The HIP-HOP proposal

draft-matthews-p2psip-hip-hop-00

Philip Matthews

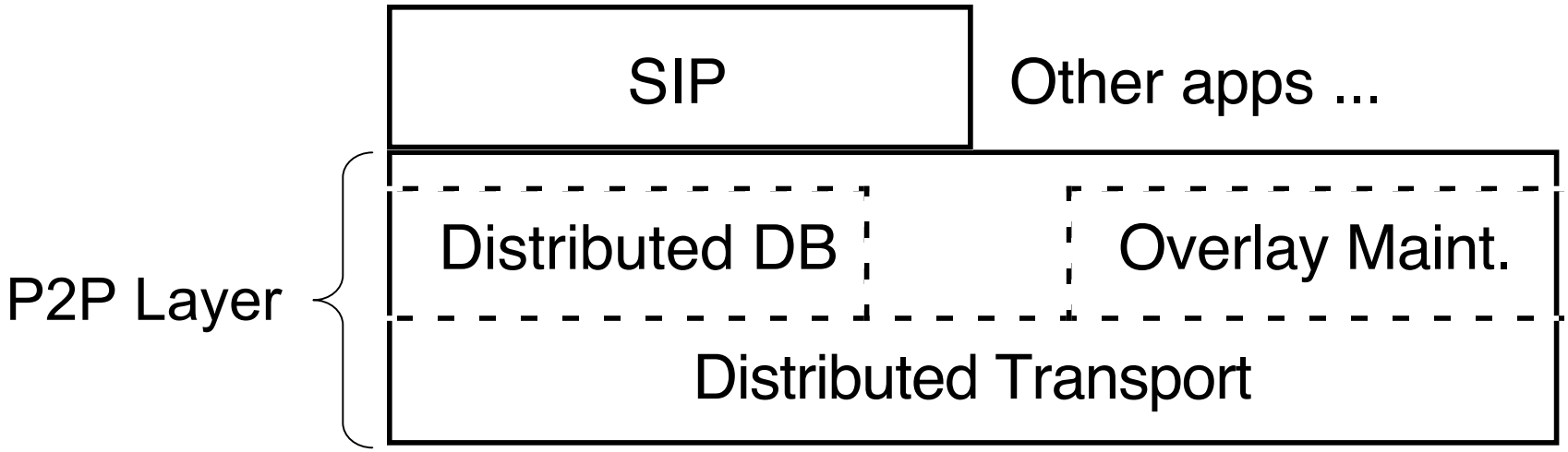
philip_matthews@magma.ca

HIP-HOP vs. the others

- Uses HIP (Host Identity Protocol) and leverages work of HIP WG.

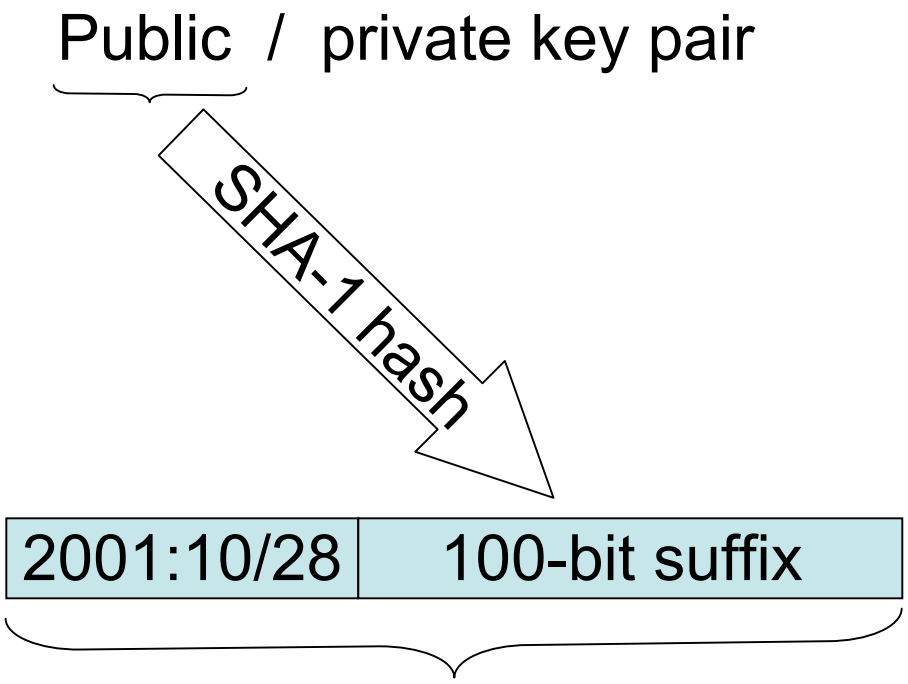
- Three key differences:
  1) Architecture of P2P layer
  2) Definition of Peer ID
  3) How forwarding at peers is done

Diff #1: Architecture



- Other proposals: monolithic P2P layer. HIP-HOP: Three separate protocols.
- Distributed Transport: deliver msg to arbitrary peer (even if behind NAT).
  - Focus of HIP-HOP proposal. Can adapt other proposals to provide Distributed DB and Overlay Maint protocols.
  - Distributed DB, Overlay Maint, SIP, and other apps use this layer for sending/receiving.
  - Supports apps other than SIP.

Diff #2: Peer IDs in HIP-HOP



- Public key is ultimate identifier of a peer.
  - Peer can prove ownership because it alone knowns private key.
- Peer ID looks like IPv6 addr, but is distinguishable due to prefix.

Process defined in RFC 4843 and draft-ietf-hip-base.

Diff #2: Peer IDs

HIP-HOP

Peer ID is special IPv6 address with crypto:

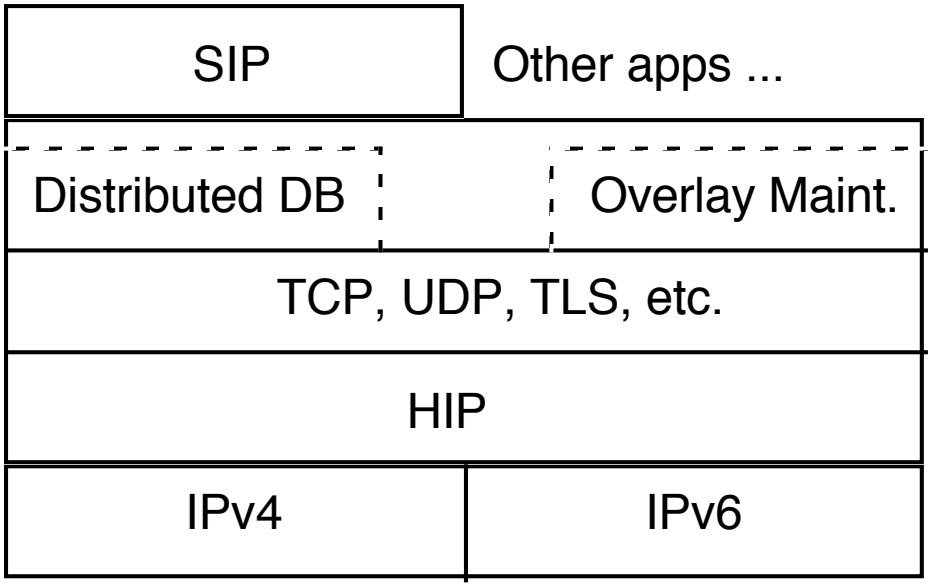
- Prevents identity theft;
- Re-uses existing APIs and IPv6 protocol work;

Other Proposals

Peer ID is 160 bits (ASP is 128), no crypto:

- Can hijack a Peer ID;
- Need new APIs and protocol extensions;

Diff #3: Forwarding



⇐ Socket API

⇐ Forwarding at each peer done at this layer

- A UDP encapsulation layer is used when necessary to transport HIP through NATs.
- HIP layer also replaced/removed in certain cases.

Diff #3: Forwarding at Peers

HIP-HOP

Forwarding done **below** transport layer:

- Use Socket API and all existing transport protocols
  - Many apps = no change
  - Transport protos work with Peer IDs, not IP addresses
- Transport conn = end-to-end;
  - TLS security / reliability / congest cntl is end-to-end
- NAT traversal and mobility handled at HIP layer.

Other Proposals

Forwarding done **above** transport layer:

- Need new APIs
  - All apps must change
- Transport conn = hop-by-hop
  - TLS security / reliability / congest cntl is hop-by-hop
    = “link layers”
  - Some proposals try to patch some of these problems
- NAT traversal and mobility handled by each app separately.

Other Details in Brief

- Inherits well-though-out security properties of HIP.
- Inherits HIP mobility support.
- High-level NAT Traversal strategy as in draft-matthews-p2psip-nats-and-overlays
  - (Same as dSIP, RELOAD, ASP)
  - Detailed NAT traversal procedures use HIP procedures (ICE)
- Bootstrap procedures as in draft-matthews-p2psip-bootstrap (adapted for HIP)
- HIP used for signaling overlay connections, and encapsulating application data
  - HIP header carries src and dst peer ID, etc.
- **Interoperable open-source implementations of HIP for Windows, MacOS, Linux, and FreeBSD.**
  - Working on HIP-HOP extensions.