Please don’t VXLAN

Please don’t VXLAN

Initially presented at NANOG 74 in 2018

History of VXLAN

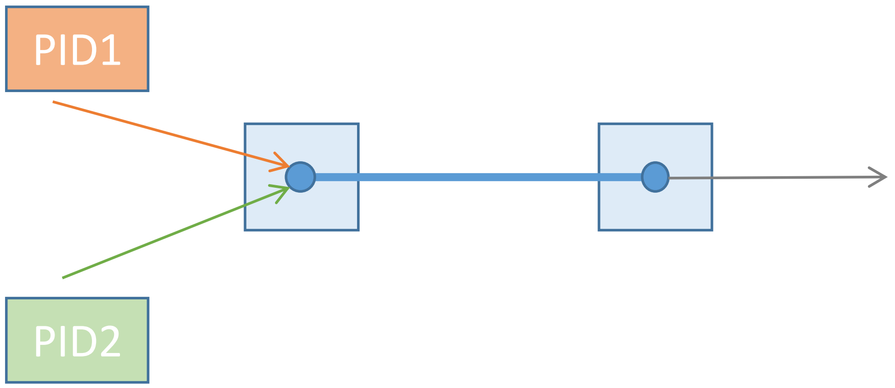
VXLAN was a successful accident

- Designed by a single vendor for a single product family.
- Designed for a specific and bounded use case.
- ISE stream, Informational level, not a product of any IETF WG.
- (Some of) the engineering tradeoffs were known and accepted at that time

VXLAN got successfully used and abused for things not even envisioned at the design time, and the limitations are now evident.

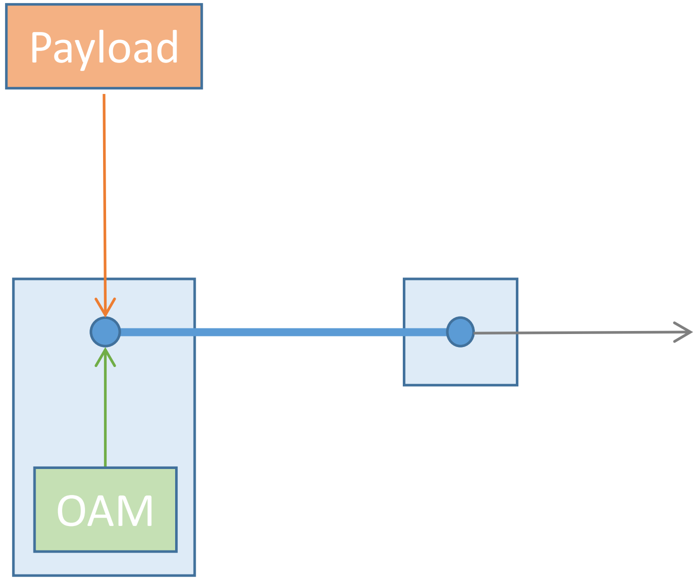
Protocol identifier

- There is no payload type identifier.
- A single tunnel cannot carry more than one payload type.
- VNI namespace is large enough, and that is not a practical scalability problem.
- Number of supported tunnels is a far bigger practical scalability problem.
- Originally VXLAN was envisioned for carrying untagged Ethernet payload only.



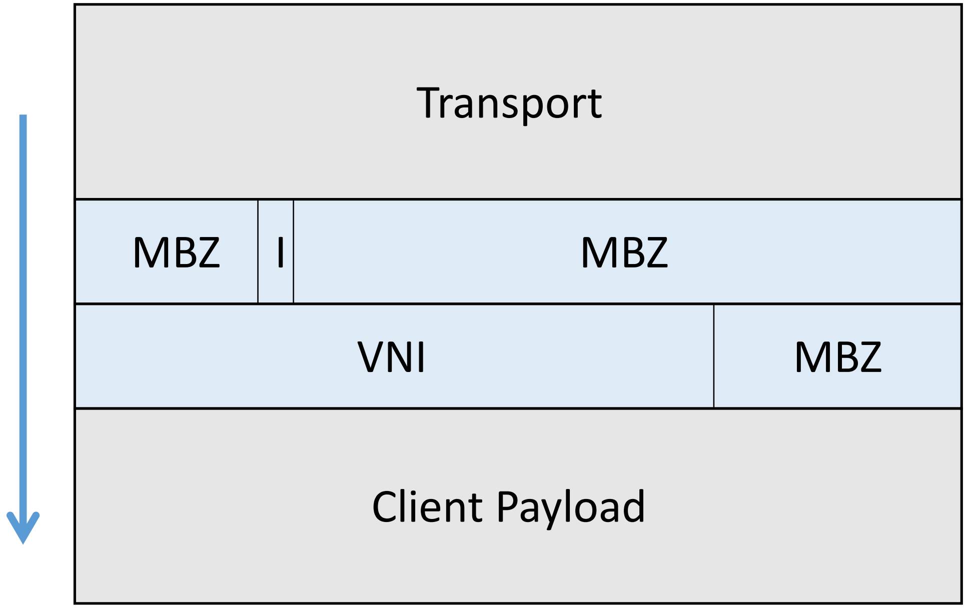
Non-client payload

- Everything in the tunnel is a payload. A client payload.
- BFD, MPLS (G)ACh, Y.1731 all require injecting corouted OAM messages into the tunnel.
- This rules out majority of common OAM mechanisms.
- Running traditional OAM toolkits over VXLAN may provide you with some data. The quality and reliability of that data is questionable.
- Client cooperation is required and assumed, OAM operations are not transparent.
- OAM mechanisms cannot just be added on top of some data plane capsulator – that needs to be architected in from the start.
- Can be partially solved by using a different PID for OAM payload.



No extensibility

- A relatively small VXLAN header with a few fields.
- While only a small fraction of space actually in use.
- No versioning.
- No practical way to add extensions in an interoperable manner.
- Header appears to be too small for major extensibility.
- There are proprietary VXLAN extensions in the field, not interoperable.



Transport

MBZ | I | MBZ

VNI | MBZ

Client Payload

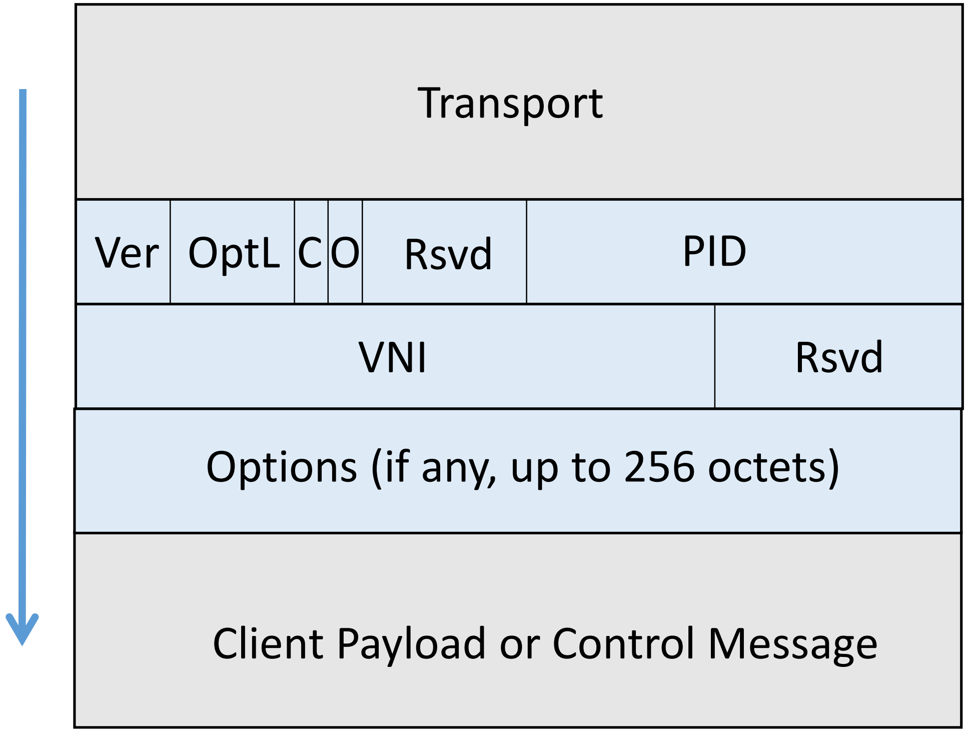
Geneve, RFC8926

- IETF stream, Proposed Standard level.
- TLV based extension headers, IANA codepoint registry.
- A compromise of extensibility and HW friendliness for vendor-specific functionality.
- Payload type indicator.
- OAM indicator.
- Transit node alert indicator.
- Base capsulator header is 8 octets, plus up to 256 octets for extensions.
- Works like VXLAN, just better.

- Geneve is not the only option, several other capsulator choices are available.
- Geneve is a final IETF selection from a set of options.

Geneve, RFC8926

- IETF stream, Proposed Standard level.
- TLV based extension headers, IANA codepoint registry.
- A compromise of extensibility and HW friendliness for vendor-specific functionality.
- Payload type indicator.
- OAM indicator.
- Transit node alert indicator.
- Base capsulator header is 8 octets, plus up to 256 octets for extensions.
- Works like VXLAN, just better.



Transport

Ver | OptL | C | O | Rsvd | PID

VNI | Rsvd

Options (if any, up to 256 octets)

Client Payload or Control Message

Design aspects (a view back from 2018)

VXLAN is starting to show its age.

- Be careful with new designs.
- Especially if OAM (interoperability) is needed.
- Component vendors are ready.
- System vendors are getting there.
- Architects and operators need to be aware.
- The changes are in data plane. Control plane components stay the same.