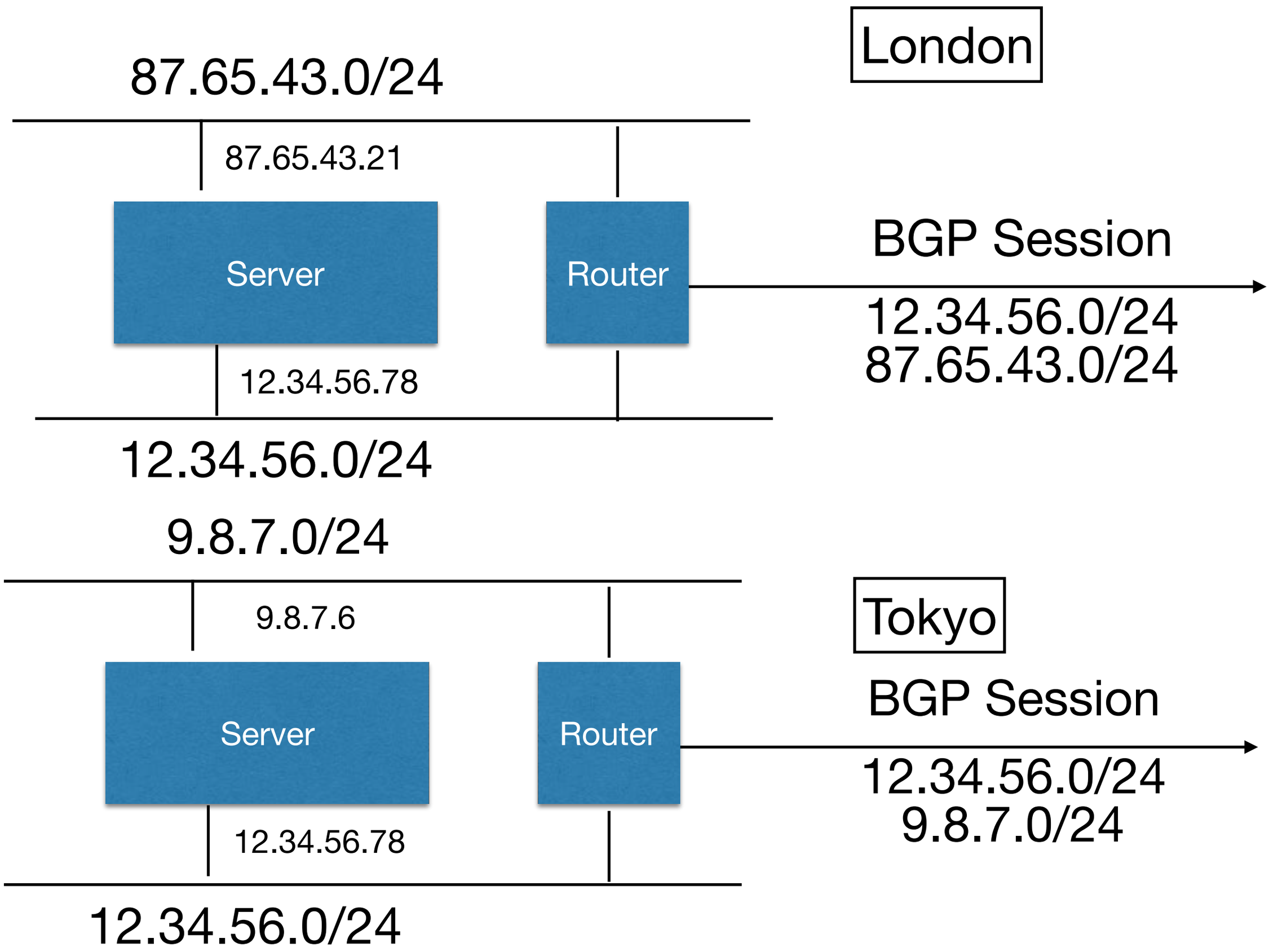
Security 16: Anycasting Redux, VPNs

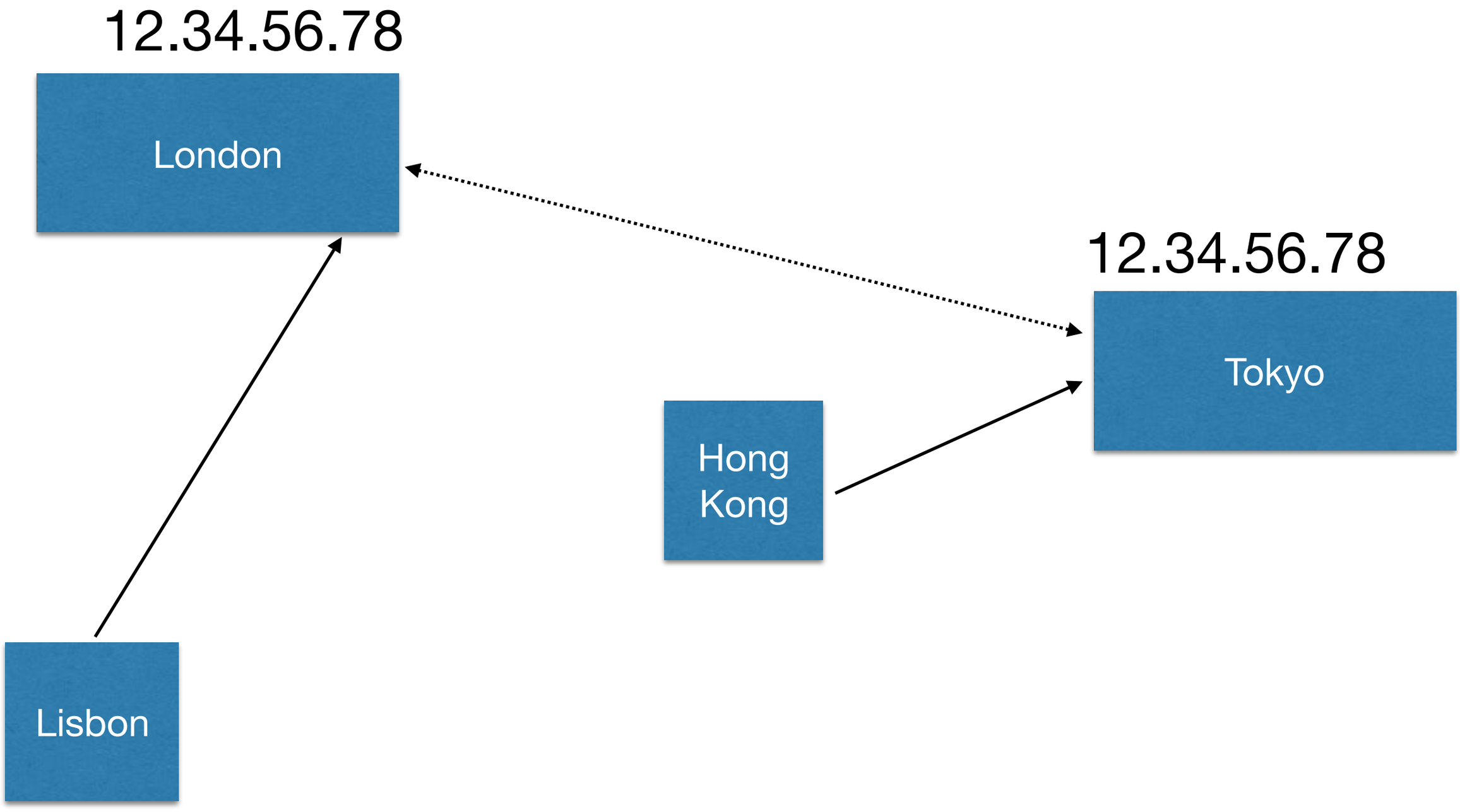
i.g.batten@bham.ac.uk

Worth Reading (esp. MSc Cyber Security Students)

- https://www.ncsc.gov.uk/blog-post/tls-13-better-individuals-harder-enterprises
- https://www.ncsc.gov.uk/node/2252
  - Serious guy (NCSC Technical Director, ex-GCHQ Technical Director) on balancing improved crypto against enterprise issues.
    - I disagree with his conclusions, but it's still very much worth reading.


London
87.65.43.0/24
87.65.43.21
Server
Router
BGP Session
12.34.56.0/24
87.65.43.0/24
12.34.56.78
12.34.56.0/24
9.8.7.0/24
Tokyo
9.8.7.6
Server
Router
BGP Session
12.34.56.0/24
9.8.7.0/24
12.34.56.78
12.34.56.0/24

Client



Anycasting

- 12.34.56.78 is “Anycast”: lots of alternative locations (two in this case, London and Tokyo) for the same service (presumably)
- 9.8.7.6 and 87.65.43.21 are standard “Unicast”: uniquely identify a machine.

Anycasting

- You can replicate data between the instances using their unique IP numbers, and then access it using the shared address.
- Failover is the BGP convergence time, a few minutes worst case
- Alternative is multiple A records, but failover requires removing the machine from the RRSet and waiting for caches to flush (can take up to an hour, as some systems override short TTLs)
    - And Multiple A records doesn’t give geographic sorting

Anycasting

- Best fit to UDP services, as service will survive route instability
- Bad fit to long-lived TCP connections, as routing changes will leave connections in a bad state
- Acceptable fit to short-lived TCP connections, such as for HTTP, so used by CDNs.

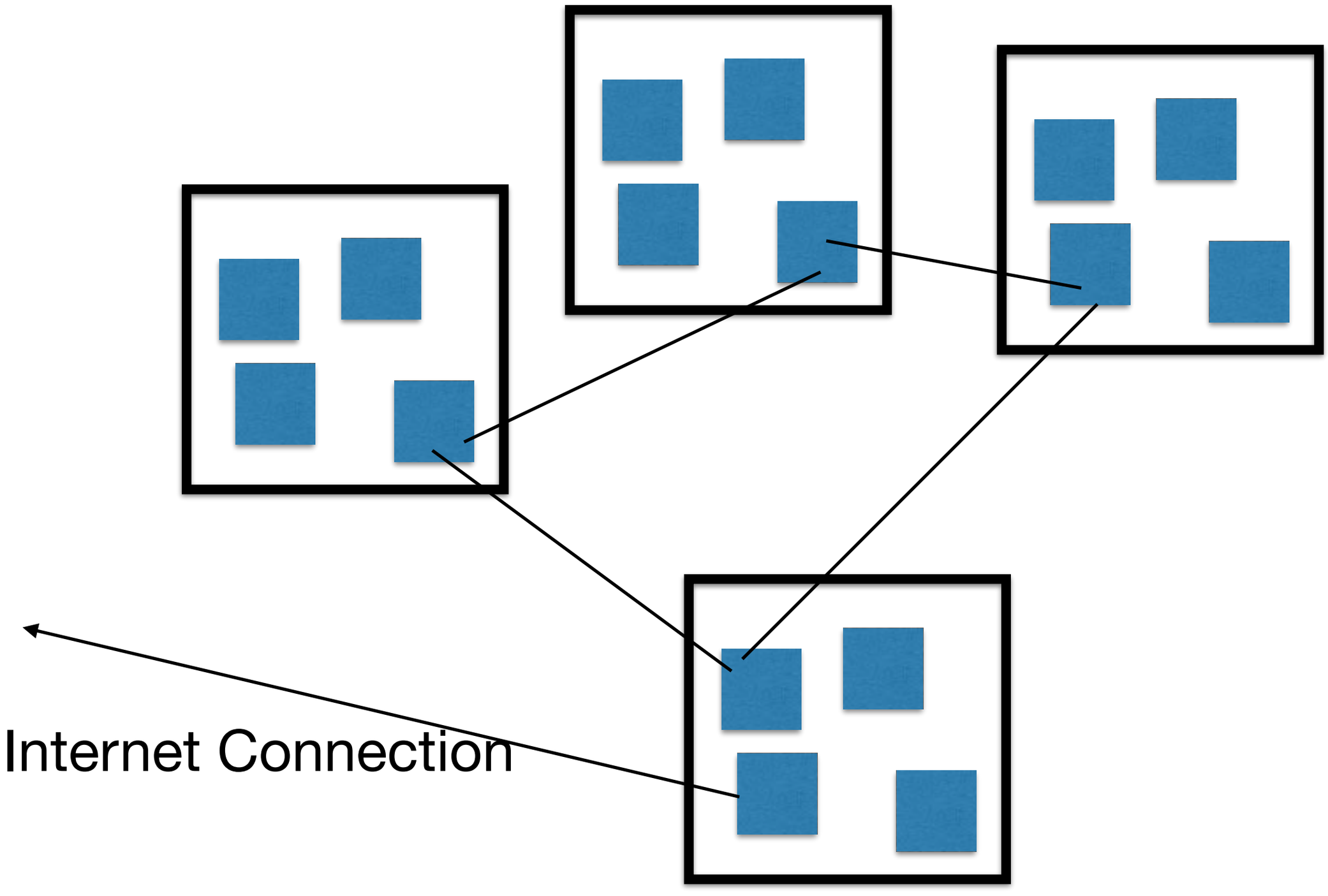
Database Replication

- If you want to run a database in this sort of architecture, you will probably run the front end application with Anycasting and then commit the data to a redundant pair using some other technology (VRRP, or a proprietary distributed database)
- Sometimes, the best solution is still a mainframe with “five nines” availability (99.999% = 5 minutes per year down time) or “six nines” (30 seconds per year downtime).
- Running one instance on highly available hardware much, much easier than attempting to do redundancy while maintaining absolute data integrity (banking, airline booking); different if willing to accept inconsistent results during failover (search, shopping).

VPNs

- Quite a good technology for linked replication servers using Anycasting :-)
- “Virtual Private Network”
- Has come to be linked with remote access for “road warriors” (yes, really) and home working, but is more general than that.

Multi-Site Company



Private Networks (not Virtual)

- 1980s/1990s style
  - You buy leased lines (kilostream, megastream, E1, DS1,T3: lots of names, lots of speeds, all lots of money) and roll your own network over the top
  - Closed user group: as secure as the telco’s network

Dial In

- Remote access 1990s style done with dedicated banks of modems (we had a second phone line at home for my wife to dial into work until about 2002)
- Remote mobile access 1990s/2000s style involved dedicated APNs and the mobile operator delivering data over a leased line to the company
  - If you have to ask the price, you can’t afford it

Internet Changes It

- Internet connectivity much cheaper than international leased lines (although probably not as reliable)
- Reasonable to assume all staff have broadband Internet, but cannot use it to “dial in”: it’s Internet only
    - We tried to put this functionality into the original Project Ascot, but lost
- Mobile internet massively, massively cheaper than previous solutions
- Security issues substantial.
- So we can run it all over the Internet, yes?

Security Issues

- Running “line of business” applications over the internet extremely risky
  - Not engineered for it, either as protocols or as implementations
- No encryption
- Servers unlikely to survive exposure to outside world for long
  - Address filtering doesn’t work for home users or mobile users

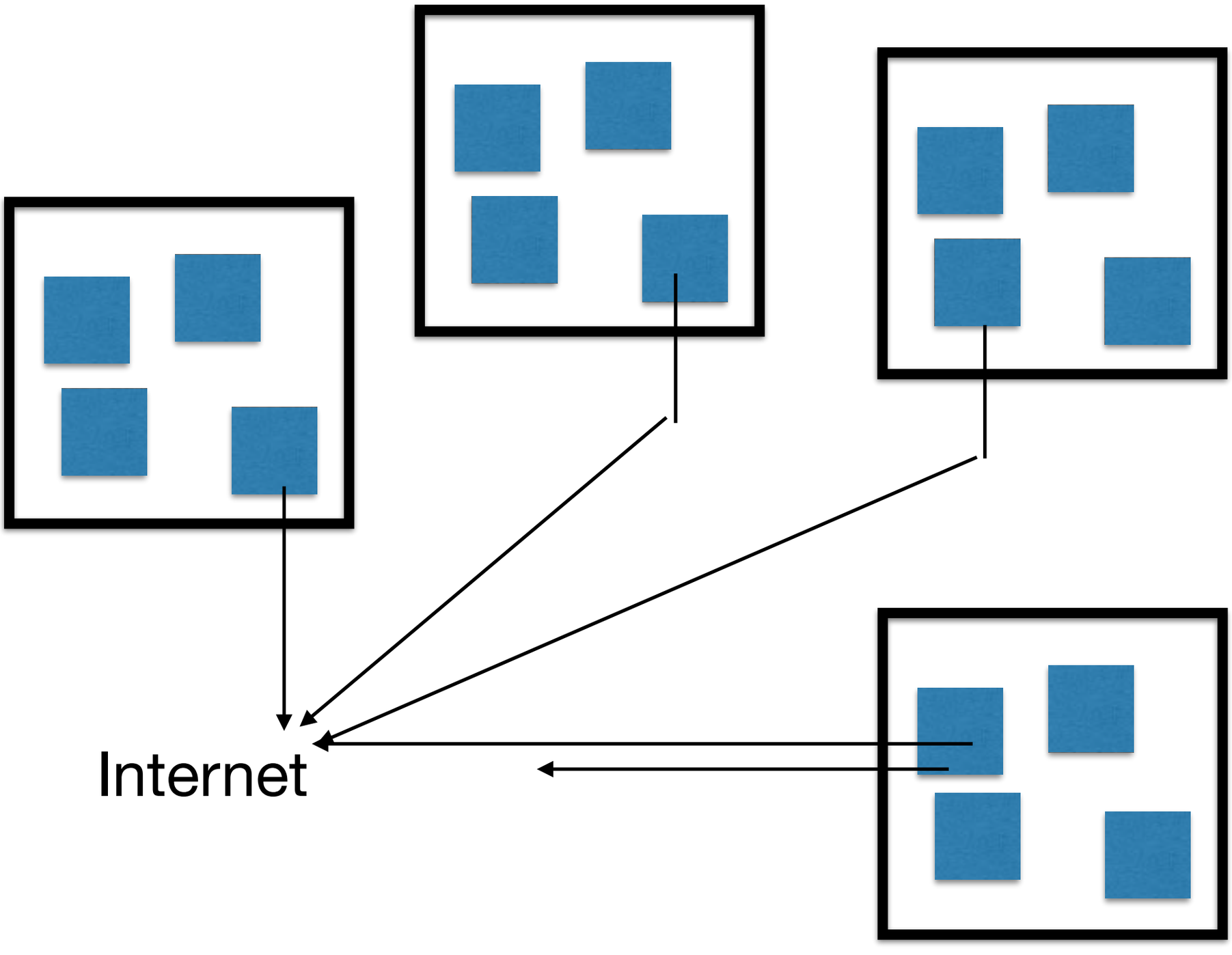
Networking/OS Issues

- Probably using RFC1918 IP numbers internally, including on LoB services
- May be forced to run old, obsolete, unpatched (or unpatchable) servers for old applications
- May not support any sort of secure authentication

Hence, VPN

- **Virtual** Private Network
- Tunnel traffic between sites by embedding the packets into IP packets

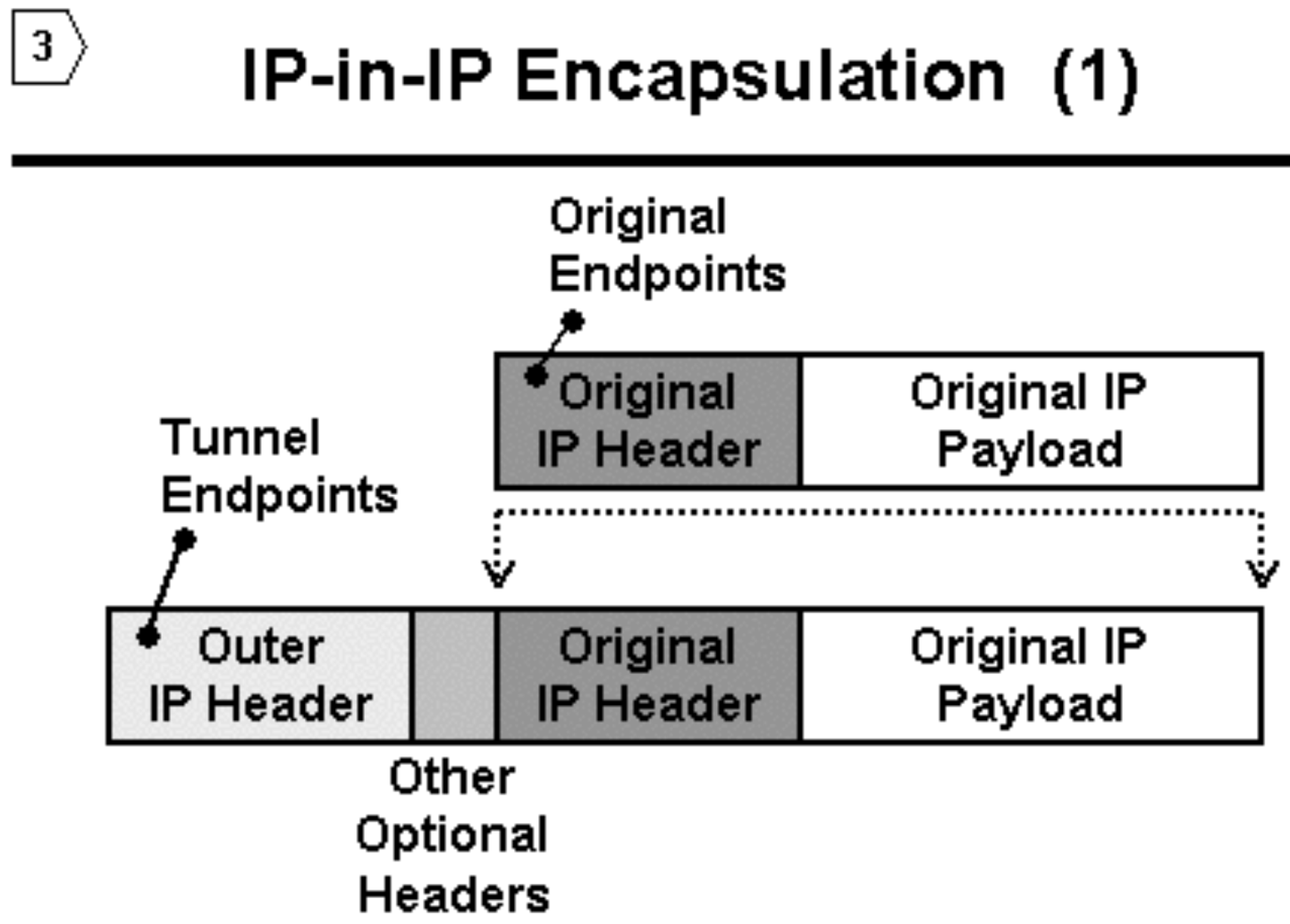
Multi-Site Company



IP-in-IP

3

IP-in-IP Encapsulation (1)



IP in IP

Site Left

Site Right

Header: A->D

Payload for App

Header: B->C

Header: A->D

Payload for App

Header: A->D

Payload for App

Alternatives to IPIP

- GRE (Cisco)
- Tunnelling PPP over TCP (Messy and unnecessary)
- L2TP (Microsoft, Apple)
- All work roughly similarly: a packet and its header become the payload in another packet
  - Can also carry raw ethernet over similar techniques

Encapsulation

- Edge router of local network has routes to other networks in the VPN, which it seems as being directly attached, one hop away
- Packets from site left to site right are used as the payload of a packet going from router B to router C

Benefits

- Sites are linked without needing to expose systems to Internet (firewall set to “pass traffic to other end of tunnel, drop everything else”)
- Can have single central point of Internet egress and ingress
  - In practice, web traffic often happens locally
- RFC1918 is OK so long as the sites don’t overlap
- But…

Security?

- In this case, non-existent
- Anyone who knows of the existence of the tunnel can inject packets into it undetectably
- Anyone who can see packets can forge connections undetectably
- If the tunnel medium is TCP it’s slightly harder, but still easy enough
  - TCP connection attacks: insert packet at current sequence number, and modify all subsequent packets to include the offset.

What can we do about it?

- Depends on our attacker model
- We might just need to authenticate packets, so that insertion is impossible
- We might want to encrypt packets for confidentiality
- We probably want to do both

Technologies

- IPsec (“IP Security”) – last week
  - either directly in tunnel mode (Cisco) or to protect something like L2TP (Apple)
- SSL-VPN: various uses of SSL-type encryption to handle a stream of packets
- Proprietary boxes (after all, so long as it’s reversible and the headers are IP, the payloads can be processed how you like)
  - Crypto.AG and others sell wire-speed AES boxes

Home / Mobile Use

- Special case where one of the “sites” is a home or mobile user
- Normally we don’t want to link the whole network, just the single machine
  - And arguably, we want to isolate/minimise contact with other machines on the home network

VPN addresses

- It will usually be difficult to use a VPN to connect to a network using a particular address if you are coming from a machine with the same address.
- Concrete, nasty examples: 192.168.0.0/24, 192.168.1.0/24, 10.0.0.0/8
- For home/SME use, pick either 10.X.Y.0/24 for random X and Y both 1..254, or better (I wish I'd done this) 172.X.Y.0/24, where X 16..31 and Y 1..254.

Personal VPN

- So single-hop route from local machine to central site
- With **no** IP forwarding on local machine, so only traffic to/from the local machine goes to/from central site
- But…

Split Tunnelling

- Big debate
- Traffic from employee’s laptop can go to company network
- But can company laptop access (for example) printers on home network while connection is up?
- And what about google and so on. Where does the traffic go: via centre, or “direct”?

Split Tunnelling

- Usually, VPNs are configured to block all traffic to and from the laptop other than from the VPN server, and default route points down VPN connection.
- This is the default on iPhones, for example
- I think arguments are nuanced, but I am in a minority