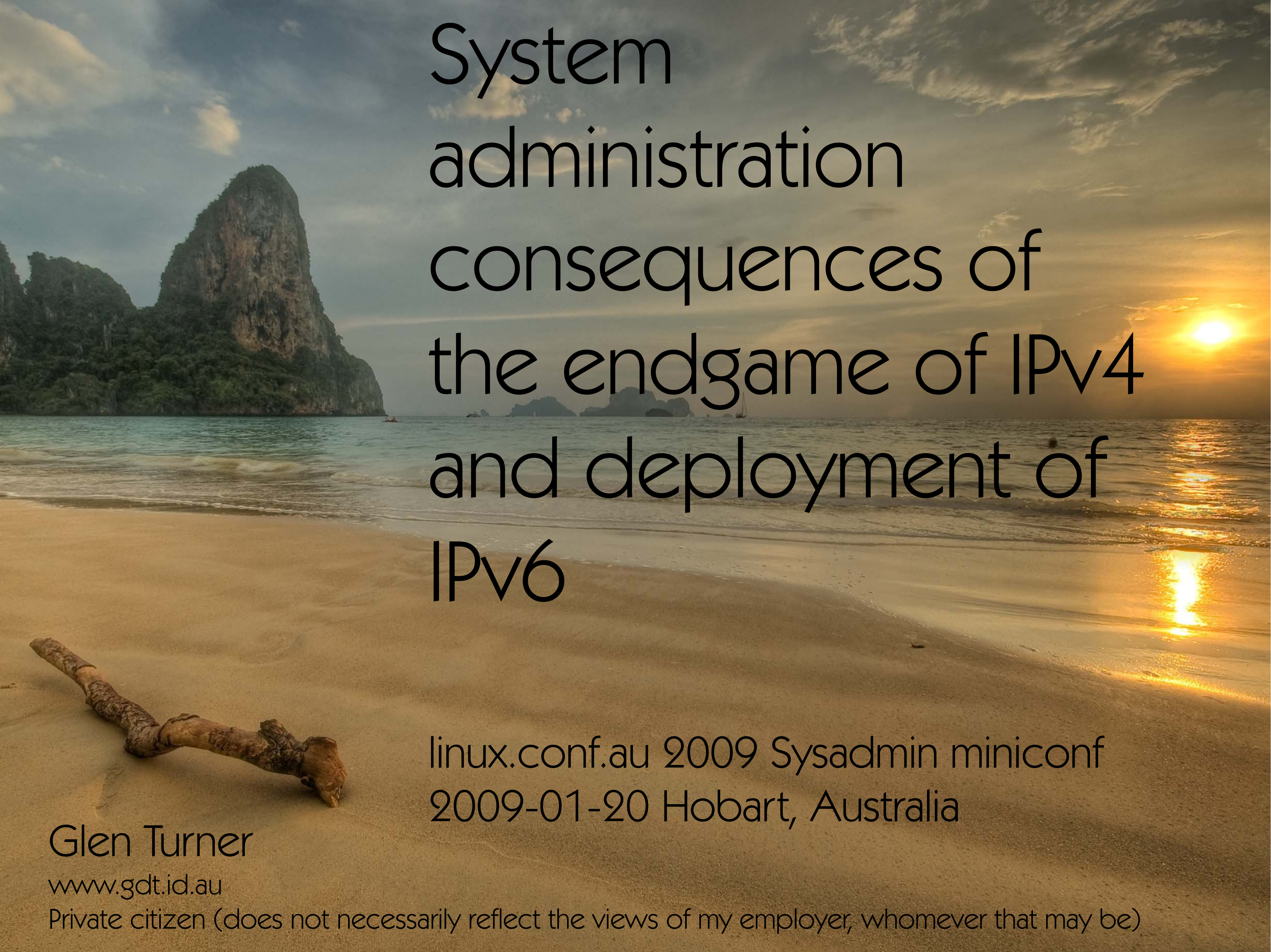
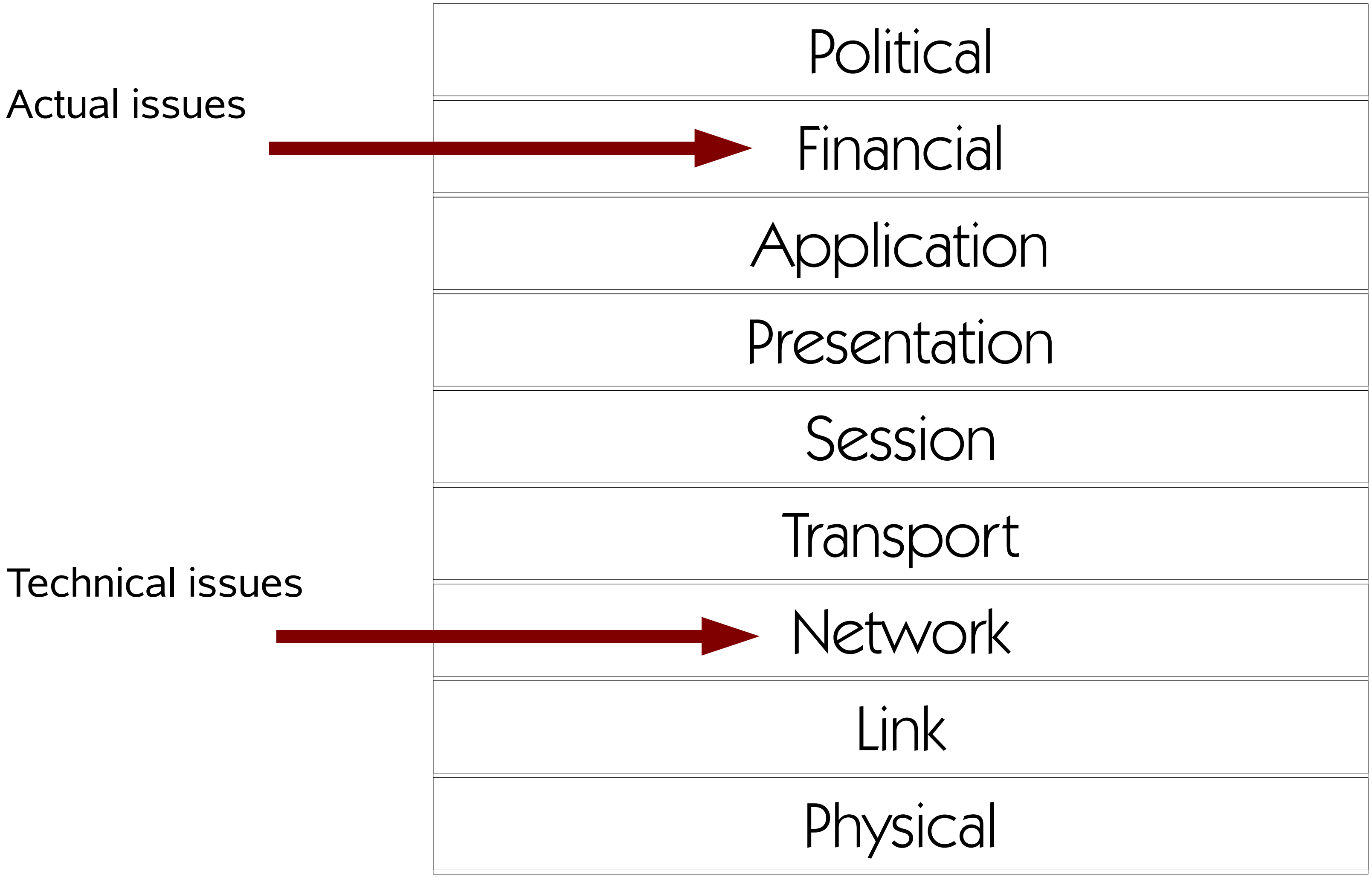


System administration consequences of the endgame of IPv4 and deployment of IPv6

linux.conf.au 2009 Sysadmin miniconf
2009-01-20 Hobart, Australia

Glen Turner
www.gdt.id.au
Private citizen (does not necessarily reflect the views of my employer, whomever that may be)

ISO model, amended for reality



Actual issues

Technical issues

Political

Financial

Application

Presentation

Session

Transport

Network

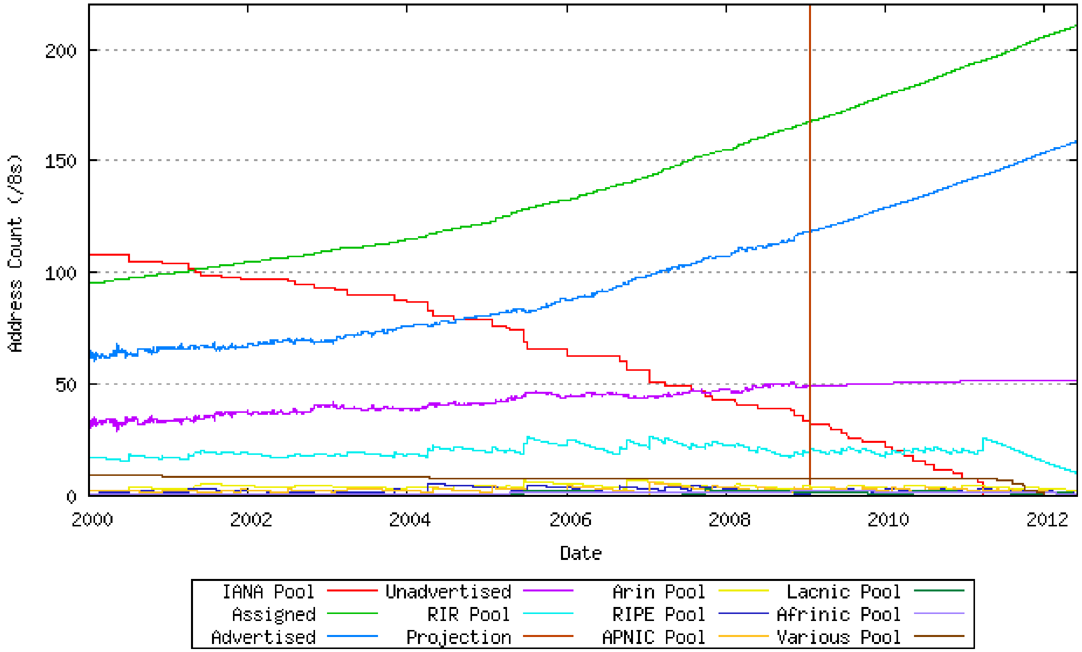
Link

Physical

Part 1
Address exhaustion

IPv4 addresses

- Are a limited resource
- Are about to become scarce as the pool of unallocated IPv4 addresses empties

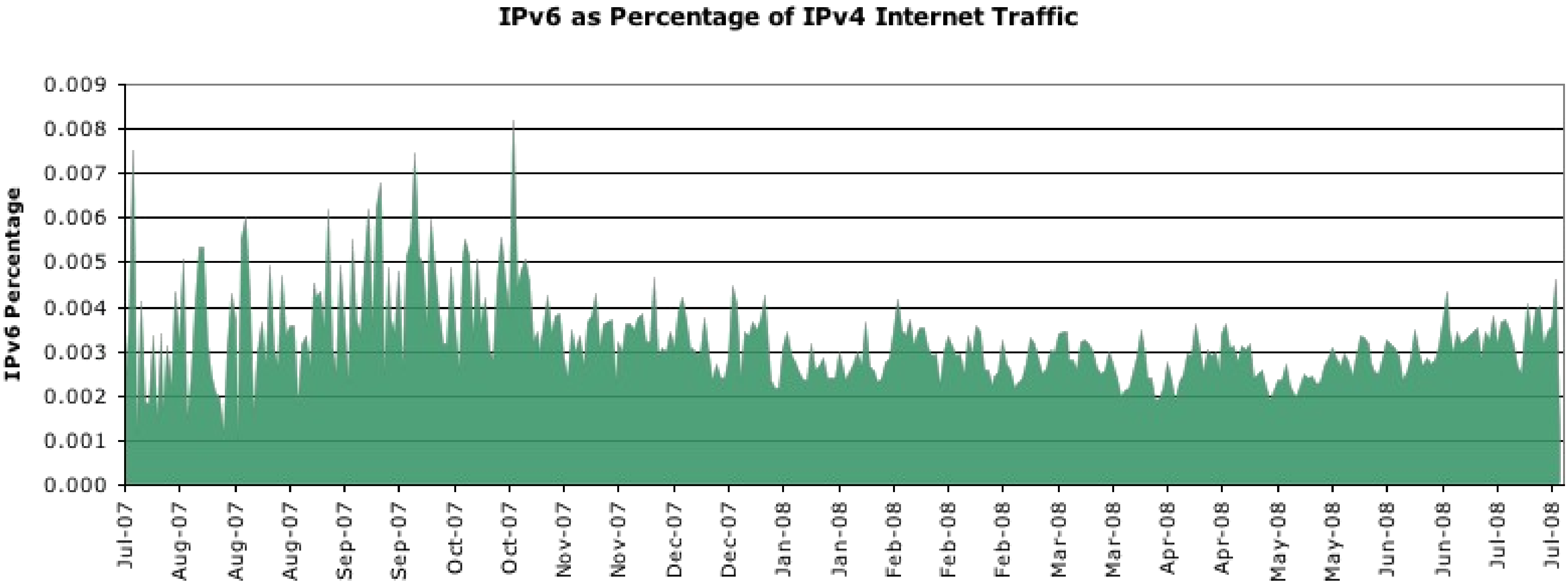


Ref: www.potaroo.net/tools/ipv4

Plan A, circa 1994

- IPv6 to replace IPv4
- Well prior to IPv4 address exhaustion
- A migration from IPv4 to IPv6

Plan A, circa 2008



- “It is now clear the original optimistic IPv6 deployment plans have failed” –Craig Labovitz

asert.arbornetworks.com/2008/08/the-end-is-near-but-is-ipv6

Plan B

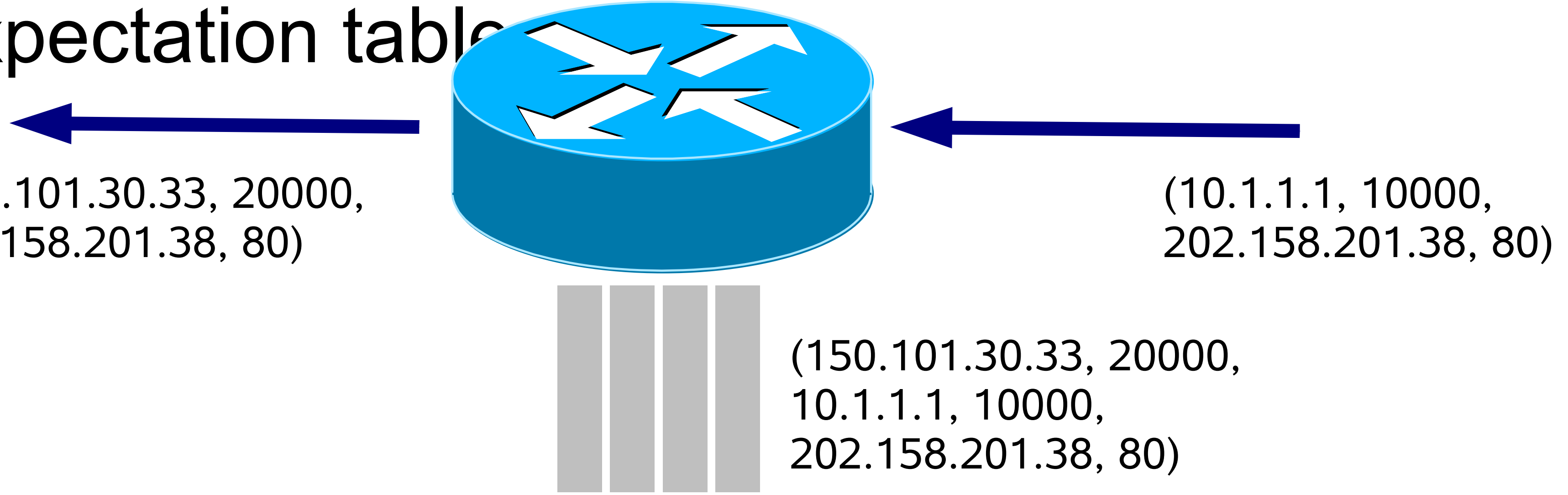
- “Carrier-class NAT”
- The ISP runs real addresses to transit and peering links, NATed addresses to retail customers
- NAT in the ISP's distribution layer of routers
- Even NAT addresses within the ISP are not unique (lots of ISPs have more than 10.0.0.0/8 of customers)

Plan B sometimes breaks IPv4

- No more end-to-end address visibility
- In reality this has been broken for quite some time
  - Can you run a globally-visible web server from your laptop?
- But now breaks the end-to-end nature of the Internet to the customer site
  - Can you run a globally-visible web server from your ADSL router?

How does NAT work?

- Inspect outgoing traffic
  - Collect (*src_addr*, *src_port*, *dst_addr*, *dst_port*)
- Re-write *src_addr* to my exterior interface, find an unused source port on my exterior interface and re-write *src_port* to that
- Record these addresses and ports in the expectation table



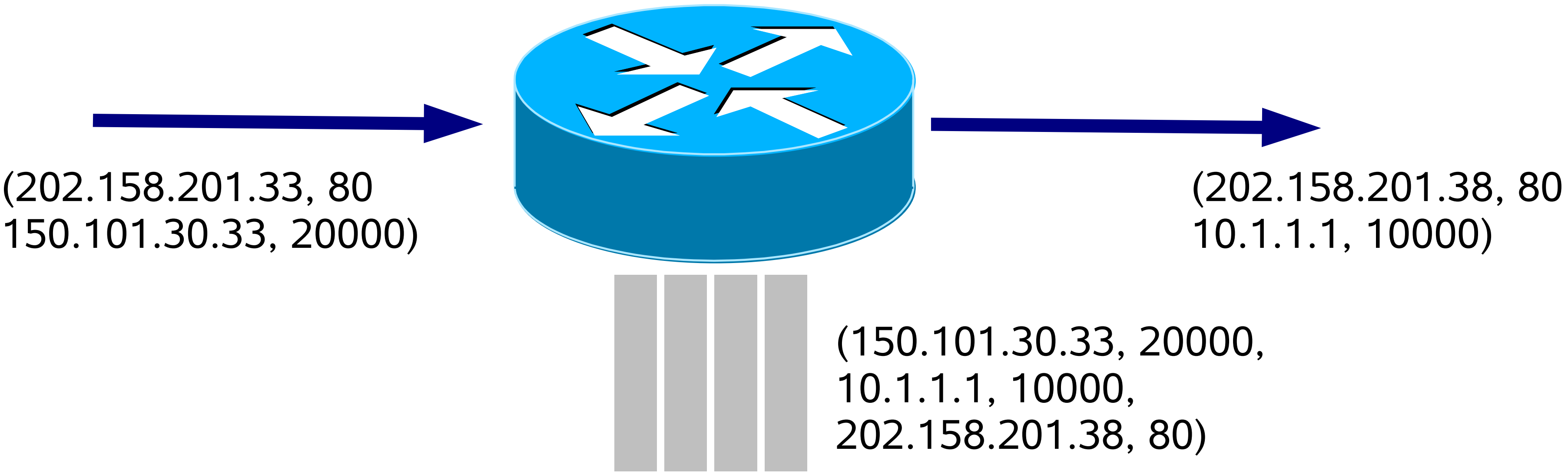
(150.101.30.33, 20000,
202.158.201.38, 80)

(10.1.1.1, 10000,
202.158.201.38, 80)

(150.101.30.33, 20000,
10.1.1.1, 10000,
202.158.201.38, 80)

How does NAT work?

- Inspect incoming traffic
- Is the incoming (*src_addr*, *src_port*, *dst_addr*, *dst_port*) in the expectation table?
- Re-write the *dst_addr* and *dst_port* to the original values in the table



(202.158.201.33, 80
150.101.30.33, 20000)

(202.158.201.38, 80
10.1.1.1, 10000)

(150.101.30.33, 20000,
10.1.1.1, 10000,
202.158.201.38, 80)

Wrinkles with NAT

- Some protocols embed IPv4 addresses
  - These need to be rewritten too
  - May be complex and thus dangerous to do in the forwarding plane
    - eg: SNMP uses ASN.1 encoding
- Some protocols embed forthcoming connection information
    - FTP, Cisco Skinny, a lot of multimedia
- These wrinkles are handled by “NAT modules”
  - inspect the traffic, add entries to the expectation table

Problems with NAT

- Complex
  - Forwarding plane moves from ASIC to CPU
- Jitter and complexity attacks
  - Some packets need a lot more work than others
- Exploits of code with errors
  - Complex code, so errors certain
- Huge amounts of *state*
  - Abundant opportunity for resource exhaustion
- Timeouts
  - Some traffic simply isn't suitable: low-power devices, sensors, episodic multimedia

Overlay networks

- NAT leads to all “finding each other” applications deploying overlay networks
- You can view STUN and the like as attempts to generalise these
  - But generalised solutions are remarkably like onion routers, and thus attract misuse
- Overlay networks do not follow the topology
  - But latency is *the* performance issue
  - Gamers will love IPv6

Consumers and carrier-class NAT

- The consumer is forced to take a hosting service, since they can no longer host services themselves
  - Innovation. You can't buy services that haven't been invented yet
  - Pricing. No competition
- Again, this has been happening for some time
  - Part of the reason for the popularity of Picassa, Flickr, YouTube rather than hosting the content on your own machine

Return of the Walled Garden?

- The ISP can control which hosting services are used
- And can force the use of their own expensive services
  - Eg: carrier does not NAT H.323 and SIP, but places H.323 gatekeepers and SIP proxies on its network for its services. Regulators are told that this is necessary to preserve the reliability of telephony.

Results of scarcity

- A market for IPv4 addresses
  - Address holders are sellers
  - ISPs and colo providers are purchasers
- Who owns IPv4 address allocations?
  - Poor record-keeping of large historical allocations
  - ISP contracts lack clarity on this point
  - Lots of registry entries are unsecured or have outdated contacts
- Ample scope for fraud

Results of scarcity on ISPs

- Ownership versus control
  - Does it matter if you own an address if a backbone ISPs insist otherwise?
- Routing and forwarding tables size explodes
  - As address allocations are sliced and diced to use every last IPv4 address

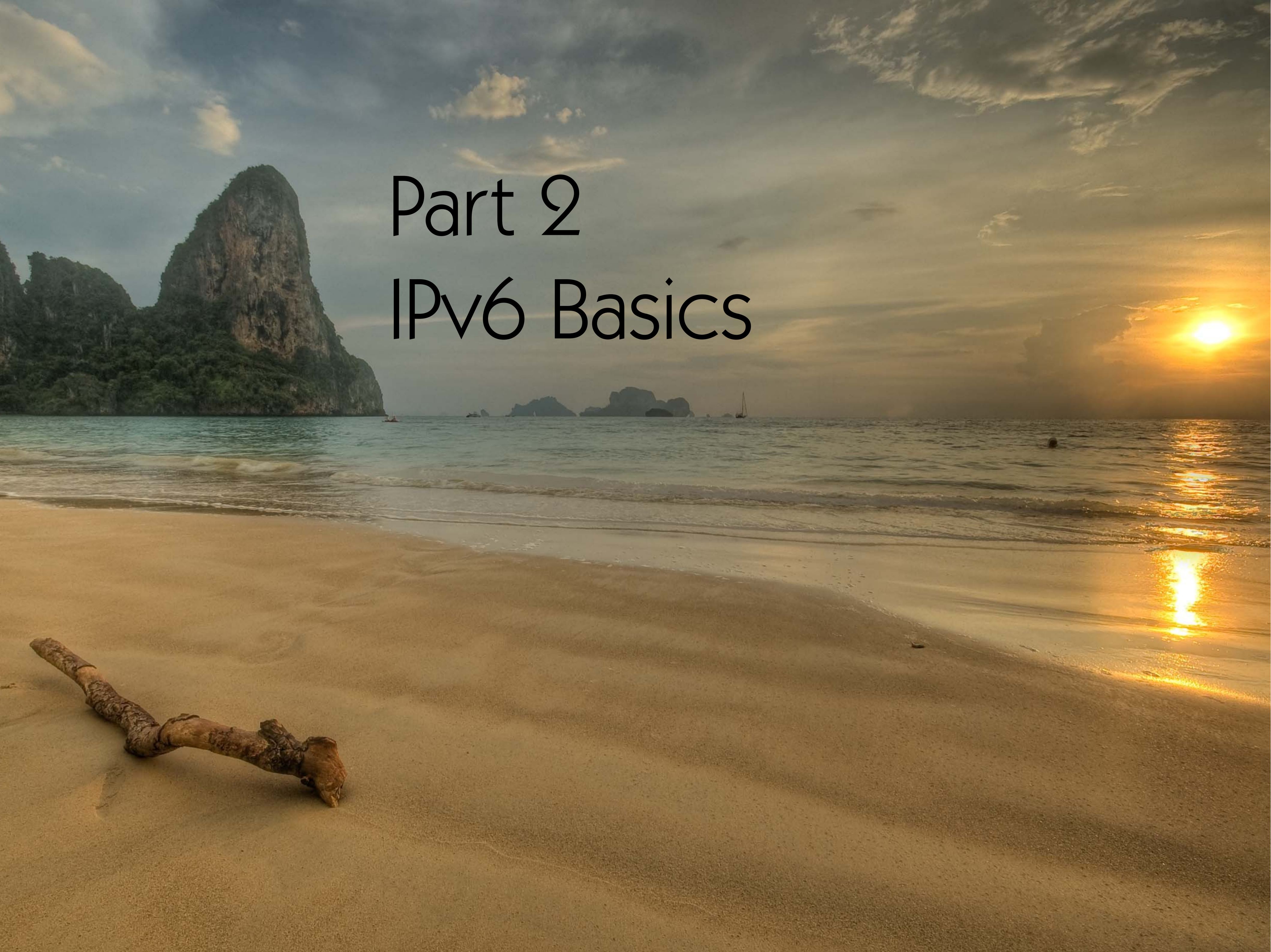
Security of retained IPv4 addresses

- Real IPv4 addresses are precious. They are the resource necessary if two NATed clients are to find each other
  - For videoconferencing
  - For peer-to-peer
  - For evil purposes
- Real IPv4 hosts become the worthwhile target

Summary

- The Internet is going to change for IPv4 from 2010. IPv4 will continue, but not as we know it
- The underutilised, barely deployed IPv6 is the only way to retain the *status quo*
- That is, the “killer application” of IPv6 is Internet access; technically, the end-to-end visibility of Internet addresses
- “No one is going to make any money from IPv6” –Simon Hackett, Internode

Part 2
IPv6 Basics



Design goals

- A bug fix release
  - Larger addresses
  - Automated configuration
  - Default route failover
  - Security
  - QoS
- Security and QoS were back-ported to IPv4 and are substantial topics by themselves
  - IPsec
  - DiffServ

IPv6 builds on IPv4

- Apart from the bug fixes, everything you know about IPv4 works with IPv6
- But IPv6 is a different protocol
  - Its own address family
    - inet6
  - Its own forwarding tables
    - ip -family inet6 route list
  - Its own routing
    - OSPFv3, extensions to BGP

Larger addresses

- IPv6 addresses are 128 bits
- Like IPv4 these addresses consist of three parts
  - Network
  - Subnet
  - Host
- ISPs are allocated a /40, they give /48s to customers

New address syntax

- 2001:388:a001:1:217:f2ff:feca:792e/64

– 2001:388:a001::/48	Site allocation
– …:1:217::/64	Subnet
– …:f2ff:feca:792e/128	Host

- Leading zeros are omitted
- :: is used to represent a run of :0:
- You can give a IPv4 dotted-quad address in the last 32 bits

Special addresses

::/128	Unassigned
::1/128	Loopback
fe80::/10	Link scope
ff00::/8	Multicast
::ffff:1.2.3.4	IPv4 in IPv6
2001:db8::/32	Documentation
::/0	Default route

Larger addresses allow autoconf

- The EUI-64 IPv6 address format uses the lowest 64 bits to hold the interface's MAC address
- Leaving the middle 16 bits for subnetting a site
  - Which the site can in turn split up for area aggregation, as with IPv4
- So the host part of the address is known from the MAC address
- How do we find the network and subnet parts of the address?

Router advertisement

- Router advertisments are send by routers periodically and upon request
- They contain
  - The method for address determination
  - The address prefix and length
  - The default router address
- Multiple announcements are fine
  - They provide redundant default routes

Router advertisement

- Linux uses radvd
- Configured from */etc/radvd.conf*

```
interface eth0 {
  AdvSendAdvert on;
  MinRtrAdvInterval 3;
  MaxRtrAdvInterval 10;
  prefix 2001:388:a001:1::/64 {
    AdvOnLink on;
    AdvAutonomous on;
    AdvRouterAddr on;
  };
};
```

Multiple addresses

- IPv6 interfaces are used to holding down multiple addresses
- This was originally intended as a migration measure for moving from one ISP's addresses to another, but it is useful for more than that

```
$ ip addr show
1: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state
UP qlen 1000
    link/ether 00:17:f2:ca:79:2e brd ff:ff:ff:ff:ff:ff
    inet 169.222.10.27/22 brd 169.222.11.255 scope global eth0
    inet6 2001:388:a001:1:217:f2ff:feca:792e/64 scope global dynamic
       valid_lft 2579159sec preferred_lft 591959sec
    inet6 2001:388:d000:d00:217:f2ff:feca:792e/64 scope global dynamic
       valid_lft 2577683sec preferred_lft 590483sec
    inet6 fe80::217:f2ff:feca:792e/64 scope link
       valid_lft forever preferred_lft forever
```

Remove poor ideas

- IPv4 packet fragmentation
  - Could be totally removed if links simply sent back the size of packets they support
    - Routers don't need to fragment
    - Hosts don't need to guess, thus using a smaller packet size than needed
  - Blocking ICMP6 packets is a really poor idea
- Identifier
  - Does do anything of value
- Small packets
  - 64KB is looking too small for fast links

Remove poor ideas

- Simplify option handling
- Single default route
  - The host listens to Router Advertisements and can select the best of them, but use the next best upon a failure
  - IPv4 hack is VRRP
- Assumption of a single IP address per interface
  - Interfaces hold multiple IPv6 addresses

DNS

- A new record type

```
www.example.org.  IN  AAAA  2001:388:1:4001:20a:e4ff:fe0d:be04
```

- The reverse uses PTR, to a different domain

```
4.0.e.b.d.0.e.f.f.f.4.e.a.0.2.0.1.0.0.4.1.0.0.0.8.8.3.0.1.0.0.2.ip6.arpa.
IN  PTR  www.example.org.
```

- Using autoconfigured addresses for well-known services is not a good idea — what if the MAC address changes?

DNS and services

- Let each machine have a autoconfigured address, use that for machine-related things
- Give each service on that machine its own address too

```
www.example.org.  IN  AAAA  2001:388:1:4001:20a::101
```

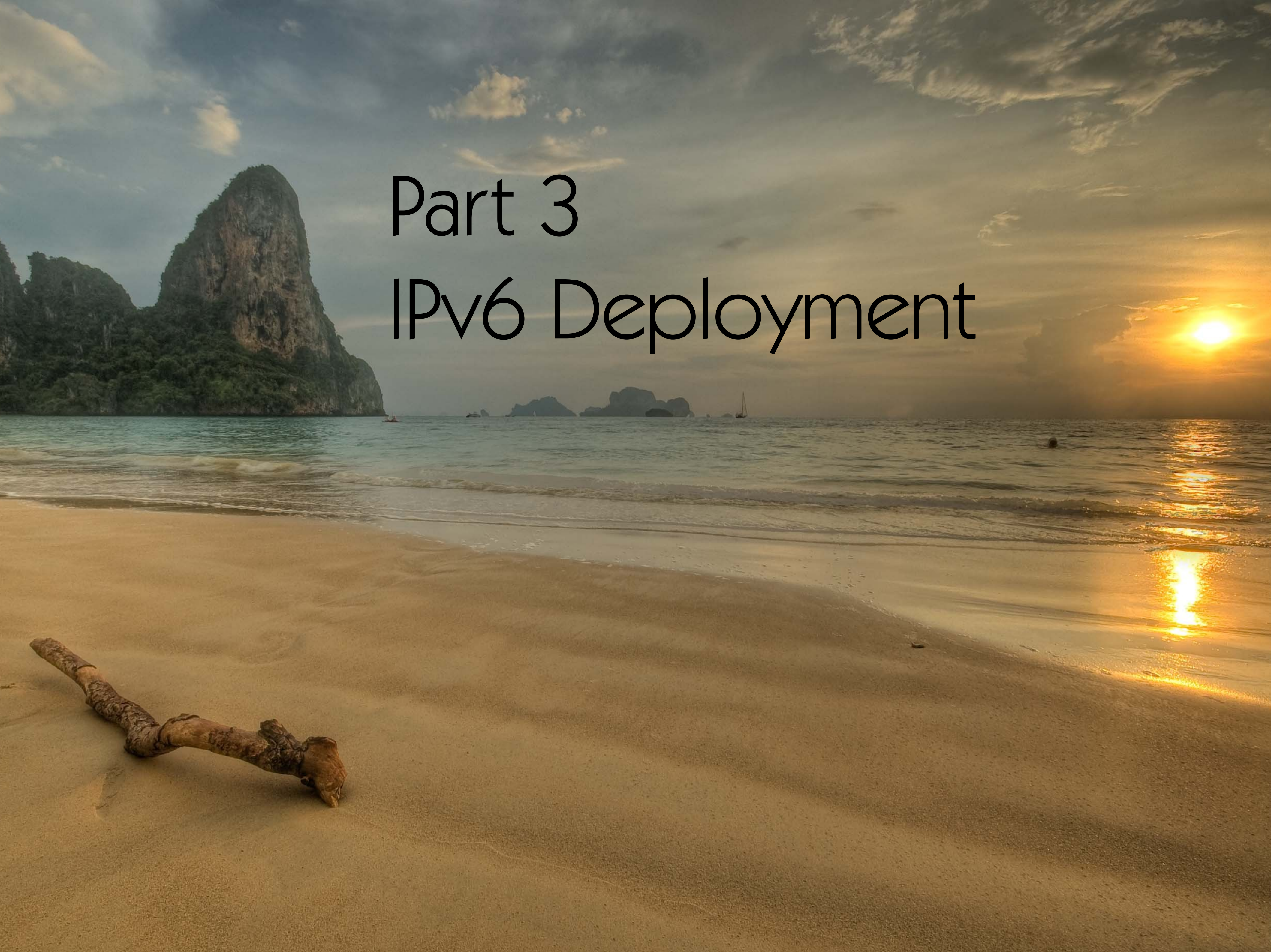
- This allows the MAC address to change without needing to update the DNS for public services
- A lot of applications software lacks a “bind” configuration option

DNS and autoconfiguration

- Static DNS entries and IPv6 autoconfiguration don't really mix
- Dynamic DNS is the answer
- Also saves a huge amount of typing

DNS

- DNS does the migration magic
- When a host has a global scope IPv6 address on a running interface it queries for a AAAA record prior to a A record
    - These queries can be transported over IPv4



Part 3
IPv6 Deployment

1. Paperwork

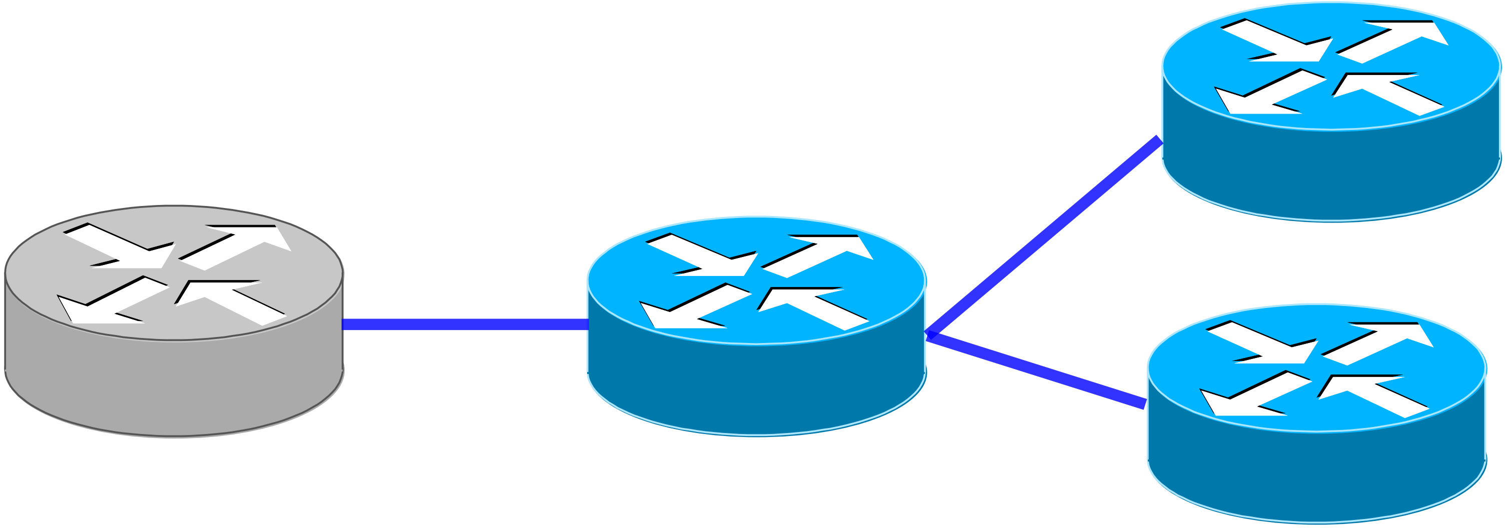
- Allocate IPv6 prefix
- Develop addressing plan
  - Lay IPv6 design over IPv4 design
  - There are 16 bits for subnetting, use the top 4 or so for site aggregation, leaving about 12 for subnets per site
  - Allocate a /64 per leaf subnet

2. Link to ISP

- Configure a IPv6 address and routing on existing ISP link
  - copying design from IPv4
- Static routing or BGP, depending upon site and ISP requirements
- Create or inject interior default route

3. Activate IPv6 on backbone

- This brings the first problem: the poor quality of IPv6 support on some firewalls and other middleboxes
- Don't use EUI-64, but be compatible

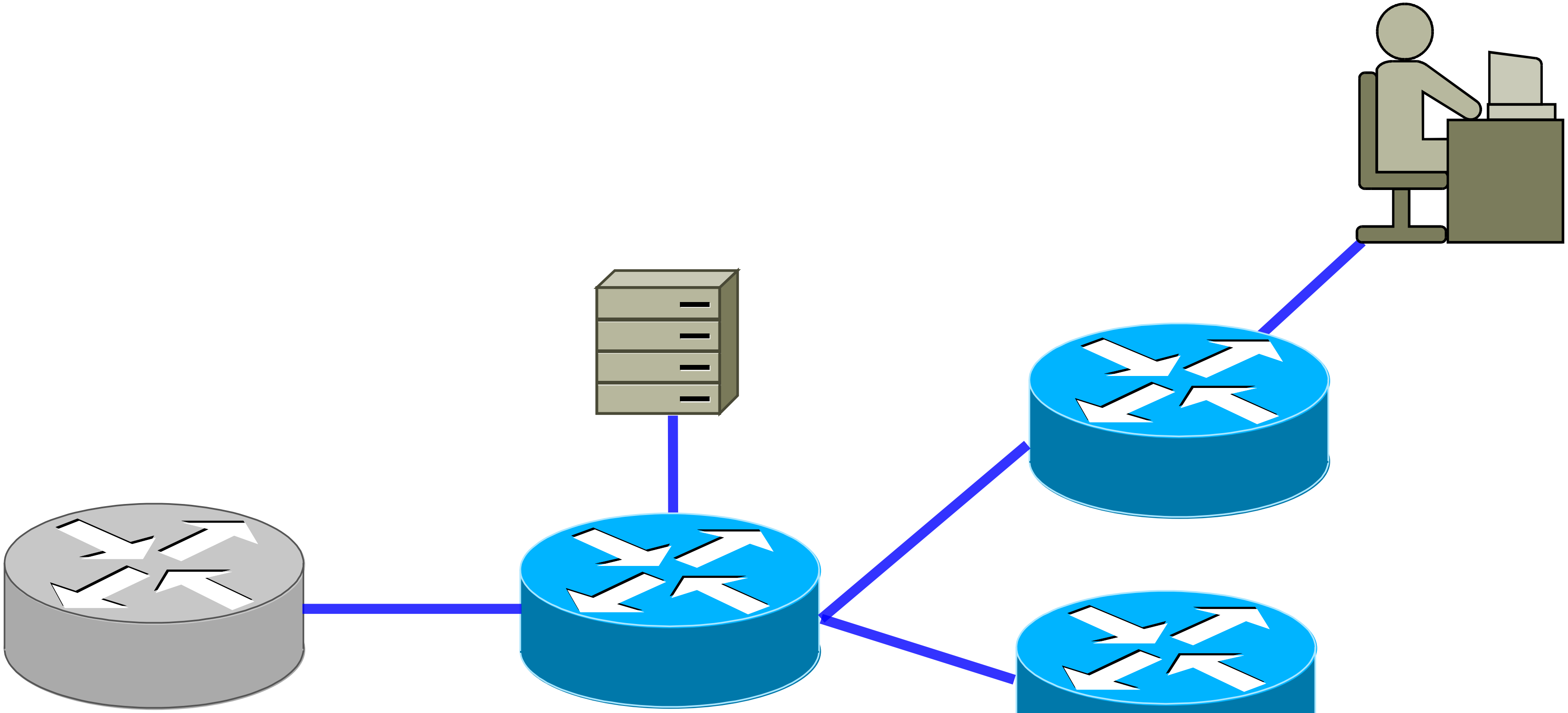


4. Establish networking servers

- Unless good reason otherwise use autoconfiguration (EUI-64 addressing) with stateless DHCP
- Stateless DHCP provides DNS and NTP server addresses
  - These will be IPv4 addresses, because of Windows Xp
- Use Dynamic DNS for the average host
- If you plan on IPv6-only devices then use an anycast IPv6 server on the well-known addresses

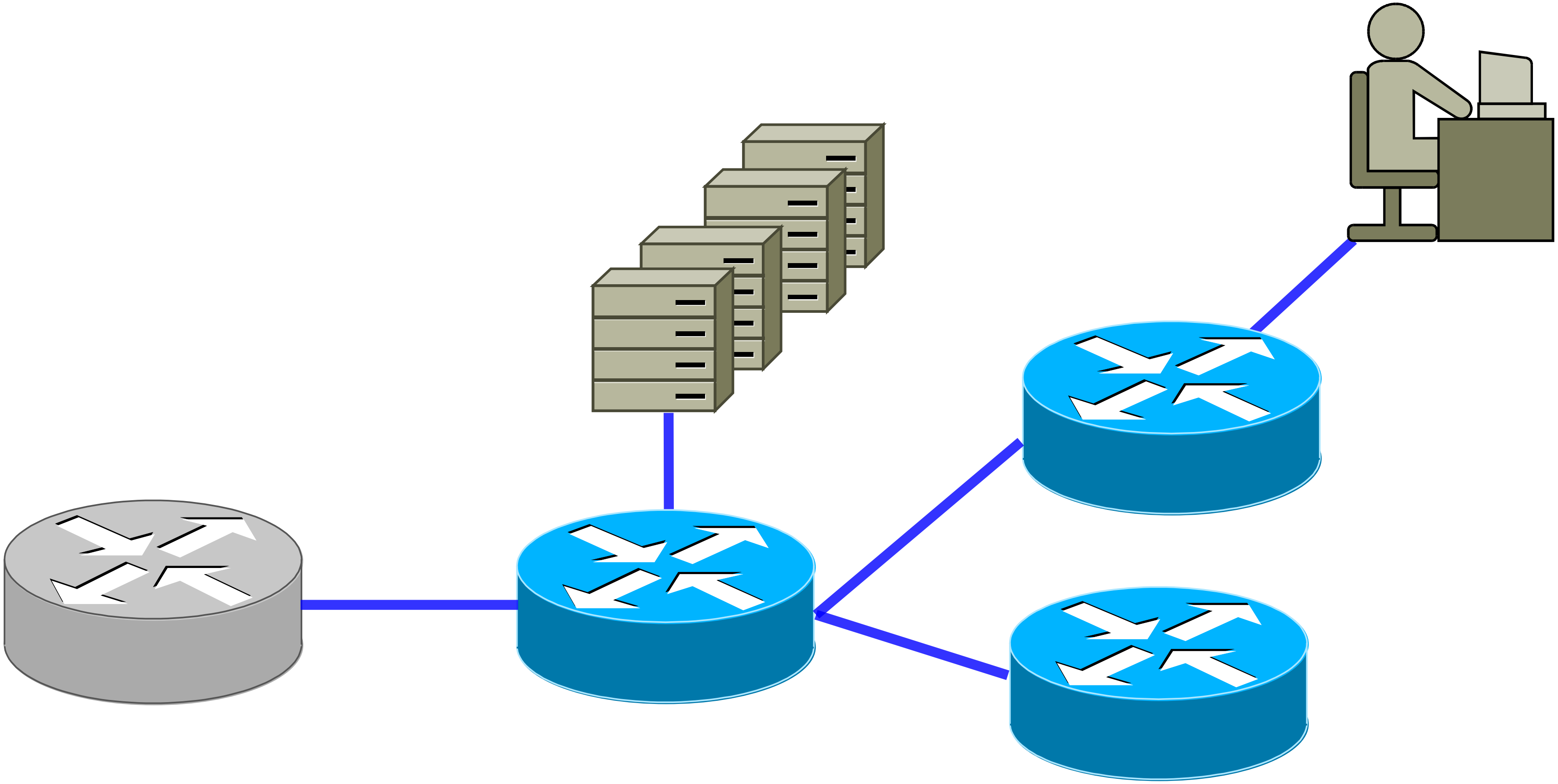
5. Find a ~~sucker~~ early adopter

- Computing hobbyists, ourselves
- System administrators



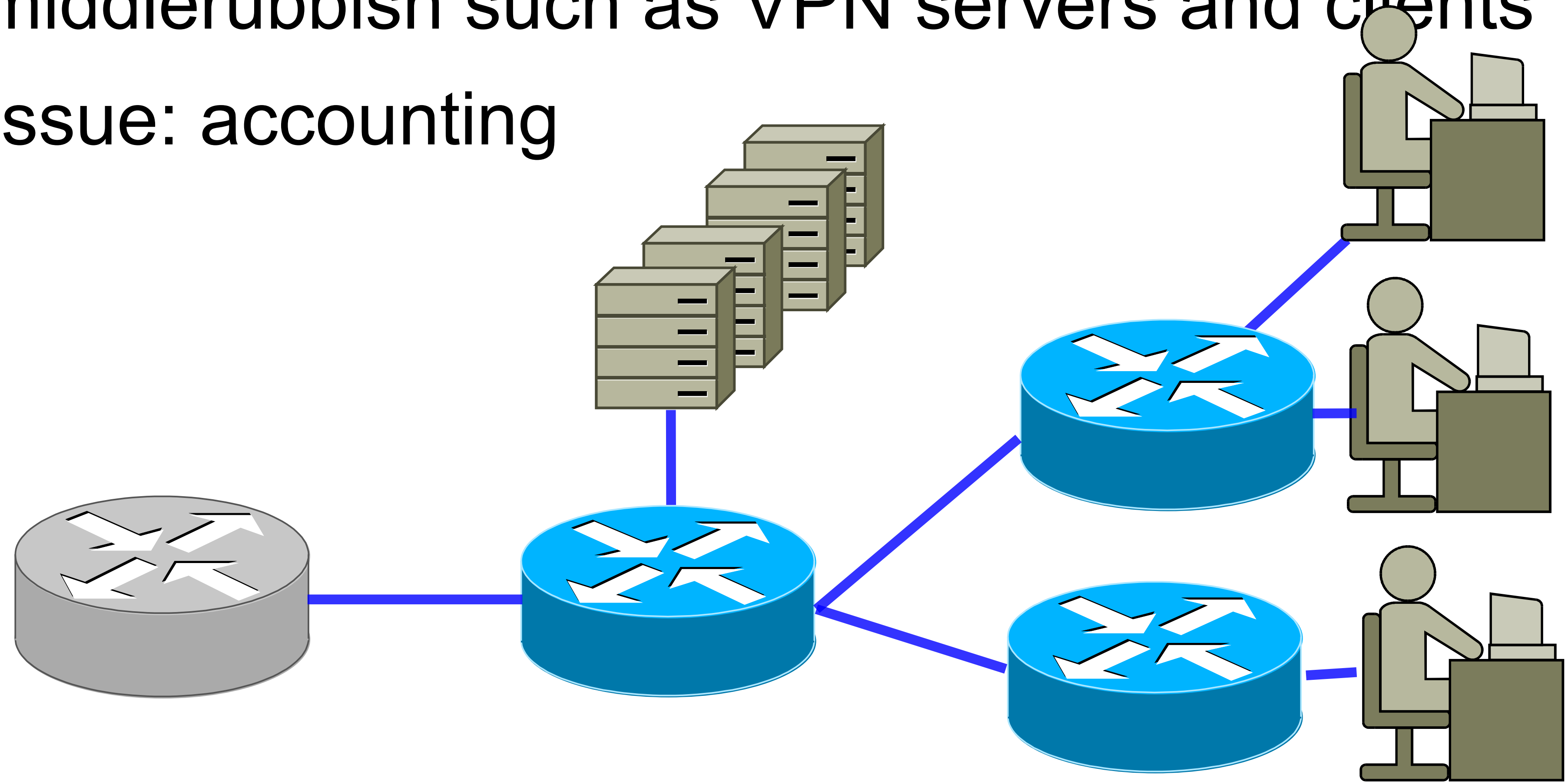
6. Transition public-facing services

- Web, e-mail, …
- Issue: Microsoft Exchange 2003
- Decision: EUI-64 or fixed address in the /64



7. Transition the masses

- Issue: people how travel to other sites which have IPv6 configured but no connectivity
- Issue: another round of fighting with middlerubbish such as VPN servers and clients
- Issue: accounting



8. Transition inward-facing services

- Problem: disconnect between network engineering and applications
  - Upgrade cycle of interior applications
  - Convincing applications programmers that the work is necessary, a hard task since they are being asked to learn something new
  - Package upgrades
    - Disruption
    - Money

9. Finish the job

- Delegation using IPv6 to DNS servers
  - Not available to edu.au
- Activate equivalent IPv6 features on switches as used on IPv4
  - To prevent address spoofing and so on
- Be careful not to deploy services which really only make sense for IPv4
  - VRRP
- Monitoring systems

Deployment ≠ technical issue

36%	Cost, time, business case
23%	Vendor support, back-office
18%	Knowledge, education
17%	User demand
17%	Upstream transit
14%	Dual-stack interoperability
4%	Multihoming
2%	Allocation policy
2%	Performance

Trickle down effect

- ISP can deploy IPv6 in a month
  - Only some equipment doesn't work
- Sysadmins can deploy IPv6 in a quarter
  - Only some software doesn't work
- Applications programmers may have one outage period per year
  - Almost no bespoke software works
  - Upgrading an application is very different to upgrading server software

Tunnel broker and opportunistic

- AARNet runs a tunnel broker: IPv6 connectivity for testing purposes
  - broker.aarnet.edu.au
- Use it in advance of native IPv6 connectivity to
  - Gain experience with IPv6
  - Test equipment and applications with Ipv6
- Plagued by mystery authentication issue
  - We are not going to solve this, we'll rip and replace the box with a more modern Hexago broker

Issue: Textual representation

- Addresses are longer :-)
- And in a different address family
  ipv4:1.2.3.4 ≠ ipv6:::ffff:1.2.3.4
- IPv6 is the only network address without a regular form

IPv4	1.2.3.4
AppleTalk	12345.67:890
IPX	12345678 00aabbccddeeff
DECnet	1.2345

- So you are looking at rewriting functions rather than adding a one-liner

Issue: Security

- Hosts
  - Not all firewall products understand IPv6, even when the host is running IPv6. You can guess the OS.
- Routers
  - It's a second protocol

    -
      ```
      ipv6 routing
      line vty 0 4
       ip access-group VTY-LIST
       ip access-group VTY-LIST6
      ```
- The real problem is support in corporate firewalls
  - And upgrade plans for those firewalls

Issue: Monitoring

- How a connection works:
  - Do I have a global address on default route interface?
  - Yes, look up DNS name using AAAA
    - Present, use that IPv6 address
    - Absent, try to look up the A record
  - No, try to look up the A record
  - Got a AAAA, try for IPv6 connection
    Got a A, try for IPv4 connection
- What happens if we have a black hole on IPv6?
  - IPv6 traffic dies, IPv4-based monitoring system says all well

Issue: Reality of corporate networks

- Inadequate
  - Configuration control
  - Monitoring
  - Change control
  - Lab scenarios
- Firewalls are the new voodoo
  - Configuration changes induce fear
  - IPv6 changes the sense of firewall rules: match against lower /64
    - ::1 to ::ff Network
    - ::ff00 to ::ffff Servers
    - ::1234:1234:1243:1234 Autoconfed MAC

Issue: Accounting systems

- Traffic accounting systems are not used to dealing with addresses as dynamic as IPv6 addresses

Issue: Enterprise switches

- Enterprise switches do a lot more than just bridge
    - Snoop DHCP requests and enforce allocations
- Look for parallel features under IPv6

Issue: Windows Xp

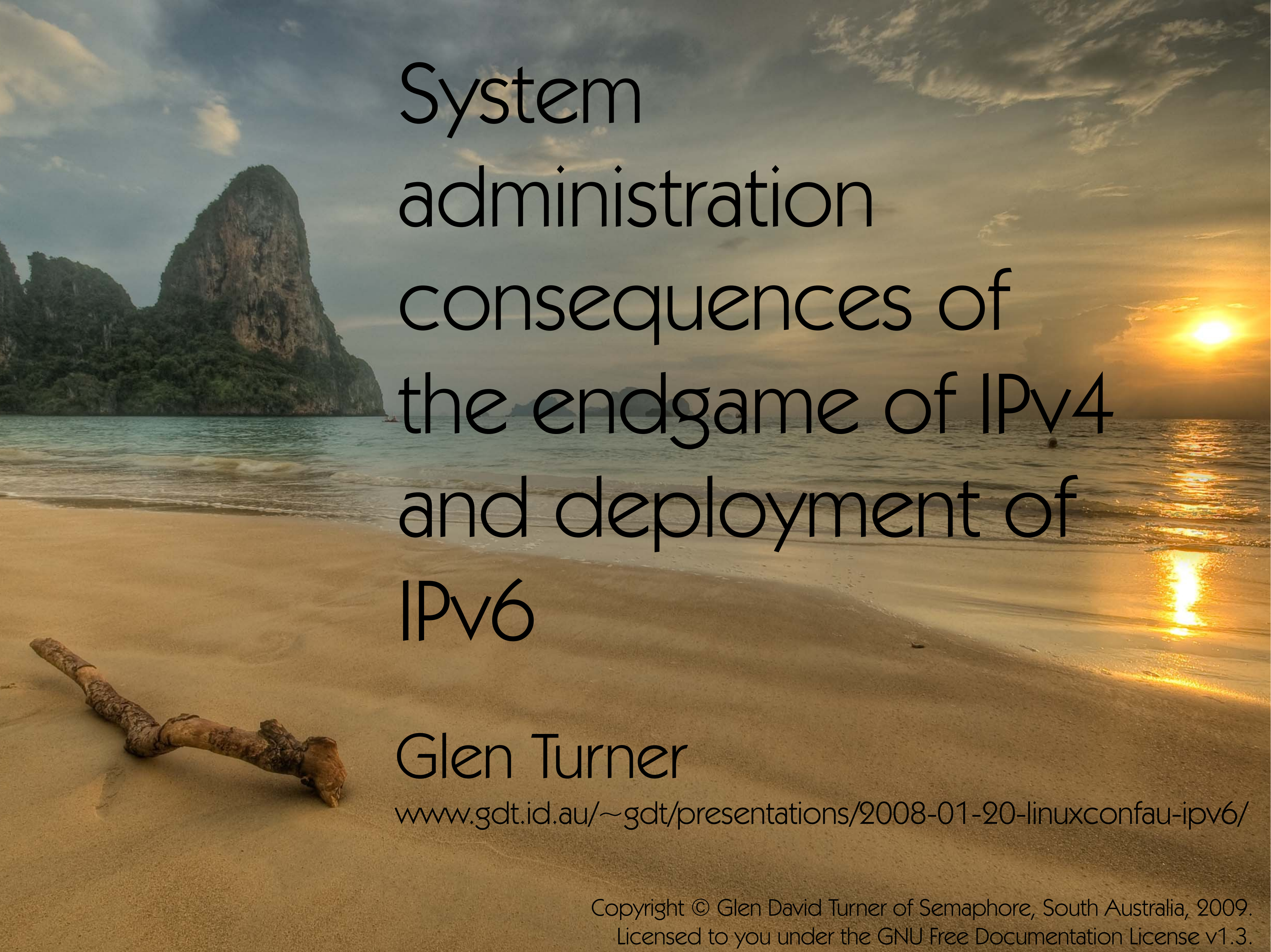
- Needs an activation command
  - ipv6 install
- Can only use a IPv4 DNS forwarder, so can't operate in a IPv6-only environment
- Linux, MacOS and Windows Vista all have IPv6 switched on

Issue: Training

- University computer science courses hardly never show students an IPv6 address
- TAFE ditto
- Vendor training (MSCE, RHCE) ditto

Summary

- IPv4, as we know it today, is ending
- NAT is the obvious way forward
- IPv6 offers a solution to the economic and technical problems with NAT, but has little deployment
- IPv6 can be deployed quickly by ISPs
- Both IPv4 NAT and a rapid IPv6 deployment leaves sysadmins holding the baby
  - Revenge for the BOFH



System administration consequences of the endgame of IPv4 and deployment of IPv6

Glen Turner

www.gdt.id.au/~gdt/presentations/2008-01-20-linuxconfau-ipv6/