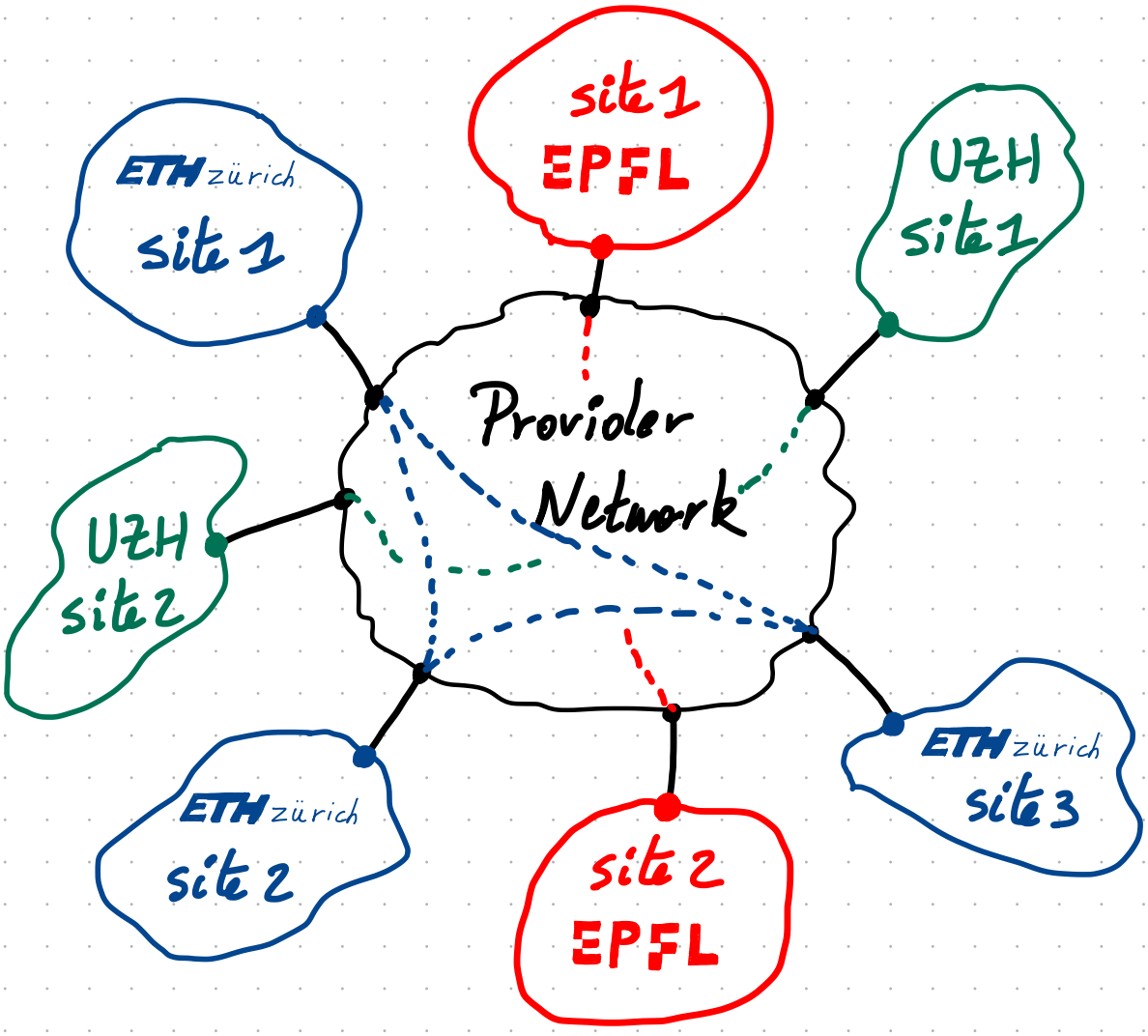
Advanced Topics in Communication Networks

L7: Virtual Private Networks / 27.10.2020

Prof. Laurent VANBEVER - nsg.ee.ethz.ch



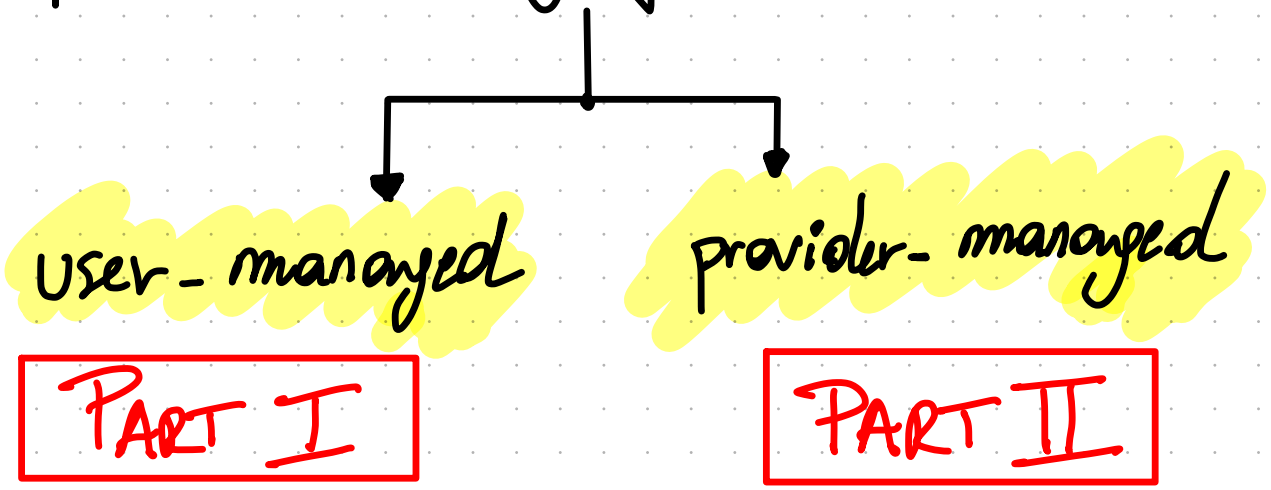
How do we provide a private network on top of a shared infrastructure?

How do we interconnect geographically-distributed sites with the "same" privacy and guarantees as a private network?

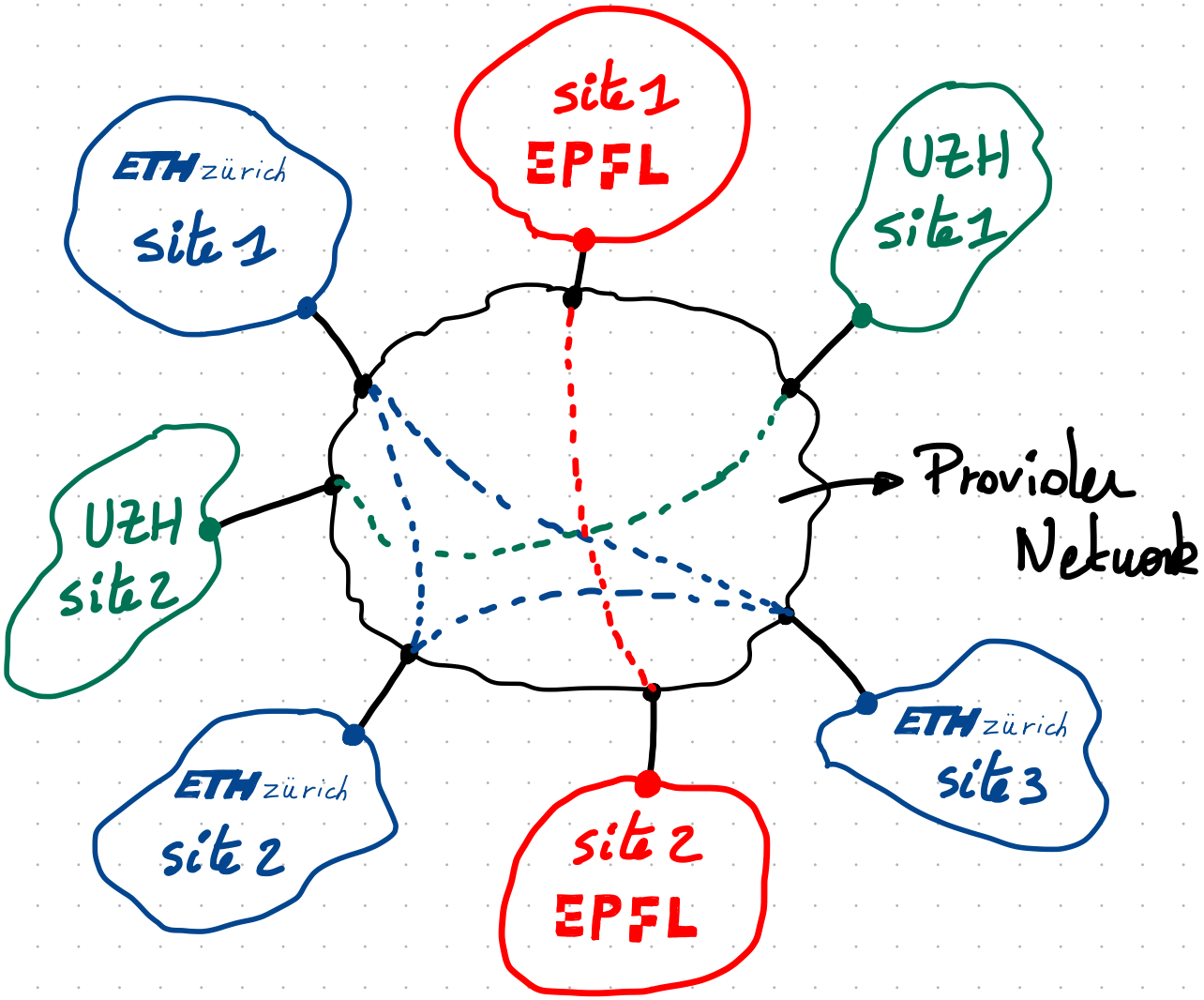
High-level goals

1. Support multiple customers.
2. Provide QoS guarantees.
3. Easy to use and manage for the customer and the provider.

Today's VPN solutions differ according to whom is responsible for setting them up and managing them



Part I: Customer-Managed VPNs



In a customer managed VPN, the provider is completely agnostic to the VPN service. It simply provides dedicated physical connections (leased lines) or IP connectivity.

Solution 1: LEASED LINES

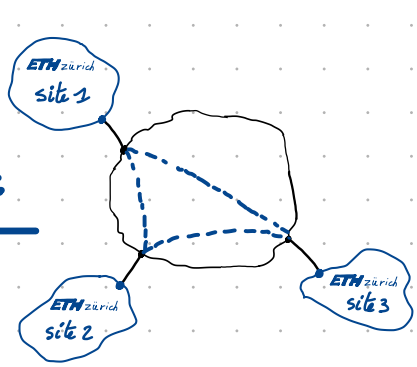
Definition: A Leased Line (LL) is a dedicated (private) connection between two geographically distant sites according to a commercial contract.

Since the connection is dedicated, the provider can provide QoS guarantees such as guaranteed bandwidth.

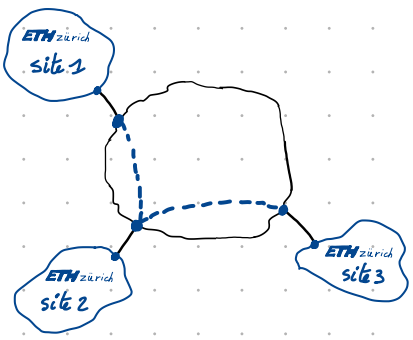
Example: A 100-Mbps connection set-up between Zürich and Geneva and provided by e.g. Swisscom.

Principle: By contracting LL between all its sites, a customer can easily establish a VPN.

Typical organizations:



FULL-MESH



HUB-AND-SPOKE

Pros:

- The quality of the connections is (typically) very good.

Cons:

- The number of LLs can be high (and therefore $$$):
  - $\frac{n(n-1)}{2}$ LLs for a full mesh with n sites
  - That also means n interfaces on each router (again this is $$$).
- LL-based VPNs are not flexible:
  - Adding a LL can take months!
  - Adding / Removing one site quickly becomes a logistic-nightmare.

Solution 2: IP-BASED VPN.

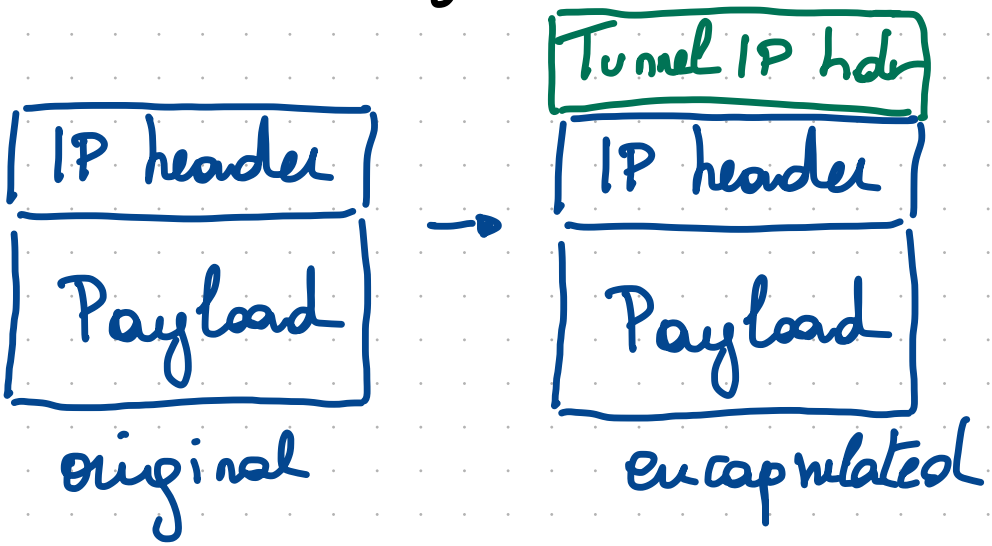
Principle: On each site, the customer contracts simple IP connectivity with one or more providers.

The customer then creates IP tunnels between its sites.

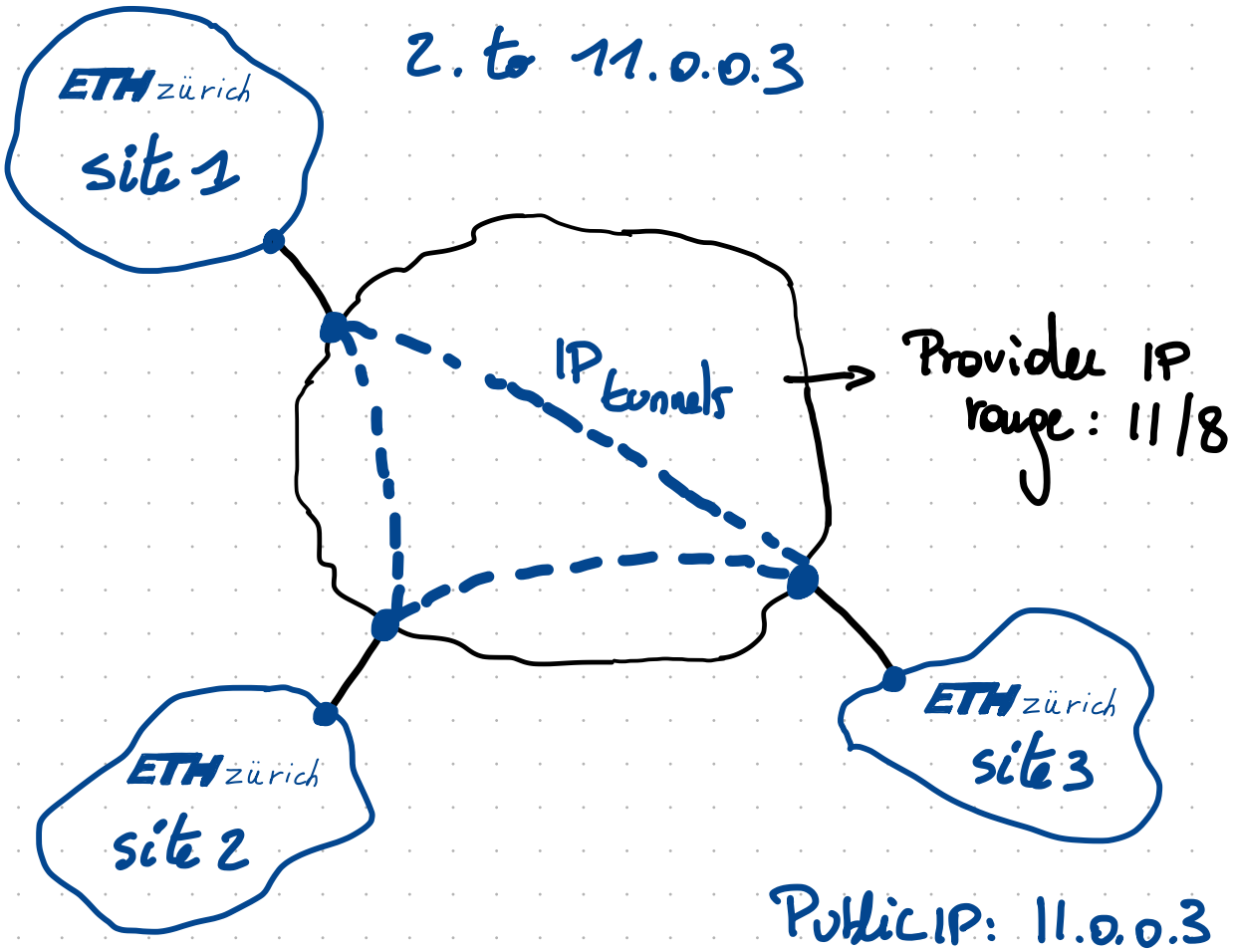
Tunnel: A tunnel is a mechanism to forward traffic across a network that wouldn't normally support it.

Tunneling is done by encapsulating the traffic to carry with a different header that the carrier network can process.

An IP tunnel uses another IP header on top of the original one (if any).



Public IP: 11.0.0.1
Private IP range: 10.1.0.0/24
IP tunnels:
1. to 11.0.0.2
2. to 11.0.0.3



Public IP: 11.0.0.2
Private IP range: 10.2.0.0/24
IP tunnels:
1. to 11.0.0.1
2. to 11.0.0.3

Public IP: 11.0.0.3
Private IP range: 10.3/24
IP tunnels:
1. to 11.0.0.1
2. to 11.0.0.2

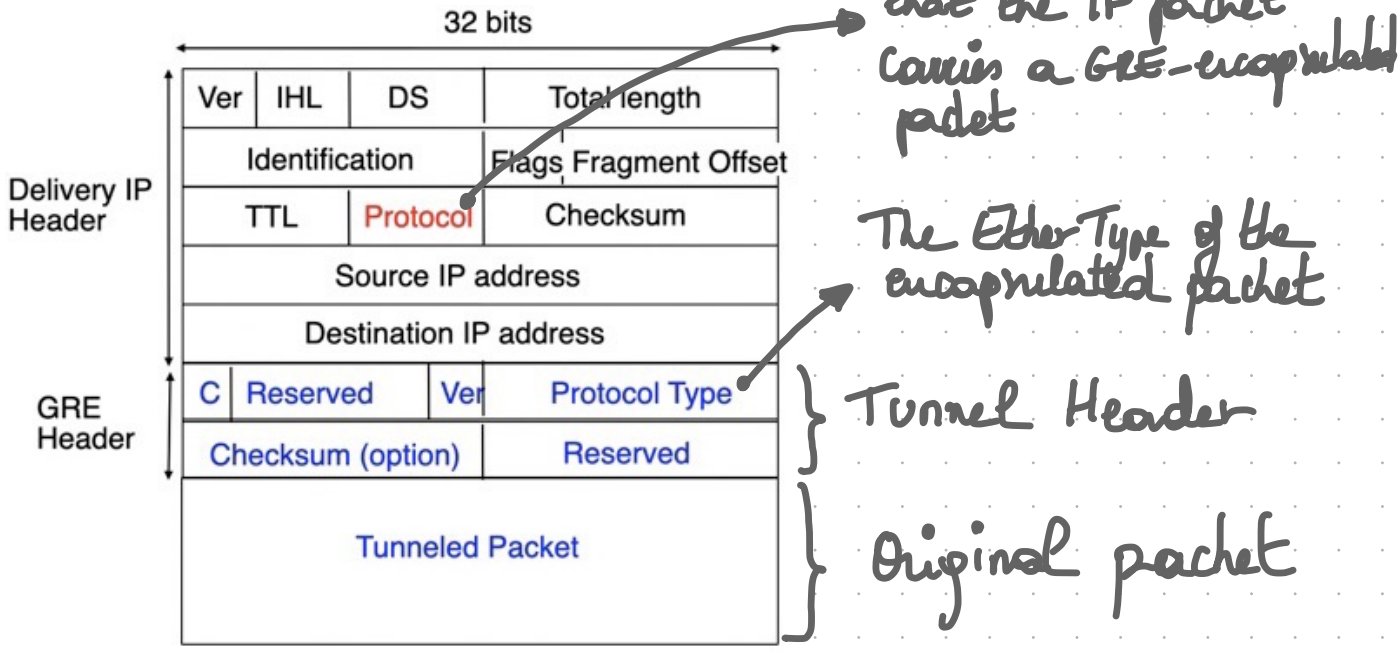
Note that, from the provider's viewpoint, all the packets are exchanged between {11.0.0.1, 11.0.0.2, 11.0.0.3}.

=> does not need to know anything about the internal IP ranges.

In practice, there exist a flurry of tunneling protocols including: IPinIP, GRE, L2TP, IPSEC

Some protocols, like IPSEC, encrypts the original IP packet - making it impossible for the provider to look at it anymore.

GRE Header:



Pros:

- Each router (in each site) only requires one interface to connect to all the others (2 for redundancy).
  Traffic for the ≠ sites is multiplexed.

Cons:

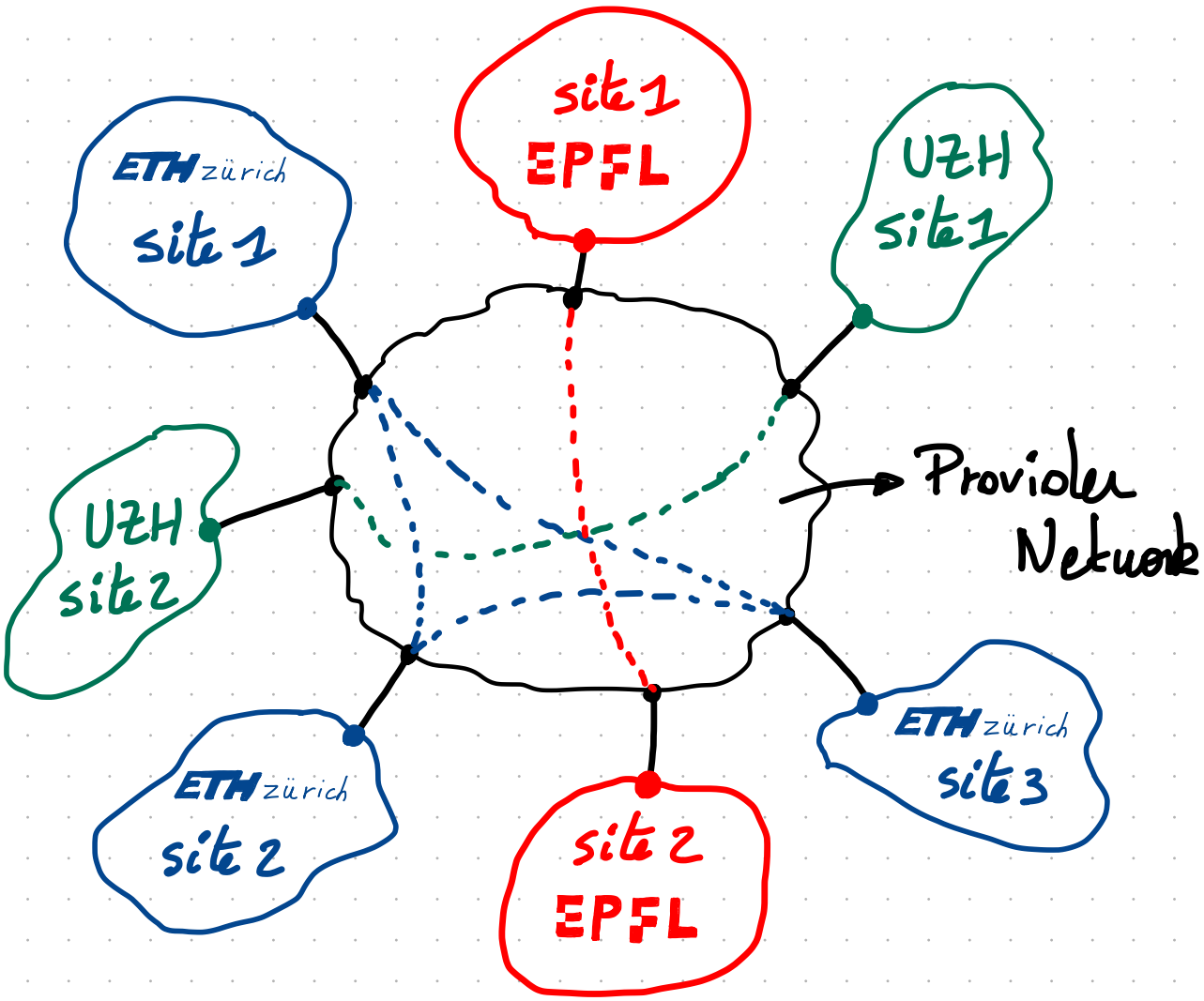
- The total number of tunnels can be high.
  - $\frac{n(n-1)}{2}$ for a full mesh of tunnels.

- Adding a new site requires to modify the configuration of all the others (assuming a full-mesh).

- These VPNs provide few guarantees. In particular, the provider is completely unaware of the fact that this is VPN traffic (it sees it as normal, plain, best effort traffic).

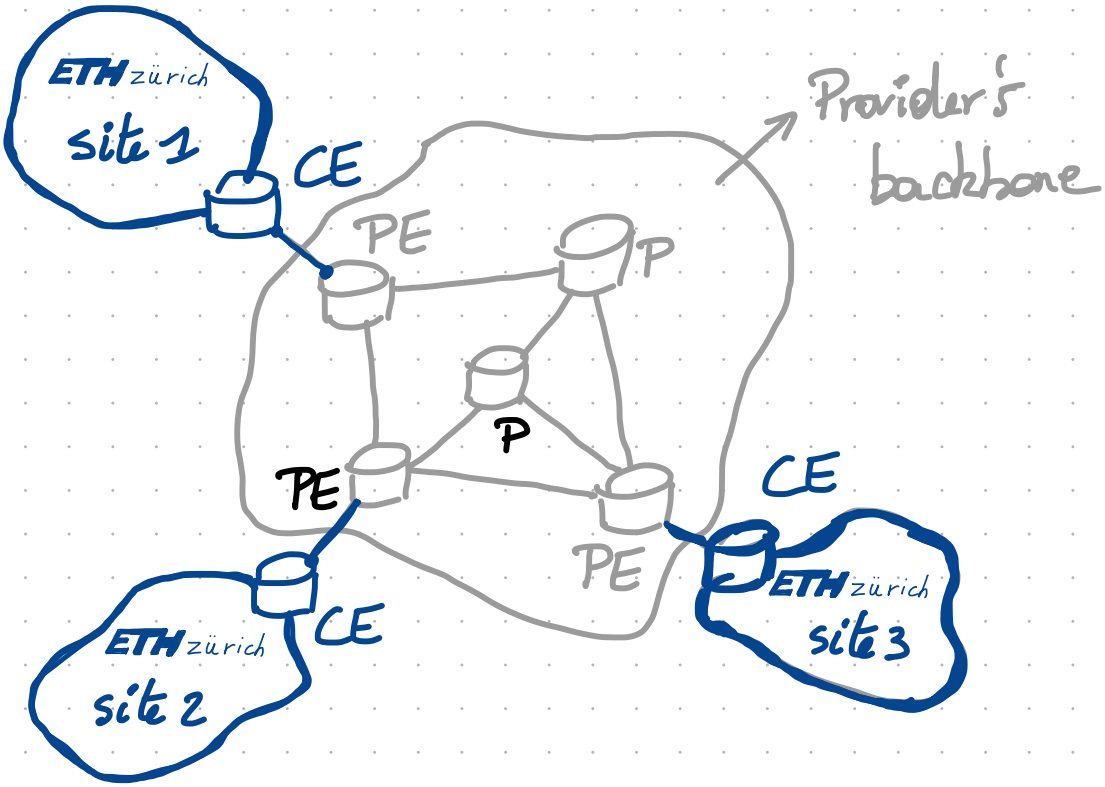
- The VPN security depends on the tunneling mechanism.
  weak for GRE, good for IPSEC.

PART II: PROVIDER-MANAGED VPNS



In a provider-managed VPN, the customer this time is agnostic to the service. For the customer, it is like its different sites are directly connected together through the provider.

Some terminology first:



- Customer Edge (CE): sends IP packets through the ISP backbone to reach the other sites of its VPN. A CE router does not know any details of the ISP backbone.
- Provider Edge (PE): maintains per-VPN configuration and ensures that the packets sent by a site are delivered to the PE attached to the PE router of the dest. site.
- Provider (P) router are within the ISP backbone and do not know anything about the VPN service.

There are 2 main problems to solve

1 Routing:

1.1. What kind of routes do we need where?

1.2. How do we distribute these routes?

1.3. How do we deal with conflicting route information?

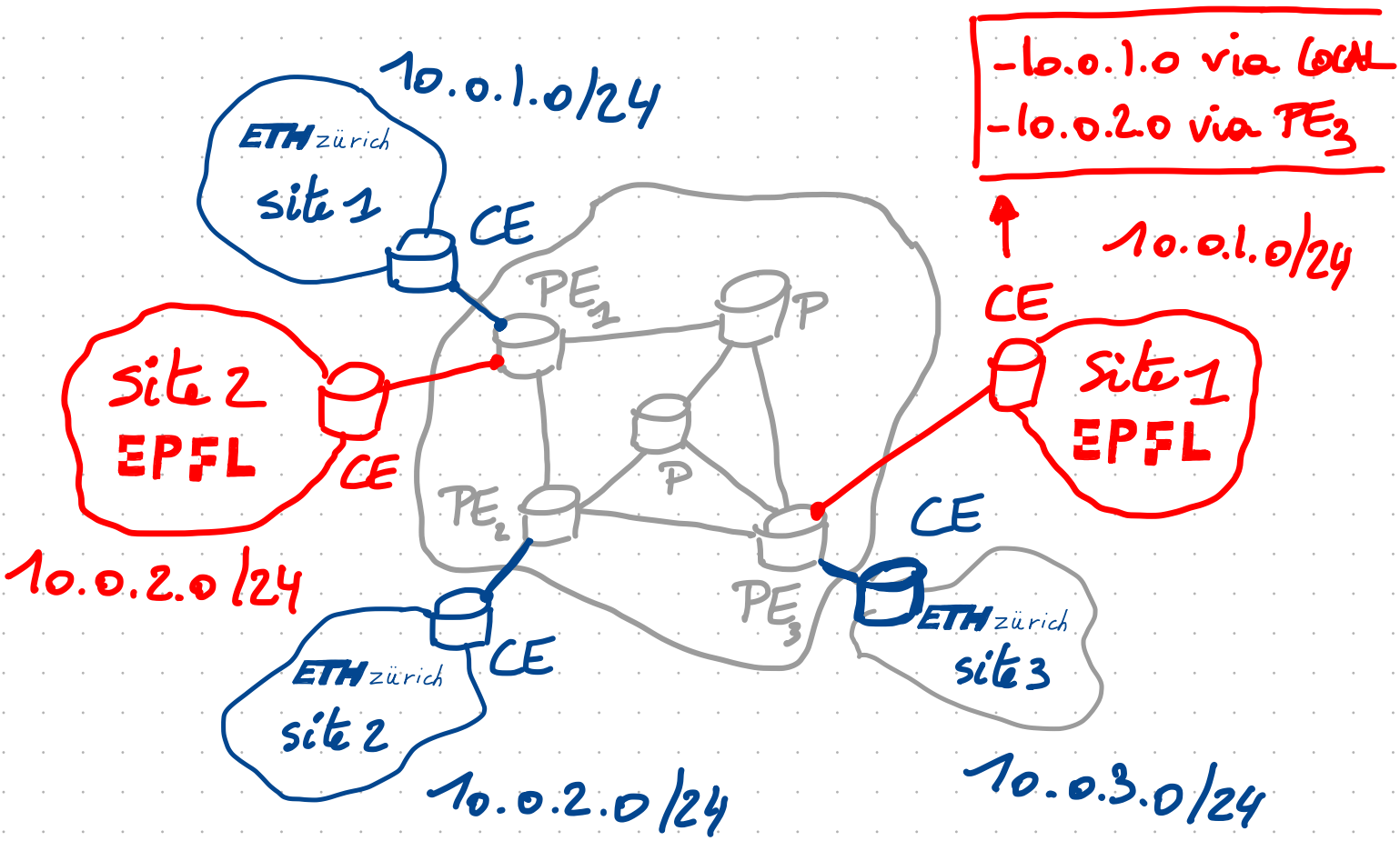
2 Forwarding:

2.1. How do we forward traffic in such a network?

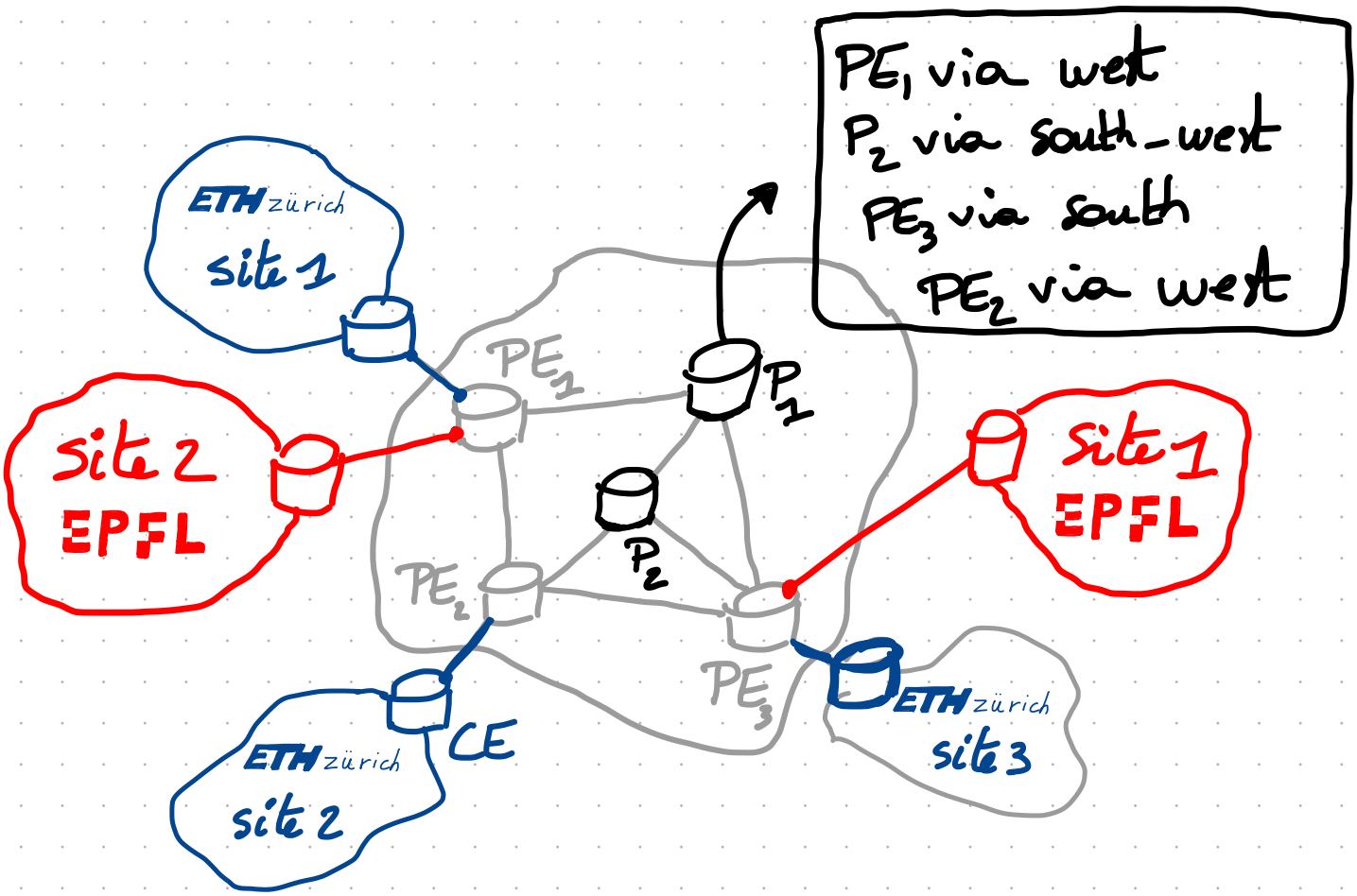
ROUTING IN A BGP/MPLS VPN:

1.1. What kind of routes do we need on each router?

CE routers: They only need one routing table containing the routes of their own VPN.



Prouters: They only need one routing table containing the routes of the backbone



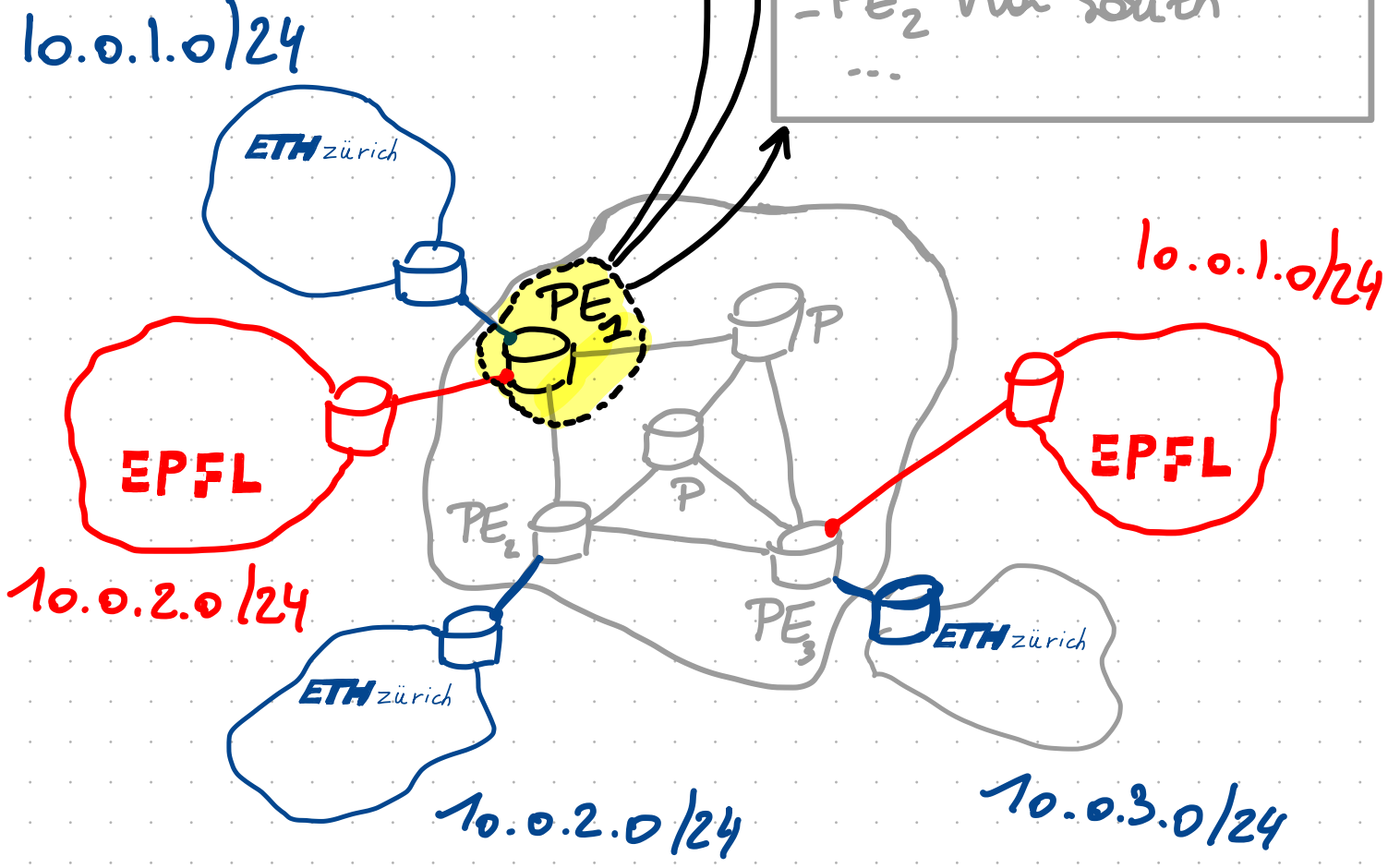
PE routers: In addition to the default routing table containing backbone routes, PE routers maintain one dedicated routing table for each VPN it is attached to.

This table is known as a VRF: VPN Routing and Forwarding table

VRF ETH:
- 10.0.1.0/24 via local
- 10.0.2.0/24 via PE_2
- 10.0.3.0/24 via PE_3

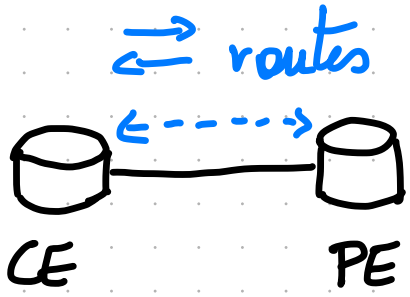
VRF EPFL:
- 10.0.1.0/24 via PE_3
- 10.0.2.0/24 via local

Default Routing Table:
- PE_2 via south
- ...



1.2. How do we distribute the routing information to the CE, PE, and P routers?

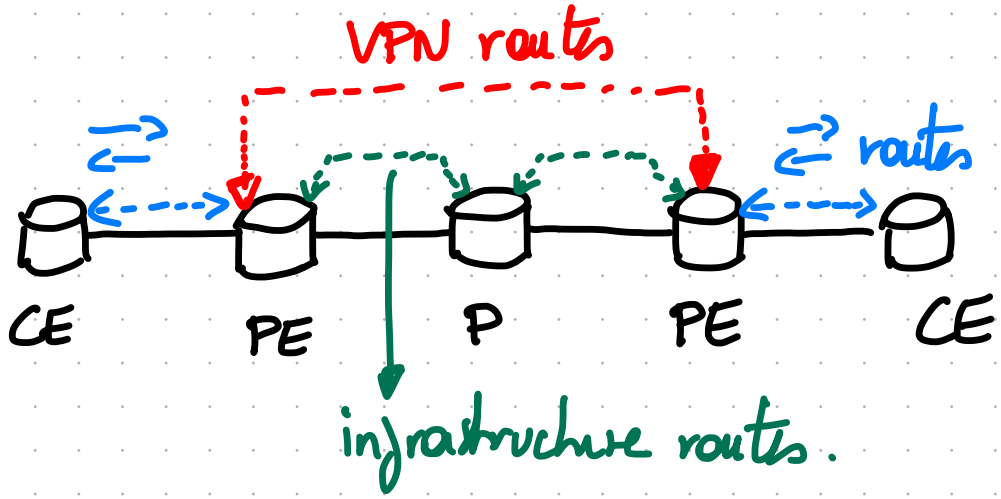
1)



Each CE must advertise its local routes to its connected PEs. These PEs in turn advertise the remote VPNs routes to the CE.

Common routing protocols spoken between the CE and the PE: static routes, eBGP, RIP, ...

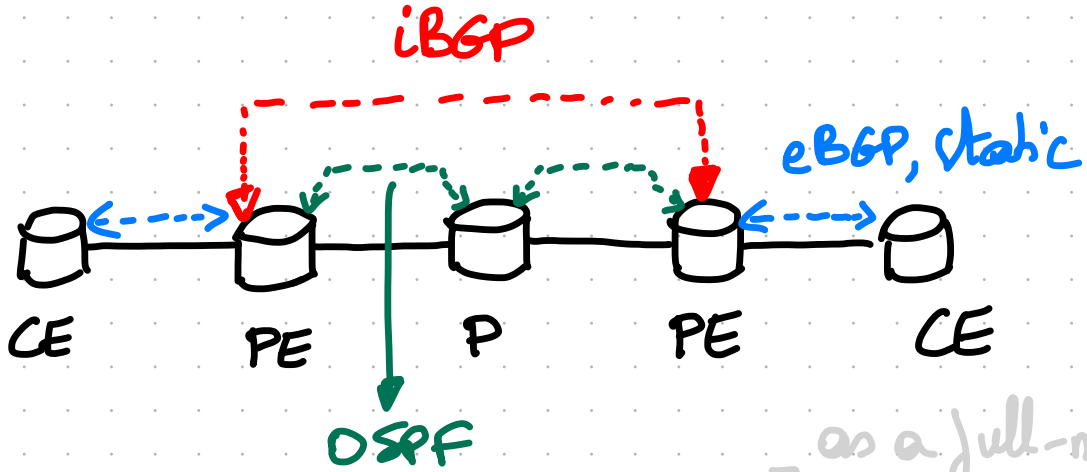
2)



Each PE must receive two types of routing info:

1. per-VPN routes reachable through local CE routers and remote PE routers.
2. internal ISP routes to reach other PE routers.

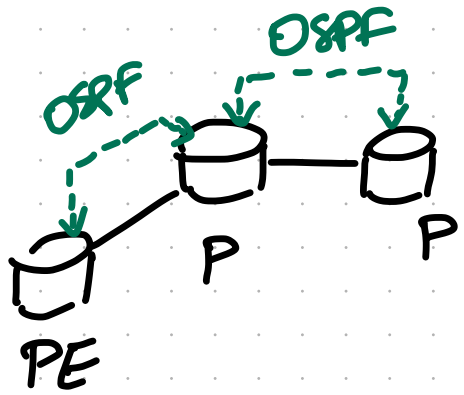
2)



as a full-mesh.

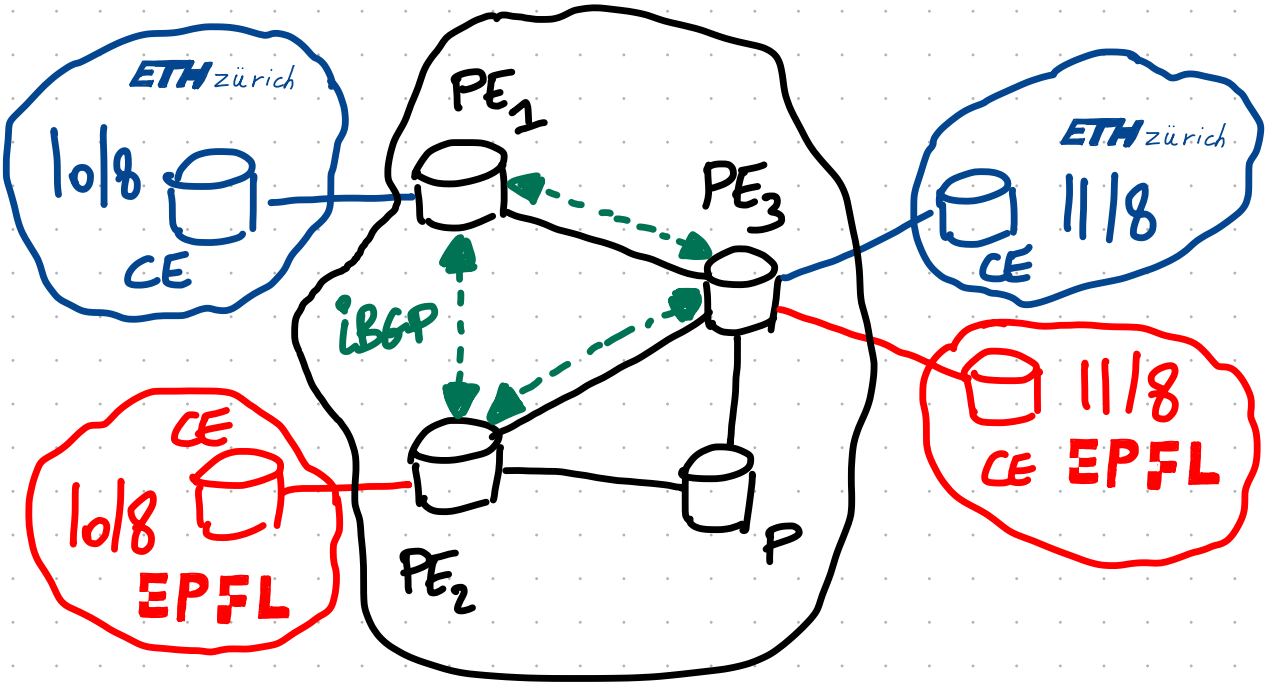
Typically, PEs talk iBGP with other PEs, alongside with a link-state routing protocol (OSPF, IS-IS) to learn the backbone route.

3)



PE routers only advertise/receive internal routes. It does not know anything about VPN routes.

1.3. How do we deal with conflicting routing information?



Problem 1: Since ≠ VPNs can use overlapping IP space, how do PEs distinguish between 10/8 advertised by ETH and 10/8 advertised by EPFL?

Problem 2: How do we ensure that the CEs of ETH and EPFL only learn the routes pertaining to their own VPN.

Solution I : Extend the notion of IP prefix

Insight: Ensure the uniqueness of the addresses by prepending them with a unique per-VPN identifier.

↳ The VPN-IPv4 address family:
8-bytes route distinguisher +
4-bytes IPv4 prefix.

Typical Route Distinguisher (RD):

* ASNumber : value
* IPv4Address : value.

PEs routers exchange VPN-IPv4 routes using BGP. More specifically, using MultiProtocol BGP.

MP-BGP: An extension to BGP that allows a BGP router to announce non-IPv4 routes such as:

- IPv6;
- Multicast;
- VPN routes.

Solution II: Restrict the distribution of VPN-IPv4 routes using labels.

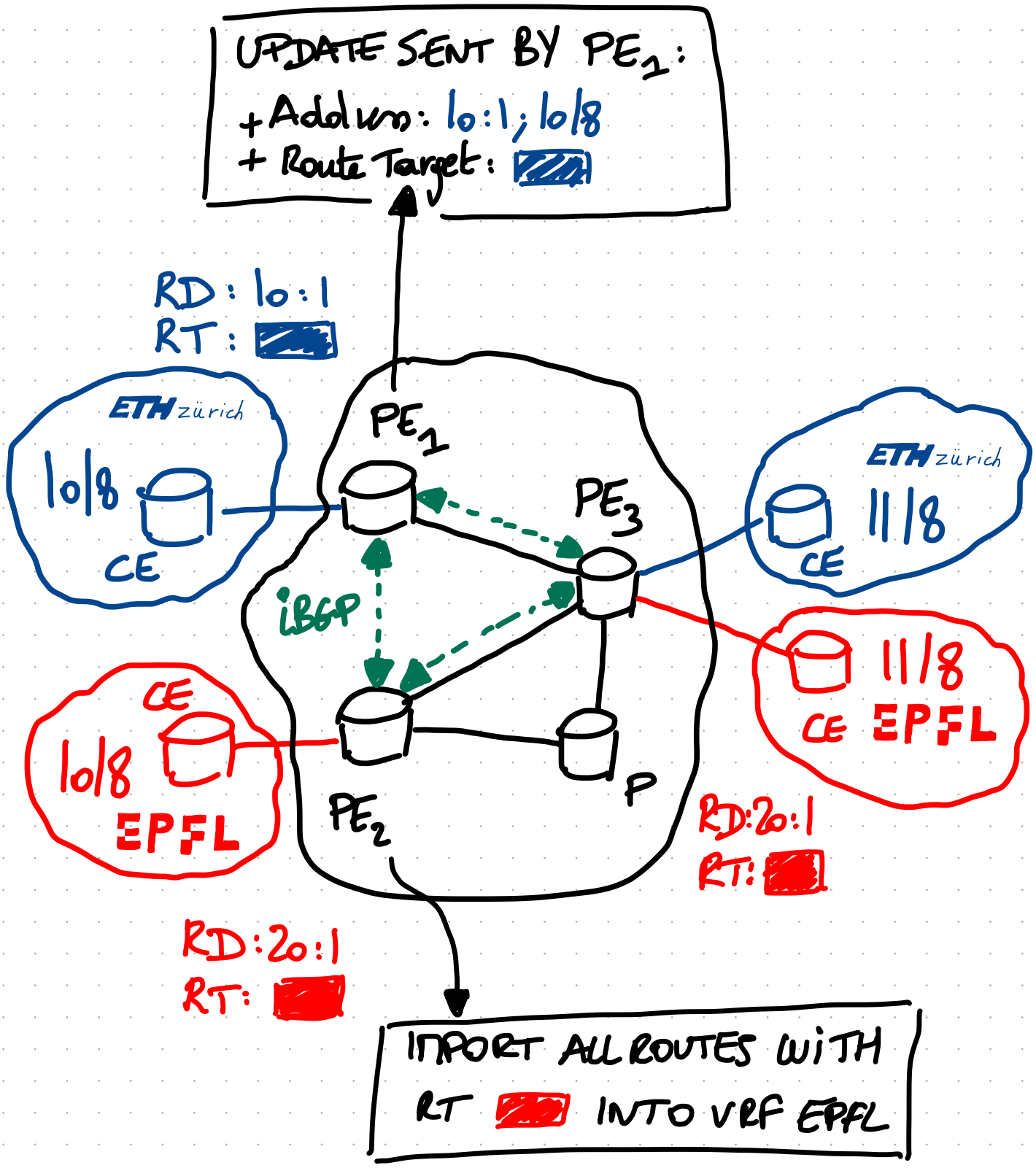
Insight:

- Assign a unique label to each VPN (ex. 1 for ETH, 2 for EPFL).
- Have the PE attach the tag to the BGP route before propagating it on iBGP.
- A PE router only imports routes associated with a given tag if they have a connected CE with this tag.

↳ This tag is known as a **Route Target**

Typical Route Targets (RT):

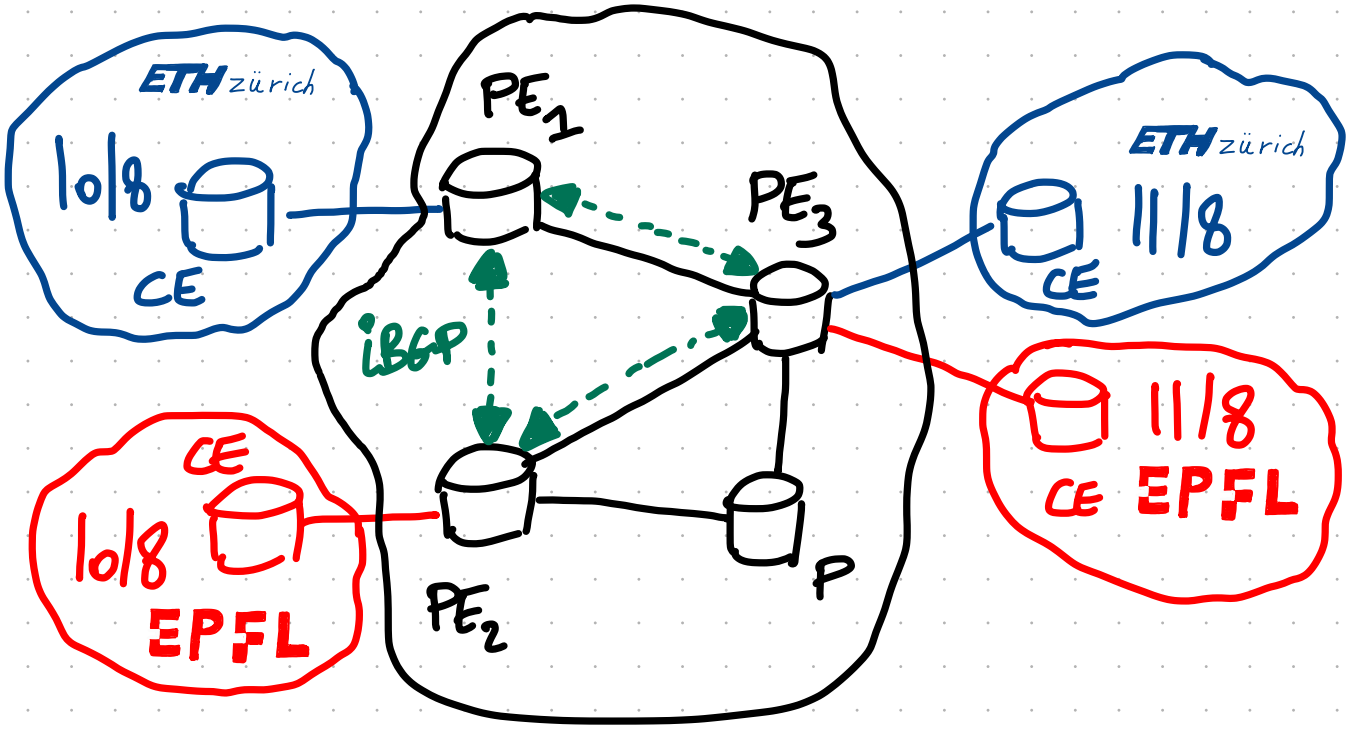
* ASNumber: value
* IPaddress: value



Remember that each PE maintains one VRF per VPN. Here, PE_1 and PE_2 have one VRF while PE_3 has two.

Routers use RT to figure out in which VRF to import each route.

FORWARDING IN A BGP/MPLS VPN:



2.1.

How do we forward traffic in such a network?

Insight:

- CE routers send pure IP pkt.
- PE routers encapsulate these IP packets with MPLS labels.

Each PE pushes two labels:

1) an outer label which identifies the next-hop PE;
2) an inner label which identifies the VRF to use in the remote PE.

Each PE learns the:

- outer label ... from LDP
- inner label ... from iBGP.
  Labels are piggybacked in the BGP UPDATE message using extension attribute.