Virtualization in Networks

- Virtualization of resources:
  - Powerful abstraction in systems engineering
  - Computing examples: virtual memory, virtual devices, virtual Oses
  - Layering of abstractions: don’t sweat the details of the lower layer, only deal with lower layers abstractly

- Virtualization in the Internet:
  - Virtual address spaces: NATs
  - Virtual links: Overlay routing
  - Virtual networks: PlanetLab, GENI
  - Virtual private networks (VPNs)

- Internet is a virtualized network!

Virtual Private Networks (VPN)

- VPNs
  - Networks perceived as being private networks by customers using them, but built over shared infrastructure owned by service provider (SP)

- SP infrastructure:
  - Backbone
  - Provider edge devices

- Customer:
  - Customer edge devices (communicating over shared backbone)
VPN Reference Architecture

VPN: Logical View

virtual private network
Leased-line VPN

Customer sites interconnected via static virtual channels (e.g., ATM VCs), leased lines

customer site connects to provider edge

Customer Premise VPN

All VPN functions implemented by customer

customer sites interconnected via tunnels
- tunnels encrypted typically
- SP treats VPN packets like all other packets
Drawbacks

- Leased-line VPN:
  - Configuration costs, maintenance by SP – long time, much manpower
- CPE-based VPN:
  - Expertise by customer to acquire, configure, manage VPN

- Network-based VPN
  - Customer’s routers connect to SP routers
  - SP routers maintain separate (independent) IP contexts for each VPN
    - Sites can use private addressing
    - Traffic from one VPN cannot be injected into another

Network-based Layer 3 VPNs

Tunnel encapsulation/de-capsulation performed in provider edge equipment

Normal IP access to PE CEs are not tunneling

multiple virtual routers in single provider edge device
Tunneling

VPN: Why?

- Privacy
- Security
- Works well with mobility (looks like you are always at home)
- Cost: newer forms of VPNs lower cost
  - Ability to share at lower layers (even though logically separate) lowers cost
  - Exploit multiple paths, redundancy, fault-recovery in lower layers
  - Need isolation mechanisms to ensure resources shared properly
- Abstraction and Manageability:
  - All machines with addresses that are “in” are trusted no matter where they are