Setting Up a Basic Linux Network Namespace on One Host

Introduction

This is a demonstration of how to create two namespaces, assign IP addresses, and perform a ping test between them on a single Linux host.

You can follow my GitHub repository, BuildNetWorkToCloud,
to see examples of automated network creation
using open-source tools

https://github.com/zenithsoul

Requirement

1) ------

a Linux Host

2) ------

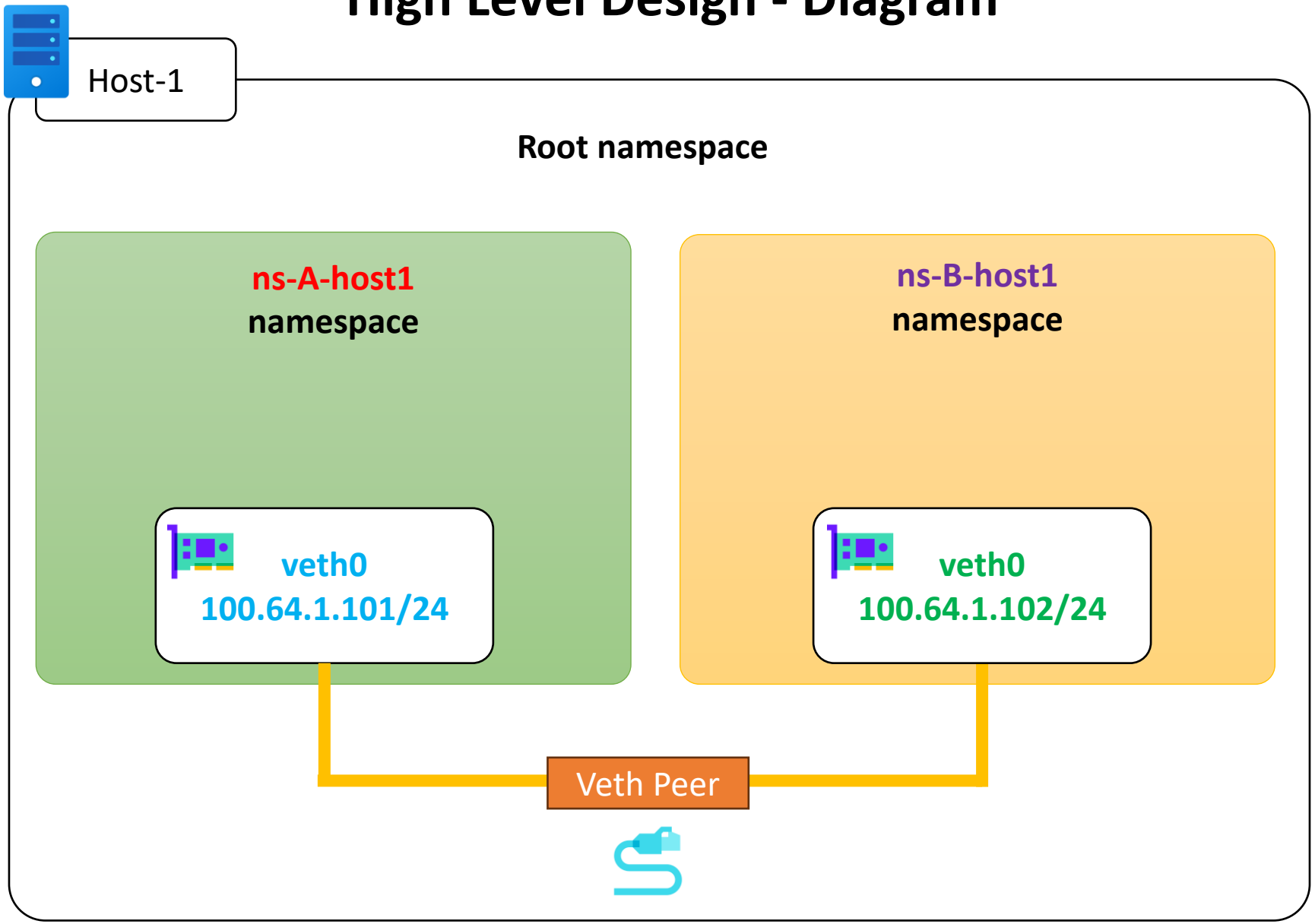
Ubuntu / Debian: iproute2

CentOS / RHEL / Fedora: iproute

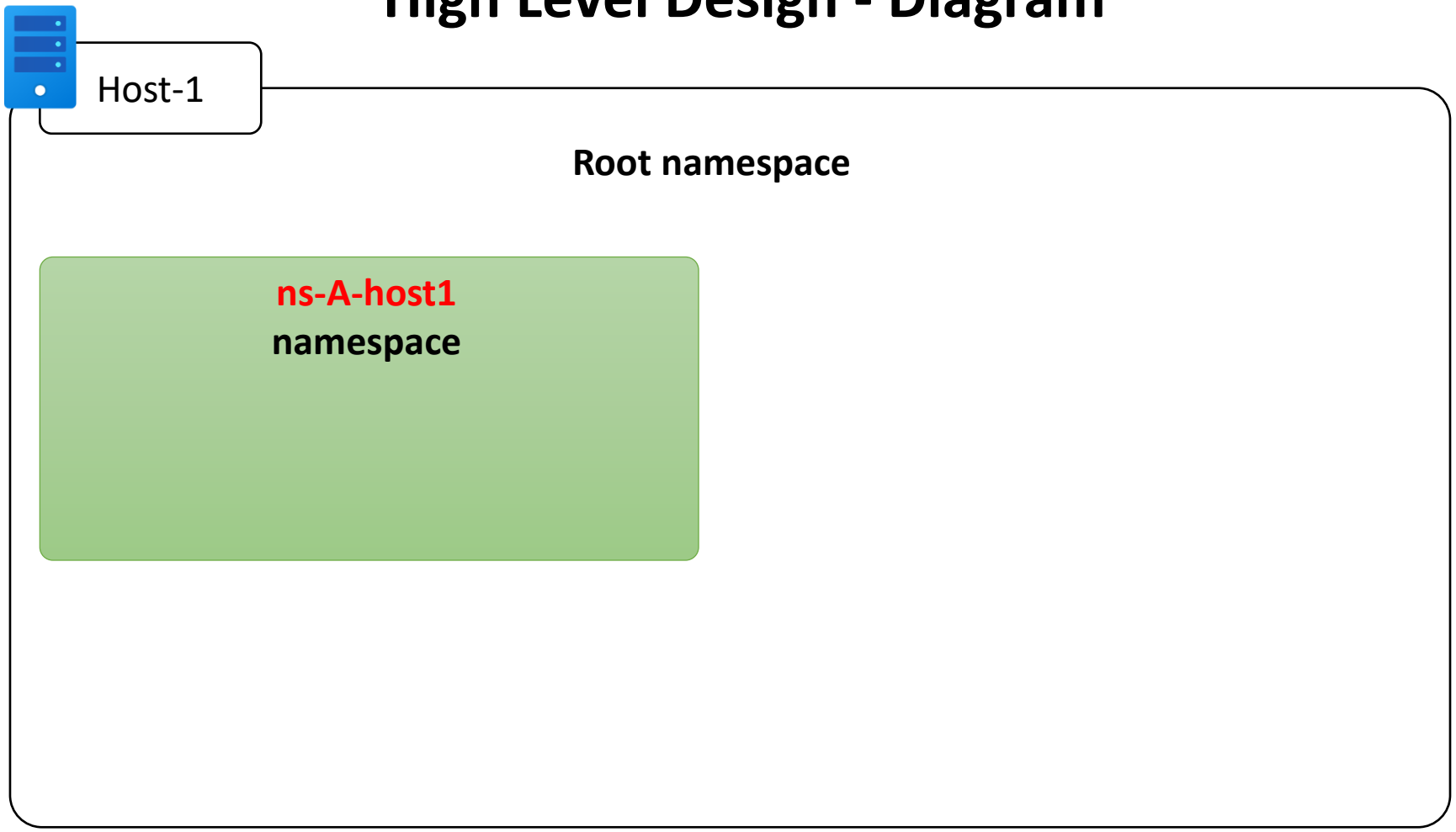
3) ------

Run with root account (sudo)

High Level Design - Diagram



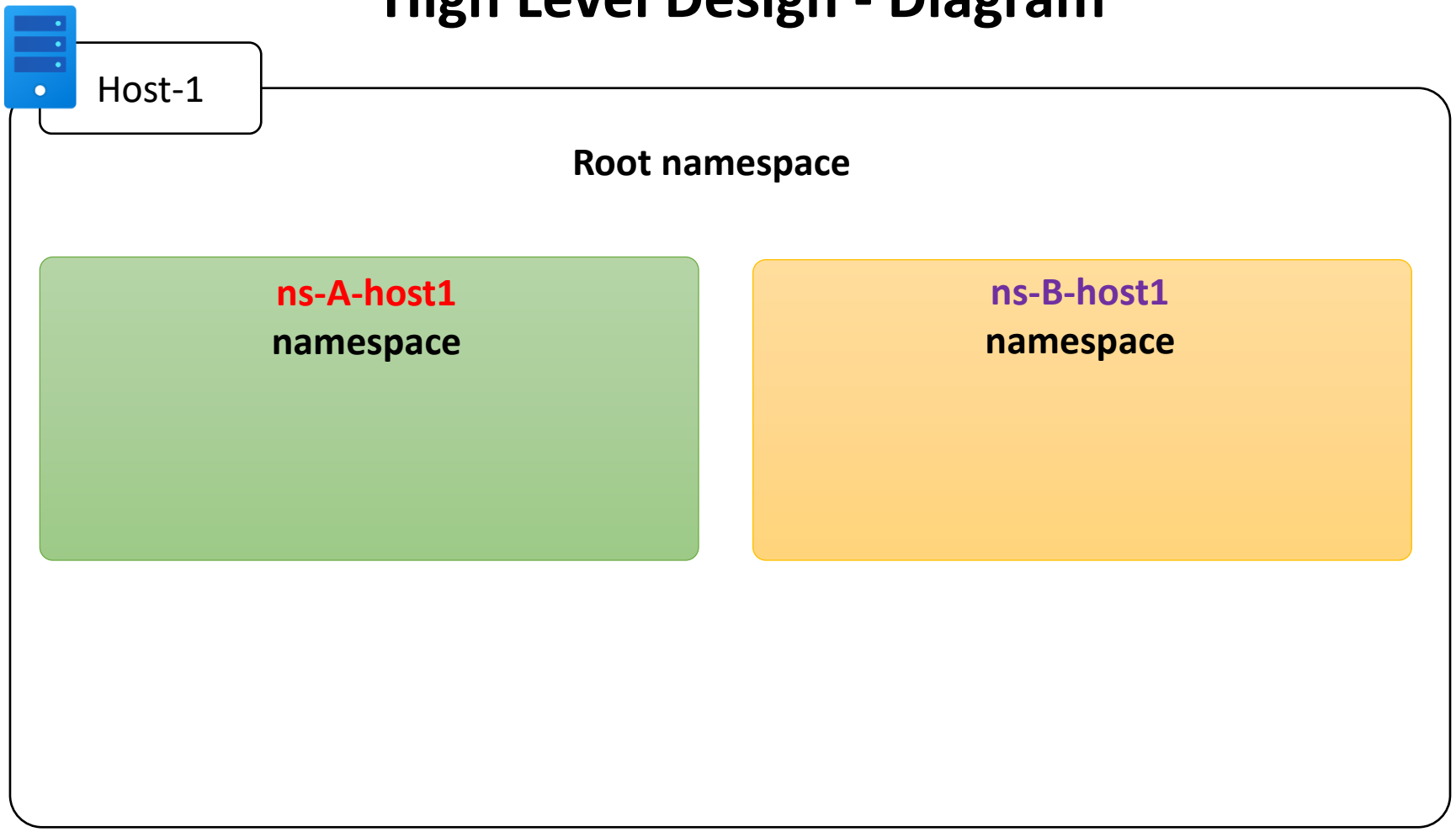
High Level Design - Diagram



Create a namespace > ns-A-host1

Host-1 # ip netns add ns-A-host1

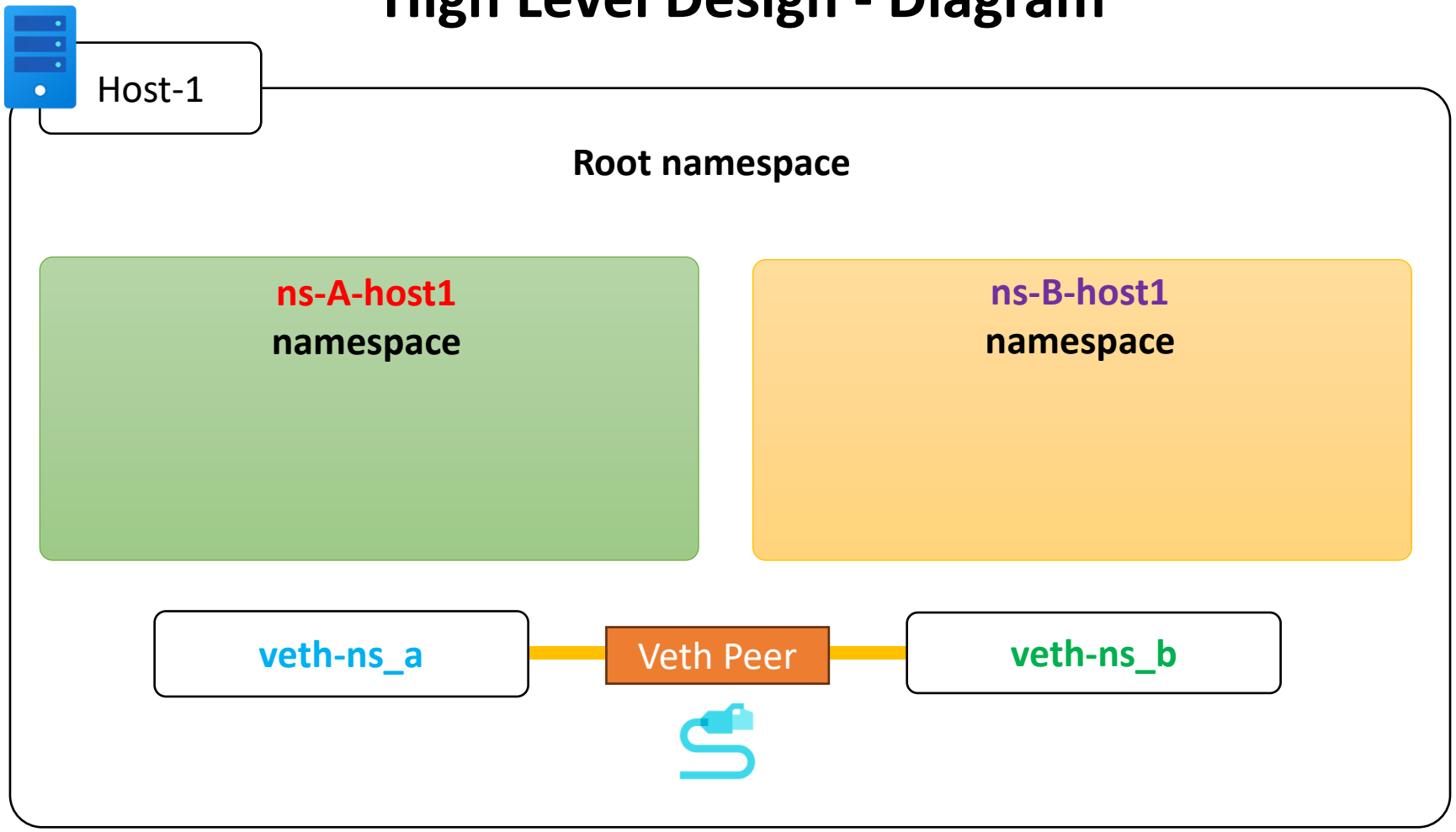
High Level Design - Diagram



Create a namespace > ns-B-host1

Host-1 # ip netns add ns-B-host1

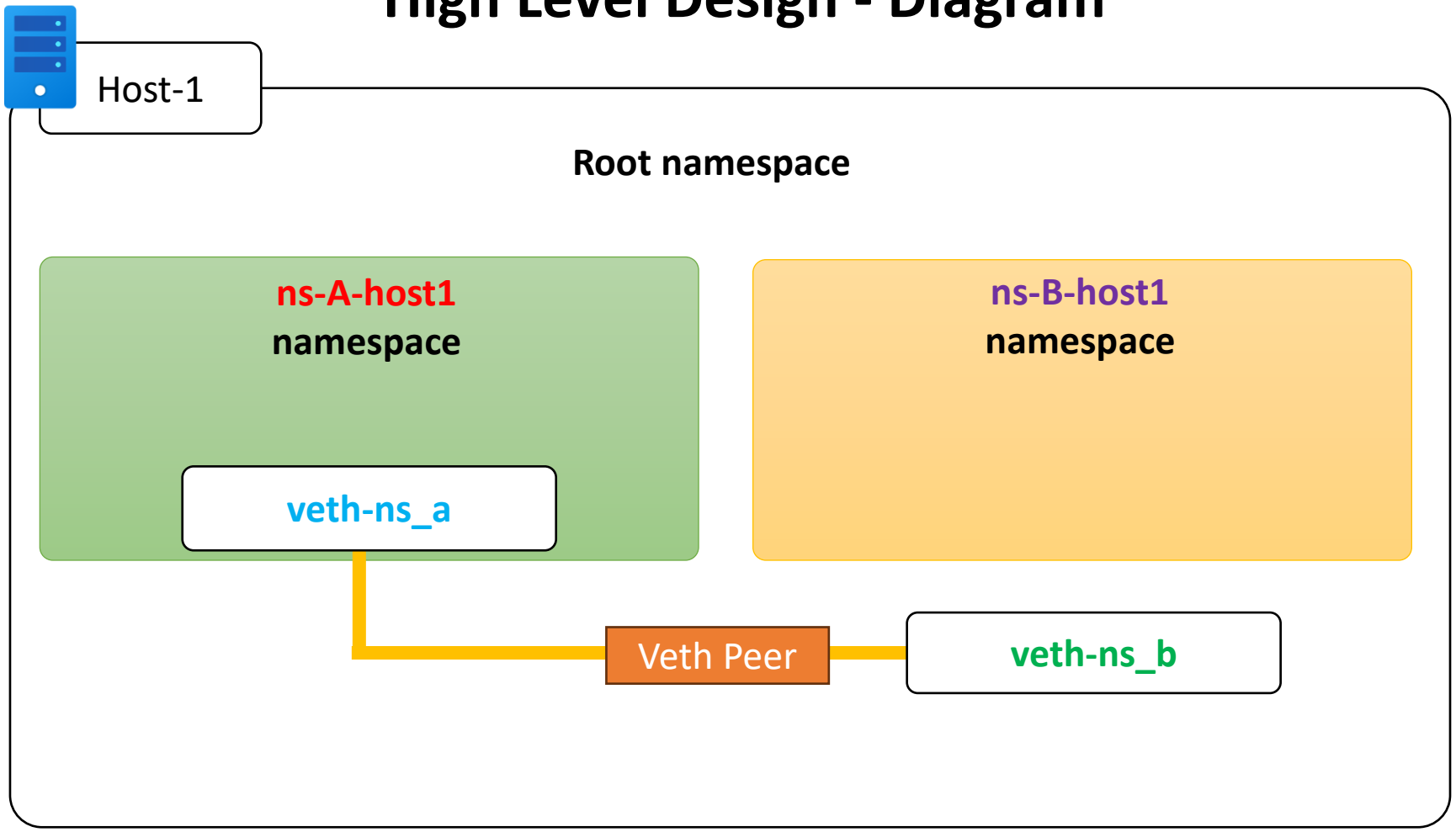
High Level Design - Diagram



Create the Veth Peer

Host-1 # ip link add veth-ns_a type veth peer name veth-ns_b

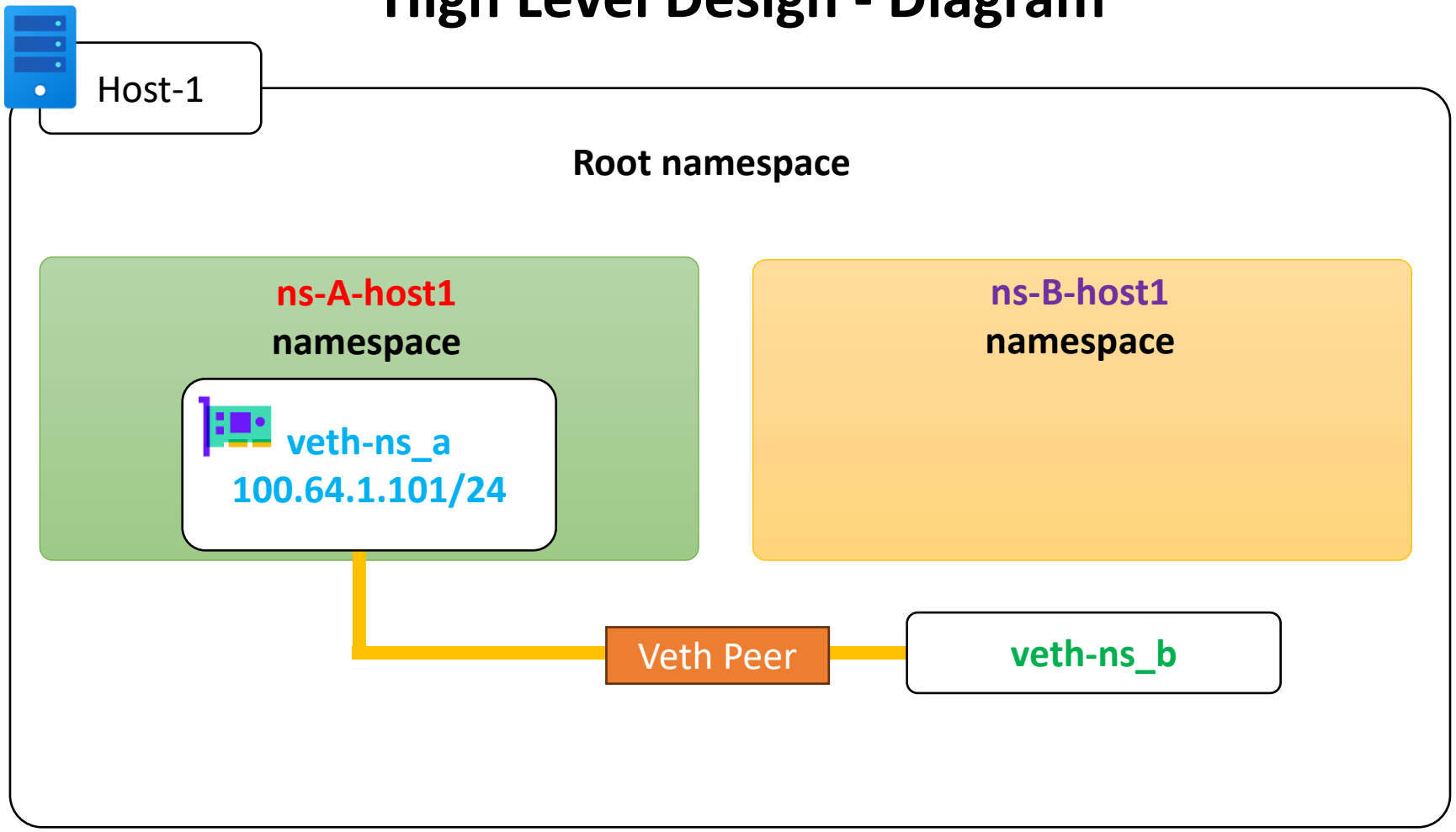
High Level Design - Diagram



Assign veth-ns_a to ns-A-host1

Host-1 # ip link set veth-ns_a netns ns-A-host1

High Level Design - Diagram

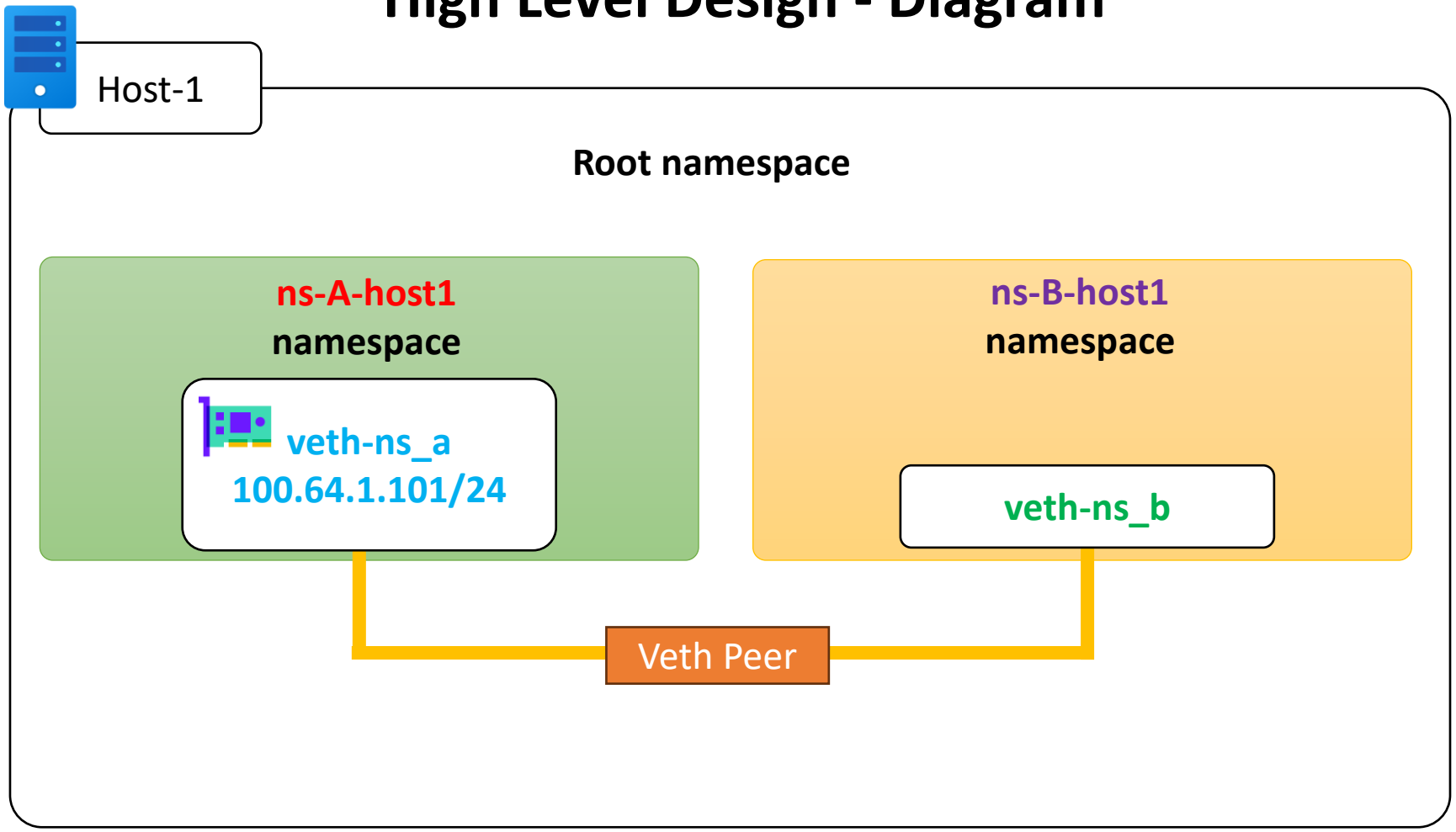


Configure ip address at veth-ns_a

Host-1 # ip netns exec ns-A-host1 ip addr add 100.64.1.101/24 dev veth-ns_a

Host-1 # ip netns exec ns-A-host1 ip link set veth-ns_a up

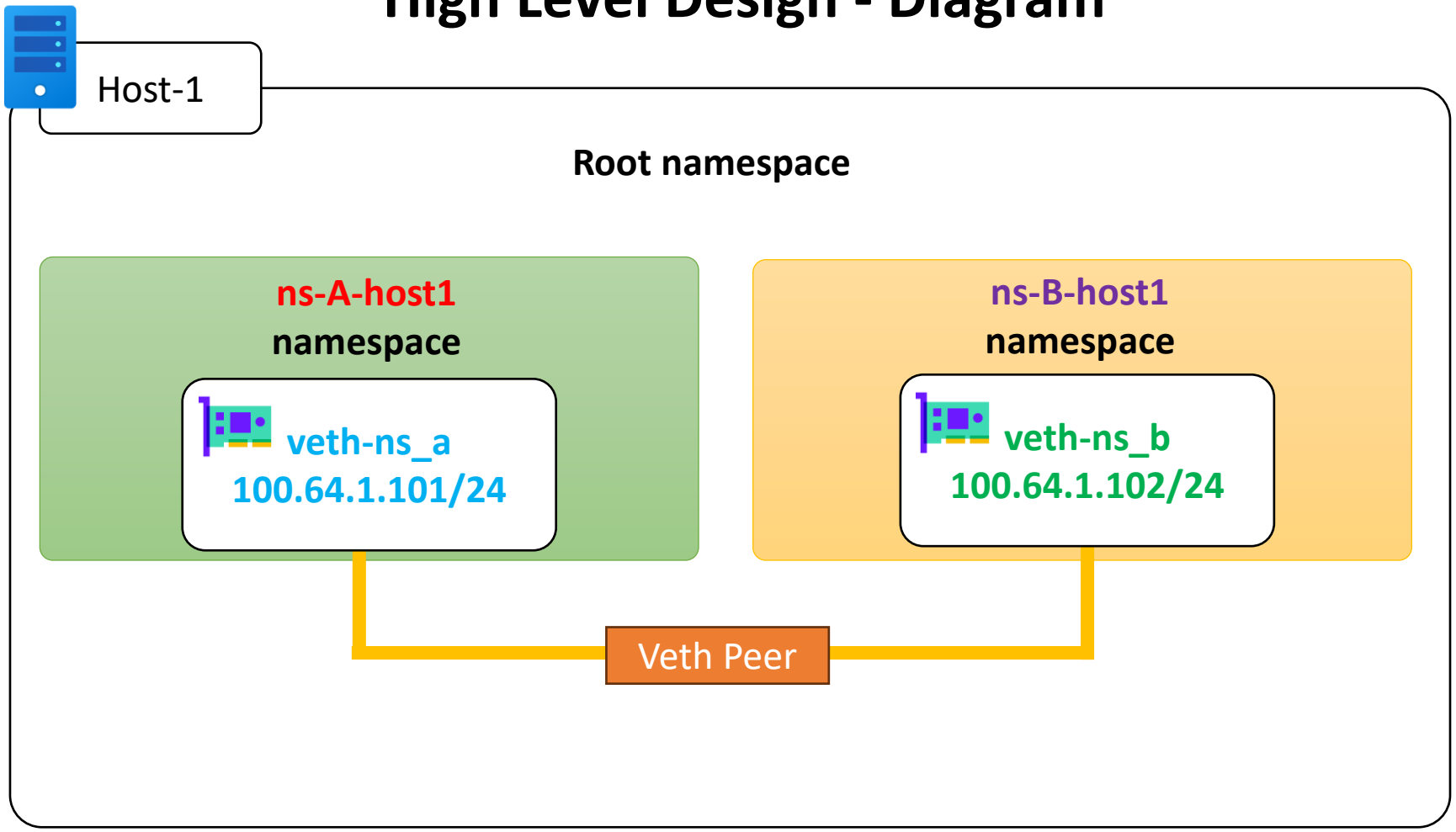
High Level Design - Diagram



Assign veth-ns_b to ns-B-host1

Host-1 # ip link set veth-ns_b netns ns-B-host1

High Level Design - Diagram

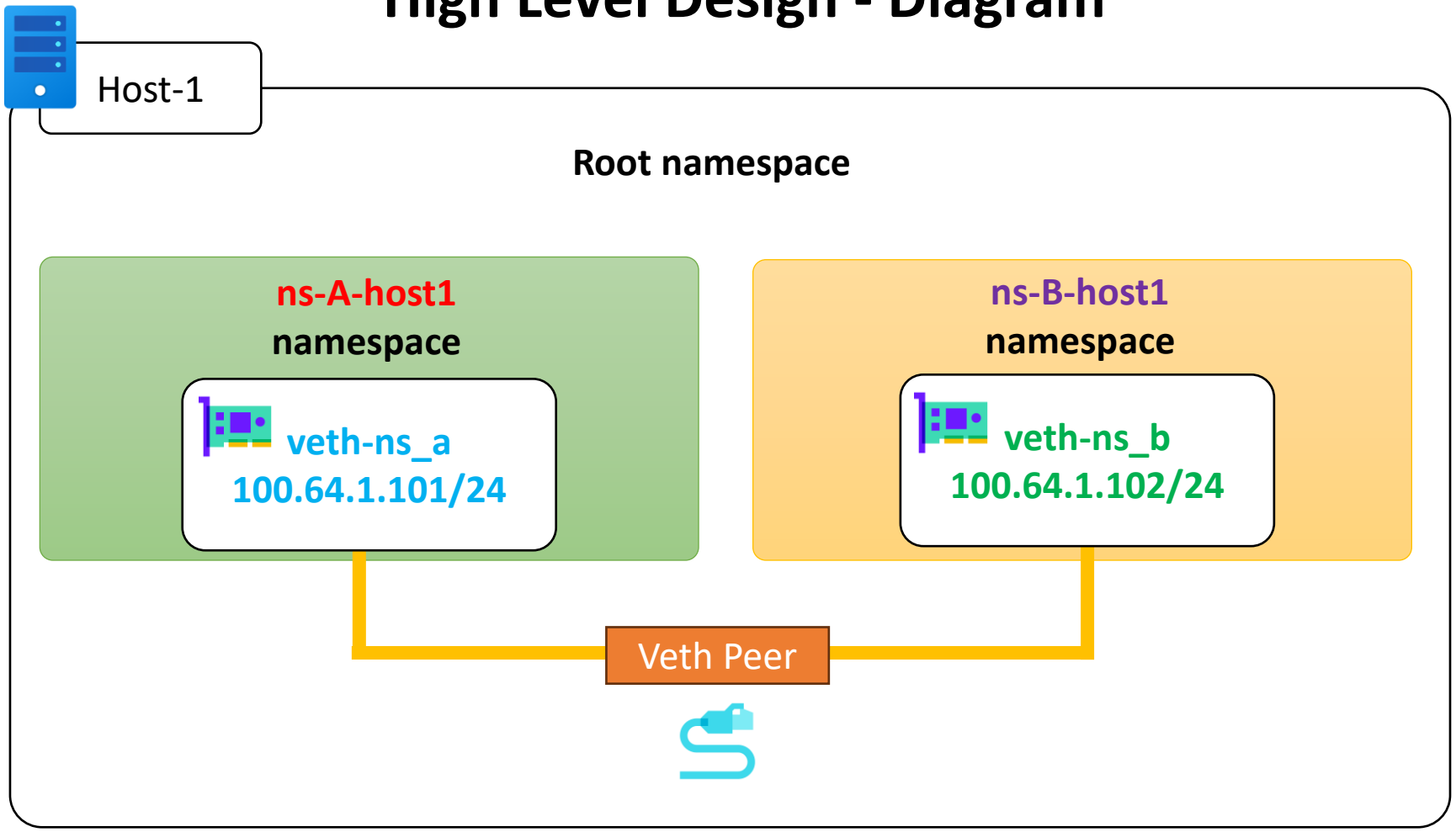


Configure ip address at veth-ns_b

Host-1 # ip netns exec ns-B-host1 ip addr add 100.64.1.102/24 dev veth-ns_b

Host-1 # ip netns exec ns-B-host1 ip link set veth-ns_b up

High Level Design - Diagram



Ping to ns-B-host1 from ns-A-host1	Ping to ns-A-host1 from ns-B-host1
ip netns exec ns-A-host1 ping 100.64.1.102 –c 3	ip netns exec ns-B-host1 ping 100.64.1.101 –c 3