An Autonomic Control Plane

draft-ietf-anima-autonomic-control-plane-05/06

98th IETF, Mar 2017

Michael Behringer (editor), Toerless Eckert, Steinthor Bjarnasson

Changelog -04 to -05 – *grey = detail slides following*

Clarifications:

5.3 - ACP neighbor selection: Intent can override only AFTER initial default ACP connection.

5.8.1 - Addressing: ACP addresses permanent, no temporary addresses (RFC4941)

5.2.3 - pointing to GRASP objective defined in I-D.carpenter-anima-ani-objectives

5.9 - routing: prescriptive (aka: mandate RPL) (prior versions was "SHOULD")

6.1 - Connecting non-AN NOC: Eg: introduced term "ACP connect" as name for it.

6.2 - non-ANI-traversal: (from mcr) thoughts on traversing a L3 cloud automated.

10 - Security considerations: No protection inside ACP against source spoofing.

Various formatting changes as supposed by Mcr
https://mailarchive.ietf.org/arch/msg/anima/nZpEphrTqDCBdzsKMpaIn2gsIzI

Functional changes:

5.5 - CRL check - Added, open item.

5.8.2 - Addressing: MD5 -> SHA for addr calculation: MD5 obsolete

11. - IANA: Added address sub-scheme assignment (one value)

New discuss in -05:
ACP peer in CRL == certificate became invalid.

- Q: OCSP or CRL ?
  - Q: Does ACP doc need to make a choice ? Can this be per-device deployment choice or do all devices need to support a common scheme ?
- Q: Do we need to support (graceful) recovery ?
  - Eg: peer disconnected (ok). But also: Peer "lost" – admin adds it to CRL
  - Peer found again in some secure storage. Should be possible to reactivate.
  - Current working solution (in ACP spec): factory-reset peer, re-enroll.
  - Possible soft recovery (similar to what is done in VPN deployments):
    - Intent to ignore CRL entry
    - Revoked peer can build ACP
    - Intent shows troubled peer that it immediately need to do cert renewal (via ACP)

Open: Various

https://mailarchive.ietf.org/arch/msg/anima/ nZpEphrTqDCBdzsKMpaIn2gsIzI

Most suggestions included in text. Remaining ones:

1. More explicit text how to deal with admin down interfaces
   - A) If possible through implementation locally -> implementation detail
   - B) if requiring additional negotation: Extension draft/RFC
   - *Selection 7 (right place ?): It is desirable to continue being able to run ACP over interfaces that are administratively down or to inquire explicit operator approval upon actions that would administratively bring down an interface and the ACP running across it, especially if bringing down the ACP is known to disconnect the operator from he device*

2. Text still claims IPsec/GRE is incompatible with Ipsec (IP/IP)

   Note that without explicit negotiation (eg: via GRASP/TLS), this method is incompatible to direct ACP via IPsec, so it must only be used as an option during GRASP/TLS negotiation.

   Sentence will be removed if one more IPsec expert beside MCR chimes in.

   By default use IKEv2 to negotiate IPsec transport mode with next protocol equal 41 (IPv6).

   If initiator supports and prefers GRE, it should offer GRE (47), it should offer GRE,IPv6.

   Use GRE according to RFC7676 . Set MTU to link level MTU minus IPsec/GRE.

   Application via the ACP need to be built assuming the available MTU is not larger than 1280.

Open: Various

https://mailarchive.ietf.org/arch/msg/anima/ nZpEphrTqDCBdzsKMpaIn2gsIzI

3. Two stage ACP channel security selection – MCR does not like it
   - Try supported ACP mechanisms in parallel including GRASP/TLS as one option
   - If GRASP/TLS supported, its negotiation result selects channel security

Primary problem example:

Current target ANIMA device may all like IPsec/IKEv2

Constrained devices that like CoAP will not have TCP and will have dTLS

Gateway device (current target ANIMA device) need to support both options

Current proposed solution supports this.

Only gateway device needs to support two channel options

Implementation can be simplified by making CoAP instead of IPsec a per-interface or intent choice – no need to support both channels on same interface!

Open: Various

https://mailarchive.ietf.org/arch/msg/anima/nZpEphrTqDCBdzsKMpaIn2gsIzI

4. 5.5.3. ACP via dTLS

So, it's UDP and then... ? GRE inside UDP? (there is a draft tsvwg-gre-in-udp-encap-19)

- Goal was to use just IPv6 ACP packets over UDP
  - No additional intervening headers
- Ask was for constrained devices
  - They have CoAP and dTLS code basis

- Any reason why this simple solution should not work ?
  - UDP port number assignment assumed.

06: Added text to reaffirm that we do not use additional security negotiation such as in eg: OpenConnect VPN or the like.

Open: Various

https://mailarchive.ietf.org/arch/msg/anima/nZpEphrTqDCBdzsKMpaIn2gsIzI

5. 5.5.3. ACP security profile

- Several suggestions that would require more/longer work to finalize ?!.
  - Eg: there are lot of networks where constrained devices are midpoints in RPL, not only leafs.
- Consideration:
  - Existing text for “constrained devices” == like to do dTLS
    is an attempt to show how to modify ANIMA for a specific device class
  - If this approach MAY ME incomplete and require follow-on work, suggest to leave it in and do that work in followup draft when desired.
  - If this approach is so far off that it is more confusing, let’s remove all “constrained devices” text from ACP doc
  - If the thoughts are useful… move to appendix/summarize there as incomplete..

5.9 Open EDNOTE: RPL parameters / profile

Need to decide: storing / non-storing mode; mcr suggests storing mode. Need to define more parameters in detail

- Text added to 06. MCR suggesting to translaet into standard RPL profile definition in -07
- Suggestion from Pascal Thubert / Toerless:
  - RPL Mode of Operations (MOP): mode 3 “Storing Mode of Operations with multicast support”. Implementations should support also other modes. *Note: Root indicates mode in DIO flow.*
  - Objective Function (OF): Use OF0 [RFC6552]. No use of metric containers, Default RPLInstanceID = 0.
    - stretch_rank: none provided (“not stretched”).
    - rank_factor: Derived from link speed: <= 100Mbps: LOW_SPEED_FACTOR(5), else HIGH_SPEED_FACTOR(1)
  - Trickle: Not used; Data Path Validation: Not used
  - Proactive, aggressive DAO state maintenance:
    - Use K-flag in unsolicited DAO indicating change from previous information (to require DAO-ACK). Retry such DAO DAO-RETRIES(3) times with DAO-ACK_TIME_OUT(256ms) in between.
  - Administrative Preference (RFC6550, 3.2.6 – to become root) (toerless):
    - Indicated in DODAGPreference field of DIO message.
    - Explicit configured ”root” registrar: 0b100; Registrar (Default): 0b011; AN-connect (non registrar): 0b010; Default: 0b001.
- Extensibility:
  - RPL/Root (direct or via intent) may create additional RPL instances with other OF/metrics.

Old discuss from -03 (Berlin)

5.1.1 – Format of domain Certificate with ANIMA information

https://mailarchive.ietf.org/arch/msg/anima/nTgDaP4_-TKbQShmfb_BWu8kFp8

A) Text is in ACP draft (not bootstrap draft) because

No ACP specific text in bootstrap -> easier to reuse in non ANIMA solutions
Easier to read ACP (without having to follow references)
Thought this was agreed with bootstrap team ?!

B) (strange but “clever”?!) formatting choosen because…

Example: anima.acp+<acp-addr>@<acp-domain> subjectAltname RFC822format
Existing subjectAlname type = works with any (badly implemented) ASN.1 parser
Tagged: “anima.acp” – immediately possible to distinguish from other subjectAltnames
RFC822format allows to carry both <addr> AND <acp-domain> in one subjectAltname
Even if address is mistaken by some system, worst case an email gets sent to address
- And ACP administrator could catch those -> set up email anima.acp@<acp-domain>
- RFC822 mailbox behavior: extensions after + are part of preceding mailbox name.

06: added more detailed version of above explanations.

Candidate TBD

- Do we like document structure:
  - I-D.carpenter-anima-ani-objectives to be a dependency of ACP and other drafts ?
    - If we move ACP objectives into ACP draft, appropriate to add co-authors ?

- TODO: Merge text into ACP text
  - Section 5.2.3
  - See BRSKY draft 3.1.1 to compare

11 – security considerations, source spoofing

- If malicious device get access to ACP -> spoofing possible
  - Sounds bad, but its not…
- Intentionally no further protection to keep solution lightweight:
  - Hop-by-hop ACP security as first layer of protection for protocols running inside of it (GRASP, RPL, any “user/operator” protocols: NTP, SNMP, DNS, …)
- This is not worse than todays IGP routing (even with security)
  - For bidirectional traffic you need to also spoof routes or be near the root
  - Spoofing routes as difficult as hacking IGP with (cert) security
- Full Solution: ACP-cert authenticated RPL messages (transitive)
  - Eg: Similar to BGP… Never done intradomain... ETOOMUCHTROUBLE ?!
- How to best state this in security section ?!

Next steps

- Close what we can in Chicago
- Candidate last call version before next IETF.