u_int16_t holdTime)`

Sets the time intervals in seconds for peer's Hold Timer and KeepAlive Timer. `smi_peer_timers_set_sdkapi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address BGP peer IP address or Tag
- ← *keepAlive* BGP keepalive time <0|3-65535>
- ← *holdTime* BGP holdtime <0|1-21845>
- ← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_INVALID_VALUE
 BGP_API_SET_ERR_INVALID_FOR_PEER_GROUP_MEMBER
 BGP_API_SET_ERR_PEER_GROUP_HAS_THE_FLAG
 BGP_API_SET_ERR_INFINITE_HOLD_TIME_VALUE
 BGP_API_SET_WARN_HOLD_AND_KEEPALIVE_INVALID
 BGP_API_SET_ERR_INVALID_HOLD_TIME

2.1.2.314 `int smi_peer_timers_unset_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, char * peerAddr)`

Sets the time intervals to default in seconds for peer's Hold Timer and KeepAlive Timer. `smi_peer_timers_unset_sdkapi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address BGP peer IP address or Tag
- ← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_INVALID_FOR_PEER_GROUP_MEMBER
 BGP_API_SET_ERR_PEER_GROUP_HAS_THE_FLAG

2.1.2.315 `s_int32_t smi_peer_transport_connection_passive_set (struct smiclient_globals * azg, u_int32_t vrId, char * vrfName, char * peerAddr)`

sets neighbor passive `smi_peer_transport_connection_passive_set`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.316 `s_int32_t smi_peer_transport_connection_passive_set_validate` (struct smiclient_globals * *azg*, char * *peerAddr*, u_int32_t *vrId*)

sets neighbor passive smi_peer_transport_connection_passive_set_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *vrId* Virtual router-id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.317 `s_int32_t smi_peer_transport_connection_passive_unset` (struct smiclient_globals * *azg*, u_int32_t *vrId*, char * *vrfName*, char * *peerAddr*)

unsets neighbor passive smi_peer_transport_connection_passive_unset

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *vrId* Virtual router id
- ← *vrfName* VRF name (default/ VRF name)

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_GET_ERROR

2.1.2.318 `s_int32_t smi_peer_transport_connection_passive_unset_validate` (struct smiclient_globals * *azg*, u_int32_t *vrId*, char * *peerAddr*)

unsets neighbor passive smi_peer_transport_connection_passive_unset_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address
- ← *vrId* Virtual router-id

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.319 `int smi_peer_unsuppress_map_set_sdkapi_validate` (struct smiclient_globals * *azg*, u_int32_t *bgpAs*, char * *peerAddr*, char * *unSuppressAclInfo*, u_int32_t *vrId*)

Sets the Route-Map to selectively unsuppress suppressed routes. smi_peer_unsuppress_map_set_sdkapi

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address BGP peer IP address or Tag
- ← *afi* Address family identifier Address family identifier <1-2> (1=IP | 2=IP6)
- ← *safi* Sub-address family identifier Sub-Address family identifier <1-4> (UC|MC|UC_MC|MPLS)
- ← *aclInfo* Access-list info Access-list number
- ← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
BGP_API_SET_ERR_INVALID_FOR_PEER_GROUP_MEMBER
BGP_API_SET_ERR_PEER_GROUP_HAS_THE_FLAG
BGP_API_SET_ERR_PEER_INACTIVE

2.1.2.320 `int smi_peer_unsuppress_map_unset_sdkapi_validate` (struct smiclient_globals * *azg*, u_int32_t *bgpAs*, char * *peerAddr*, u_int32_t *vrId*)

Unsets the Route-Map to selectively unsuppress suppressed routes. smi_peer_unsuppress_map_unset_sdkapi

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address BGP peer IP address or Tag
- ← *afi* Address family identifier Address family identifier <1-2> (1=IP | 2=IP6)
- ← *safi* Sub-address family identifier Sub-Address family identifier <1-4> (UC|MC|UC_MC|MPLS)
- ← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_INVALID_FOR_PEER_GROUP_MEMBER
 BGP_API_SET_ERR_PEER_GROUP_HAS_THE_FLAG
 BGP_API_SET_ERR_PEER_INACTIVE

2.1.2.321 `int smi_peer_update_routing_source_set_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, char * peerAddr, char * sourceId)`

Sets the source for routing updates. smi_peer_update_routing_source_set_sdkapi

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address BGP peer IP address or Tag
- ← *sourceId* Source for routing updates
- ← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_PEER_GROUP_HAS_THE_FLAG

2.1.2.322 `int smi_peer_version_set_sdkapi_validate (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, char * peerAddr, u_int32_t bgpVersion)`

Sets the Neighbor's BGP version. smi_peer_version_set_sdkapi

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address BGP peer IP address or Tag
- ← *bgpVersion* BGP Version
- ← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_INVALID_VALUE
 BGP_API_SET_ERR_INVALID_FOR_PEER_GROUP_MEMBER

2.1.2.323 int smi_peer_version_unset_sdkapi_validate (struct smiclient_globals * *azg*, u_int32_t *vrId*, u_int32_t *bgpAs*, char * *peerAddr*)

This function unsets the BGP version. smi_peer_version_unset_sdkapi

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address BGP peer IP address or Tag
- ← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success.

2.1.2.324 int smi_peer_weight_set_sdkapi_validate (struct smiclient_globals * *azg*, u_int32_t *vrId*, u_int32_t *bgpAs*, char * *peerAddr*, u_int16_t *weight*)

Sets the default weight for routes from this port of neighbors. smi_peer_weight_set_sdkapi

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address BGP peer IP address or Tag
- ← *weight* Weight for routes <0-65535>
- ← *afi* Address family identifier Address family identifier <1-2> (1=IP | 2=IP6)
- ← *safi* Sub-address family identifier Sub-Address family identifier <1-4> (UC|MC|UC_MC|MPLS)
- ← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
 BGP_API_SET_ERR_PEER_GROUP_HAS_THE_FLAG

2.1.2.325 `int smi_peer_weight_unset_sdkapi (struct smiclient_globals * azg, u_int32_t vrId, u_int32_t bgpAs, char * peerAddr)`

Unsets the default weight for routes from this port of neighbors. `smi_peer_weight_unset_sdkapi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address BGP peer IP address or Tag
- ← *afi* Address family identifier Address family identifier <1-2> (1=IP | 2=IP6)
- ← *safi* Sub-address family identifier Sub-Address family identifier <1-4> (UC|MC|UC_MC|MPLS)
- ← *vrId* Virtual Router Id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
BGP_API_SET_ERR_PEER_GROUP_HAS_THE_FLAG

2.1.2.326 `int smi_show_bgp_afi_regexp_safi (struct smiclient_globals * azg, char * vrfName, char * bgpRegExp, char * af, char * saf, struct list * showList, int(*) (struct list * showlist) callback)`

show ip bgp paths displays the ipv4/ipv6 routes matching the AS path regular expression `smi_show_bgp_afi_regexp_safi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrfName* VRF Name.
- ← *regExp* identifies regular expression.
- ← *af* Address family identifier (ipv4/ipv6)
- ← *saf* Sub-Address family identifier (multicast/unicast)
- *showList, Pointer* to the linked list of structure `bgpProcessInfo`
- ← *callbackFunc* Callback func pointer

Returns:

BGP_API_SHOW_SUCCESS on success, otherwise error codes: SMI_ERROR

2.1.2.327 `int smi_show_bgp_afi_route_map_safi (struct smiclient_globals * azg, char * vrfName, char * bgpRegExp, char * af, char * saf, struct list * showList, int(*) (struct list * showlist) callback)`

show ip bgp paths displays the ipv4/ipv6 routes matching the route-map `smi_show_bgp_afi_route_map_safi`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrfName* VRF Name.
- ← *regExp* identifies regular expression.
- ← *af* Address family identifier (ipv4/ipv6)
- ← *saf* Sub-Address family identifier (multicast/unicast)
- *showList,Pointer* to the linked list of structure bgpProcessInfo
- ← *callbackFunc* Callback func pointer

Returns:

BGP_API_SHOW_SUCCESS on success, otherwise error codes: SMI_ERROR

2.1.2.328 `int smi_show_bgp_dampening_parameters (struct smiclient_globals * azg, int afi, int safi, struct rfdConfigData * rfdOutInfo, struct list * bgpDampeningParaOutList, u_int32_t(*) (struct list * bgpDampeningParaOutList) callbackFunc)`

show bgp info about dampening, smi_show_bgp_dampening_parameters

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *afi* Address family identifier Address family identifier <1-2> (1=IP | 2=IP6)
- ← *safi* Sub-address family identifier Sub-Address family identifier <1-4> (UC|MC|UC_MC|MPLS)
- *rfdOutInfo* struct rfdConfigData pointer containing all information about dampening parameters
- *bgpDampeningParaOutList* Pointer to the linked list of structure rfdListLoop-Data
- ← *callbackFunc* Callback func pointer

Returns:

0 on success, otherwise error codes: SMI_ERROR

2.1.2.329 `int smi_show_bgp_inconsistent_as (struct smiclient_globals * azg, char * af, char * saf, struct list * showList, int(*) (struct list * showlist) callback)`

show bgp multicast/unicast route with inconsistent AS path for IPv4/IPv6 environment
smi_show_bgp_inconsistent_as

Parameters:

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

- ← **afi** Address family identifier Address family identifier (ipv4/ipv6)
- ← **safi** Sub-address family identifier Sub-Address family identifier (multicast/unicast)
- **showList,Pointer** to the linked list of structure bgpProcessInfo
- ← **callbackFunc** Callback func pointer

Returns:

0 on success, otherwise error codes: SMI_ERROR

2.1.2.330 `int smi_show_bgp_ip_neighbor_routes (struct smiclient_globals * azg, char * vrfName, char * peerAddr, char * af, char * saf, struct list * showList, int(*) (struct list * showlist) callback)`

show ip bgp paths displays the information on TCP and BGP ipv4/ipv6 neighbor connections smi_show_bgp_ip_neighbor_routes

Parameters:

- ← **azg** Pointer to the SMI client global structure
- ← **vrfName** VRF Name.
- ← **peerAddr** Peer IP address
- ← **af** Address family identifier (ipv4/ipv6)
- ← **saf** Sub-Address family identifier (multicast/unicast)
- **showList,Pointer** to the linked list of structure bgpProcessInfo
- ← **callbackFunc** Callback func pointer

Returns:

BGP_API_SHOW_SUCCESS on success, otherwise error codes: SMI_ERROR

2.1.2.331 `int smi_show_bgp_neighbor_advertised_routes (struct smiclient_globals * azg, char * ipAddr, char * name, char * af, char * saf, struct list * bgpShowList, int(*) (struct list * showlist) callback)`

show advertised routes for all neighbors smi_show_bgp_neighbor_advertised_routes

Parameters:

- ← **azg** Pointer to the SMI client global structure
- ← **name** VRF name
- ← **ipAddr** address Peer IP address
- ← **afi** Address family identifier Address family identifier (ipv4/ipv6)
- ← **safi** Sub-address family identifier Sub-Address family identifier (multicast/unicast)

- *showList,Pointer* to the linked list of structure BgpPeerAdjInfo
- ← *callbackFunc* Callback func pointer

Returns:

0 on success, otherwise error codes: SMI_ERROR

2.1.2.332 `int smi_show_bgp_neighbor_recieved_routes (struct smiclient_globals * azg, char * ipAddr, char * name, char * af, char * saf, struct list * bgpShowList, int(*)(struct list *showlist) callback)`

show recieved routes for all neighbors smi_show_bgp_neighbor_recieved_routes

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *name* VRF name
- ← *ipAddr* address Peer IP address
- ← *afi* Address family identifier Address family identifier (ipv4/ipv6)
- ← *safi* Sub-address family identifier Sub-Address family identifier (multicast/unicast)
- *bgpShowList,Pointer* to the linked list of structure BgpPeerAdjInfo
- ← *callbackFunc* Callback func pointer

Returns:

0 on success, otherwise error codes: SMI_ERROR

2.1.2.333 `int smi_show_bgp_neighbors_rcv_prefix_filter (struct smiclient_globals * azg, char * ipAddr, char * af, char * saf, struct list * bgpShowList, int(*)(struct list *showlist) callback)`

neighbors matching the given prefix filter smi_show_bgp_neighbors_rcv_prefix_filter

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *ipAddr* address passed for the neighbor
- ← *afi* Address family identifier Address family identifier (ipv4/ipv6)
- ← *safi* Sub-address family identifier Sub-Address family identifier (multicast/unicast)
- *bgpShowList,Pointer* to the linked list of structure smiBGPNeighborPrefixInfo
- ← *callbackFunc* Callback func pointer

Returns:

0 on success, otherwise error codes: SMI_ERROR

2.1.2.334 `int smi_show_bgp_regex (struct smiclient_globals * azg, char * bgpRegExp, struct list * showList, int(*)(struct list *showlist) callback)`

show ip bgp paths displays the routes matching the AS path regular expression smi_show_bgp_regex

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrfName* VRF Name.
- ← *regExp* identifies regular expression.
- ← *af* Address family identifier (ipv4/ipv6)
- ← *saf* Sub-Address family identifier (multicast/unicast)
- ← *option* vrf_option can be passed as default, all or VRF name
- *showList,Pointer* to the linked list of structure bgpProcessInfo
- ← *callbackFunc* Callback func pointer

Returns:

BGP_API_SHOW_SUCCESS on success, otherwise error codes: SMI_ERROR

2.1.2.335 `int smi_show_bgp_route_map (struct smiclient_globals * azg, char * bgpRegExp, struct list * showList, int(*)(struct list *showlist) callback)`

show ip bgp paths displays the routes matching the route-map smi_show_bgp_route_map

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrfName* VRF Name.
- ← *regExp* identifies regular expression.
- *showList,Pointer* to the linked list of structure bgpProcessInfo
- ← *callbackFunc* Callback func pointer

Returns:

BGP_API_SHOW_SUCCESS on success, otherwise error codes: SMI_ERROR

2.1.2.336 `int smi_show_bgp_sessions (struct smiclient_globals * azg, char * name, struct list * showList, int(*)(struct list *showlist) callback)`

show bgp established session info smi_show_bgp_session

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *name* VRF name
- *showList,Pointer* to the linked list of structure bgpSession
- ← *callbackFunc* Callback func pointer

Returns:

0 on success, otherwise error codes: SMI_ERROR

2.1.2.337 `int smi_show_bgp_summary (struct smiclient_globals * azg, char * vrfName, u_int8_t safi, struct list * bgpSummaryList, u_int32_t(*) (struct list * bgpSummaryList) callbackFunc)`

show bgp summary info of neighbor status for IPv4 environment smi_show_bgp_summary

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrfName* VRF name
- ← *safi* Sub-Address family identifier <1,2> (MC|UC)
- *bgpSummaryList* Pointer to the linked list of structure showBgpSummary
- ← *callbackFunc* Callback func pointer

Returns:

0 on success, otherwise error codes: SMI_ERROR

2.1.2.338 `int smi_show_bgp_V6_neighbors_rcv_prefix_filter (struct smiclient_globals * azg, char * ipAddr, char * af, char * saf, struct list * bgpShowList, int(*) (struct list * showlist) callback)`

neighbors matching the given prefix filter fro ipv6 address smi_show_bgp_V6_neighbors_rcv_prefix_filter

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *ipAddr* address passed for the neighbor
- ← *afi* Address family identifier Address family identifier (ipv4/ipv6)
- ← *safi* Sub-address family identifier Sub-Address family identifier (multicast/unicast)
- *bgpShowList,Pointer* to the linked list of structure smiBGPNeighborPrefixInfo
- ← *callbackFunc* Callback func pointer

Returns:

0 on success, otherwise error codes: SMI_ERROR

2.1.2.339 `int smi_show_ip_bgp (struct smiclient_globals * azg, char * vrfName, char * af, char * saf, struct list * showList, int(*) (struct list * showlist) callback)`

show bgp info about neighbor instance of the given instance `smi_show_ip_bgp`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrfName* VRF name
- ← *afi* Address family identifier Address family identifier (ipv4/ipv6)
- ← *safi* Sub-address family identifier Sub-Address family identifier (multicast/unicast)
- *showList,Pointer* to the linked list of structure `bgpProcessInfo`
- ← *callbackFunc* Callback func pointer

Returns:

0 on success, otherwise error codes: `SMI_ERROR`

2.1.2.340 `int smi_show_ip_bgp_cidr_only (struct smiclient_globals * azg, char * af, char * saf, struct list * showList, int(*) (struct list * showlist) callback)`

show bgp routes info with non-natural network mask `smi_show_ip_bgp_cidr_only`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *afi* Address family identifier Address family identifier (ipv4/ipv6)
- ← *safi* Sub-address family identifier Sub-Address family identifier (multicast/unicast)
- *showList,Pointer* to the linked list of structure `bgpProcessInfo`
- ← *callbackFunc* Callback func pointer

Returns:

0 on success, otherwise error codes: `SMI_ERROR`

2.1.2.341 `int smi_show_ip_bgp_community (struct smiclient_globals * azg, char * vrfName, char * af, char * saf, struct list * showList, int(*) (struct list * showlist) callback)`

show bgp info about neighbor instance of the given instance `smi_show_ip_bgp_community`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrfName* VRF name
- ← *afi* Address family identifier Address family identifier (ipv4/ipv6)
- ← *safi* Sub-address family identifier Sub-Address family identifier (multicast/unicast)
- *showList,Pointer* to the linked list of structure *bgpProcessInfo*
- ← *callbackFunc* Callback func pointer

Returns:

0 on success, otherwise error codes: SMI_ERROR

2.1.2.342 `int smi_show_ip_bgp_dampening_dampend_paths (struct smiclient_globals *azg, char *vrfName, struct list *bgpDampendPathList, u_int32_t (*)(struct list *bgpDampendPathList) callbackFunc)`

show bgp information about dampening dampened parameters info `smi_show_ip_bgp_dampening_dampend_paths`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrfName* VRF name
- *bgpDampendPathList* Pointer to the linked list of structure *bgpDampenedPathInfo*
- ← *callbackFunc* Callback func pointer

Returns:

0 on success, otherwise error codes: SMI_ERROR

2.1.2.343 `int smi_show_ip_bgp_dampening_flap_statistics (struct smiclient_globals *azg, char *vrfName, int vrfOption, struct list *bgpDampingFlapList, u_int32_t (*)(struct list *bgpDampingFlapList) callbackFunc)`

show bgp information about dampening flapping statistics `smi_show_ip_bgp_dampening_flap_statistics`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrfName* VRF name

- ← *vrfOption* VRF option which can be passed as follows : 1 = SMI_CONFIG_VRF_ALL 2 = SMI_CONFIG_VRF_DEFAULT
- *bgpDampingFlapList* Pointer to the linked list of structure *bgpFlappingStatsInfo*
- ← *callbackFunc* Callback func pointer

Returns:

0 on success, otherwise error codes: SMI_ERROR

2.1.2.344 `int smi_show_ip_bgp_filter_list_exact_match (struct smiclient_globals *azg, char *filterList_name, char *vrfName, char *afi, char *safi, int type, struct list *showList, int(*) (struct list *showlist) callback)`

show bgp multicast/unicast filter list for IPv4/IPv6 environment *smi_show_ip_bgp_filter_list_exact_match*

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *filterList_name* Name of the filter list
- ← *vrfName* VRF name
- ← *afi* Address family identifier Address family identifier (ipv4/ipv6)
- ← *safi* Sub-address family identifier Sub-Address family identifier (multicast/unicast)
- ← *type* Flags for choosing the type of output 0 = SMI_EXACT_MATCH_LIST For the exact-match prefix list 1 = SMI_NORMAL_LIST For the normal prefix list
- *showList,Pointer* to the linked list of structure *bgpProcessInfo*
- ← *callbackFunc* Callback func pointer

Returns:

0 on success, otherwise error codes: SMI_ERROR

2.1.2.345 `int smi_show_ip_bgp_ipv6_dampening_parameters (struct smiclient_globals *azg, char *vrfName, int afi, int safi, struct rfdConfigData *rfdOutInfo, struct list *bgpDampeningParaOutList, u_int32_t(*) (struct list *bgpDampeningParaOutList) callbackFunc)`

show bgp info about dampening parameters info *smi_show_ip_bgp_ipv6_dampening_parameters*

Parameters:

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

- ← *vrfName* VRF name
- ← *afi* Address family identifier Address family identifier <1,2> (1=IP | 2=IP6)
- ← *safi* Sub-address family identifier Sub-Address family identifier <1,2> (MC|UC)
- *rfdOutInfo* struct rfdConfigData pointer containing all information about dampening parameters
- *bgpDampeningParaOutList* Pointer to the linked list of structure rfdListLoop-Data
- ← *callbackFunc* Callback func pointer

Returns:

0 on success, otherwise error codes: SMI_ERROR

2.1.2.346 `int smi_show_ip_bgp_longer_prefixes (struct smiclient_globals * azg, char * prefix, char * vrfName, u_int8_t afi, struct list * bgpPrefixList, u_int32_t(*) (struct list * bgpPrefixList) callbackFunc)`

show bgp network information from the mask information smi_show_ip_bgp_longer_prefixes

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *prefix* IPv4 prefix address
- ← *vrfName* VRF name
- ← *afi* Address family identifier <1,2> (1=IP | 2=IP6)
- *bgpPrefixList* Pointer to the linked list of structure bgpProcessInfo
- ← *callbackFunc* Callback func pointer

Returns:

0 on success, otherwise error codes: SMI_ERROR

2.1.2.347 `int smi_show_ip_bgp_neighbors_HKC (struct smiclient_globals * azg, char * name, char * ipAddr, char * type, struct list * bgpShowHKCList, int(*) (struct list * showlist) callback)`

show hold-time | keepalive-interval | connection-retry for all neighbors smi_show_ip_bgp_neighbors_HKC

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *name* VRF name

- ← *option* type which can be passed as follows : 2= show_hold_time 3= show_keeplive_interval 4= show_connection_retrytime
- ← *ipAddr* address passed for the neighbor
- *bgpShowHKCList,Pointer* to the linked list of structure BgpHKCNeighborInfo
- ← *callbackFunc* Callback func pointer

Returns:

0 on success, otherwise error codes: SMI_ERROR

2.1.2.348 `int smi_show_ip_bgp_paths (struct smiclient_globals * azg, struct list * showList, int(*) (struct list * showlist) callback)`

show ip bgp paths displays the path information smi_show_ip_bgp_paths

Parameters:

- ← *azg* Pointer to the SMI client global structure
- *showList,Pointer* to the linked list of structure bgpPathInfo
- ← *callbackFunc* Callback func pointer

Returns:

BGP_API_SHOW_SUCCESS on success, otherwise error codes: SMI_ERROR

2.1.2.349 `int smi_show_ip_bgp_prefix_list_exact_match (struct smiclient_globals * azg, char * prefixList_name, char * vrfName, char * afi, char * saf, int type, struct list * showList, int(*) (struct list * showlist) callback)`

show bgp routes matching the prefix list name otherwise show all the prefix list smi_show_ip_bgp_prefix_list_exact_match

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrfName* VRF name
- ← *afi* Address family identifier Address family identifier (ipv4/ipv6)
- ← *safi* Sub-address family identifier Sub-Address family identifier (multicast/unicast)
- ← *type* Flags for choosing the type of output 0 = SMI_EXACT_MATCH_LIST For the exact-match prefix list 1 = SMI_NORMAL_LIST For the normal prefix list
- *showList,Pointer* to the linked list of structure bgpProcessInfo

← *callbackFunc* Callback func pointer

Returns:

0 on success, otherwise error codes: SMI_ERROR

2.1.2.350 `int smi_show_ip_bgp_quote_regexp (struct smiclient_globals * azg,
char * vrfName, char * bgpRegExp, char * af, char * saf, struct list
* showList, int(*)(struct list *showlist) callback)`

show ip bgp paths displays the ipv4/ipv6 routes matching the AS path quote-regular expression word smi_show_ip_bgp_quote_regexp

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrfName* VRF Name.
- ← *regExp* identifies regular expression.
- ← *af* Address family identifier (ipv4/ipv6)
- ← *saf* Sub-Address family identifier (multicast/unicast)
- *showList,Pointer* to the linked list of structure bgpProcessInfo
- ← *callbackFunc* Callback func pointer

Returns:

BGP_API_SHOW_SUCCESS on success, otherwise error codes: SMI_ERROR

2.1.2.351 `int smi_show_ip_bgp_received_paths (struct smiclient_globals * azg,
char * vrfName, struct list * bgpReceivedPathList, u_int32_t(*)(struct
list *bgpReceivedPathList) callbackFunc)`

show bgp neighbor information for the received paths smi_show_ip_bgp_received_paths

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrfName* VRF name
- *bgpReceivedPathList* Pointer to the linked list of structure bgpShowReceived-Path
- ← *callbackFunc* Callback func pointer

Returns:

0 on success, otherwise error codes: SMI_ERROR

2.1.2.352 `int smi_show_ip_bgp_regex (struct smiclient_globals * azg, char * vrfName, char * bgpRegExp, char * af, char * saf, struct list * showList, int(*) (struct list * showlist) callback)`

show ip bgp paths displays the routes matching the AS path regular expression smi_show_ip_bgp_regex

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrfName* VRF Name.
- ← *regExp* identifies regular expression.
- ← *af* Address family identifier (ipv4/ipv6)
- ← *saf* Sub-Address family identifier (multicast/unicast)
- *showList,Pointer* to the linked list of structure bgpProcessInfo
- ← *callbackFunc* Callback func pointer

Returns:

BGP_API_SHOW_SUCCESS on success, otherwise error codes: SMI_ERROR

2.1.2.353 `int smi_show_ip_bgp_route_map (struct smiclient_globals * azg, char * vrfName, char * bgpRegExp, char * af, char * saf, struct list * showList, int(*) (struct list * showlist) callback)`

show ip bgp paths displays the routes matching the route-map smi_show_ip_bgp_route_map

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrfName* VRF Name.
- ← *regExp* identifies regular expression.
- ← *af* Address family identifier (ipv4/ipv6)
- ← *saf* Sub-Address family identifier (multicast/unicast)
- *showList,Pointer* to the linked list of structure bgpProcessInfo
- ← *callbackFunc* Callback func pointer

Returns:

BGP_API_SHOW_SUCCESS on success, otherwise error codes: SMI_ERROR

2.1.2.354 `int smi_show_ip_bgp_safi_regexp (struct smiclient_globals * azg,
char * vrfName, char * bgpRegExp, char * af, char * saf, struct list
* showList, int(*)(struct list *showlist) callback)`

show ip bgp paths displays the ipv4 routes matching the AS path regular expression
smi_show_ip_bgp_safi_regexp

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrfName* VRF Name.
- ← *regExp* identifies regular expression.
- ← *af* Address family identifier (ipv4/ipv6)
- ← *saf* Sub-Address family identifier (multicast/unicast)
- *showList,Pointer* to the linked list of structure bgpProcessInfo
- ← *callbackFunc* Callback func pointer

Returns:

BGP_API_SHOW_SUCCESS on success, otherwise error codes: SMI_ERROR

2.1.2.355 `int smi_show_ip_bgp_safi_route_map (struct smiclient_globals * azg,
char * vrfName, char * bgpRegExp, char * af, char * saf, struct list
* showList, int(*)(struct list *showlist) callback)`

show ip bgp paths displays the ipv4 routes matching the route-map smi_show_ip_
bgp_safi_route_map

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrfName* VRF Name.
- ← *regExp* identifies regular expression.
- ← *af* Address family identifier (ipv4/ipv6)
- ← *saf* Sub-Address family identifier (multicast/unicast)
- *showList,Pointer* to the linked list of structure bgpProcessInfo
- ← *callbackFunc* Callback func pointer

Returns:

BGP_API_SHOW_SUCCESS on success, otherwise error codes: SMI_ERROR

2.1.2.356 `int smi_show_ip_bgp_summary (struct smiclient_globals * azg,
char * vrfName, int afi, int safi, struct list * bgpSummaryList,
u_int32_t(*) (struct list * bgpSummaryList) callbackFunc)`

show bgp summary info of neighbor status `smi_show_ip_bgp_summary`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrfName* VRF name
- ← *afi* Address family identifier Address family identifier <1,2> (1=IP | 2=IP6)
- ← *safi* Sub-address family identifier Sub-Address family identifier <1,2> (MC|UC)
- *bgpSummaryList* Pointer to the linked list of structure `showBgpSummary`
- ← *callbackFunc* Callback func pointer

Returns:

0 on success, otherwise error codes: SMI_ERROR

2.1.2.357 `int smi_show_ip_bgp_word_neighbors (struct smiclient_globals *
azg, char * name, char * ipAddr, struct list * bgpShowAllList,
int(*) (struct list * showlist) callback)`

show neighbors matching the neighbor name or else show for aall neighbors `smi_show_ip_bgp_word_neighbors`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *name* VRF name
- ← *ipAddr* address passed for the neighbor
- *bgpShowAllList,Pointer* to linked list of structure `BgpNeighborInfo`
- ← *callbackFunc* Callback func pointer

Returns:

0 on success, otherwise error codes: SMI_ERROR

2.1.2.358 `int smi_show_ip_bgp_word_peer_neighbors (struct smiclient_globals
* azg, char * name, char * ipAddr, struct list * bgpShowPeerList,
int(*) (struct list * showlist) callback)`

show neighbors matching the neighbor ipaddress or else show for all neighbors `smi_show_ip_bgp_word_peer_neighbors`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *name* VRF name
- ← *ipAddr* address passed for the neighbor
- *bgpShowPeerList,Pointer* to linked list of structure BgpNeighborInfo
- ← *callbackFunc* Callback func pointer

Returns:

0 on success, otherwise error codes: SMI_ERROR

2.1.2.359 `int smi_show_ip_protocol_all (struct smiclient_globals * azg, char * af, struct list * showList, int(*)(struct list *showlist) callback)`

show bgp information about protocols smi_show_ip_protocol_all

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vrfName* VRF name
- ← *vrf_option* VRF option which can be passed as follows : 1 = SMI_CONFIG_VRF_ALL 2 = SMI_CONFIG_VRF_DEFAULT
- *showList* Pointer to the returned link list of BgpProtocolInf, Which contains protocol Information.
- ← *callbackFunc* Callback func pointer

Returns:

0 on success, otherwise error codes: SMI_ERROR

2.1.2.360 `s_int32_t smi_transport_connection_passive_set_validate (struct smiclient_globals * azg, char * peerAddr)`

sets the transport connection passive smi_transport_connection_passive_set_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.1.2.361 `s_int32_t smi_transport_connection_passive_unset_validate (struct smiclient_globals * azg, char * peerAddr)`

unsets the transport connection passive smi_transport_connection_passive_unset_validate

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *peerAddr* The peer address

Returns:

BGP_API_GET_SUCCESS on success, otherwise one of the following error codes
BGP_API_GET_ERROR

2.2 smi_bgp_bfd.h File Reference

Provides APIs for managing Bidirectional Forwarding Detection(BFD) in ZebOS.

```
#include "smi_client.h"
```

```
#include "smi_bgp_bfd_msg.h"
```

Functions

- int [smi_bgp_peer_bfd_set](#) (struct smiclient_globals *azg, u_int32_t vr_id, vrf_id_t vrf_id, char *peer_str, bool_t mh)

This function Sets the BFD fall-over check for the specified peer.

- int [smi_bgp_peer_bfd_unset](#) (struct smiclient_globals *azg, u_int32_t vr_id, vrf_id_t vrf_id, char *peer_str)

This funtion Unsets the BFD fall-over check for the specified peer.

- int [smi_bgp_peer_bfd_wrap_validate](#) (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, char *peer_str, bool_t mh)
- int [smi_bgp_peer_bfd_wrap](#) (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, char *peer_str, bool_t mh)
- int [smi_bgp_peer_bfd_mh_wrap_validate](#) (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, char *peer_str, bool_t peerBfd, bool_t mh)
- int [smi_bgp_peer_bfd_mh_wrap](#) (struct smiclient_globals *azg, u_int32_t vr_id, u_int32_t bgpAs, char *peer_str, bool_t peerBfd, bool_t mh)

2.2.1 Detailed Description

Provides APIs for managing Bidirectional Forwarding Detection(BFD) in ZebOS.

2.2.2 Function Documentation

2.2.2.1 int [smi_bgp_peer_bfd_set](#) (struct smiclient_globals * azg, u_int32_t vr_id, vrf_id_t vrf_id, char * peer_str, bool_t mh)

This function Sets the BFD fall-over check for the specified peer. [smi_bgp_peer_bfd_set](#)

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vr_id* Virtual Router Id
- ← *vrf_id* Virtual Routing and Forwarding Id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes

BGP_API_SET_ERR_INVALID_BGP
BGP_API_SET_ERR_PEER_MALFORMED_ADDRESS
BGP_API_SET_ERR_PEER_SELF_ADDRESS
BGP_API_SET_ERR_PEER_UNINITIALIZED
BGP_API_SET_ERR_PEER_DUPLICATE

2.2.2.2 `int smi_bgp_peer_bfd_unset (struct smiclient_globals * azg, u_int32_t vr_id, vrf_id_t vrf_id, char * peer_str)`

This function Unsets the BFD fall-over check for the specified peer. `smi_bgp_peer_bfd_unset`

Parameters:

- ← *azg* Pointer to the SMI client global structure
- ← *vr_id* Virtual Router Id
- ← *vrf_id* Virtual Routing and Forwarding Id

Returns:

BGP_API_SET_SUCCESS on success, otherwise one of the following error codes
BGP_API_SET_ERR_INVALID_BGP
BGP_API_SET_ERR_PEER_MALFORMED_ADDRESS
BGP_API_SET_ERR_PEER_SELF_ADDRESS
BGP_API_SET_ERR_PEER_UNINITIALIZED
BGP_API_SET_ERR_PEER_DUPLICATE

Index

- smi_bgp.h, [3](#)
- smi_bgp4_get_path_attr_-
aggregator_addr_sdkapi, [70](#)
- smi_bgp4_get_path_attr_-
aggregator_as_sdkapi, [70](#)
- smi_bgp4_get_path_attr_atomic_-
aggregate_sdkapi, [70](#)
- smi_bgp4_get_path_attr_best_-
sdkapi, [71](#)
- smi_bgp4_get_path_attr_calc_-
local_pref_sdkapi, [71](#)
- smi_bgp4_get_path_attr_ip_addr_-
prefix_len_sdkapi, [72](#)
- smi_bgp4_get_path_attr_ip_addr_-
prefix_sdkapi, [72](#)
- smi_bgp4_get_path_attr_local_-
pref_sdkapi, [73](#)
- smi_bgp4_get_path_attr_multi_-
exit_disc_sdkapi, [73](#)
- smi_bgp4_get_path_attr_next_-
hop_sdkapi, [74](#)
- smi_bgp4_get_path_attr_origin_-
sdkapi, [74](#)
- smi_bgp4_get_path_attr_peer_-
sdkapi, [75](#)
- smi_bgp_address_family_set, [75](#)
- smi_bgp_af_config_check_sdkapi,
[76](#)
- smi_bgp_aggregate_nexthop_-
check_set, [76](#)
- smi_bgp_aggregate_nexthop_-
check_set_validate, [77](#)
- smi_bgp_aggregate_nexthop_-
check_unset, [77](#)
- smi_bgp_aggregate_nexthop_-
check_unset_validate, [77](#)
- smi_bgp_always_compare_med_set,
[78](#)
- smi_bgp_always_compare_med_-
set_validate, [78](#)
- smi_bgp_always_compare_med_-
unset, [78](#)
- smi_bgp_always_compare_med_-
unset_validate, [79](#)
- smi_bgp_api_address_family_unset,
[79](#)
- smi_bgp_aspath_access_list_set_-
validate, [79](#)
- smi_bgp_aspath_access_list_unset_-
validate, [80](#)
- smi_bgp_auto_summary_update_-
set_sdkapi_validate, [80](#)
- smi_bgp_bestpath_aspath_ignore_-
set, [81](#)
- smi_bgp_bestpath_aspath_ignore_-
set_validate, [81](#)
- smi_bgp_bestpath_aspath_ignore_-
unset, [81](#)
- smi_bgp_bestpath_aspath_ignore_-
unset_validate, [82](#)
- smi_bgp_bestpath_compare_-
confed_aspath_set, [82](#)
- smi_bgp_bestpath_compare_-
confed_aspath_set_validate,
[82](#)
- smi_bgp_bestpath_compare_-
confed_aspath_unset, [83](#)
- smi_bgp_bestpath_compare_-
confed_aspath_unset_validate,
[83](#)
- smi_bgp_bestpath_compare_-
router_id_set, [83](#)
- smi_bgp_bestpath_compare_-
router_id_set_validate, [84](#)
- smi_bgp_bestpath_compare_-
router_id_unset, [84](#)
- smi_bgp_bestpath_compare_-
router_id_unset_validate, [84](#)
- smi_bgp_bestpath_dont_compare_-
originator_id_set, [85](#)

- smi_bgp_bestpath_dont_compare_-
originator_id_set_validate,
85
- smi_bgp_bestpath_dont_compare_-
originator_id_unset, 85
- smi_bgp_bestpath_dont_compare_-
originator_id_unset_validate,
86
- smi_bgp_bestpath_med_set, 86
- smi_bgp_bestpath_med_set_-
validate, 86
- smi_bgp_bestpath_med_unset, 87
- smi_bgp_bestpath_med_unset_-
validate, 87
- smi_bgp_bestpath_tie_break_on_-
age_set, 87
- smi_bgp_bestpath_tie_break_on_-
age_set_validate, 88
- smi_bgp_bestpath_tie_break_on_-
age_unset, 88
- smi_bgp_bestpath_tie_break_on_-
age_unset_validate, 88
- smi_bgp_check_instance, 89
- smi_bgp_clear_gen_sdkapi, 89
- smi_bgp_cluster_id_digit_set_-
sdkapi_validate, 90
- smi_bgp_cluster_id_set_sdkapi_-
validate, 90
- smi_bgp_cluster_id_unset_sdkapi_-
validate, 91
- smi_bgp_community_list_entry_-
unset, 91
- smi_bgp_community_list_set, 91
- smi_bgp_community_list_unset_-
validate, 92
- smi_bgp_confederation_id_set_-
sdkapi_validate, 92
- smi_bgp_confederation_id_unset_-
sdkapi_validate, 93
- smi_bgp_confederation_peer_-
check_sdkapi, 93
- smi_bgp_confederation_peers_-
add_sdkapi_validate, 94
- smi_bgp_confederation_peers_-
remove_sdkapi_validate, 94
- smi_bgp_create_instance_set_-
sdkapi_validate, 94
- smi_bgp_debug_validate, 95
- smi_bgp_default_ipv4_unicast_set,
95
- smi_bgp_default_ipv4_unicast_-
set_validate, 96
- smi_bgp_default_ipv4_unicast_-
unset, 96
- smi_bgp_default_ipv4_unicast_-
unset_validate, 96
- smi_bgp_default_local_preference_-
set_sdkapi_validate, 97
- smi_bgp_default_local_preference_-
unset_sdkapi_validate, 97
- smi_bgp_deterministic_med_set, 97
- smi_bgp_deterministic_med_set_-
validate, 98
- smi_bgp_deterministic_med_unset,
98
- smi_bgp_deterministic_med_-
unset_validate, 98
- smi_bgp_disable_adj_out_set, 99
- smi_bgp_disable_adj_out_set_-
validate, 99
- smi_bgp_disable_adj_out_unset, 99
- smi_bgp_disable_adj_out_unset_-
validate, 100
- smi_bgp_enforce_first_as_set, 100
- smi_bgp_enforce_first_as_set_-
validate, 100
- smi_bgp_enforce_first_as_unset,
101
- smi_bgp_enforce_first_as_unset_-
validate, 101
- smi_bgp_extcommunity_list_-
entry_unset_validate, 101
- smi_bgp_extcommunity_list_set,
102
- smi_bgp_extcommunity_list_unset,
103
- smi_bgp_fast_external_failover_set,
103
- smi_bgp_fast_external_failover_-
set_validate, 103
- smi_bgp_fast_external_failover_-
unset, 104
- smi_bgp_fast_external_failover_-
unset_validate, 104
- smi_bgp_get_address_family, 104
- smi_bgp_get_grst_restart_time, 105
- smi_bgp_get_grst_stalepath_time,
105
- smi_bgp_get_identifier, 106
- smi_bgp_get_local_as, 106

- smi_bgp_get_nbr_address_family, 106
- smi_bgp_get_peer_admin_status, 107
- smi_bgp_get_peer_connect_retry_interval, 107
- smi_bgp_get_peer_fsm_established_time, 108
- smi_bgp_get_peer_fsm_established_transitions, 108
- smi_bgp_get_peer_hold_time, 109
- smi_bgp_get_peer_hold_time_configured, 109
- smi_bgp_get_peer_identifier, 110
- smi_bgp_get_peer_in_total_messages, 110
- smi_bgp_get_peer_in_update_elapsed_time, 111
- smi_bgp_get_peer_in_updates, 111
- smi_bgp_get_peer_keep_alive, 112
- smi_bgp_get_peer_keep_alive_configured, 112
- smi_bgp_get_peer_last_error, 113
- smi_bgp_get_peer_local_addr, 113
- smi_bgp_get_peer_local_port, 114
- smi_bgp_get_peer_min_as_origination_interval, 114
- smi_bgp_get_peer_min_route_advertisement_interval, 115
- smi_bgp_get_peer_negotiated_version, 115
- smi_bgp_get_peer_out_total_messages, 116
- smi_bgp_get_peer_out_updates, 116
- smi_bgp_get_peer_remote_addr, 117
- smi_bgp_get_peer_remote_as, 117
- smi_bgp_get_peer_remote_port, 117
- smi_bgp_get_peer_state, 118
- smi_bgp_get_peer_timers, 118
- smi_bgp_get_update_delay_val, 119
- smi_bgp_get_version, 119
- smi_bgp_grst_restart_time_set_validate, 119
- smi_bgp_grst_restart_time_unset_validate, 120
- smi_bgp_grst_set_validate, 120
- smi_bgp_grst_stalepath_time_set_validate, 120
- smi_bgp_grst_stalepath_time_unset_validate, 121
- smi_bgp_grst_unset_validate, 121
- smi_bgp_instance_unset_sdkapi_validate, 121
- smi_bgp_maximum_paths_set, 122
- smi_bgp_maximum_paths_set_validate, 122
- smi_bgp_maximum_paths_unset, 123
- smi_bgp_maximum_paths_unset_validate, 123
- smi_bgp_multiple_instance_set, 123
- smi_bgp_multiple_instance_set_validate, 124
- smi_bgp_multiple_instance_unset, 124
- smi_bgp_multiple_instance_unset_validate, 124
- smi_bgp_nbr_address_family_set, 125
- smi_bgp_nbr_address_family_unset, 125
- smi_bgp_network_sync_set_sdkapi_validate, 126
- smi_bgp_network_sync_unset_sdkapi_validate, 126
- smi_bgp_no_debug_validate, 126
- smi_bgp_option_check_sdkapi, 127
- smi_bgp_option_set, 127
- smi_bgp_option_unset_validate, 128
- smi_bgp_peer_group_bind_sdkapi_validate, 129
- smi_bgp_peer_group_delete_unset_sdkapi_validate, 129
- smi_bgp_peer_group_remote_as_delete_unset_sdkapi_validate, 130
- smi_bgp_peer_group_unbind_sdkapi_validate, 130
- smi_bgp_peer_remote_as_set_sdkapi, 130
- smi_bgp_peer_unset_sdkapi_validate, 131
- smi_bgp_rfc1771_path_select_set, 131
- smi_bgp_rfc1771_path_select_set_validate, 132
- smi_bgp_rfc1771_path_select_unset, 132

- smi_bgp_rfc1771_path_select_-unset_validate, [132](#)
- smi_bgp_router_id_set_sdkapi_-validate, [133](#)
- smi_bgp_router_id_unset_sdkapi_-validate, [133](#)
- smi_bgp_set_peer_admin_status_-validate, [133](#)
- smi_bgp_set_peer_connect_retry_-interval_validate, [134](#)
- smi_bgp_set_peer_hold_time_-configured_validate, [134](#)
- smi_bgp_set_peer_keep_alive_-configured_validate, [135](#)
- smi_bgp_set_peer_min_as_-origination_interval_validate, [135](#)
- smi_bgp_set_peer_min_route_-advertisement_interval_-validate, [136](#)
- smi_bgp_show_bgp, [136](#)
- smi_bgp_show_bgp_-extcommunity_list, [137](#)
- smi_bgp_show_ip_bgp, [137](#)
- smi_bgp_show_ip_bgp_community, [138](#)
- smi_bgp_show_ip_bgp_-community_list, [138](#)
- smi_bgp_show_ip_bgp_-extcommunity_list_exact_-match, [139](#)
- smi_bgp_show_ip_bgp_-extcommunity_list_exact_-match_vrf, [139](#)
- smi_bgp_static_network_set_-sdkapi_validate, [140](#)
- smi_bgp_static_network_unset_-sdkapi_validate, [140](#)
- smi_bgp_synchronization_set_-sdkapi_validate, [141](#)
- smi_bgp_synchronization_unset_-sdkapi_validate, [141](#)
- smi_bgp_timers_set_sdkapi, [142](#)
- smi_bgp_timers_unset_sdkapi_-validate, [142](#)
- smi_bgp_update_delay_val_set_-validate, [142](#)
- smi_bgp_update_delay_val_unset_-validate, [143](#)
- smi_bgp_vrf_neighbor_as_-override_set, [143](#)
- smi_bgp_vrf_neighbor_as_-override_set_validate, [144](#)
- smi_bgp_vrf_neighbor_as_-override_unset, [144](#)
- smi_bgp_vrf_neighbor_as_-override_unset_validate, [144](#)
- smi_filter_list_set_validate, [145](#)
- smi_filter_list_unset_validate, [145](#)
- smi_neighbor_attr_unchanged_as_-path_set, [146](#)
- smi_neighbor_attr_unchanged_as_-path_unset, [146](#)
- smi_neighbor_attr_unchanged_-med_set, [147](#)
- smi_neighbor_attr_unchanged_-med_unset, [147](#)
- smi_neighbor_attr_unchanged_-nexthop_set, [147](#)
- smi_neighbor_attr_unchanged_-nexthop_unset, [148](#)
- smi_neighbor_capability_grst_set, [148](#)
- smi_neighbor_capability_grst_set_-validate, [149](#)
- smi_neighbor_capability_grst_-unset, [149](#)
- smi_neighbor_capability_grst_-unset_validate, [150](#)
- smi_neighbor_capability_orf_-prefix_set, [150](#)
- smi_neighbor_capability_orf_-prefix_set_validate, [150](#)
- smi_neighbor_capability_orf_-prefix_unset, [151](#)
- smi_neighbor_capability_orf_-prefix_unset_validate, [151](#)
- smi_neighbor_capability_route_-refresh_set, [152](#)
- smi_neighbor_capability_route_-refresh_set_validate, [152](#)
- smi_neighbor_capability_route_-refresh_unset, [152](#)
- smi_neighbor_capability_route_-refresh_unset_validate, [153](#)
- smi_neighbor_collide_established_-set, [153](#)

- smi_neighbor_collide_established_-
set_validate, [154](#)
- smi_neighbor_collide_established_-
unset, [154](#)
- smi_neighbor_collide_established_-
unset_validate, [154](#)
- smi_neighbor_connection_retry_-
time_unset_validate, [155](#)
- smi_neighbor_disallow_infinite_-
timer_set_validate, [155](#)
- smi_neighbor_disallow_infinite_-
timer_unset_validate, [156](#)
- smi_neighbor_dont_capability_-
negotiate_unset_validate, [156](#)
- smi_neighbor_enforce_multihop_-
set, [156](#)
- smi_neighbor_enforce_multihop_-
set_validate, [157](#)
- smi_neighbor_enforce_multihop_-
unset, [157](#)
- smi_neighbor_enforce_multihop_-
unset_validate, [157](#)
- smi_neighbor_filter_list_set_-
validate, [158](#)
- smi_neighbor_filter_list_unset_-
validate, [158](#)
- smi_neighbor_g_shut_time_set, [159](#)
- smi_neighbor_g_shut_time_set_-
validate, [159](#)
- smi_neighbor_g_shut_time_unset,
[159](#)
- smi_neighbor_g_shut_time_unset_-
validate, [160](#)
- smi_neighbor_local_as_set_-
validate, [160](#)
- smi_neighbor_local_as_unset_-
validate, [161](#)
- smi_neighbor_override_capability_-
set, [161](#)
- smi_neighbor_override_capability_-
set_validate, [161](#)
- smi_neighbor_override_capability_-
unset, [162](#)
- smi_neighbor_override_capability_-
unset_validate, [162](#)
- smi_neighbor_remove_private_as_-
set, [162](#)
- smi_neighbor_remove_private_as_-
set_validate, [163](#)
- smi_neighbor_remove_private_as_-
unset, [163](#)
- smi_neighbor_remove_private_as_-
unset_validate, [164](#)
- smi_neighbor_route_reflector_-
client_set_validate, [164](#)
- smi_neighbor_route_reflector_-
client_unset_validate, [165](#)
- smi_neighbor_route_server_client_-
set, [165](#)
- smi_neighbor_route_server_client_-
set_validate, [166](#)
- smi_neighbor_route_server_client_-
unset, [166](#)
- smi_neighbor_route_server_client_-
unset_validate, [166](#)
- smi_neighbor_strict_capability_set,
[167](#)
- smi_neighbor_strict_capability_-
set_validate, [167](#)
- smi_neighbor_strict_capability_-
unset, [168](#)
- smi_neighbor_strict_capability_-
unset_validate, [168](#)
- smi_neighbor_transparent_as_set_-
validate, [168](#)
- smi_neighbor_transparent_-
nexthop_set_validate, [169](#)
- smi_peer_activate_set_sdkapi_-
validate, [169](#)
- smi_peer_advertise_interval_set_-
sdkapi_validate, [170](#)
- smi_peer_advertise_interval_unset_-
sdkapi_validate, [170](#)
- smi_peer_af_flag_config_check, [171](#)
- smi_peer_af_flag_set_sdkapi_-
validate, [171](#)
- smi_peer_af_flag_unset_sdkapi_-
validate, [172](#)
- smi_peer_allowas_in_set_sdkapi_-
validate, [173](#)
- smi_peer_allowas_in_unset_-
sdkapi_validate, [174](#)
- smi_peer_aslist_set_sdkapi_-
validate, [174](#)
- smi_peer_aslist_unset_sdkapi_-
validate, [175](#)
- smi_peer_asorig_interval_set_-
sdkapi_validate, [175](#)

- smi_peer_asorig_interval_unset_-
 sdkapi_validate, 176
- smi_peer_deactivate_sdkapi_-
 validate, 176
- smi_peer_default_originate_set_-
 sdkapi_validate, 176
- smi_peer_default_originate_unset_-
 sdkapi, 177
- smi_peer_description_set_sdkapi_-
 validate, 177
- smi_peer_description_unset_-
 sdkapi_validate, 178
- smi_peer_disallow_hold_timer_-
 set_sdkapi, 178
- smi_peer_disallow_hold_timer_-
 unset_sdkapi, 178
- smi_peer_distribute_set_sdkapi_-
 validate, 179
- smi_peer_distribute_unset_sdkapi,
 179
- smi_peer_dont_capability_-
 negotiate_set, 180
- smi_peer_dont_capability_-
 negotiate_set_validate, 180
- smi_peer_dont_capability_-
 negotiate_unset, 181
- smi_peer_dynamic_capability_set,
 181
- smi_peer_dynamic_capability_set_-
 validate, 181
- smi_peer_dynamic_capability_-
 unset, 182
- smi_peer_dynamic_capability_-
 unset_validate, 182
- smi_peer_ebgp_multihop_set_-
 sdkapi_validate, 183
- smi_peer_ebgp_multihop_unset_-
 sdkapi_validate, 183
- smi_peer_flag_config_check, 183
- smi_peer_flag_set_sdkapi_validate,
 184
- smi_peer_flag_unset_sdkapi_-
 validate, 184
- smi_peer_get_advertise_interval,
 185
- smi_peer_get_allowas_in, 186
- smi_peer_get_asorig_interval, 186
- smi_peer_get_description, 186
- smi_peer_get_ebgp_multihop, 187
- smi_peer_get_interface, 187
- smi_peer_get_timers, 187
- smi_peer_get_timers_connect, 188
- smi_peer_get_update_source_info,
 188
- smi_peer_interface_set_sdkapi_-
 validate, 189
- smi_peer_interface_unset_sdkapi_-
 validate, 189
- smi_peer_maximum_prefix_set_-
 sdkapi_validate, 189
- smi_peer_next_hop_self_set, 190
- smi_peer_next_hop_self_set_-
 validate, 190
- smi_peer_next_hop_self_unset, 191
- smi_peer_next_hop_self_unset_-
 validate, 191
- smi_peer_password_set_validate,
 192
- smi_peer_password_unset_sdkapi_-
 validate, 192
- smi_peer_port_set_sdkapi, 192
- smi_peer_port_unset_sdkapi_-
 validate, 193
- smi_peer_prefix_list_set_sdkapi,
 193
- smi_peer_prefix_list_unset_-
 sdkapi_validate, 194
- smi_peer_route_map_set_sdkapi_-
 validate, 194
- smi_peer_route_map_unset_-
 sdkapi_validate, 195
- smi_peer_route_reflector_client_set,
 195
- smi_peer_route_reflector_client_-
 unset, 196
- smi_peer_shutdown_set, 196
- smi_peer_shutdown_set_validate,
 197
- smi_peer_shutdown_unset, 197
- smi_peer_shutdown_unset_validate,
 197
- smi_peer_soft_reconfiguration_-
 inbound_set, 198
- smi_peer_soft_reconfiguration_-
 inbound_set_validate, 198
- smi_peer_soft_reconfiguration_-
 inbound_unset, 199
- smi_peer_soft_reconfiguration_-
 inbound_unset_validate, 199

- smi_peer_timers_connect_set_-
 sdkapi_validate, 200
- smi_peer_timers_connect_unset_-
 sdkapi_validate, 200
- smi_peer_timers_set_sdkapi_-
 validate, 200
- smi_peer_timers_unset_sdkapi_-
 validate, 201
- smi_peer_transport_connection_-
 passive_set, 201
- smi_peer_transport_connection_-
 passive_set_validate, 202
- smi_peer_transport_connection_-
 passive_unset, 202
- smi_peer_transport_connection_-
 passive_unset_validate, 202
- smi_peer_unsuppress_map_set_-
 sdkapi_validate, 203
- smi_peer_unsuppress_map_unset_-
 sdkapi_validate, 203
- smi_peer_update_routing_source_-
 set_sdkapi_validate, 204
- smi_peer_version_set_sdkapi_-
 validate, 204
- smi_peer_version_unset_sdkapi_-
 validate, 205
- smi_peer_weight_set_sdkapi_-
 validate, 205
- smi_peer_weight_unset_sdkapi, 205
- smi_show_bgp_afi_regexp_safi, 206
- smi_show_bgp_afi_route_map_safi,
 206
- smi_show_bgp_dampening_-
 parameters, 207
- smi_show_bgp_inconsistent_as, 207
- smi_show_bgp_ip_neighbor_routes,
 208
- smi_show_bgp_neighbor_-
 advertised_routes, 208
- smi_show_bgp_neighbor_-
 recieved_routes, 209
- smi_show_bgp_neighbors_rcv_-
 prefix_filter, 209
- smi_show_bgp_regexp, 209
- smi_show_bgp_route_map, 210
- smi_show_bgp_sessions, 210
- smi_show_bgp_summary, 211
- smi_show_bgp_V6_neighbors_-
 rcv_prefix_filter, 211
- smi_show_ip_bgp, 211
- smi_show_ip_bgp_cidr_only, 212
- smi_show_ip_bgp_community, 212
- smi_show_ip_bgp_dampening_-
 dampend_paths, 213
- smi_show_ip_bgp_dampening_-
 flap_statistics, 213
- smi_show_ip_bgp_filter_list_-
 exact_match, 214
- smi_show_ip_bgp_ipv6_-
 dampening_parameters, 214
- smi_show_ip_bgp_longer_prefixes,
 215
- smi_show_ip_bgp_neighbors_HKC,
 215
- smi_show_ip_bgp_paths, 216
- smi_show_ip_bgp_prefix_list_-
 exact_match, 216
- smi_show_ip_bgp_quote_regexp,
 217
- smi_show_ip_bgp_received_paths,
 217
- smi_show_ip_bgp_regexp, 217
- smi_show_ip_bgp_route_map, 218
- smi_show_ip_bgp_safi_regexp, 218
- smi_show_ip_bgp_safi_route_map,
 219
- smi_show_ip_bgp_summary, 219
- smi_show_ip_bgp_word_neighbors,
 220
- smi_show_ip_bgp_word_peer_-
 neighbors, 220
- smi_show_ip_protocol_all, 221
- smi_transport_connection_passive_-
 set_validate, 221
- smi_transport_connection_passive_-
 unset_validate, 221
- smi_bgp4_get_path_attr_aggregator_-
 addr_sdkapi
 smi_bgp.h, 70
- smi_bgp4_get_path_attr_aggregator_as_-
 sdkapi
 smi_bgp.h, 70
- smi_bgp4_get_path_attr_atomic_-
 aggregate_sdkapi
 smi_bgp.h, 70
- smi_bgp4_get_path_attr_best_sdkapi
 smi_bgp.h, 71
- smi_bgp4_get_path_attr_calc_local_-
 pref_sdkapi
 smi_bgp.h, 71

- smi_bgp4_get_path_attr_ip_addr_-
 prefix_len_sdkapi
 smi_bgp.h, [72](#)
- smi_bgp4_get_path_attr_ip_addr_-
 prefix_sdkapi
 smi_bgp.h, [72](#)
- smi_bgp4_get_path_attr_local_pref_-
 sdkapi
 smi_bgp.h, [73](#)
- smi_bgp4_get_path_attr_multi_exit_-
 disc_sdkapi
 smi_bgp.h, [73](#)
- smi_bgp4_get_path_attr_next_hop_-
 sdkapi
 smi_bgp.h, [74](#)
- smi_bgp4_get_path_attr_origin_sdkapi
 smi_bgp.h, [74](#)
- smi_bgp4_get_path_attr_peer_sdkapi
 smi_bgp.h, [75](#)
- smi_bgp_address_family_set
 smi_bgp.h, [75](#)
- smi_bgp_af_config_check_sdkapi
 smi_bgp.h, [76](#)
- smi_bgp_aggregate_nexthop_check_set
 smi_bgp.h, [76](#)
- smi_bgp_aggregate_nexthop_check_-
 set_validate
 smi_bgp.h, [77](#)
- smi_bgp_aggregate_nexthop_check_-
 unset
 smi_bgp.h, [77](#)
- smi_bgp_aggregate_nexthop_check_-
 unset_validate
 smi_bgp.h, [77](#)
- smi_bgp_always_compare_med_set
 smi_bgp.h, [78](#)
- smi_bgp_always_compare_med_set_-
 validate
 smi_bgp.h, [78](#)
- smi_bgp_always_compare_med_unset
 smi_bgp.h, [78](#)
- smi_bgp_always_compare_med_unset_-
 validate
 smi_bgp.h, [79](#)
- smi_bgp_api_address_family_unset
 smi_bgp.h, [79](#)
- smi_bgp_aspath_access_list_set_validate
 smi_bgp.h, [79](#)
- smi_bgp_aspath_access_list_unset_-
 validate
 smi_bgp.h, [80](#)
- smi_bgp_auto_summary_update_set_-
 sdkapi_validate
 smi_bgp.h, [80](#)
- smi_bgp_bestpath_aspath_ignore_set
 smi_bgp.h, [81](#)
- smi_bgp_bestpath_aspath_ignore_set_-
 validate
 smi_bgp.h, [81](#)
- smi_bgp_bestpath_aspath_ignore_unset
 smi_bgp.h, [81](#)
- smi_bgp_bestpath_aspath_ignore_-
 unset_validate
 smi_bgp.h, [82](#)
- smi_bgp_bestpath_compare_confed_-
 aspath_set
 smi_bgp.h, [82](#)
- smi_bgp_bestpath_compare_confed_-
 aspath_set_validate
 smi_bgp.h, [82](#)
- smi_bgp_bestpath_compare_confed_-
 aspath_unset
 smi_bgp.h, [83](#)
- smi_bgp_bestpath_compare_confed_-
 aspath_unset_validate
 smi_bgp.h, [83](#)
- smi_bgp_bestpath_compare_router_id_-
 set
 smi_bgp.h, [83](#)
- smi_bgp_bestpath_compare_router_id_-
 set_validate
 smi_bgp.h, [84](#)
- smi_bgp_bestpath_compare_router_id_-
 unset
 smi_bgp.h, [84](#)
- smi_bgp_bestpath_compare_router_id_-
 unset_validate
 smi_bgp.h, [84](#)
- smi_bgp_bestpath_dont_compare_-
 originator_id_set
 smi_bgp.h, [85](#)
- smi_bgp_bestpath_dont_compare_-
 originator_id_set_validate
 smi_bgp.h, [85](#)
- smi_bgp_bestpath_dont_compare_-
 originator_id_unset
 smi_bgp.h, [85](#)
- smi_bgp_bestpath_dont_compare_-
 originator_id_unset_validate
 smi_bgp.h, [86](#)

- smi_bgp_bestpath_med_set
 - smi_bgp.h, [86](#)
- smi_bgp_bestpath_med_set_validate
 - smi_bgp.h, [86](#)
- smi_bgp_bestpath_med_unset
 - smi_bgp.h, [87](#)
- smi_bgp_bestpath_med_unset_validate
 - smi_bgp.h, [87](#)
- smi_bgp_bestpath_tie_break_on_age_set
 - smi_bgp.h, [87](#)
- smi_bgp_bestpath_tie_break_on_age_-set_validate
 - smi_bgp.h, [88](#)
- smi_bgp_bestpath_tie_break_on_age_-unset
 - smi_bgp.h, [88](#)
- smi_bgp_bestpath_tie_break_on_age_-unset_validate
 - smi_bgp.h, [88](#)
- smi_bgp_bfd.h, [223](#)
 - smi_bgp_peer_bfd_set, [223](#)
 - smi_bgp_peer_bfd_unset, [224](#)
- smi_bgp_check_instance
 - smi_bgp.h, [89](#)
- smi_bgp_clear_gen_sdkapi
 - smi_bgp.h, [89](#)
- smi_bgp_cluster_id_digit_set_sdkapi_-validate
 - smi_bgp.h, [90](#)
- smi_bgp_cluster_id_set_sdkapi_validate
 - smi_bgp.h, [90](#)
- smi_bgp_cluster_id_unset_sdkapi_-validate
 - smi_bgp.h, [91](#)
- smi_bgp_community_list_entry_unset
 - smi_bgp.h, [91](#)
- smi_bgp_community_list_set
 - smi_bgp.h, [91](#)
- smi_bgp_community_list_unset_validate
 - smi_bgp.h, [92](#)
- smi_bgp_confederation_id_set_sdkapi_-validate
 - smi_bgp.h, [92](#)
- smi_bgp_confederation_id_unset_-sdkapi_validate
 - smi_bgp.h, [93](#)
- smi_bgp_confederation_peer_check_-sdkapi
 - smi_bgp.h, [93](#)
- smi_bgp_confederation_peers_add_-sdkapi_validate
 - smi_bgp.h, [94](#)
- smi_bgp_confederation_peers_remove_-sdkapi_validate
 - smi_bgp.h, [94](#)
- smi_bgp_create_instance_set_sdkapi_-validate
 - smi_bgp.h, [94](#)
- smi_bgp_debug_validate
 - smi_bgp.h, [95](#)
- smi_bgp_default_ipv4_unicast_set
 - smi_bgp.h, [95](#)
- smi_bgp_default_ipv4_unicast_set_-validate
 - smi_bgp.h, [96](#)
- smi_bgp_default_ipv4_unicast_unset
 - smi_bgp.h, [96](#)
- smi_bgp_default_ipv4_unicast_unset_-validate
 - smi_bgp.h, [96](#)
- smi_bgp_default_local_preference_set_-sdkapi_validate
 - smi_bgp.h, [97](#)
- smi_bgp_default_local_preference_-unset_sdkapi_validate
 - smi_bgp.h, [97](#)
- smi_bgp_deterministic_med_set
 - smi_bgp.h, [97](#)
- smi_bgp_deterministic_med_set_validate
 - smi_bgp.h, [98](#)
- smi_bgp_deterministic_med_unset
 - smi_bgp.h, [98](#)
- smi_bgp_deterministic_med_unset_-validate
 - smi_bgp.h, [98](#)
- smi_bgp_disable_adj_out_set
 - smi_bgp.h, [99](#)
- smi_bgp_disable_adj_out_set_validate
 - smi_bgp.h, [99](#)
- smi_bgp_disable_adj_out_unset
 - smi_bgp.h, [99](#)
- smi_bgp_disable_adj_out_unset_validate
 - smi_bgp.h, [100](#)
- smi_bgp_enforce_first_as_set
 - smi_bgp.h, [100](#)
- smi_bgp_enforce_first_as_set_validate
 - smi_bgp.h, [100](#)
- smi_bgp_enforce_first_as_unset
 - smi_bgp.h, [101](#)

- smi_bgp_enforce_first_as_unset_validate
smi_bgp.h, 101
- smi_bgp_extcommunity_list_entry_-
unset_validate
smi_bgp.h, 101
- smi_bgp_extcommunity_list_set
smi_bgp.h, 102
- smi_bgp_extcommunity_list_unset
smi_bgp.h, 103
- smi_bgp_fast_external_failover_set
smi_bgp.h, 103
- smi_bgp_fast_external_failover_set_-
validate
smi_bgp.h, 103
- smi_bgp_fast_external_failover_unset
smi_bgp.h, 104
- smi_bgp_fast_external_failover_unset_-
validate
smi_bgp.h, 104
- smi_bgp_get_address_family
smi_bgp.h, 104
- smi_bgp_get_grst_restart_time
smi_bgp.h, 105
- smi_bgp_get_grst_stalepath_time
smi_bgp.h, 105
- smi_bgp_get_identifier
smi_bgp.h, 106
- smi_bgp_get_local_as
smi_bgp.h, 106
- smi_bgp_get_nbr_address_family
smi_bgp.h, 106
- smi_bgp_get_peer_admin_status
smi_bgp.h, 107
- smi_bgp_get_peer_connect_retry_-
interval
smi_bgp.h, 107
- smi_bgp_get_peer_fsm_established_time
smi_bgp.h, 108
- smi_bgp_get_peer_fsm_established_-
transitions
smi_bgp.h, 108
- smi_bgp_get_peer_hold_time
smi_bgp.h, 109
- smi_bgp_get_peer_hold_time_configured
smi_bgp.h, 109
- smi_bgp_get_peer_identifier
smi_bgp.h, 110
- smi_bgp_get_peer_in_total_messages
smi_bgp.h, 110
- smi_bgp_get_peer_in_update_elapsed_-
time
smi_bgp.h, 111
- smi_bgp_get_peer_in_updates
smi_bgp.h, 111
- smi_bgp_get_peer_keep_alive
smi_bgp.h, 112
- smi_bgp_get_peer_keep_alive_-
configured
smi_bgp.h, 112
- smi_bgp_get_peer_last_error
smi_bgp.h, 113
- smi_bgp_get_peer_local_addr
smi_bgp.h, 113
- smi_bgp_get_peer_local_port
smi_bgp.h, 114
- smi_bgp_get_peer_min_as_origination_-
interval
smi_bgp.h, 114
- smi_bgp_get_peer_min_route_-
advertisement_interval
smi_bgp.h, 115
- smi_bgp_get_peer_negotiated_version
smi_bgp.h, 115
- smi_bgp_get_peer_out_total_messages
smi_bgp.h, 116
- smi_bgp_get_peer_out_updates
smi_bgp.h, 116
- smi_bgp_get_peer_remote_addr
smi_bgp.h, 117
- smi_bgp_get_peer_remote_as
smi_bgp.h, 117
- smi_bgp_get_peer_remote_port
smi_bgp.h, 117
- smi_bgp_get_peer_state
smi_bgp.h, 118
- smi_bgp_get_peer_timers
smi_bgp.h, 118
- smi_bgp_get_update_delay_val
smi_bgp.h, 119
- smi_bgp_get_version
smi_bgp.h, 119
- smi_bgp_grst_restart_time_set_validate
smi_bgp.h, 119
- smi_bgp_grst_restart_time_unset_-
validate
smi_bgp.h, 120
- smi_bgp_grst_set_validate
smi_bgp.h, 120

- smi_bgp_grst_stalepath_time_set_-
 validate
 smi_bgp.h, [120](#)
- smi_bgp_grst_stalepath_time_unset_-
 validate
 smi_bgp.h, [121](#)
- smi_bgp_grst_unset_validate
 smi_bgp.h, [121](#)
- smi_bgp_instance_unset_sdkapi_validate
 smi_bgp.h, [121](#)
- smi_bgp_maximum_paths_set
 smi_bgp.h, [122](#)
- smi_bgp_maximum_paths_set_validate
 smi_bgp.h, [122](#)
- smi_bgp_maximum_paths_unset
 smi_bgp.h, [123](#)
- smi_bgp_maximum_paths_unset_-
 validate
 smi_bgp.h, [123](#)
- smi_bgp_multiple_instance_set
 smi_bgp.h, [123](#)
- smi_bgp_multiple_instance_set_validate
 smi_bgp.h, [124](#)
- smi_bgp_multiple_instance_unset
 smi_bgp.h, [124](#)
- smi_bgp_multiple_instance_unset_-
 validate
 smi_bgp.h, [124](#)
- smi_bgp_nbr_address_family_set
 smi_bgp.h, [125](#)
- smi_bgp_nbr_address_family_unset
 smi_bgp.h, [125](#)
- smi_bgp_network_sync_set_sdkapi_-
 validate
 smi_bgp.h, [126](#)
- smi_bgp_network_sync_unset_sdkapi_-
 validate
 smi_bgp.h, [126](#)
- smi_bgp_no_debug_validate
 smi_bgp.h, [126](#)
- smi_bgp_option_check_sdkapi
 smi_bgp.h, [127](#)
- smi_bgp_option_set
 smi_bgp.h, [127](#)
- smi_bgp_option_unset_validate
 smi_bgp.h, [128](#)
- smi_bgp_peer_bfd_set
 smi_bgp_bfd.h, [223](#)
- smi_bgp_peer_bfd_unset
 smi_bgp_bfd.h, [224](#)
- smi_bgp_peer_group_bind_sdkapi_-
 validate
 smi_bgp.h, [129](#)
- smi_bgp_peer_group_delete_unset_-
 sdkapi_validate
 smi_bgp.h, [129](#)
- smi_bgp_peer_group_remote_as_-
 delete_unset_sdkapi_validate
 smi_bgp.h, [130](#)
- smi_bgp_peer_group_unbind_sdkapi_-
 validate
 smi_bgp.h, [130](#)
- smi_bgp_peer_remote_as_set_sdkapi
 smi_bgp.h, [130](#)
- smi_bgp_peer_unset_sdkapi_validate
 smi_bgp.h, [131](#)
- smi_bgp_rfc1771_path_select_set
 smi_bgp.h, [131](#)
- smi_bgp_rfc1771_path_select_set_-
 validate
 smi_bgp.h, [132](#)
- smi_bgp_rfc1771_path_select_unset
 smi_bgp.h, [132](#)
- smi_bgp_rfc1771_path_select_unset_-
 validate
 smi_bgp.h, [132](#)
- smi_bgp_router_id_set_sdkapi_validate
 smi_bgp.h, [133](#)
- smi_bgp_router_id_unset_sdkapi_-
 validate
 smi_bgp.h, [133](#)
- smi_bgp_set_peer_admin_status_validate
 smi_bgp.h, [133](#)
- smi_bgp_set_peer_connect_retry_-
 interval_validate
 smi_bgp.h, [134](#)
- smi_bgp_set_peer_hold_time_-
 configured_validate
 smi_bgp.h, [134](#)
- smi_bgp_set_peer_keep_alive_-
 configured_validate
 smi_bgp.h, [135](#)
- smi_bgp_set_peer_min_as_origination_-
 interval_validate
 smi_bgp.h, [135](#)
- smi_bgp_set_peer_min_route_-
 advertisement_interval_-
 validate
 smi_bgp.h, [136](#)
- smi_bgp_show_bgp

- smi_bgp.h, 136
- smi_bgp_show_bgp_extcommunity_list
 - smi_bgp.h, 137
- smi_bgp_show_ip_bgp
 - smi_bgp.h, 137
- smi_bgp_show_ip_bgp_community
 - smi_bgp.h, 138
- smi_bgp_show_ip_bgp_community_list
 - smi_bgp.h, 138
- smi_bgp_show_ip_bgp_extcommunity_-list_exact_match
 - smi_bgp.h, 139
- smi_bgp_show_ip_bgp_extcommunity_-list_exact_match_vrf
 - smi_bgp.h, 139
- smi_bgp_static_network_set_sdkapi_-validate
 - smi_bgp.h, 140
- smi_bgp_static_network_unset_sdkapi_-validate
 - smi_bgp.h, 140
- smi_bgp_synchronization_set_sdkapi_-validate
 - smi_bgp.h, 141
- smi_bgp_synchronization_unset_-sdkapi_validate
 - smi_bgp.h, 141
- smi_bgp_timers_set_sdkapi
 - smi_bgp.h, 142
- smi_bgp_timers_unset_sdkapi_validate
 - smi_bgp.h, 142
- smi_bgp_update_delay_val_set_validate
 - smi_bgp.h, 142
- smi_bgp_update_delay_val_unset_-validate
 - smi_bgp.h, 143
- smi_bgp_vrf_neighbor_as_override_set
 - smi_bgp.h, 143
- smi_bgp_vrf_neighbor_as_override_set_-validate
 - smi_bgp.h, 144
- smi_bgp_vrf_neighbor_as_override_-unset
 - smi_bgp.h, 144
- smi_bgp_vrf_neighbor_as_override_-unset_validate
 - smi_bgp.h, 144
- smi_filter_list_set_validate
 - smi_bgp.h, 145
- smi_filter_list_unset_validate
 - smi_bgp.h, 145
- smi_neighbor_attr_unchanged_as_path_-set
 - smi_bgp.h, 146
- smi_neighbor_attr_unchanged_as_path_-unset
 - smi_bgp.h, 146
- smi_neighbor_attr_unchanged_med_set
 - smi_bgp.h, 147
- smi_neighbor_attr_unchanged_med_-unset
 - smi_bgp.h, 147
- smi_neighbor_attr_unchanged_nexthop_-set
 - smi_bgp.h, 147
- smi_neighbor_attr_unchanged_nexthop_-unset
 - smi_bgp.h, 148
- smi_neighbor_capability_grst_set
 - smi_bgp.h, 148
- smi_neighbor_capability_grst_set_-validate
 - smi_bgp.h, 149
- smi_neighbor_capability_grst_unset
 - smi_bgp.h, 149
- smi_neighbor_capability_grst_unset_-validate
 - smi_bgp.h, 150
- smi_neighbor_capability_orf_prefix_set
 - smi_bgp.h, 150
- smi_neighbor_capability_orf_prefix_-set_validate
 - smi_bgp.h, 150
- smi_neighbor_capability_orf_prefix_-unset
 - smi_bgp.h, 151
- smi_neighbor_capability_orf_prefix_-unset_validate
 - smi_bgp.h, 151
- smi_neighbor_capability_route_refresh_-set
 - smi_bgp.h, 152
- smi_neighbor_capability_route_refresh_-set_validate
 - smi_bgp.h, 152
- smi_neighbor_capability_route_refresh_-unset
 - smi_bgp.h, 152
- smi_neighbor_capability_route_refresh_-unset_validate
 - smi_bgp.h, 152

- smi_bgp.h, [153](#)
- smi_neighbor_collide_established_set
 - smi_bgp.h, [153](#)
- smi_neighbor_collide_established_set_-
validate
 - smi_bgp.h, [154](#)
- smi_neighbor_collide_established_unset
 - smi_bgp.h, [154](#)
- smi_neighbor_collide_established_-
unset_validate
 - smi_bgp.h, [154](#)
- smi_neighbor_connection_retry_time_-
unset_validate
 - smi_bgp.h, [155](#)
- smi_neighbor_disallow_infinite_timer_-
set_validate
 - smi_bgp.h, [155](#)
- smi_neighbor_disallow_infinite_timer_-
unset_validate
 - smi_bgp.h, [156](#)
- smi_neighbor_dont_capability_-
negotiate_unset_validate
 - smi_bgp.h, [156](#)
- smi_neighbor_enforce_multihop_set
 - smi_bgp.h, [156](#)
- smi_neighbor_enforce_multihop_set_-
validate
 - smi_bgp.h, [157](#)
- smi_neighbor_enforce_multihop_unset
 - smi_bgp.h, [157](#)
- smi_neighbor_enforce_multihop_unset_-
validate
 - smi_bgp.h, [157](#)
- smi_neighbor_filter_list_set_validate
 - smi_bgp.h, [158](#)
- smi_neighbor_filter_list_unset_validate
 - smi_bgp.h, [158](#)
- smi_neighbor_g_shut_time_set
 - smi_bgp.h, [159](#)
- smi_neighbor_g_shut_time_set_validate
 - smi_bgp.h, [159](#)
- smi_neighbor_g_shut_time_unset
 - smi_bgp.h, [159](#)
- smi_neighbor_g_shut_time_unset_-
validate
 - smi_bgp.h, [160](#)
- smi_neighbor_local_as_set_validate
 - smi_bgp.h, [160](#)
- smi_neighbor_local_as_unset_validate
 - smi_bgp.h, [161](#)
- smi_neighbor_override_capability_set
 - smi_bgp.h, [161](#)
- smi_neighbor_override_capability_set_-
validate
 - smi_bgp.h, [161](#)
- smi_neighbor_override_capability_unset
 - smi_bgp.h, [162](#)
- smi_neighbor_override_capability_-
unset_validate
 - smi_bgp.h, [162](#)
- smi_neighbor_remove_private_as_set
 - smi_bgp.h, [162](#)
- smi_neighbor_remove_private_as_set_-
validate
 - smi_bgp.h, [163](#)
- smi_neighbor_remove_private_as_unset
 - smi_bgp.h, [163](#)
- smi_neighbor_remove_private_as_-
unset_validate
 - smi_bgp.h, [164](#)
- smi_neighbor_route_reflector_client_-
set_validate
 - smi_bgp.h, [164](#)
- smi_neighbor_route_reflector_client_-
unset_validate
 - smi_bgp.h, [165](#)
- smi_neighbor_route_server_client_set
 - smi_bgp.h, [165](#)
- smi_neighbor_route_server_client_set_-
validate
 - smi_bgp.h, [166](#)
- smi_neighbor_route_server_client_unset
 - smi_bgp.h, [166](#)
- smi_neighbor_route_server_client_-
unset_validate
 - smi_bgp.h, [166](#)
- smi_neighbor_strict_capability_set
 - smi_bgp.h, [167](#)
- smi_neighbor_strict_capability_set_-
validate
 - smi_bgp.h, [167](#)
- smi_neighbor_strict_capability_unset
 - smi_bgp.h, [168](#)
- smi_neighbor_strict_capability_unset_-
validate
 - smi_bgp.h, [168](#)
- smi_neighbor_transparent_as_set_-
validate
 - smi_bgp.h, [168](#)

- smi_neighbor_transparent_nexthop_set_-
 validate
 smi_bgp.h, [169](#)
- smi_peer_activate_set_sdkapi_validate
 smi_bgp.h, [169](#)
- smi_peer_advertise_interval_set_-
 sdkapi_validate
 smi_bgp.h, [170](#)
- smi_peer_advertise_interval_unset_-
 sdkapi_validate
 smi_bgp.h, [170](#)
- smi_peer_af_flag_config_check
 smi_bgp.h, [171](#)
- smi_peer_af_flag_set_sdkapi_validate
 smi_bgp.h, [171](#)
- smi_peer_af_flag_unset_sdkapi_validate
 smi_bgp.h, [172](#)
- smi_peer_allowas_in_set_sdkapi_-
 validate
 smi_bgp.h, [173](#)
- smi_peer_allowas_in_unset_sdkapi_-
 validate
 smi_bgp.h, [174](#)
- smi_peer_aslist_set_sdkapi_validate
 smi_bgp.h, [174](#)
- smi_peer_aslist_unset_sdkapi_validate
 smi_bgp.h, [175](#)
- smi_peer_asorig_interval_set_sdkapi_-
 validate
 smi_bgp.h, [175](#)
- smi_peer_asorig_interval_unset_sdkapi_-
 validate
 smi_bgp.h, [176](#)
- smi_peer_deactivate_sdkapi_validate
 smi_bgp.h, [176](#)
- smi_peer_default_originate_set_sdkapi_-
 validate
 smi_bgp.h, [176](#)
- smi_peer_default_originate_unset_sdkapi
 smi_bgp.h, [177](#)
- smi_peer_description_set_sdkapi_-
 validate
 smi_bgp.h, [177](#)
- smi_peer_description_unset_sdkapi_-
 validate
 smi_bgp.h, [178](#)
- smi_peer_disallow_hold_timer_set_-
 sdkapi
 smi_bgp.h, [178](#)
- smi_peer_disallow_hold_timer_unset_-
 sdkapi
 smi_bgp.h, [178](#)
- smi_peer_distribute_set_sdkapi_validate
 smi_bgp.h, [179](#)
- smi_peer_distribute_unset_sdkapi
 smi_bgp.h, [179](#)
- smi_peer_dont_capability_negotiate_set
 smi_bgp.h, [180](#)
- smi_peer_dont_capability_negotiate_-
 set_validate
 smi_bgp.h, [180](#)
- smi_peer_dont_capability_negotiate_-
 unset
 smi_bgp.h, [181](#)
- smi_peer_dynamic_capability_set
 smi_bgp.h, [181](#)
- smi_peer_dynamic_capability_set_-
 validate
 smi_bgp.h, [181](#)
- smi_peer_dynamic_capability_unset
 smi_bgp.h, [182](#)
- smi_peer_dynamic_capability_unset_-
 validate
 smi_bgp.h, [182](#)
- smi_peer_ebgp_multihop_set_sdkapi_-
 validate
 smi_bgp.h, [183](#)
- smi_peer_ebgp_multihop_unset_-
 sdkapi_validate
 smi_bgp.h, [183](#)
- smi_peer_flag_config_check
 smi_bgp.h, [183](#)
- smi_peer_flag_set_sdkapi_validate
 smi_bgp.h, [184](#)
- smi_peer_flag_unset_sdkapi_validate
 smi_bgp.h, [184](#)
- smi_peer_get_advertise_interval
 smi_bgp.h, [185](#)
- smi_peer_get_allowas_in
 smi_bgp.h, [186](#)
- smi_peer_get_asorig_interval
 smi_bgp.h, [186](#)
- smi_peer_get_description
 smi_bgp.h, [186](#)
- smi_peer_get_ebgp_multihop
 smi_bgp.h, [187](#)
- smi_peer_get_interface
 smi_bgp.h, [187](#)
- smi_peer_get_timers

- smi_bgp.h, 187
- smi_peer_get_timers_connect
 - smi_bgp.h, 188
- smi_peer_get_update_source_info
 - smi_bgp.h, 188
- smi_peer_interface_set_sdkapi_validate
 - smi_bgp.h, 189
- smi_peer_interface_unset_sdkapi_-
validate
 - smi_bgp.h, 189
- smi_peer_maximum_prefix_set_sdkapi_-
validate
 - smi_bgp.h, 189
- smi_peer_next_hop_self_set
 - smi_bgp.h, 190
- smi_peer_next_hop_self_set_validate
 - smi_bgp.h, 190
- smi_peer_next_hop_self_unset
 - smi_bgp.h, 191
- smi_peer_next_hop_self_unset_validate
 - smi_bgp.h, 191
- smi_peer_password_set_validate
 - smi_bgp.h, 192
- smi_peer_password_unset_sdkapi_-
validate
 - smi_bgp.h, 192
- smi_peer_port_set_sdkapi
 - smi_bgp.h, 192
- smi_peer_port_unset_sdkapi_validate
 - smi_bgp.h, 193
- smi_peer_prefix_list_set_sdkapi
 - smi_bgp.h, 193
- smi_peer_prefix_list_unset_sdkapi_-
validate
 - smi_bgp.h, 194
- smi_peer_route_map_set_sdkapi_-
validate
 - smi_bgp.h, 194
- smi_peer_route_map_unset_sdkapi_-
validate
 - smi_bgp.h, 195
- smi_peer_route_reflector_client_set
 - smi_bgp.h, 195
- smi_peer_route_reflector_client_unset
 - smi_bgp.h, 196
- smi_peer_shutdown_set
 - smi_bgp.h, 196
- smi_peer_shutdown_set_validate
 - smi_bgp.h, 197
- smi_peer_shutdown_unset
 - smi_bgp.h, 197
- smi_peer_shutdown_unset_validate
 - smi_bgp.h, 197
- smi_peer_soft_reconfiguration_-
inbound_set
 - smi_bgp.h, 198
- smi_peer_soft_reconfiguration_-
inbound_set_validate
 - smi_bgp.h, 198
- smi_peer_soft_reconfiguration_-
inbound_unset
 - smi_bgp.h, 199
- smi_peer_soft_reconfiguration_-
inbound_unset_validate
 - smi_bgp.h, 199
- smi_peer_timers_connect_set_sdkapi_-
validate
 - smi_bgp.h, 200
- smi_peer_timers_connect_unset_-
sdkapi_validate
 - smi_bgp.h, 200
- smi_peer_timers_set_sdkapi_validate
 - smi_bgp.h, 200
- smi_peer_timers_unset_sdkapi_validate
 - smi_bgp.h, 201
- smi_peer_transport_connection_-
passive_set
 - smi_bgp.h, 201
- smi_peer_transport_connection_-
passive_set_validate
 - smi_bgp.h, 202
- smi_peer_transport_connection_-
passive_unset
 - smi_bgp.h, 202
- smi_peer_transport_connection_-
passive_unset_validate
 - smi_bgp.h, 202
- smi_peer_unsuppress_map_set_sdkapi_-
validate
 - smi_bgp.h, 203
- smi_peer_unsuppress_map_unset_-
sdkapi_validate
 - smi_bgp.h, 203
- smi_peer_update_routing_source_set_-
sdkapi_validate
 - smi_bgp.h, 204
- smi_peer_version_set_sdkapi_validate
 - smi_bgp.h, 204
- smi_peer_version_unset_sdkapi_validate
 - smi_bgp.h, 205

- smi_peer_weight_set_sdkapi_validate
smi_bgp.h, 205
- smi_peer_weight_unset_sdkapi
smi_bgp.h, 205
- smi_show_bgp_afi_regexp_safi
smi_bgp.h, 206
- smi_show_bgp_afi_route_map_safi
smi_bgp.h, 206
- smi_show_bgp_dampening_parameters
smi_bgp.h, 207
- smi_show_bgp_inconsistent_as
smi_bgp.h, 207
- smi_show_bgp_ip_neighbor_routes
smi_bgp.h, 208
- smi_show_bgp_neighbor_advertised_
routes
smi_bgp.h, 208
- smi_show_bgp_neighbor_recieved_
routes
smi_bgp.h, 209
- smi_show_bgp_neighbors_rcv_prefix_
filter
smi_bgp.h, 209
- smi_show_bgp_regexp
smi_bgp.h, 209
- smi_show_bgp_route_map
smi_bgp.h, 210
- smi_show_bgp_sessions
smi_bgp.h, 210
- smi_show_bgp_summary
smi_bgp.h, 211
- smi_show_bgp_V6_neighbors_rcv_
prefix_filter
smi_bgp.h, 211
- smi_show_ip_bgp
smi_bgp.h, 211
- smi_show_ip_bgp_cidr_only
smi_bgp.h, 212
- smi_show_ip_bgp_community
smi_bgp.h, 212
- smi_show_ip_bgp_dampening_
dampend_paths
smi_bgp.h, 213
- smi_show_ip_bgp_dampening_flap_
statistics
smi_bgp.h, 213
- smi_show_ip_bgp_filter_list_exact_
match
smi_bgp.h, 214
- smi_show_ip_bgp_ipv6_dampening_
parameters
smi_bgp.h, 214
- smi_show_ip_bgp_longer_prefixes
smi_bgp.h, 215
- smi_show_ip_bgp_neighbors_HKC
smi_bgp.h, 215
- smi_show_ip_bgp_paths
smi_bgp.h, 216
- smi_show_ip_bgp_prefix_list_exact_
match
smi_bgp.h, 216
- smi_show_ip_bgp_quote_regexp
smi_bgp.h, 217
- smi_show_ip_bgp_received_paths
smi_bgp.h, 217
- smi_show_ip_bgp_regexp
smi_bgp.h, 217
- smi_show_ip_bgp_route_map
smi_bgp.h, 218
- smi_show_ip_bgp_safi_regexp
smi_bgp.h, 218
- smi_show_ip_bgp_safi_route_map
smi_bgp.h, 219
- smi_show_ip_bgp_summary
smi_bgp.h, 219
- smi_show_ip_bgp_word_neighbors
smi_bgp.h, 220
- smi_show_ip_bgp_word_peer_neighbors
smi_bgp.h, 220
- smi_show_ip_protocol_all
smi_bgp.h, 221
- smi_transport_connection_passive_set_
validate
smi_bgp.h, 221
- smi_transport_connection_passive_
unset_validate
smi_bgp.h, 221