Automatic Multicast Tunneling & Upipe: a Proof of Concept

Christophe Massiot (FOSDEM 2015)
Why multicast?

- Saves network bandwidth by avoiding packet duplication
- Particularly useful for television, but not only
Unicast vs. Multicast Scaling

Number of Streams

Server

Router

Server

Router

Unicast

Multicast
Who uses multicast?

- “Any Source Multicast” (RFC1112) requires assignment of limited resources
- Several (complex) subscription protocols
- Used in closed IPTV networks
Multicast over Internet Howto

- “Source Specific Multicast” (RFC4607) defined by source + multicast addresses
- Simple subscription protocol
- Also used for IPTV
- Built for Internet, but seldom used
  - Lack of incremental adoption strategy
  - Industry concentrated on IP multicast in networks that could roll it out natively
Automatic Multicast Tunneling

- RFC draft “draft-ietf-mboned-auto-multicast-18”
- Protocol between a gateway (client-side) and a relay (server-side)
  - Allow multicast distribution to extend to unicast-only connected receivers
  - Provide the benefits of multicast wherever multicast is deployed
  - Work seamlessly with existing apps
AMT—Automatic Multicast Tunneling

Additional Receivers Are Served by the AMT Relays; the Benefits of IP Multicast Are Retained by the Content Owner and All Enabled Networks in the Path
AMT—Automatic Multicast Tunneling

Creates an Expanding Radius of Incentive to Deploy Multicast

Mcast-Enabled ISP

Content Owner

Enables Multicast Content to a Large (Global) Audience

Mcast-Traffic

Mcast Join

AMT Request

Ucast Stream
What AMT doesn’t fix

• Packet loss
• Network latency, jitter, packet reordering
• Network congestion (adaptive bitrate)
PoC: Upipe + Chrome

• Display a multicast stream in a web browser, using AMT if needed
• Without AMT support from the OS, or from a local network equipment
Non-standard video in a browser

• NPAPI deprecated
• ActiveX not portable and unsecure
• Media Source Extensions very tied to “chunked” delivery
• Google’s PPAPI & Native Client best choice for a proof of concept
Choice of components

- Cisco’s open source AMT library
  - https://github.com/cisco/SSMAMTtools.git

- Multimedia framework w/ NaCl support: Upipe
  - http://upipe.org/
Limits of the current PoC

• Sockets blocked by default by NaCl sandbox
• No IGMP support in PPAPI
• No FFmpeg assembly optimizations
Future perspectives

• PPAPI now features a “video decoder” interface
• AMT should probably be integrated into MSE/other W3C stuff
Hands on!

Multicast sender → IGMP → Multicast-enabled network → AMT Relay

EBU

http://upipe.org/player_chrome/
https://workspace.ebu.ch/display/BISMulti/
cmassiot@openheadend.tv
Upipe meet-up in BOF room H.3.XXX Sunday XX:00