Impact of AAA Proxies on Security

Katrin Hoeper, NIST

Objectives

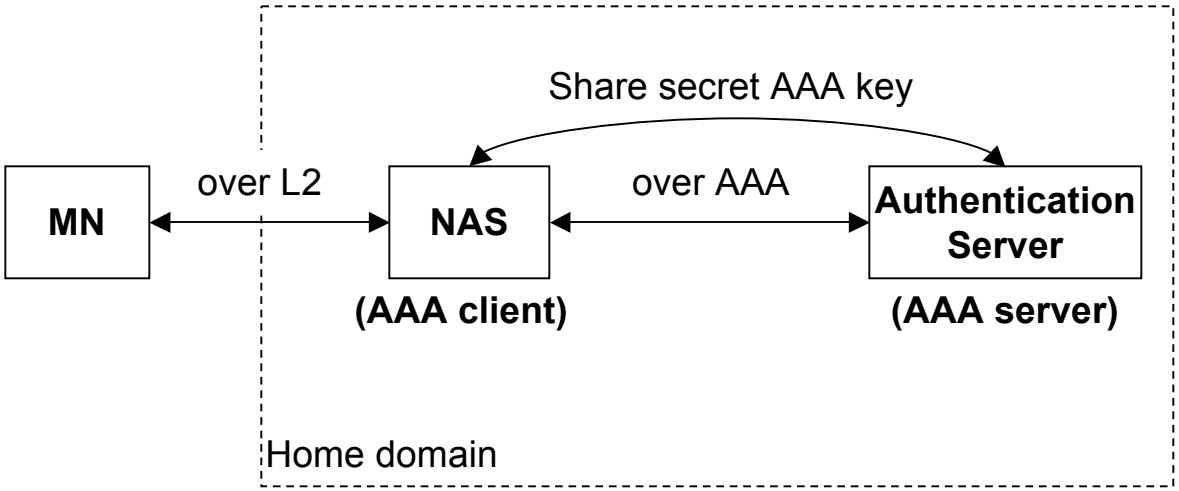
- Initiate work on a general solution to address the proxy problem that can be added as extension to “affected” drafts
  - ⇒Tackle the issue which is slowing down progress of any draft that is subject to “proxy problem”
  - ⇒Ensure secure implementations of existing & future schemes in networks with proxies

Outline

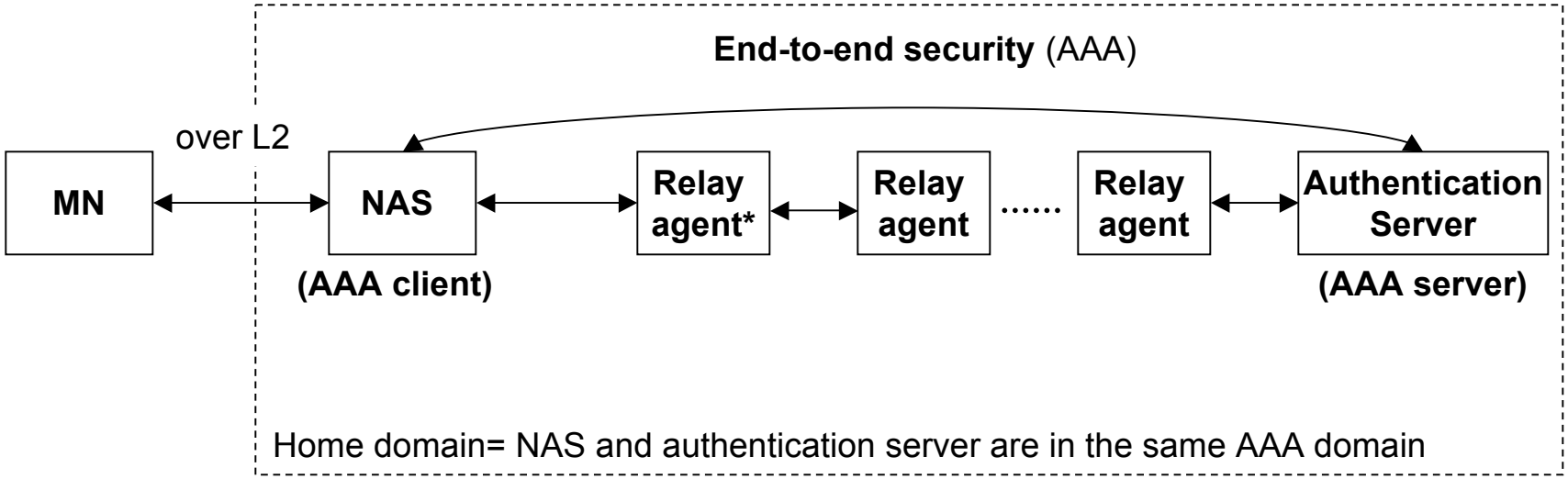
- Check communication model used in many drafts and study problems that may arise when introducing proxies
  - summarize trust model without proxies
  - define “proxy problem”
  - derive necessary security requirements to preserve trust model even with proxies

Traditional Communication Model

- 3-party model
  - with MN, NAS and authentication server
  - AAA protocols used for NAS ↔ server communications, i.e. NAS and server share a secret AAA key
- All parties are in the same administrative domain
  - Home domain = one AAA domain



This includes…



*[RFC 3588] Relay agents do not make policy decisions and neither exam nor alter non-routing AAA attributes. Hence, relay agents do not need to understand semantics of the messages they forward. In the remainder, we consider relay agents as part of the links since they do not affect the security.

Trust Model (w/o proxies)

Trust relationships	**Established by**
MN ↔ server[1]	-mutual authentication (e.g. EAP)
NAS ↔ server (AAA client ↔ AAA server)	-mutual authentication -authorization NAS -[verify NAS advertized info (e.g. channel binding)]*
MN ↔ NAS	-HS using established keys (e.g. IEEE 802.11i)

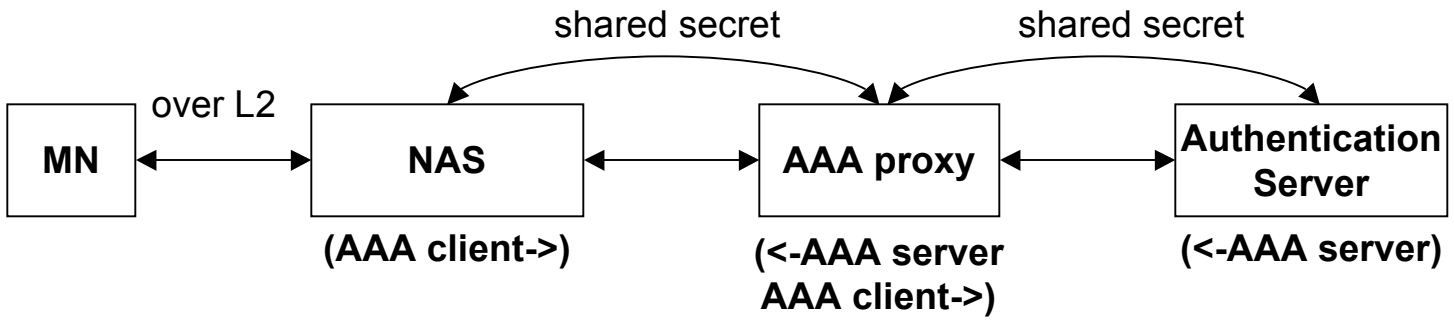
Assumptions

[1]MN trusts home server, e.g. to provide proper accounting and distribute keys only to successfully authenticated and authorized NAS

*Here: lying NAS problem out of scope

Communication Model with Proxies

- 4-party model
  - with MN, NAS, proxy and authentication server
  - a proxy appears as a server to its client and as a client to the upstream server
  - AAA protocols used for NAS ↔ AAA proxy as well as AAA proxy ↔ server communications
  - pairwise secret AAA keys ⇒ hop-by-hop security



Why Proxies?

- Scalability (enterprise scenario)
  - easier management of network access in large networks
- Mobility (service provider scenario)
  - proxies enable roaming: MN can access other networks by authenticating to its home server through local proxies

Why Do Proxies Need Keys?

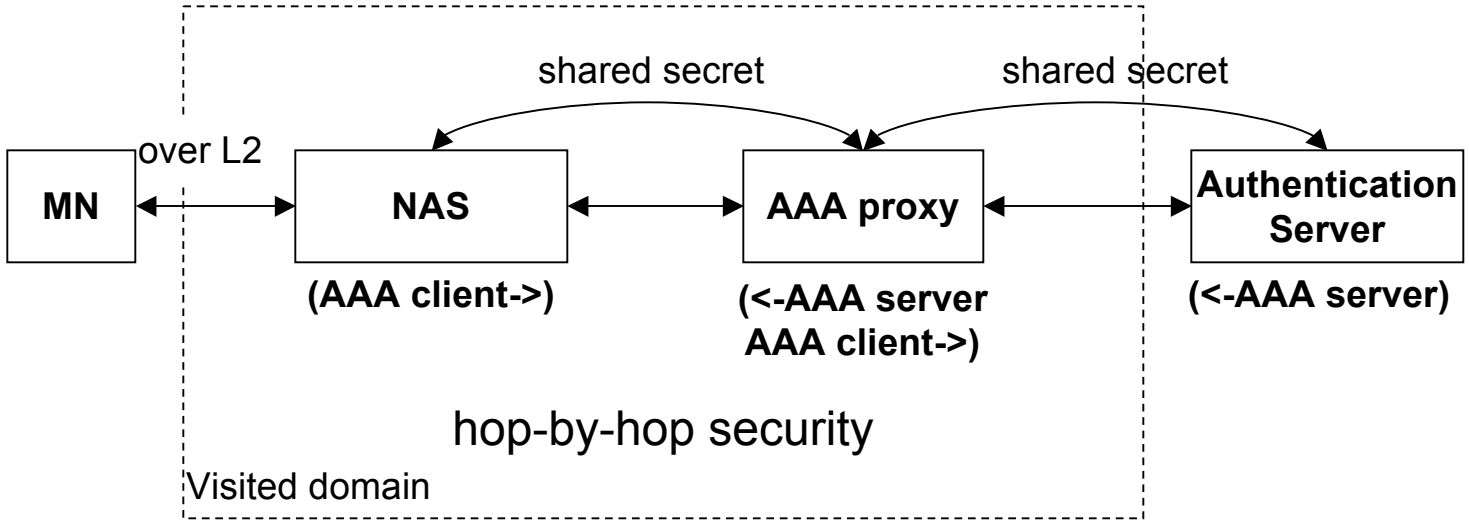
- Proxies make policy decisions relating to resource usage and provisioning, because NAS and server cannot directly negotiate,
  - adjust to locally offered services/capabilities
  - comply with local policies
- Therefore, proxies need to be able to
  - generate reject messages when policy was violated
  - understand semantics of messages
  - be able to modify attributes

What's the Problem?

- Proxies break the trust relationship between NAS and the authentication server in the trust model w/o proxies
  - No longer end-to-end security, instead hop-by-hop security
- Keys are shared among server and proxies
  - intentional or unintentional misconfigurations may lead to security vulnerabilities
  - even less controllable if keys are shared across domains
- Possible attacks by proxies include
  - *fraudulent accounting*, e.g. charge extra roaming fees
  - *attribute modifications*, e.g. downgrade ciphersuites
  - *replay attacks,* .e.g. replay passwords
  - *data theft*, e.g. steal passwords or accounting data

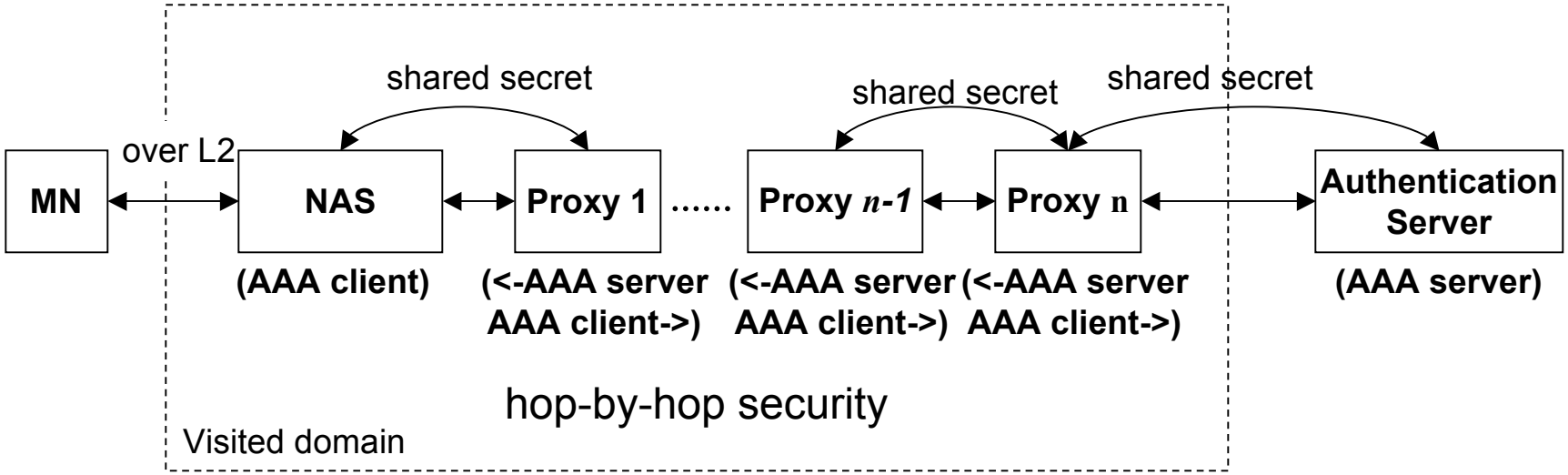
Use Case: Roaming, single proxy

- Service provider scenario
  - NAS and local proxy both belong to visited access network
  - server shares pairwise keys with proxies in visited domains with roaming agreements ⇒ server can validate authentication and authorization of proxy
  - established trust may imply that local proxy validates authentication and authorization of local NAS



Use Case: Roaming, proxy chain

- In visited domain
  - server can only validate last proxy
  - NAS and proxies 1 to n-1 not directly validated by server, no trust relationships established
  - difficult to trust large number of intermediate proxies in visited domain
- Note: server can't distinguish between single proxy or proxy chain!
  ⇒ any proxy scenario requires special protection!



Security Requirements (?)

- To preserve trust model, the end-to-end security feature between NAS ↔ server must be preserved
  - Can this be done by AAA protocols that provide end-to-end security for at least some of the attributes exchanged between NAS and server?
  - Does the scenario get more complicated in multiple domains networks?
    - Same as single domain if server is able to authenticate and authorize all NASes in every domain with roaming agreements
    - Otherwise(?): hop-by-hop security between proxy *n* ↔ home server and end-to-end security between NAS ↔ proxy *n*
  - Does anything change if proxy n is in the home domain?

Design Constraints*

- How much additional code will a solution add to a AAA server, proxy or NAS?
- What will the performance impact be?
- Is incremental deployment possible?
  - For example, can a single operator role the solution out on a subset of NAS devices, or do multiple operators have to support the solution in order for it to work?
- What is the operational impact of the solution?

* from Bernard's comments

Possible Approaches (?)

- Some trust assumptions seem unavoidable
  - the home server acts as anchor of trust
  - trust in proxies that directly communicate with the server (one-hop)
- First steps towards a solution
  - derive trust model for communication models with proxies incl. a set of general (security) requirements
  - then show how requirements could be achieved
  - this may make modifying *each* existing scheme redundant and help designing future methods

Existing Work

- <draft-ietf-aaa-diameter-cms-sec-04>, expired
- <draft-kaushik-radius-sec-ext-06>, expired

- If end-to-end AAA is the only requirements, are we done already?
- Why not deployed?

Questions?

Volunteers?