Connected and Static IP Route

Introduction

We have seen in many customer's usage of applications on EVE devices, sometimes the applications can not reach the local network hosts due to them missing the device's external connected routes; also the applications sometimes can not reach the Internet due to we blindly forward the 'router' option in DHCP to applications. Some customers have a need to set static IP routes towards their gateway for the applications.

When Apps use more customized network functions on EVE devices, e.g. Firewall app, SD-WAN App, other Apps may need to have static routes to redirect data traffic to their own gateways.

Scope

Only IPv4 routes are involved in this design, since we are using DHCP options to 'program' the IP routes into the application side, and only IPv4 has such options. For 'Local' type of Network Instance, only IPv4 is supported today anyway. We can reevaluate this when we see IPv6 requirements in the future.

Unlike the 'Network' object, the 'Network Instance' object does not have a template for multiple edge-nodes to share; but the 'Network Instance' has the built-in deployment policy which can be shared by multiple edge-nodes. For this feature, we are not going to create another 'template' for the 'Network Instance' object.

User Interface

Users should be able to add one or more IPv4 prefix and gateway IP address pair in the 'Network Instance' object page. This is similar to the current 'Host to IP' mapping entries in the same object page. E.g. if user wants to add two Static IP Routes for this NI:

Prefix	Gateway
20.1.1.0/24	*192.168.1.1*
10.2.0.0/16	*192.168.1.1*

Users should be able to use 'ZCLI' to create/update/delete the static routes for the Network Instance. The 'zcli network-instance create/show/update /delete' needs modification to support this feature.

The requirement for the **connected** IPv4 routes of uplink device ports does not need explicit UI or ZCLI modifications, it is handled implicitly on EVE devices in this feature.

Terminology

NI - Network Instance Object

IP - IPv4 in the context, unless specified otherwise

Route - IPv4 Route

Implementation Considerations

- **Limit**
  1. Support up to 10 Static IP Routes for NI, or whatever a reasonable limit
- **NI Type**
  1. This feature applies only to 'Local' NI type
- **Device ports**
  1. A NI can have one or more external device ports
  2. The uplink port can be either 'Mgmt' or 'App-Shared' type
  3. The route must have a valid gateway IP address
  4. Zedcloud side will not validate the gateway IP address(whether it matches the port subnet or not), it is up to the EVE device to check
  5. The gateway IP address can not be the device's own interface IP address
  6. The gateway IP address must be part of the outbound port interface IP prefix
  7. When have multiple uplink ports, only one of them is selected at a time, thus the valid static routes may only be partially applied at a time in the IP routing tables
- **Default route 0.0.0.0/0**

i. On NI where the uplink is ‘Mgmt’ type, default route is ‘relayed’ into application automatically (as is in today’s behavior)
ii. On NI where the uplink is ‘App-Shared’, the default route is not blindly ‘relayed’ into application (this is different from today’s behavior). In order to avoid breaking the existing customer application behavior, the EVE side needs to make this decision based on if the App-Shared port does have a default route that goes through the port or not, mainly with the external DHCP server advertisement. If the uplink port does not have the default route on it, then the default route is suppressed for ‘relaying’ to the application using the NI.
iii. If the ‘App-Shared’ NI of the App does require the default route to be installed for application, the user can explicitly add the static route for the 0.0.0.0/0 for the NI.
iv. For NI with ‘App-Shared’, the lease time may be reduced to allow certain dynamics to be updated more quickly (as in above point 4b).

- **Connected routes**

i. A NI can have one or multiple uplink ports
ii. All the port prefix(es) is always implicitly ‘relayed’ into application with the IP gateway to the NI bridge interface IP address (e.g. bn2 IP address) to be the connected IP route

- **Static routes**

i. User defined static IP routes are always being ‘relayed’ into the applications, through DHCP, regardless of the route is actually installed into the linux ip routing table or not. If the external port is down, the application has no other alternative way to reach the destination anyway in general.
ii. The static route’s gateway address is replaced by the NI’s bridge interface IP address when being ‘relayed’ through the DHCP option.
iii. Multiple NI may share the same uplink port(s), and have different sets of static IP routes, they should all work independently

- **Air-gap NI**

i. This feature also works in the case of air-gap NI, where the uplink is ‘None’
ii. There is no ‘connected’ IP routes in this case, but can have static IP routes
iii. The IP route gateway IP address must be part of the NI bridge subnet (e.g. if the bridge prefix is 10.1.0.0/16, the route’s gateway IP may be **10.1.0.130**), and the gateway IP address must not be the bridge Interface address (such as bn1’s 10.1.0.1).
iv. Host side does not need to install the static ip route into IP routing table
v. If the air-gap NI has static IP routes enabled, then the ‘All-Ones-Mask’ must be disabled on this bridge for DHCP to assign a real mask on the application interface subnet.
vi. Although the App(e.g. a firewall, or SD-Wan app), we can call this **gateway-app**, with the gateway IP address (e.g. the App has eth0 IP address 10.1.0.130 as above) also receives this IP route advertisement through DHCP option, it does not seem to be harmful. If the gateway-app needs to redirect the data traffic towards their uplink connections, it can use ‘ip route’ to define a lower metrics for the route internally inside the app.

- **Network Object**

a. This is ‘Network’ and **NOT** ‘Network Instance’ object related
b. Today if the ‘Network’ object requires a static IP configuration, and Gateway IP address is required, and it can not be 0.0.0.0, this needs to be modified
c. On the zedcloud, we should allow the IP address of the Network Object ‘gateway’ address to be ‘0.0.0.0’
d. On the EVE device side, if the user manually enters the device physical interface gateway IP address of ‘0.0.0.0’, then no default IP route will be installed for the ‘Network’ on the device. This is related to the ***above point in (4b),*** the ‘relaying’ of the default IP route for ‘App-Shared’ type is based on if the route exists on the physical uplink port, we need to give users the option of suppressing this default IP route in their local network.
e. This is mainly useful when an ‘App-Shared’ port is statically configured with an IP address, usually it is not desirable to install a default IP route on it.