(Preliminary) Gap Analysis

<draft-tschofenig-ace-overview>

Hannes Tschofenig

Goal of this Presentation

- The IETF has developed a number of security technologies that are applicable to the presented use cases.
- Is there possibility for re-use?
- Go through a few selected technologies to identify gaps.

Non-Goals

- Design the solution in this room.

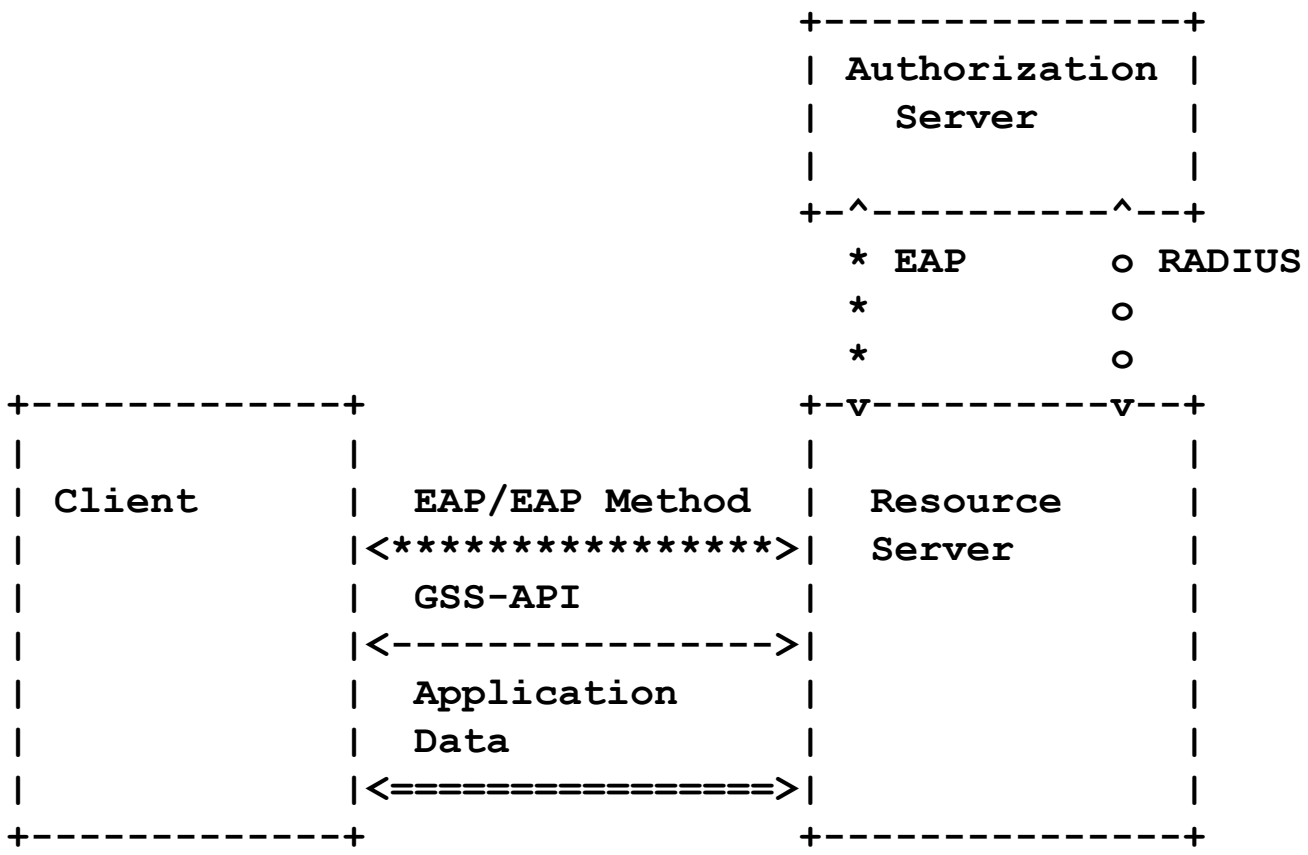
→ Don’t get hung up on the details.

Tutorials

- Kerberos
  - Slides
  - Recording
- OAuth
  - Slides
  - Recording
- "PKI/Certificate Model"
  - Slides
  - Recording
- AAA
  - Slides: http://www.ietf.org/edu/tutorials/IETF89-Tutorial-AAA.pdf

Note: .arf files are Webex recordings. You might need to use a Webex player. See http://www.webex.com/play-webex-recording.html

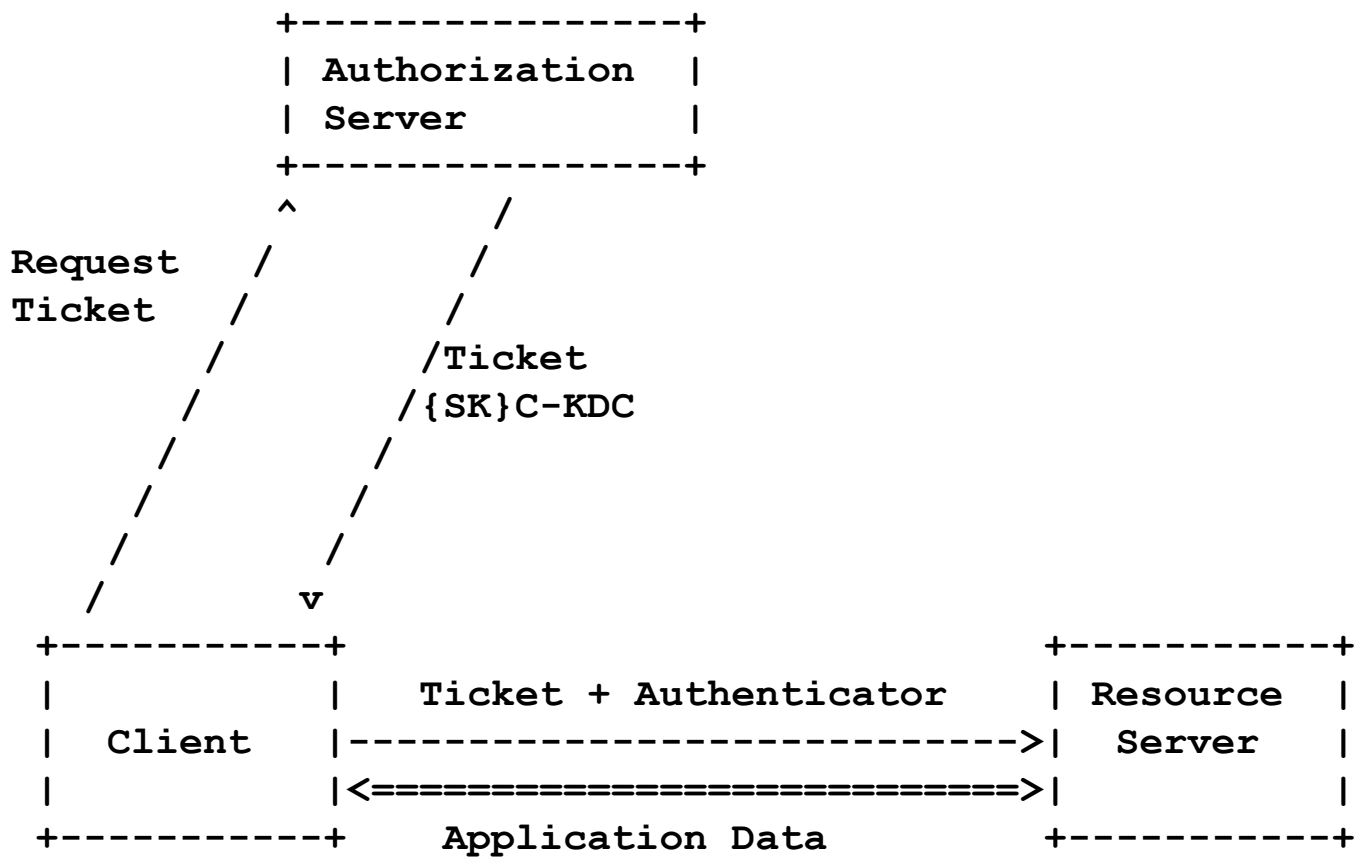
ABFAB


Authorization Server
* EAP
o RADIUS
Client
EAP/EAP Method
GSS-API
Application Data
Resource Server

Gaps

- Real-time interaction between the AAA server and the resource server.
- ABFAB architecture uses layering of EAP within the GSS-API, which adds additional overhead.
- A binding for the transport of EAP payloads in CoAP, for example, does not exist.
- No unified authorization policy language has been defined for the AAA/EAP architecture. Instead, RADIUS attributes carry information about access control decisions.

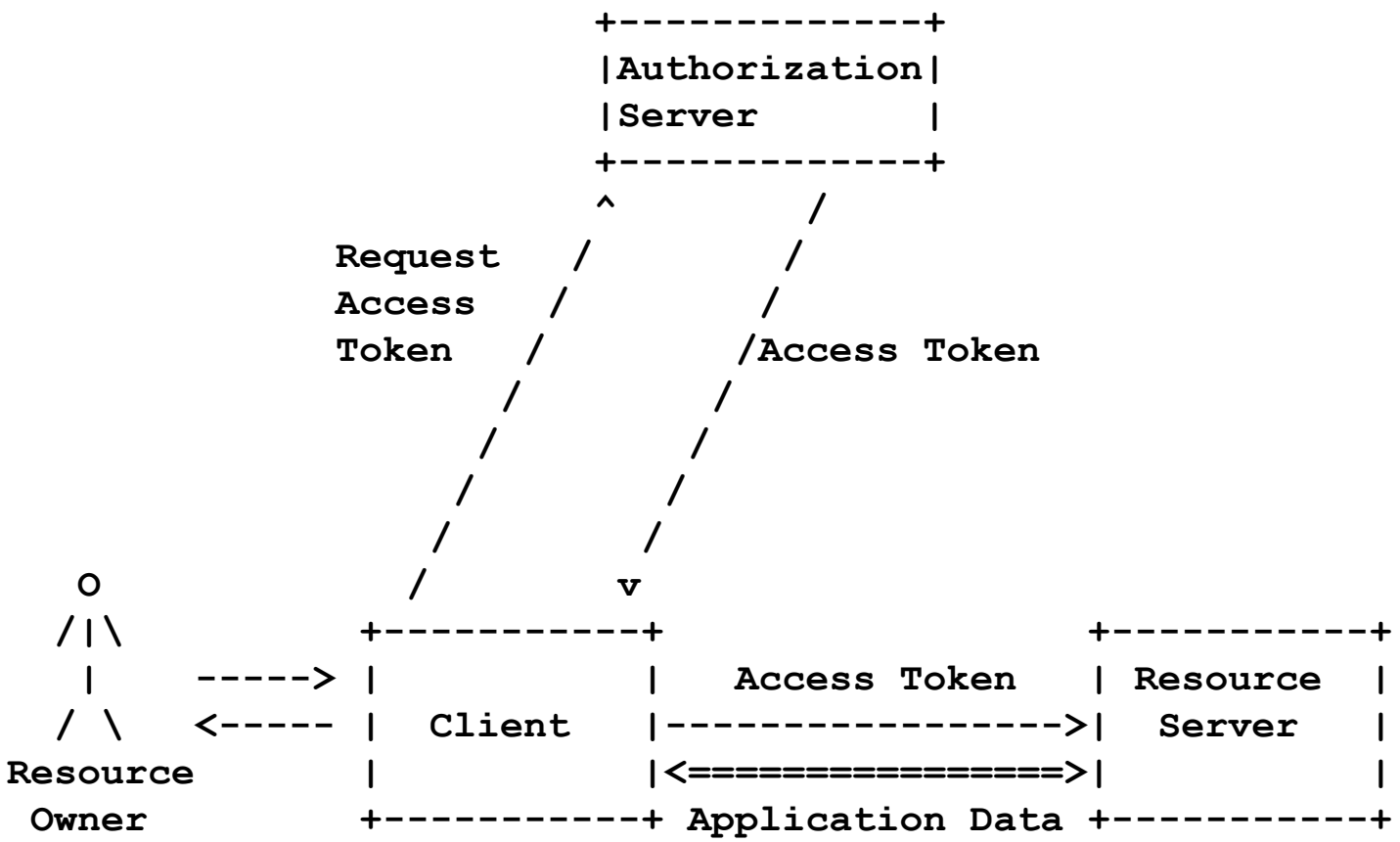
Kerberos


Authorization
Server
Request
Ticket
Ticket
{SK}C-KDC
Client
Ticket + Authenticator
Resource
Server
Application Data

Gaps

- Each ticket is only usable for a single service (intentionally).
- Kerberos uses ASN.1 for encoding of the ticket and various messages.
- No access control policy language has been standardized.
  - Standardization in KITTEN in progress.
  - Proprietary policies are, however, used in real-world deployments.
- A CoAP binding for the KRB_PRIV and the KRB_SAFE message exchanges not been defined.
- Ticket and Authenticator rely on symmetric key only.

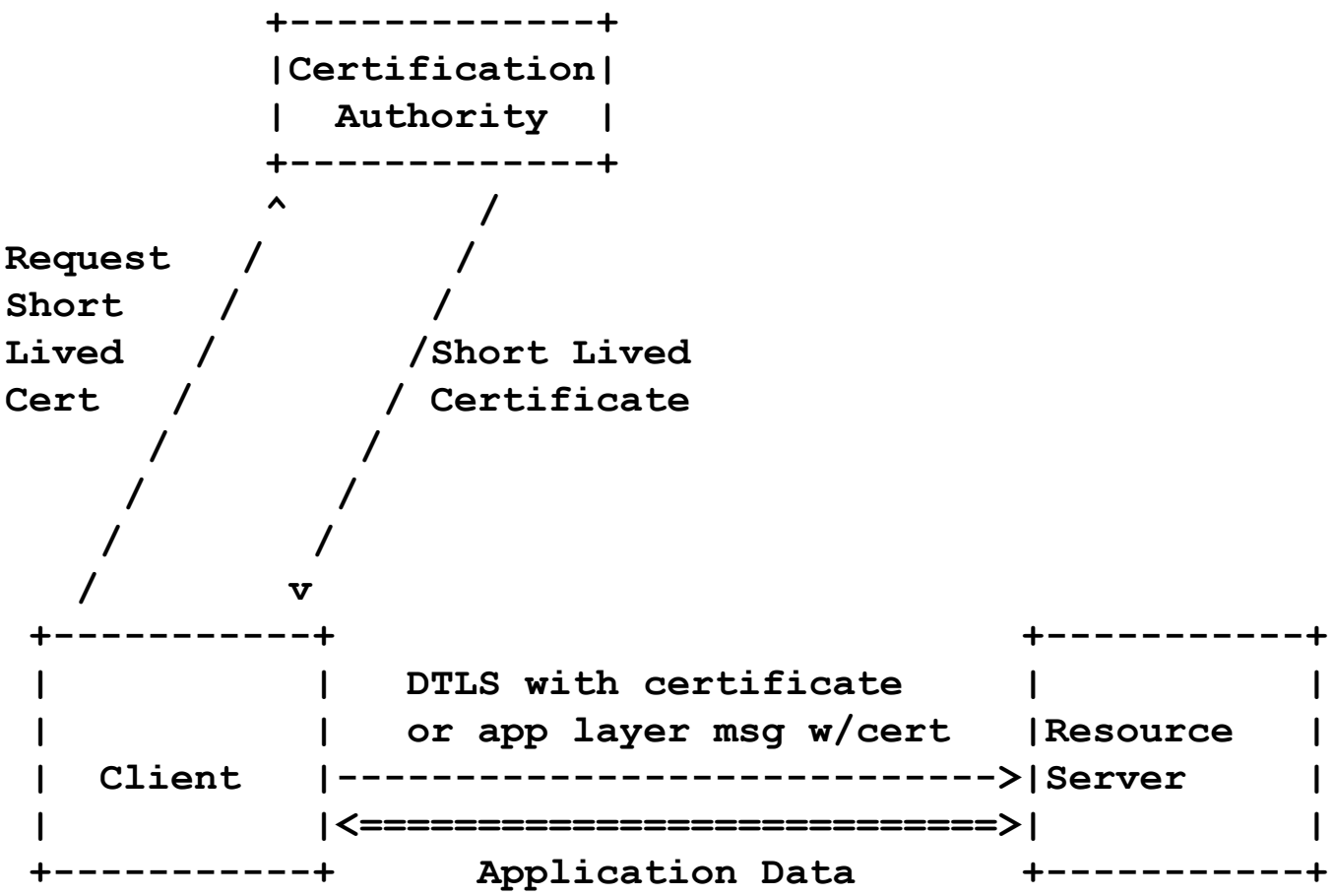
OAuth


+-------------+
|Authorization|
|Server |
+-------------+
Request
Access
Token
Access Token
O
/|\
/ \
Resource
Owner
-----> <-----
+-----------+
| |
| Client |
| |
+-----------+
Access Token
Application Data
+-----------+
| Resource |
| Server |
| |
+-----------+

Gaps

- Support for cross-realm interaction has not been standardized.
- A binding for CoAP does not exist for the client to authorization server nor for the client to resource server.
- The OAuth architecture does not standardize the authentication procedure of the resource owner to the authorization server itself.
- Profile is needed to navigate through the options (since OAuth provides a lot of flexibility).
- CoAP/DTLS bindings currently not defined.

“PKI/Certificate Model”



Gaps

- The certificate format and the PKI management protocols use ASN.1.
- No UDP or CoAP transport is defined for CMC/ CMP/SCEP. For PKCS#10 no transport is defined at all.
- Asymmetric cryptography is computationally more expensive than symmetric cryptography but offers additional security benefits.

Conclusion

- … need to agree on the requirements first.
  - There may also be other relevant security technologies as well.
- Our preliminary analysis makes us believe that some work is needed to get these security protocols to work on constrained devices.
- Need a venue to have a dialog.