RELOAD Status/Open Issues

draft-ietf-p2psip-base-11

IETF 79

Eric Rescorla

ekr@rtfm.com

Overall Status

- draft -10 (Aug 3), draft-11 (Oct 12)
  - Resolved most known open issues
  - Thanks to Eric Burger for a detailed review
- Second WGLC ended November 4
  - Some minor new issues raised
- General plan
  - Resolve remaining issues here
  - Confirm on the list
  - Generate a finished draft by December 10

Variable-length node-ids

- Enacts WG consensus
- Fixed per overlay
- Range of 16-20 bytes
- Set in configuration document

Non-TLS security modes

- Enacts WG consensus: (D)TLS for now with room for other prototocols in future
- Requirements for future link protocols in §5.6.1:
  - Endpoint authentication
  - Traffic origin authentication and integrity
  - Traffic confidentiality
- Set in configuration document

Direct Response Routing

- Permitted on a single overlay basis
- Set in configuration document

Minor Changes

- Provided a definition of `AppAttachReq` and `AppAttachAns` in §5.5.2.1 and 5.5.2.2.
- no ICE → NoICE
- Added a `send_update` flag to `AttachReqAns` to facilitate requests for immediate updates

Minor Changes: RFC 2119 issues

- Removed MUST-level requirement for generation counter on opaque `Destination` values as unenforceable [Eric Burger]
- Made setting `FORWARD_CRITICAL` and `DESTINATION_CRITICAL` MUST-level with `DirectResponseForwardingOption`. (interop requirement)
- Recipients now MAY process messages with unknown non-critical extensions (was SHOULD) [Eric Burger]
- Clarified what the MUST requirement is for processing `Attach` (you can refuse and throw an error) [Eric Burger]
- Strengthened requirements on which STUN servers to use (MUST use one from the same group) in §5.5.1.4.

Known Uncontroversial TODOs

- Add padding to `PING` to facilitate MTU discovery
- Unwind misguided leap-second correction in §5.5.3.2

ICE: Nomination Level

- §5.5.1.10.2 formerly required regular nomination
  - Regular nomination is very slower than aggressive
  - There are already a lot of round-trips
- Original rationale was to ensure consistent state
  - Don't believe this is needed: ICE naturally converges

Proposed Resolution: Leave as-is in the draft

Mandatory to Implement Signature/Hash Algorithms

- None specified
- Need some for interop

Proposed Resolution: RSA with SHA-256

Direct Response Routing and ICE

- Specified in §5.3.2.4

> This option can only be used if the direct-return-response-permitted flag in the configuration for the overlay is set to TRUE. The RESPONSE_COPY flag SHOULD be set to false while the FORWARD_CRITICAL and DESTINATION_CRITICAL MUST be set to true. When a node that supports this forwarding options receives a request with it, it acts as if it had send an Attach request to the the requesting_node and it had received the connection_information in the answer. This causes it to form a new connection directly to that node.

- This doesn't work with ICE because the sender of the request doesn't have your information

Proposed Resolution: DRR can only be used with No-ICE

Node-Ids in JOIN/LEAVE

- Currently JoinReq and LeaveReq have the joining Node-Id

```
struct {
   NodeId                     joining_peer_id;
   opaque                     overlay_specific_data<0..2^16-1>;
} JoinReq;
```

- This is unnecessary because the Node-Id is provided by the security protocol
- Just one more thing to check

Proposed Resolution: Remove Node-ID from these messages

Specifying Counter Values for `NODE-MULTIPLE`

§6.3.4:

> `In the NODE-MULTIPLE policy, a given value MUST be written (or overwritten) if and only if the request is signed with a key associated with a certificate containing a Node-ID such that H(Node-ID || i) is equal to the Resource-ID for some small integer value of i. When this policy is in use, the maximum value of i MUST be specified in the kind definition.`

- `i` is not carried on the wire anywhere
- Maximum value is specified in the configuration document
- Possible approaches
  - Verifier iterates through `i` values (not that slow but annoying)
  - Add syntax to carry `i` (kind of a gross special case)

Proposed Resolution: Verifier iterates (with regrets)

Pings while Joining (§9.4)

- Current procedure requires sending Pings to populate the table (step 2)
- These are unnecessary since Attach automatically discovers the right node

Proposed Resolution: Remove Pings as proposed on-list by BBL (Nov 1)

Join race condition I (Michael Chen)

- §9.4:
  - Step 7: routing table from AP → JP
  - Step 8: routing table from AP → NP
- In some cases (e.g., Chord predecessors) this may cause simultaneous connects between JP and it's new neighbors

Proposed Resolution: Tiebreaker when multiple connections are established between a pair of nodes. Smallest Node-Id seems like a natural choice.

Join Attach timing (Michael Chen)

- Proposal is to skip step 3 in which JP sends Attaches to its expected nodes.
- Argument for this is that the logic is simpler since no need to do incremental probing.
- Argument against is that it then takes longer to get fully established. Client has multiple ways to get AP's routing table which would allow unified logic for the neighbor set.

Proposed Resolution: Leave as-is but add discussion of the option to get AP's routing table rather than probe.