Topics

- Main changes since draft-ietf-anima-grasp-01
- Prototype code
- Open issues
- Discussion, next steps
Main Changes (1)

- draft-ietf-anima-grasp-02:
  - Resolved issues according to WG discussions
  - Added optional error string to DECLINE Option
  - Redefined naming rule for Objectives so that PEN is one option among several others (MUST -> MAY)
  - Added FLOOD & SYNCH messages to simplify the message coding
  - Added initiator id to DISCOVERY, RESPONSE and FLOOD messages
Main Changes (2)

- **draft-ietf-anima-grasp-03**:  
  - Split REQUEST message into two (Request Negotiation + Request Synchronization) and updated other message names for clarity.  
  - Removed initiator id from DISCOVERY, RESPONSE and FLOOD messages

- **draft-ietf-anima-grasp-04**:  
  - Added initiator id to DISCOVERY, RESPONSE and FLOOD messages and finally made the loop detection for relayed multicasts correct.
Current message names

• Discovery (link-local multicast)
  Discovery Response
• Request Synchronization
  Synchronization
• Flood Synchronization (link-local multicast)
• Request Negotiation
  Negotiation
  Confirm Waiting
  Negotiation End
• No Operation Message (only for practical)

*Unicast except where noted*
Multicast relaying (1)

If there is a physical loop in the network...

Source of Discovery or Flood

Relay

Relay

Relay

Relay
Multicast relaying (2)

• To detect such a loop and kill it:
  - the message carries a unique ID (Session ID + Initiator ID)
  - each relay MUST cache the ID when it relays a link-local multicast
  - each relay MUST check the cache when it receives a link-local multicast
  - clear the cache after a suitable timeout (at least GRASP_DEF_TIMEOUT)

• Worst case:
  - the looped message arrives later than the cache timeout

• The GRASP_LOOP_COUNT will act as a backup
Python prototype

• A Python 3 implementation of GRASP as a module `grasp.py`
  – About 1100 lines of code
• A test suite to exercise as many code paths as possible, `grasptests.py`
• Two toy ASAs to test operation across the network, `Briggs.py` and `Gray.py`
• [https://www.cs.auckland.ac.nz/~brian/graspy/](https://www.cs.auckland.ac.nz/~brian/graspy/)
Tests (they worked!)

- Building switch (supports IPv6 but defective MLD snooping)
  - Netgear switch (just a bridge)
    - ASA Gray in nuc9 (a neat little Debian Linux box)
    - ASA Briggs in BEC (Brian’s Windows 7 laptop)
Open Issues

• 7. Cross-check against other ANIMA WG documents for consistency and gaps.
• 43. Rapid mode is currently limited to a single objective for simplicity. A future consideration is to allow multiple objectives in rapid mode for greater efficiency.
• 48. Should the Appendix "Capability Analysis of Current Protocols" be deleted before RFC publication?
• Should the Reference Model talk about multiple instances of GRASP (e.g. an insecure instance as well as the secure instance)?
Discussion + next steps

• We need more reviews of the draft.
• We need people to think about implementation issues. Either play with the prototype or write your own!

• GRASP API
  - draft-liu-anima-grasp-api