IP-based communications in a dispersed environment

PITA Workshop
Suva, Fiji
23 - 27 September 2013

Christopher Liljenstolpe
cdl@asgaard.org

Objective

The intersection of multiple concurrent developments, including software-based routing, p2p SIP, NGN, NFV, and *Cloud* provide an opportunity to re-evaluate service delivery architectures.

We will discuss practical aspects of these developments and place them in the context of the Western Pacific environment.

Who are we?
Christopher

- Chief Architect - C&W and Telstra
- CTO IP Div, APAC - Alcatel
- Network Architect - USARP
- Advisor to multiple Cloud and SDN start-ups.
- If it's not "interesting" - it's not rewarding.

Who are we?
Michael

- VoIP applications developer @ Metaswitch
- Technical lead for Metaswitch's APAC expansion
- Director for Systems Engineering, APAC @ Metaswitch

Who are you?

Topics to be covered

- IP Networking
- SIP
- NGN
- NFV
- Cloud
- Colo/IXP
- Synthesis

Proposed workshop model

- Interactive sessions - ask questions as we go
  - If a topic becomes too involved we will put it in the queue.
- At the end of every section we can have a "practice assignment" that we review after.
  - During the practice, we will run side-discussions to empty the queue.
- "Office hours"

IP Networking

A Review

IP Networking Review

- Overview of IP networking
  - What about MPLS?
- IPv4 and IPv6
- IP addressing - LPM and summarization
- IGP vs. iBGP vs. eBGP
- eBGP peering issues

IP Networking Basics (some are even, mostly, true)

- It's packets.
  - MPLS allows for circuit simulation
  - Flows - usually more fine-grained than MPLS.
- Destination based forwarding
  - Policy is possible - may be expensive
- Per - hop forwarding

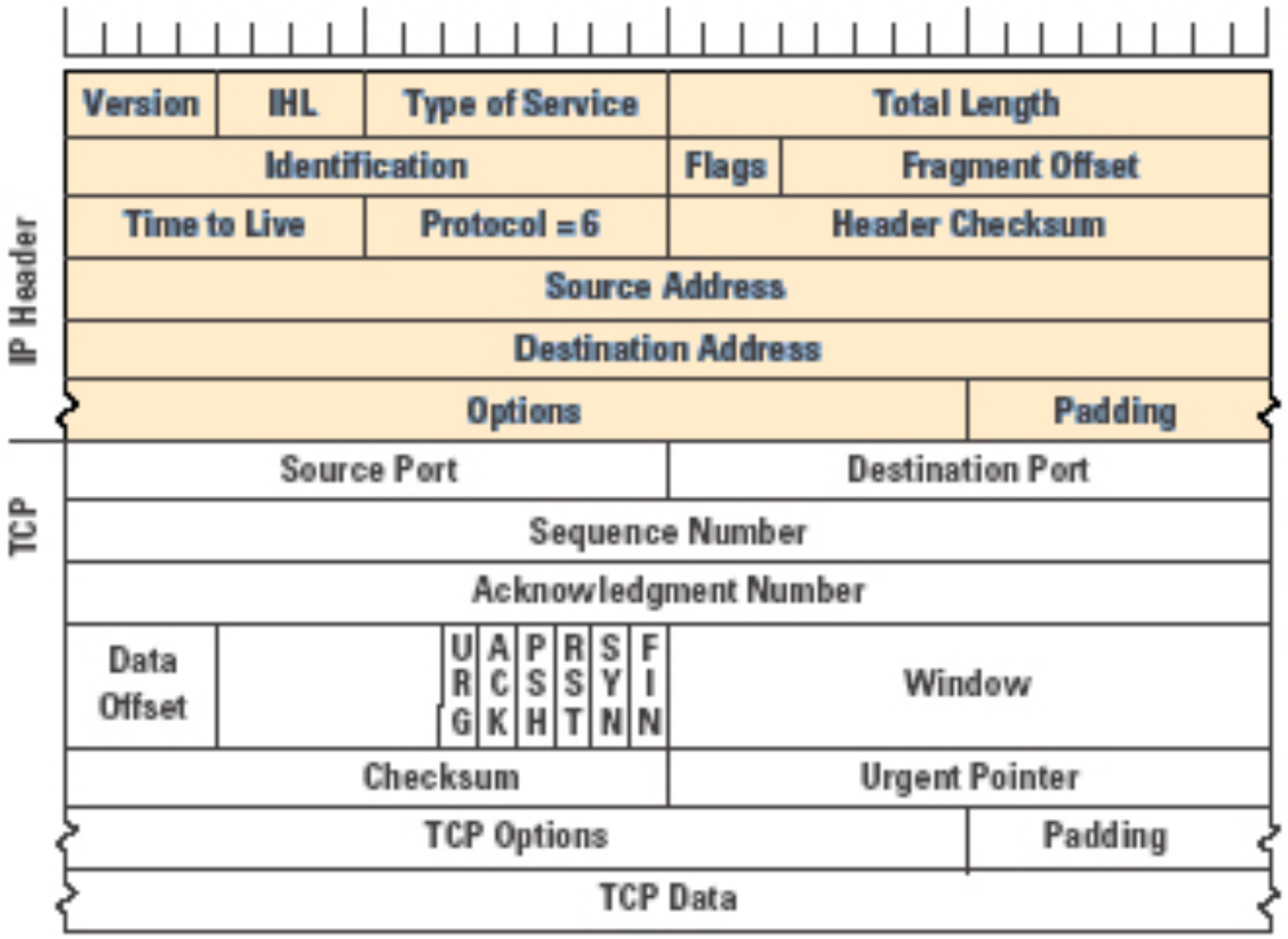
What is IP?

- A common networking framework for applications independent of distance, latency, link layer, or administrative boundary. *

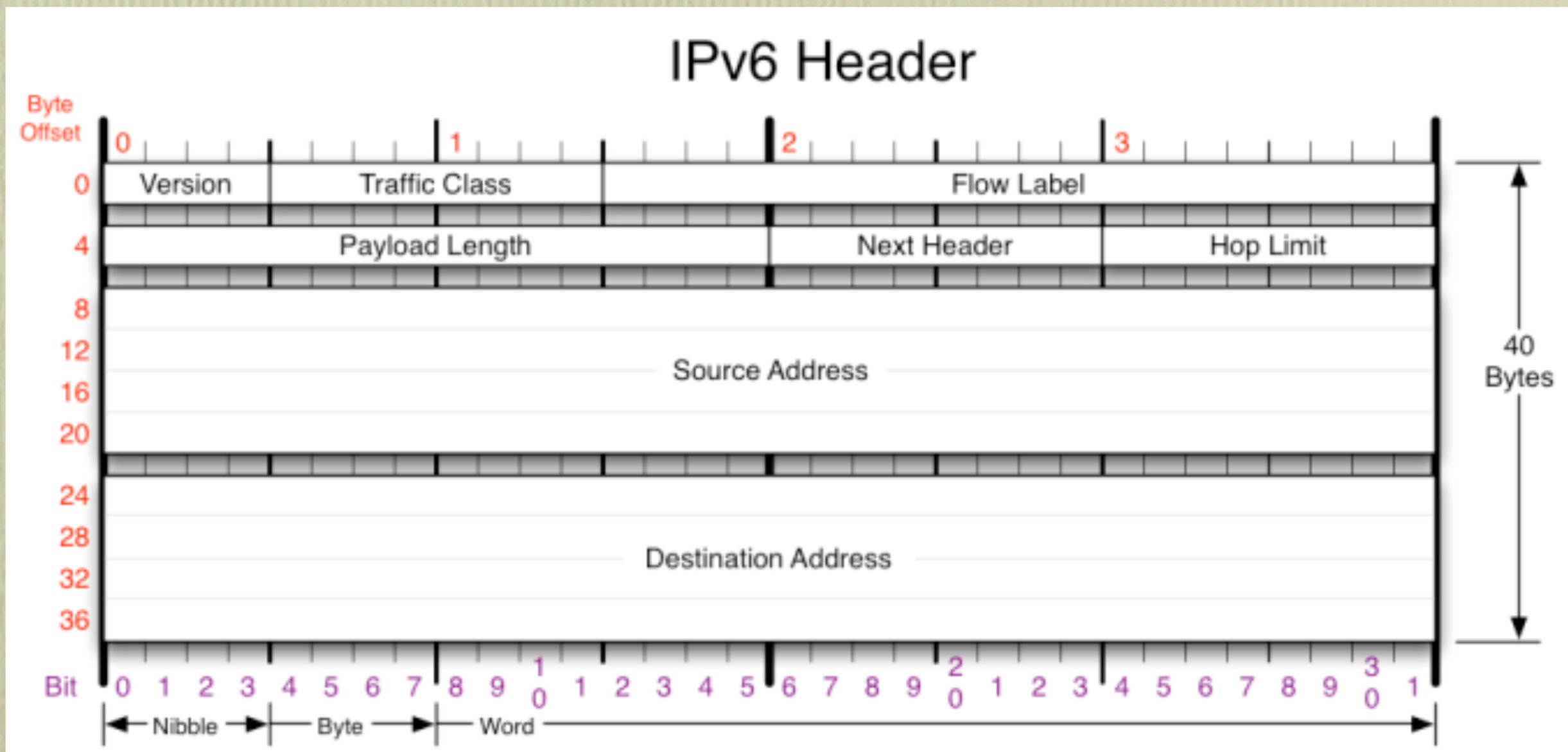
- Dependent on a link layer for transport.
- The only ubiquitous network ‘infrastructure’ today.
- Plesiodeterministic.

Why IPv6?

- Address space exhaustion
- *Easier* configuration and setup *
- *Better* security *



Construction of an TCP/IPv4 packet



IPv6 packet structure

How does IP work?

- At each ‘hop’ in an IP network, a packet’s destination address * is looked up in a routing table and the best *next hop* is selected.
- The correct link layer network is selected for that next hop and the packet is forwarded.
- When a packet reaches it’s destination, it is delivered to a higher layer (TCP, UDP, ICMP, *etc.*) for processing and/or delivery to the application stack.
- The next hop MUST * be *local* on the link layer.

How is the best path selected

- Absent application layer *policy routing*, it's *longest prefix match.*
- An example:
  - Destination = 192.0.2.13
  - Target A = 192.0.2.0/24 (192.0.2.0 - 192.0.2.255) -> nh = 198.51.100.4
  - Target B = 192.0.2.8/29 (192.0.2.8 - 192.0.2.15) -> nh = 198.51.100.32
  - 198.51.100.32 will be selected as next hop.

What's with the notation?

- Used to be class A, B, C, D, and E addresses
  - D &E are *special*
- CIDR came along - needed better way of indicating masks
  - A mask is a way of telling an IP node what part of the address is the *host* and which is the *network* - *i.e.* which local link the host is on.
  - The network portion can be further divided by *administrative* boundary or aggregated space.
- The notation of x.x.x.x/y says that the address x.x.x.x has a y-bit address mask

RFC 1918 and NAT

- Some address blocks set aside for *private* use in IPv4 - controlled by RFC 1918.
- NAT allows many addresses (usually 1918) behind a smaller group (or single) address.
  - 1918 collisions
  - Breaks the end-to-end model - and some applications.
- No equivalent in IPv6
- Does NOT provide security
- Another v4 block is set aside for CGN by carriers.

IPv6 Notation

- While a v4 address is 32 bits, a v6 address is 128 bits.
- v4 addresses are represented by decimal digits on byte boundaries.
- v6 addresses are represented by hex digits on nibble boundaries.
- 2620:0:3f0::47ab:3245/64
  - less than 4 digits between :: - left pad with 0's.
  - :: Fill with :0000:'s - can only be used once.
  - /64 = 64 bits of netmask
  - 2620:0000:03f0:0000:0000:0000:47ab:3245.
    - network = 2620:0000:03f0:0000
    - host = 0000:0000:47ab:3245

v6 Addressing rules

- All *subnets* should get a /64 - allows for *stateless autoconfiguration*
- Specific networks (*e.g.* p2p) can be a /127 or /128
  - All sites should get a /48. Small sites (*i.e.* residential) could get a /56. Remember, a mobile phone may become a router.
- Carriers should get a minimum of a /32 - /20 - /28 more likely.

Address Plans

- Summarization is your friend
- Allow for expansion
- Divide by function or geography? Both?
- Address patterns help at 03:00 on a Sunday morning.
- Uniformity between v4 and v6?

What about MPLS? It's come full circle

- Originally designed to simplify backbone routing (routing table exhaustion and the switching = fast, routing = slow mythology).
- Now, far from simple.
- If you have a hammer…
- Nothing this week requires MPLS.
- Think LONG and HARD about MPLS if you haven't already deployed. Many reasons are no longer valid.

End of IP Overview

- Questions?
- Practical Exercise?
- Break

Multi-path IP Networks

There doesn't need to be just a single exit

Multi-path selection in IP networks

- Default behavior is longest prefix match and then shortest path first
- Policy can perturb the selection
  - These destinations are forced down a given interface (more efficient)
  - These protocols are forced down a given interface (more selective)
  - These priorities are forced down a given interface (in between)

Asymetry

- As IP is a destination-based forwarding system, a router can only influence an outbound packet (i.e. inbound packets may take a different path).
- This can be mediated by changing what routes are announced out of what interfaces
  - Only works for IP addresses (not protocols or priorities)
  - Potentially more fragile, unless it is done right.

Failover

- If done right, it “just works”
  - Make sure covering routes are announced, if appropriate, over all links.
- May take some time
  - Internal - IGP - seconds
  - External - BGP - 1-2 minutes

Link imbalance

- Not all links need to be the same speed, latency, reliability
  - IP has no way of knowing * the characteristics of the link, unless you tell it (weights, policies, *etc.*)
  - "High" quality links (fibre, Intelsat, *etc.*) can be supplemented by "lower" quality links (VSAT, microwave, *etc.*).
    - Many of those lower quality links are as good or better than some high quality links.
- Allows the balance of CAPEX/OPEX to circuit functionality (*i.e.* All services can go over a high quality, $$ link, but some services may go over a lower quality, $ link to improve service or diversity).

End of Multipathing

- Questions?
- Practical Exercise?
- Queue?
- Lunch

IP BGP Routing

One week of training in 3 hours

Remember the *node* we mentioned earlier?

- IP *nodes* are
  - hosts (a device that originates or terminates a packet)
  - routers (a device that forwards a packet that it did not originate or terminate). Routers can also be hosts
  - There are also "gateways" that are a bit of both (closer to hosts, however).

Routers = Routing Protocols?

- Hosts can (but rarely do) run routing protocols
- Routers don't need to run routing protocols.
- Terminology
  - Routing = running a routing protocol
  - L2 forwarding = Ethernet switching
  - L3 forwarding = decisions based on IP addresses

Routing protocols come in two flavors

- IGPs like OSPF and IS-IS
- EGPs like BGP
  - iBGP is internal
  - eBGP is external
- External to what - administrative boundary.
- Administrative boundaries are identified by ASN's in IP networks.

What is an IGP

- Is used only internally.
- There are no policy controls - anyone who exchanges IGP routes with you can completely control your network.
- Used in service providers to locate NH for internal devices/ services (including eBGP routers).
- MAY cary customer networks if those are assigned by the carrier (i.e. 3G, DSL, *etc.*).
- Used for liveness detection (BFD may replace over time).
- Fast cycle time.
- Should be stable.

What is an EGP = BGP?

- Used to distribute/learn external routes (those from outside your AS).
- Two flavors
  - iBGP - distributes eBGP learned routes internally. Does NOT set NHS.
  - eBGP - at the AS boundary. Sets NHS
- Policy enforcement - can protect peerings with foreign networks.
- Longer time scale.
- More “dynamic”.

What's an AS and how does it relate to BGP?

- Autonomous System Number
  - Every address in the GRT originates from exactly * one ASN.
  - Are contiguous
  - Identify an administrative boundary
  - ASNs connect to other ASNs

ASNs and BGP routing

- A LPM match is done on a destination.
- The NH is equated to an ASN
- The shortest path to that ASN is used
  - The next ASN is expected to be able to route to any point in the ASN
  - This is hot-potato routing
  - Can be adjusted if policy steps (or MEDs are used).

What's a MED

- Multi Exit Descriptor
- Allows an AS to tell it's peer to send traffic for certain prefixes to peering connection A, and other prefixes to peering connection B.
- Usually not exchanged unless the business arrangement allows for it (more hassle = more $ $).
- Failures are handled by covering routes.

FIB vs. RIB

- RIB = Routing Information Base = All valid paths to a given target.
- FIB = Forwarding Information Base = Best path for a given target.
- in DFZ/GRT, FIB ~400K routes, RIB ~ 3M routes for v4

BCP

- Route aggregation
  - Reduce the size of the RIB and FIB. More detail the closer you are to a destination, less specific as you get further away.
- Route filters and registries
  - Ensure that a peer can't swamp your router, or send you a route that is incorrect.

Effects of Policy Routing

- Increase FIB/RIB size
- Allows for routing on other than pure dest routes using standard selection (shortest path).
- Allows for the enforcement of business or technology rules.

Options for Application Specific Peering

- Dedicated address space
  - More scalable and performant
  - Increases FIB/RIB size
- ACL based forwarding
  - Obverse of the address space solution

What does a router do?

- Run routing protocols
  - Requires memory and CPU
- Forward traffic
  - Requires performant dataplane

Routing Protocols

- Run in an RE on a hardware based router
- RE's are usually pretty "weak" computers & not updated frequently. A modern server is almost always more powerful than an RE (*i.e.* more memory and RAM).
- Really just userspace software. The *host* part of a router.

Forwarding Plane

- Started off in software in early routers
- Became dedicated hardware (ASIC/FPGA/NP)
  - Still necessary for > *N*x10Gbps data flows.
- GP CPUs can now do ~20Gbps in optimal usage.

Routing Protocol Software

- Dedicated hardware vendors (Cisco, Juniper, Alcatel, *etc*.)
- Commercial Software solutions (Vyatta, Metaswitch, *et. al.*).
- FOSS solutions (Quagga, XORP, BiRD, *etc.*).
- Not all routers in use are CJA - more and more software solutions being used for policy and administrative routing functions.

What is peering?

- In routing terms - a pair of BGP routers exchanging routes (BGP peer).
- In connectivity terms - a connection between two networks where traffic is exchanged between those networks' customers. A transit connection is where one network sends traffic to another network that is not destined for that networks traffic (*i.e.* is destined for the wider Internet).
- In commercial terms - a connection between two networks where the connecting parties see equivalent value, and do not *settle* on the connection. Each side bears their own costs.

How do BGP peers or networks interconnect?

- Private peering
- Common meet-me facilities - Internet Exchange Points, Carrier Hotels, *etc.*
  - Probably more cost effective if at least three networks are interconnecting.

End of BGP/Routing Overview

- Questions?
- Practical Exercise?
- Queue?
- Office hours
- Bar?

VoIP

Basics and Considerations

Agenda for VoIP Overview

- Overview
- SIP
- RTP
- Peering
- Building Blocks
- Software vs. Hardware

VoIP Random Considerations

Some random things to consider on this journey

Emulation vs. Equivalency

- Do you really need to emulate existing services?
  - Billing and packaging?
  - Orphaning customer equipment and services
  - e.g. Modem vs. IP services for an alarm?
- Some revenue is not worth chasing

SIP vs. H.248/Megaco

- H.248 is only a signaling infrastructure for a media gateway.
- SIP is an application framework as well as a signaling gateway.
- Services can be built on top of SIP, not so much on H.248

IMS/RCS vs. OTT

- IMS's RCS was a great idea, 10 years ago..
- OTT services have really embedded over that time.
- Would subscribers accept a service tied to a network and/or specific terminal?
- OTT works anywhere, on any network, on any terminal.
- Is the deployable, commercially viable RCS solution really OTT (*i.e.* WebRTC)?

VoIP Peering

Above the protocols

TDM-based Interconnect

- Well understood
- Need to do translation
- Increased latency in circuit
- Possible that you could see VoIP-TDM-VoIP or even VoIP-TDM-VoIP-TDM-VoIP-TDM-VoIP
- Need Media Gateway hardware (cards or devices)
- OSS/BSS support and interworking/settlement well developed.

IP-level Interconnect

- No additional latency
- No signaling translation
- Less understood
- Strongly recommended to run SBC
- Media transcoding possible if no previous agreement on codecs (end-to-end).
- Layer 8 & 9 issues

IP interconnect, continued

- Bulk IP (i.e. all IP peering, including VoIP)
- Dedicated VoIP peering
- Long-line interconnect vs. colo
- Shared resources?

Why and SBC?

- Much like BGP - you can apply protective policies.
- Attenuate fraud attempts.
- Common protocol point to collect usage statistics.
- Can provide transcoding.
- Can provide NAT-like function.

NFV Overview

Telco meets Webscale

What is NFV?

- Initiative by a number of ‘tier 1’ carriers to leverage web scale infrastructure for telco and SP services.
- Under active development in ETSI with additional work being done in other bodies
- More than just IMS

Existing SIP Platforms

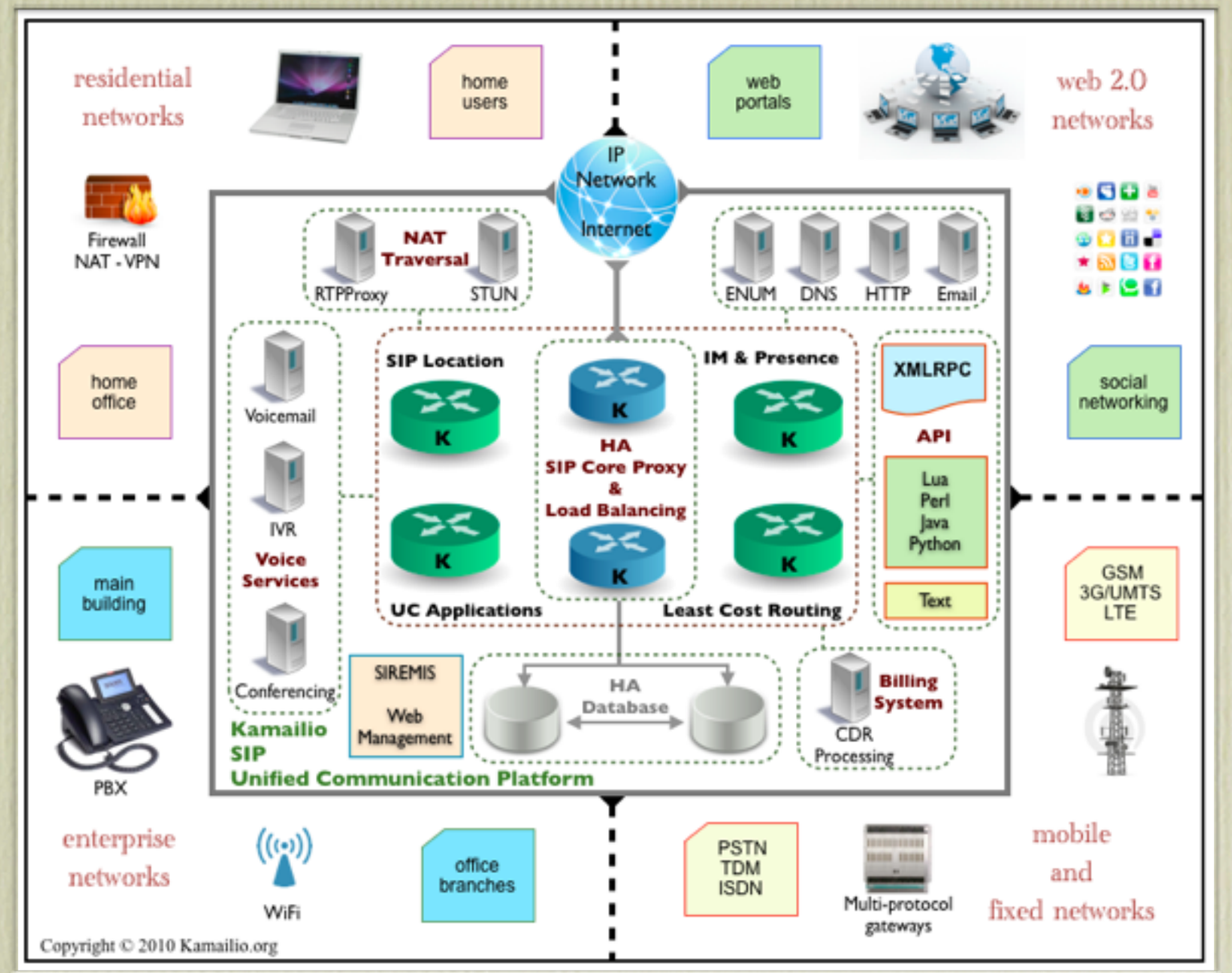
The use of existing FOSS platforms in NFV

Asterisk

- Most well-known SIP software stack
- Large number of add-ons and tooling
- Mainly PBX focused
- An “interesting” configuration environment
- Architecture is best described as “organic”
- Commercial support is available

“OpenSER” / Kamailio / OpenSIPS

- More modern, modular architecture compared to Asterisk
- More easily horizontally scalable
- Not as much a PBX as general SIP platform
- Can provide many of the services that have been discussed.
- Commerical support is available.



Kamailio Architecture

Some other NfV applications

It's not just the Network...

Network Services

- Virtual Carrier / VISP / MVNO?
- Virtual PBX
- Managed Firewall/VPN/DPI/*etc*?
- Virtual CPE

IT Services
It's not just for business

- vHD - caching
- cloud Bursting / HA
- cloud Hosting
- vIT - Mail, Wiki, *etc*.

Scalable Fabrics

Cloudy with a hint of clarity

My slides to date:

http://www.asgaard.org/pita/dispersed-nfv.pdf

What is *Cloud*?

- A technology?
- An operational model?
- A prime component of buzzword bingo?
- YES

Isn't it just virtualization?

- In a virtualized environment, I can still point to the server hosting the guest, the disk array hosting the data, and the network port that the traffic is flowing over. In a *cloud*, or, more appropriately, a *fabric*, it's "over there, somewhere." More importantly, it's not necessary to know. The service is disjunct from the underlying platform.

Types of Service / *X*aaS

- IaaS - Infrastructure as a Service - Selling raw compute, storage, and pipe.
- PaaS - Platform as a Service - Selling a development platform. IaaS indirectly inclusive.
- SaaS - Software as a Service - Selling a software based service. IaaS and, potentially, PaaS indirectly inclusive.

Multiple constituents

- Compute
- Storage
- Network
- VAS
- OA&M
- Physicals

Compute

Types of Compute - Bare Metal

- Primarily used to host VM microkernels / Hypervisors.
- Some used for base administrative functions.
- Sometimes used for specific requirements (Database servers, high-capacity storage, *etc*.)
- Starting to be offered to consumers.
- Should be able to be re-provisioned

Types of Compute - Virtual Machines

- Provides a logical computer that is completely isolated at the OS layer - a virtual server.
- Most common compute model in the cloud.
- Continual innovation - overhead reduction
- Ballooning and vcores / vram

Hypervisors

- VMWare - largest footprint, $$, more enterprise focused
- KVM - integrated in Linux kernel, most active development.
- Xen - predates KVM, finally in the kernel - still some features not in KVM.
- HyperV - if you have an all MS environment, *maybe*.

Containers

- Shared OS
- Isolated application environment
  - resource containment
  - library and environment isolation
  - ACLs
- Very efficient
- UML, Jails, Erlang

VM Migration

- Live migration
  - Storage and networking implications
  - Wide area vs. local
  - Pre-planed vs. emergency

Storage

NAS

- Can only serve files
- Issues with global name space
- Issues with clustering
- $$
- Primarily vertical scale

SAN

- Primarily Blocks
- $$$
- iSCSI and AOE - lessor of the evils
- FC - requires special hardware on compute

DAS

- Leverage the disks already on the servers
- Horizontal scale if done right
- $
- Can be high IOP - depends on load
- Can be very fault tolerant
- Erasure encoding, replication, vs. RAID
- Tape?

Object stores

- Ceph, SWIFT, S3
- Can provide blocks and files
- Hashing and buckets

Databases

- Atomicity, Consistency, Isolation, Durability
- Horizontal vs. Vertical
- Key/Value, column, graph stores
- SQL vs. NoSQL

Network

Classical Datacenter 3-tier

- L2 edge, L3 core - where's the state?
- Congestion and hairpin
- Blocked bandwidth

Flat L2 fabrics

- Physical
  - Folded Clos
  - Toroid
- Native - VLANs
- Tunneled - VXLAN/NVGRE
- OpenFlow / SDN
- Remember the switch in the server

L3 fabrics

- Common to TOR
- Maybe further down?
- IP only services

Value Added Services

Other services

- DBaaS
- FWaaS / DPIaaS / VPNaaS
- *etc.*
- Service Chaining

OA&M

Infrastructure Provisioning

- Discovery, classification, boot and deploy
- Scale out, scale in
- Hardware alert and alarm
  - Support model

Platform basic OA&M - Cloud OS

- VCenter *et.al.*
- CloudStack
- OpenStack
  - keystone, swift, nova, neutron, glance, horizon, cinder
  - AWS vs OS APIs

More OpenStack development

- Bare metal
- heat
- DNSaaS
- one-click install
- Chef & Puppet libraries

Physicals

Cloud-friendly “normal servers”

- Only what you need
  - Do you need to hot-swap drives?
  - Do you need redundant power?
- No video - serial or SOL
- IPMI / BMC - via SSH or HTTPS
- Higher temp friendly
- Front IO? Consider switches.

Higher-density servers

- dual twin, fat twin, etc.
  - Shared power - ok, shared anything else, probably not.
  - Chassis should be considered a failure domain.

Open Compute

- Facebook
- Datacentre to Server - everything in between
- Gathering momentum
- Possibilities
  - New Rack, new power
  - micro servers

New DC ideas

- HV DC
- HV AC
- free-air cooling
- low PUE

Parting Thoughts

- Don't build a cloud based on external product demand
- Do build a cloud for internal use
- Do use it to sell captive services
- Do sell excess capacity to external customers
  - It may become a stand-alone product set(s).