

Guidelines for Autonomic Service Agents

<https://datatracker.ietf.org/doc/draft-ietf-anima-asa-guidelines>

Brian Carpenter

Laurent Ciavaglia

Sheng Jiang

Pierre Peloso

IETF 110 – ANIMA WG

draft-ietf-anima-asa-guidelines-00

- Adopted as WG document (Nov 2020)
- Document shepherd: Toerless Eckert
- Diff from draft-carpenter-anima-asa-guidelines-09
 - Only editorial fixes: updating references

draft-ietf-anima-asa-guidelines-00

- Detailed review from Toerless E.
 1. Introduction
 - Suggestions for clarification, consider example (later in the doc)
 2. Logical Structure of an Autonomic Service Agent
 - Suggestions for clarification, diverse approach for base design of ASA
 3. Interaction with the Autonomic Networking Infrastructure
 - Discussion on security bootstrapping
 - Suggestions for clarification, consider examples
 - Request to update text on policy/intent
 4. Interaction with Non-Autonomic Components
 - Suggestion to move part of the section to security considerations
 5. Design of GRASP Objectives
 - Suggestions for improvement on GRASP, Dry Run mode, JSON/CBOR, NETCONF/YANG considerations

draft-ietf-anima-asa-guidelines-00

- Detailed review from Toerless E. (bis)

- 6. Life Cycle

- Suggestions to fix/revise use of requirements wordings, missing definitions, use of term “objectives” -> GRASP objectives, and various text improvements

- 7. Not commented (but will need to be fixed anyway)

- 8. Coordination with Traditional Management Functions

- Suggestions for text improvements/new section

- 9. Robustness

- Useful observation ASA robustness design

- 10. Security Considerations

- Suggestions for text improvements and clarifications on “Authorization”

- 13. References

- Need to fix normative/informative references

draft-ietf-anima-asa-guidelines-00

- Feedback from ASA life cycle proof-of-concept, Brian C. [1]
 - ... "ASA loader" that can load (and unload) multiple ASAs onto the same GRASP instance.
- **Conclusion: the only new feature needed**, as far as the ASA programmer is concerned, is one function that must be **called regularly in the main loop**. I called it checkrun(), and **if it returns True the ASA may continue**, and if it returns **False, the ASA must exit gracefully** (i.e. conclude all current actions first).

[1]: <https://mailarchive.ietf.org/arch/msg/anima/u1TCJQ0PpOJSVs9bdqr5akwA25w/>

Objectives for version -01

- **Address review comments**
 - BTW, additional reviews and feedback are most welcome
- **Refine content**
 - Document has been quite stable, even before WG adoption
 - Few request for technical change expressed/received
- **Set good foundations for future work in ANIMA WG (“ASA ecosystem”)**
 - Link to use cases and applications, detailing examples of different AFs and ASAs
 - Design, implementation and deployment considerations
- **Target: end of March 2021**