

Deprecated NSX-T Data Center and Policy Manager API Calls and Properties

The following API calls and properties are deprecated. They are marked as deprecated in the API reference. You can continue to use them at your discretion, but be aware that they will be removed from NSX-T Data Center in a future release.

Deprecated NSX-T Data Center Type Definitions

IpfixObsPointConfig:

Global switch IPFIX config is deprecated. Use `IpfixSwitchUpmProfile` and `IpfixCollectorUpmProfile` instead. Switch IPFIX profile can be applied to specific entities, such as logical switch, logical port, and so on.

Logical:

Services are now defined in service-specific APIs, such as the DHCP relay service API, the NAT API, and others.

LogicalServiceResourceTypes:

Resource types of logical services

LogicalService:

Services are now defined in service-specific APIs, such as the DHCP relay service API, the NAT API, and others.

ServiceProfileResourceTypes:

Resource types of service profiles

NodeActionParameters:

fabric node action parameters

IpfixObsPointsListResult:

Use `UpmProfileListResult` instead.

LogicalServiceListResult:

Services are now defined in service-specific APIs, such as the DHCP relay service API, the NAT API, and others.

ServiceProfile:

Services are now defined in service-specific APIs, such as the DHCP relay service API, the NAT API, and others.

ServiceProfileListResult:

Services are now defined in service-specific APIs, such as the DHCP relay service API, the NAT API, and others.

IpfixSwitchConfig:

Use `IpfixSwitchUpmProfile` instead which can specify its own collectors and observation ID.

Deprecated NSX-T Data Center API Property Definitions

LbVirtualServer.default_pool_member_port:

Use 'default_pool_member_ports' instead. If default_pool_member_port is configured and default_pool_member_ports are not specified, both default_pool_member_port and default_pool_member_ports in response payload would return the same port value. If both are specified, default_pool_member_ports setting would take effect with higher priority.

LbVirtualServer.port:

Use 'ports' instead. Port setting could be single port for both L7 mode and L4 mode. For L4 mode, a single port range is also supported. The port setting could be a single port or port range such as "80", "1234-1236". If port is configured and ports are not specified, both port and ports in response payload would return the same port value. If both port and ports are configured, ports setting would take effect with higher priority.

BgpNeighbor.filter_in_routemap_id:

Use 'address_family' instead.

BgpNeighbor.remote_as:

Use 'remote_as_num' instead.

BgpNeighbor.filter_out_ipprefixlist_id:

Use 'address_family' instead.

BgpNeighbor.filter_out_routemap_id:

Use 'address_family' instead.

BgpNeighbor.source_address:

Do not provide a value for this field. Use source_addresses instead.

BgpNeighbor.filter_in_ipprefixlist_id:

Use 'address_family' instead.

LabelValueConfiguration.weight:

Specify relative weight in WidgetItem for placement in a view. See WidgetItem for details.

LabelValueConfiguration.shared:

Use the property 'shared' of View instead. The widgets of a shared view are visible to other users.

HostSwitch.static_ip_pool_id:

ID of configured Static IP Pool. If specified allocate IP for Endpoints from Pool. Otherwise, assume IP will be assigned for Endpoints from DHCP. Use ip_assignment_spec field instead.

LbService.access_log_enabled:

Whether access log is enabled

AddControllerNodeSpec.control_plane_server_certificate:

Do not supply a value for this property.

ContainerConfiguration.weight:

Specify relative weight in WidgetItem for placement in a view. See WidgetItem for details.

ContainerConfiguration.shared:

Use the property 'shared' of View instead. The widgets of a shared view are visible to other users.

RealizationStateQueryParameters.barrier_id:

The system checks to ensure that the logical entity is realized or not at least at the given barrier number. This parameter has been deprecated. Use `request_id` instead.

StatsConfiguration.weight:

Specify relative weight in `WidgetItem` for placement in a view. See `WidgetItem` for details.

StatsConfiguration.shared:

Use the property `'shared'` of `View` instead. The widgets of a shared view are visible to other users.

BgpConfig.as_number:

Use `'as_num'` instead.

LogicalRouterLinkPortOnTIER1.edge_cluster_member_index:

Use logical router API to pass NSX Edge cluster members indexes manually.

TransportNode.host_switches:

Use `'host_switch_spec'` instead. Property `'host_switches'` can only be used for NSX managed transport nodes. `'host_switch_spec'` can be used for both NSX-T managed or manually preconfigured host switches.

NsxRole.permissions:

Use the `'/user-info/permissions'` API to get the permission that the user has on each feature.

WidgetConfiguration.weight:

Specify relative weight in `WidgetItem` for placement in a view. See `WidgetItem` for details.

WidgetConfiguration.shared:

Use the property `'shared'` of `View` instead. The widgets of a shared view are visible to other users.

DonutConfiguration.weight:

Specify relative weight in `WidgetItem` for placement in a view. See `WidgetItem` for details.

DonutConfiguration.shared:

Use the property `'shared'` of `View` instead. The widgets of a shared view are visible to other users.

MultiWidgetConfiguration.weight:

Specify relative weight in `WidgetItem` for placement in a view. See `WidgetItem` for details.

MultiWidgetConfiguration.shared:

Use the property `'shared'` of `View` instead. The widgets of a shared view are visible to other users.

PortConnectionHypervisor.pnics:

Use the `pnics_list` property instead.

GridConfiguration.weight:

Specify relative weight in `WidgetItem` for placement in a view. See `WidgetItem` for details.

GridConfiguration.shared:

Use the property `'shared'` of `View` instead. The widgets of a shared view are visible to other users.

PrincipalIdentity.permission_group:

Use the `'role'` field instead and pass in `'auditor'` for `read_only_api_users` or `'enterprise_admin'` for the others.

Deprecated NSX-T Data Center APIs

GetLogicalRouterRouteTableInCsvFormat (/logical-routers/<logical-router-id>/routing/route-table?format=csv):

Use /logical-routers/<logical-router-id>/routing/routing-table for RIB and /logical-routers/<logical-router-id>/routing/forwarding-table for FIB. Returns the route table in CSV format for the logical router on a node of the given transport-node-id. Query parameter "transport_node_id=<transport-node-id>" is required. The query parameter "source=cached" is not supported.

IncrementRealizationStateBarrier (/realization-state-barrier/current?action=increment):

Increment the current barrier number by one for NSX-T. To track realization state, use X-NSX-REQUESTID request header instead.

UpdateServiceProfile (/service-profiles/<service-profile-id>):

Modifies the specified service profile. The PUT request must include the resource_type parameters. Modifiable parameters include description and display_name. Other parameters might be modifiable, depending on the specified service type.

DeleteService (/services/<service-id>):

Deletes the specified logical router service.

ListServices (/services):

Returns information about all configured logical router services that can be applied to one or more logical router ports. You must create a service-profile before you can create a service. Currently, only DhcpRelayService is supported.

DeleteServiceProfile (/service-profiles/<service-profile-id>):

Deletes the specified service profile.

ListServiceProfiles (/service-profiles):

Returns information about all service profiles. A service profile is a configuration that you can use to create a service, which is then applied to one or more logical router ports. Currently, only the DhcpRelayProfile is supported.

UpdateSwitchIpflixConfig (/ipfix-obs-points/switch-global):

Use /ipfix-profiles/<ipfix-profile-id> for switch IPFIX profile and /ipfix-collector-profiles/<ipfix-collector-profile-id> for IPFIX collector profile.

GetCurrentBarrier (/realization-state-barrier/current):

Returns the current global realization barrier number for NSX-T. To track realization state, use X-NSX-REQUESTID request header instead.

CreateServiceProfile (/service-profiles):

Creates a service profile, which can then be used to create a service. Services are then applied to one or more logical router ports.

DeleteLicense (/licenses/<license-key>):

Use POST /licenses?action=delete API instead.

GetLicense (/license):

Deprecated. Use the GET /licenses API instead.

GetIpflixObsPoints (/ipfix-obs-points):

Use /ipfix-profiles for switch IPFIX profile and /ipfix-collector-profiles for IPFIX collector profile.

ReadServiceProfile (/service-profiles/<service-profile-id>):

Returns information about the specified service profile.

UpdateLicense (/license):

Use the POST /licenses API instead

CreateService (/services):

Creates a service that can be applied to one or more logical router ports. For some service types, you must create a service-profile before you can create a service.

UpdateService (/services/<service-id>):

Modifies the specified logical router service. The resource_type parameter is required. The modifiable parameters depend on the service type.

ReadService (/services/<service-id>):

Returns information about the specified service.

GetSwitchIpflixConfig (/ipfix-obs-points/switch-global):

Use /ipfix-profiles/<ipfix-profile-id> for switch IPFIX profile and /ipfix-collector-profiles/<ipfix-collector-profile-id> for IPFIX collector profile.

GetLicenseByKey (/licenses/<license-key>):

Use GET /licenses API instead.

GetLogicalRouterRouteTable (/logical-routers/<logical-router-id>/routing/route-table):

Use /logical-routers/<logical-router-id>/routing/routing-table for RIB and /logical-routers/<logical-router-id>/routing/forwarding-table for FIB. Returns the route table for the logical router on a node of the given transport-node-id. Query parameter "transport_node_id =<transport-node-id>" is required. The query parameter "source=cached" is not supported.

PerformNodeAction (/fabric/nodes/<node-id>):

The supported fabric node actions are enter_maintenance_mode, exit_maintenance_mode for EdgeNode. Use TransportNode maintenance mode API to update maintenance mode, refer to "Update transport node maintenance mode".

Deprecated Policy Manager Type Definitions

IpflixObsPointConfig:

Global switch IPFIX config is deprecated. Use IpflixSwitchUpmProfile and IpflixCollectorUpmProfile instead. Switch IPFIX profile can be applied to specific entities, such as logical switch, logical port and so on.

Logical:

Services are now defined in service-specific APIs, such as the DHCP relay service API, the NAT API, and others.

LogicalServiceListResult:

Services are now defined in service-specific APIs, such as the DHCP relay service API, the NAT API, and others.

LogicalService:

Services are now defined in service-specific APIs, such as the DHCP relay service API, the NAT API, and others.

LogicalServiceResourceTypes:

Resource types of logical services.

ServiceProfileListResult:

Services are now defined in service-specific APIs, such as the DHCP relay service API, the NAT API, and others.

IpfixObsPointsListResult:

Use `UpmProfileListResult` instead.

ServiceProfile:

Services are now defined in service-specific APIs, such as the DHCP relay service API, the NAT API, and others.

ServiceProfileResourceTypes:

Resource types of service profiles.

IpfixSwitchConfig:

Use `IpfixSwitchUpmProfile` instead which can specify its own collectors and observation ID.

Deprecated Policy Manager Property Definitions

LbVirtualServer.default_pool_member_port:

Use `'default_pool_member_ports'` instead. If `default_pool_member_port` is configured and `default_pool_member_ports` are not specified, both `default_pool_member_port` and `default_pool_member_ports` in response payload would return the same port value. If both are specified, `default_pool_member_ports` setting would take effect with higher priority.

LbVirtualServer.port:

Use `'ports'` instead. Port setting could be single port for both Layer 7 mode and Layer 4 mode. For Layer 4 mode, a single port range is also supported. The port setting could be a single port or port range such as "80", "1234-1236". If port is configured and ports are not specified, both ports in the response payload would return the same port value. If both ports are configured, port setting would take effect with higher priority.

BgpNeighbor.filter_in_routemap_id:

Use `'address_family'` instead.

BgpNeighbor.remote_as:

Use `'remote_as_num'` instead.

BgpNeighbor.filter_out_ipprefixlist_id:

Use `'address_family'` instead.

BgpNeighbor.filter_out_routemap_id:

Use `'address_family'` instead.

BgpNeighbor.source_address:

Do not provide a value for this field. Use `source_addresses` instead.

BgpNeighbor.filter_in_ipprefixlist_id:

Use `'address_family'` instead.

LabelValueConfiguration.weight:

Specify relative weight in `WidgetItem` for placement in a view. See `WidgetItem` for details.

LabelValueConfiguration.shared:

Use the property `'shared'` of `View` instead. The widgets of a shared view are visible to other users.

instead. The widgets of a shared view are visible to other users.

HostSwitch.static_ip_pool_id:

ID of configured Static IP Pool. If specified allocate IP for Endpoints from Pool. Else assume IP will be assigned for Endpoints from DHCP. Use ip_assignment_spec field instead.

LbService.access_log_enabled:

Whether access log is enabled.

AddControllerNodeSpec.control_plane_server_certificate:

Do not supply a value for this property.

MultiWidgetConfiguration.weight:

Specify relative weight in WidgetItem for placement in a view. See WidgetItem for details.

MultiWidgetConfiguration.shared:

Use the property 'shared' of View

ContainerConfiguration.weight:

Specify relative weight in WidgetItem for placement in a view. See WidgetItem for details.

ContainerConfiguration.shared:

Use the property 'shared' of View instead. The widgets of a shared view are visible to other users.

LbService.access_log_enabled:

Verify whether access log is enabled.

WidgetConfiguration.weight:

Specify relative weight in WidgetItem for placement in a view. See WidgetItem for details.

WidgetConfiguration.shared:

Use the property 'shared' of View instead. The widgets of a shared view are visible to other users.

StatsConfiguration.weight:

Specify relative weight in WidgetItem for placement in a view. See WidgetItem for details.

StatsConfiguration.shared:

Use the property 'shared' of View instead. The widgets of a shared view are visible to other users.

BgpConfig.as_number:

Use 'as_num' instead.

TransportNode.host_switches:

Use 'host_switch_spec' instead. Property 'host_switches' can only be used for NSX managed transport nodes. 'host_switch_spec' can be used for both NSX managed or manually preconfigured host switches.

NsxRole.permissions:

Use the /user-info/permissions api to get the permission that the user has on each feature.

DonutConfiguration.weight:

Specify relative weight in WidgetItem for placement in a view. See WidgetItem for details.

DonutConfiguration.shared:

Use the property 'shared' of View instead. The widgets of a shared view are visible to other users.

GridConfiguration.weight:

Specify relative weight in WidgetItem for placement in a view. See WidgetItem for details.

GridConfiguration.shared:

Use the property 'shared' of View instead. The widgets of a shared view are visible to other users.

PrincipalIdentity.permission_group:

Use the 'role' field instead and pass in 'auditor' for read_only_api_users or 'enterprise_admin' for the others.

Deprecated Policy Manger APIs

DeleteLicense (/licenses/<license-key>):

Use POST /licenses?action=delete API instead.

GetLicense (/license):

Use the GET /licenses API instead.

UpdateLicense (/license):

Use the POST /licenses API instead.

GetLicenseByKey (/licenses/<license-key>):

Use GET /licenses API instead.