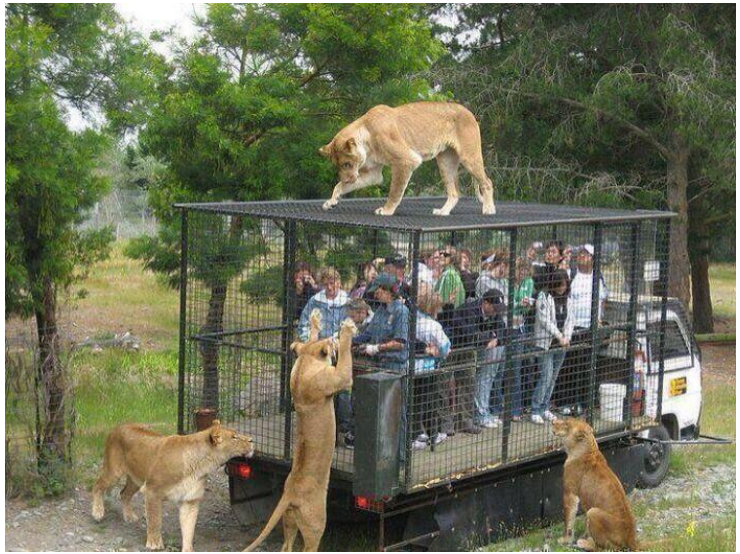
Encapsulation Considerations

Design team report
draft-rtg-dt-encap-01.txt

Design team members

Albert Tian
Erik Nordmark
Jesse Gross
Jon Hudson
Larry Kreeger
Pankaj Garg
Pat Thaler
Tom Herbert

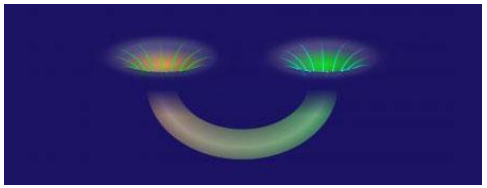


Charter http://www.ietf.org/mail-archive/web/rtgwg/current/msg04715.html

Motivation for design team

- IETF doing new encaps - NVO3, SFC, BIER
  - And multiple might be used in the same packet
- Each encap has its own information, but also needs to handle common issues
  - Explore more common ways to handle those issues
  - Each proponent/WG doesn't need to reinvent
- Focus is on encaps packet format - *not* on control plane

What this IS



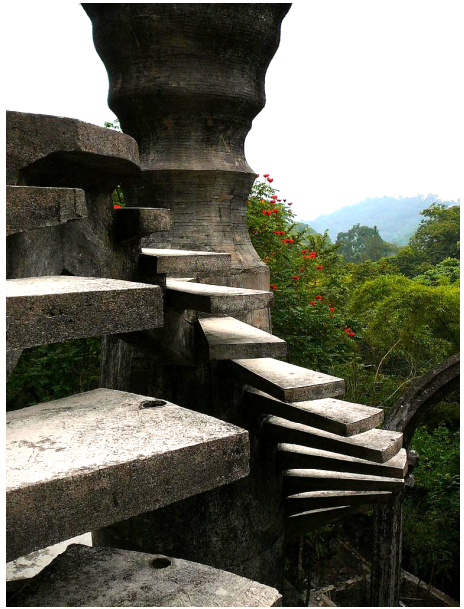
- A look across the three new encapsulations
  - While taking lots of previous work into account
- Focus on encaps that run over IP/UDP
  - Many encaps desire to run at least over IP
  - Avoided diving into control-plane interaction
- Turns out some “transport” independence fell out as a result
  - E.g., MPLS entropy label fits in

What this is NOT

- A design of a new encaps to rule them all
- A design of a new NVO3 encaps
- A selection from existing encapsulations
- An evaluation of existing and proposed encapsulations
- A floor wax and/or dessert topping

Set of common issues

A twelve-step program



1. How to provide entropy for ECMP
2. Next header indication
3. Packet size and fragmentation/reassembly
4. OAM - what support needed in an encapsulation format?
5. Security and privacy
6. QoS
7. Congestion Considerations
8. Header and data protection - UDP or header checksums
9. Extensibility - for OAM, security, and/or congestion control
10. Layering of multiple encapsulations
11. Service model
12. Hardware Friendly

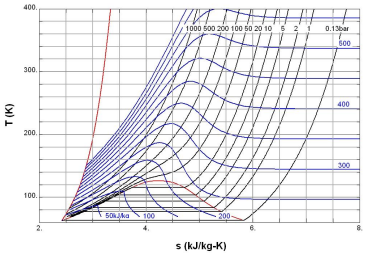
Different encaps - different information

- NVO3 needs to carry at least a VNI-ID
  - Carried edge-to-edge unmodified
  - Optional OAM info like timestamps modified?
- SFC carries service path and meta-data
  - Index modified at each hop for loop prevention
  - Service meta-data may be modified by SF
- BIER carries a bitmap of egress routers
  - Bitmap modified as packet is forwarded

Assumptions

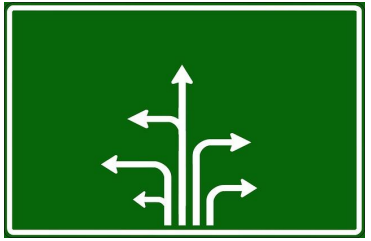
- Underlay MTU is managed and configured
  - Encaps can make packets larger
- TE/traffic management differs from TCP CC
  - The underlay is well-provisioned, policed
  - Due to multi-tenancy, endpoint CC is not trusted
- Implementable in hardware and software

Entropy for ECMP



- UDP source port for hash of inner headers
  - Provides >=14 bits (ephemeral range) plus IP src, dst
  - IPv6 will provide more IP src, dst bits, flow label
- Q: Allowed to look inside for more entropy?
  - A: Avoid messing up OAM frames and extensions
- Entropy field belongs to “transport” i.e. adjunct to IP header.
  - Fits with using MPLS as another “transport” - has its own entropy label

Next header indication



- Each encap want to carry different payloads
  - Use Ethernet types? IP protocol number? Create new numbering space?
- When layering multiple encaps headers?
  - Define a common approach?
  - Define a common numbering space?
- But also needs to fit with existing schemes
  - UDP uses port numbers; GRE Ethernet types; etc.
  - Used to indicate the (first) encaps header

Packet size and fragmentation

- Deployed overlays assume underlay MTU
  - Reasonable for controlled deployments in datacenter or SP networks
- Useful to detect misconfiguration
  - Set outer don’t fragment (DF) flag
  - Report any received ICMP packet too big - syslog
  - Possible to generate overlay ICMP PTB for IPv4/6
  - For Ethernet payload - use existing LLDP TLV?
- Other encaps could do frag/reassembly

OAM

- Discussed in NVO3 and SFC and LIME
  - Rich architectural discussion
  - We only looked at effect on encaps format
- Need for in-band OAM measurements
  - Add measurement info to data packets
- Out-of-band measurements
  - OAM packets follow same path as data packets
  - Assumes same ECMP, QoS, middlebox/firewall
  - Constraints entropy use in forwarding routers

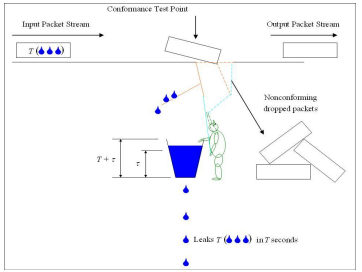
OAM support

- Avoid sending OAM frames to end stations
  - Use some “discard” next header value, or OAM bit?
- Support in-band OAM measurements
  - Bit for counter sync between ingress and egress
  - Optional timestamps etc in encaps header
- Error Reporting Protocol as part of OAM?
  - How to avoid it being filtered as ICMP often is?
  - Recommend that IETF look into error reporting that is independent of the specific encaps

Security and privacy

- At least three considerations for security
  - Anti-spoofing - prevent packet injection
  - Interaction with and use of IPsec
  - Privacy
- Different possible anti-spoofing mechanism
  - Cookie in encaps header - against off-path attacks
  - Secure hash of header fields (excluding fields modified in transit)

QoS



- Existing specifications such as RFC 2983 (Diffserv and tunnels) can be applied
- If OAM messages are used to measure latency, need to treat them the same as data payloads

Congestion Considerations

- Explicit Congestion Notification - RFC 6040
- Carrying non-congestion controlled traffic
  - “Encapsulating MPLS in UDP” draft-ietf-mpls-in-udp
  - Circuit breakers? draft-ietf-tsvwg-circuit-breaker
- Protect against malicious end stations
  - Congestion control/policing across tunnels
- Ensure fairness with multi-tenancy?
  - draft-briscoe-conex-data-centre?

Header protection

- RFC 6936 Applicability Statement for the Use of IPv6 UDP Datagrams with Zero Checksums
- Need checksum for the encaps header?
  - Misdelivery if e.g. VNI ID, BIER bitmap is corrupted
  - Using pseudo-header for important IP fields?
- Ties in with higher assurance for security
  - No need for checksum if secure hash is used?

Extensibility

- Needed semantics
  - New incompatible version
  - Stuff which can be ignored by the egress
  - Error/drop if egress doesn’t support
  - Handle on-path parsing (BIER routers, middleboxes)
- Different encodings
  - Use reserved bits/fields
  - TLVs; extension header chains
  - Flag-fields as in GRE
- Use it or lose it?

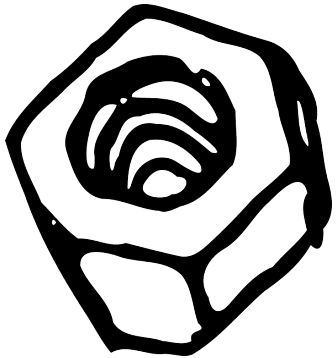
Layering of multiple encapsulations

- Might see a future with e.g.,
  - BIER+NVO3+SFC+payload
  - NVO3+NVO3+payload
- Q: Would there be multiple UDP headers?
  - A: UDP header goes with IP header
- Implications for devices in the path
  - Can inspect any layer (and drop/forward)
  - Can only modify its own layer (eg SFF, BIER router)
  - Otherwise needs to be visible i.e. decap+encap

Service model

- IP service is lossy and subject to reordering
  - Unordered for different flows - unicast vs. multicast
- Some services might desire no reordering, timeliness or drop, rate limiting, FEC, etc
  - If so, layer on top of encaps
  - Possible to reuse PWE3 [RFC3985, RFC5586]
  - Potentially relates to timestamps for OAM
- Tunnels becoming a protocol fixing place?
  - This is a slippery slope

Hardware Friendly



- Not required, but impacts deployment
  - Using existing chips; facilitate design of new chips
- Different hardware concerns for
  - Switch/router chips, vs. NIC offload
- Encap header checksum OK - not whole
  - However, NIC offload can do whole pkt checksum
- Put important info at fixed offsets
  - Unconstrained TLVs seem hard
  - Limit number of header combinations

Middlebox Considerations

- As encapsulations get widely deployed middleboxes might do more
  - Not just drop based on UDP port number
  - Gateways stitching could have similar effect
- Example would be to filter VNI IDs for NVO3
  - Better defense in depth
- Should the IETF document what not to do?
  - Avoid accidentally blocking OAM but not payload
  - Avoid interfering with ECMP?

Open Issues

- Common OAM error reporting protocol?
  - Useful or a distraction?
- Next protocol indication - common across different encapsulation headers?
- In-order-delivery service layer on top vs. sequence numbers and timestamps for OAM and CC?

Next Steps

- Gather feedback from different groups in the IETF
- RTGWG WG document? Or somewhere else?