IOT BZH

Token Logic Access Control

Purpose of token logic shift

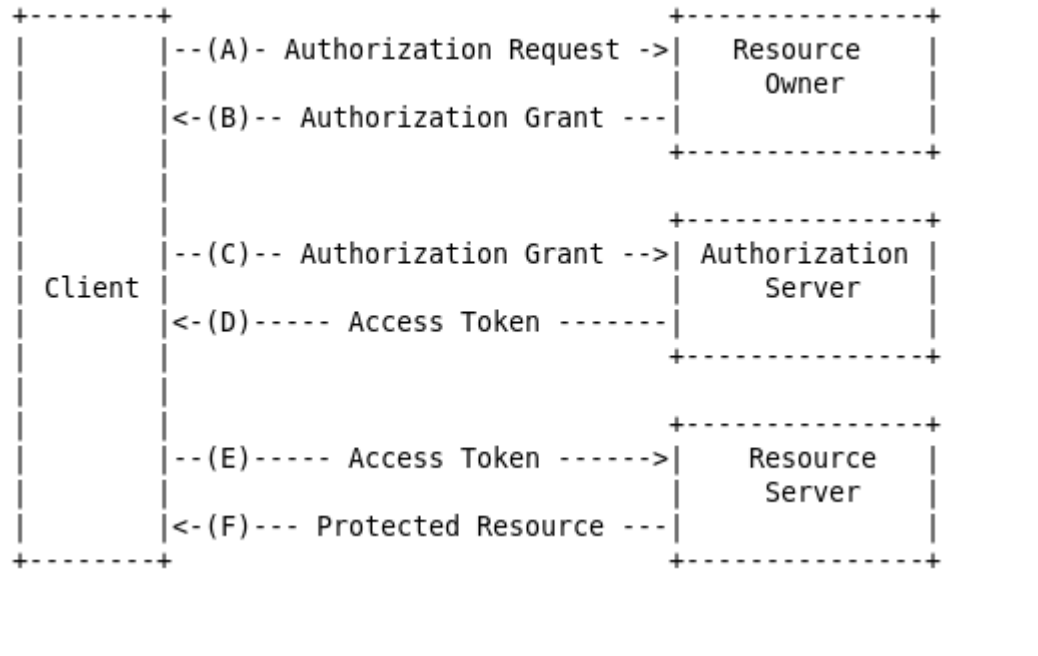
- Standardization of resource access control
  - Using open web standards: OpenId Connect, OAuth2
- Opening of AGL to external
  - Allowing access to car from smartphone or tablet
- Secure integration of scattered components
  - Single security mechanism for all ECUs of the car

Basis of token logic access control

- Secure connection
  - Exchanges between the client and the resource can’t be spied
  - Ex: TLS, IPSEC
- Bearer token
  - A text that can not be guessed or forged by malicious.
- Principle
  - A client that access a resource produces it a token proving its access rights. The resource checks the validity of the token. If the token is valid access to the resource is granted, or, otherwise, if invalid, forbidden.

OAuth2 access control

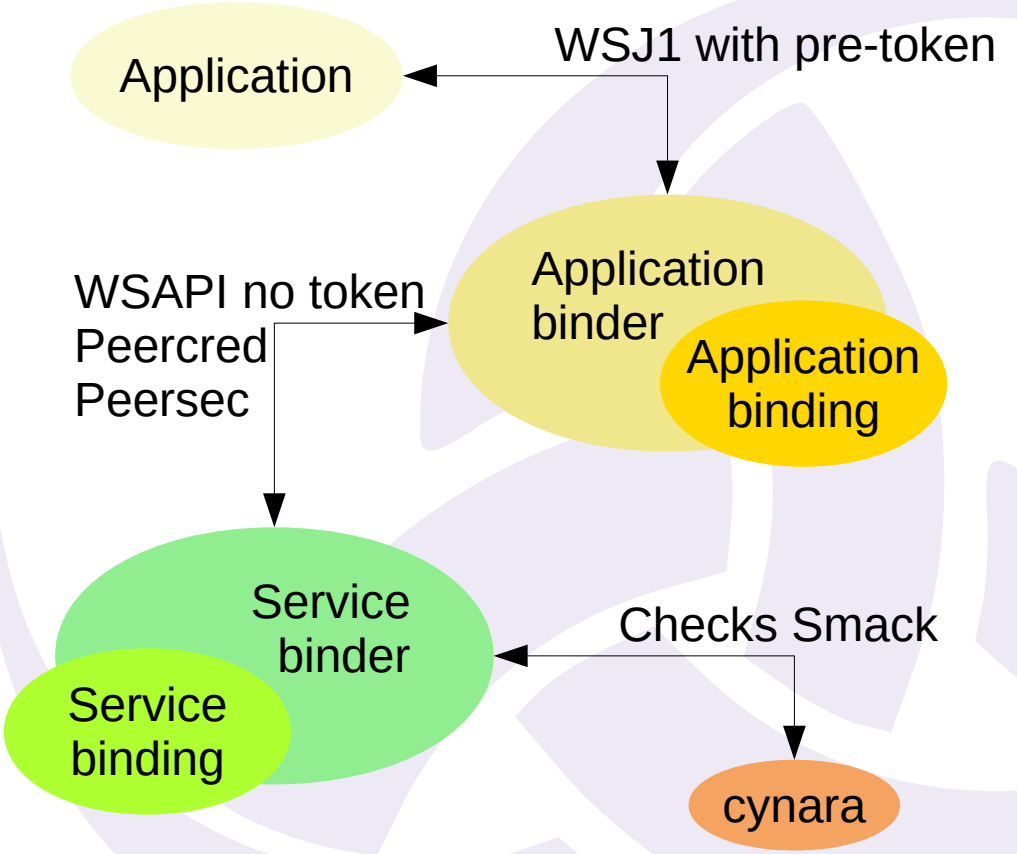
- RFC6749 normalizes the protocol flow
- RFC6750 normalizes the access token usage as bearer
- RFC7519 normalizes a rich signed and self-describing JSON Web Token



OAuth2 abstract protocol flow

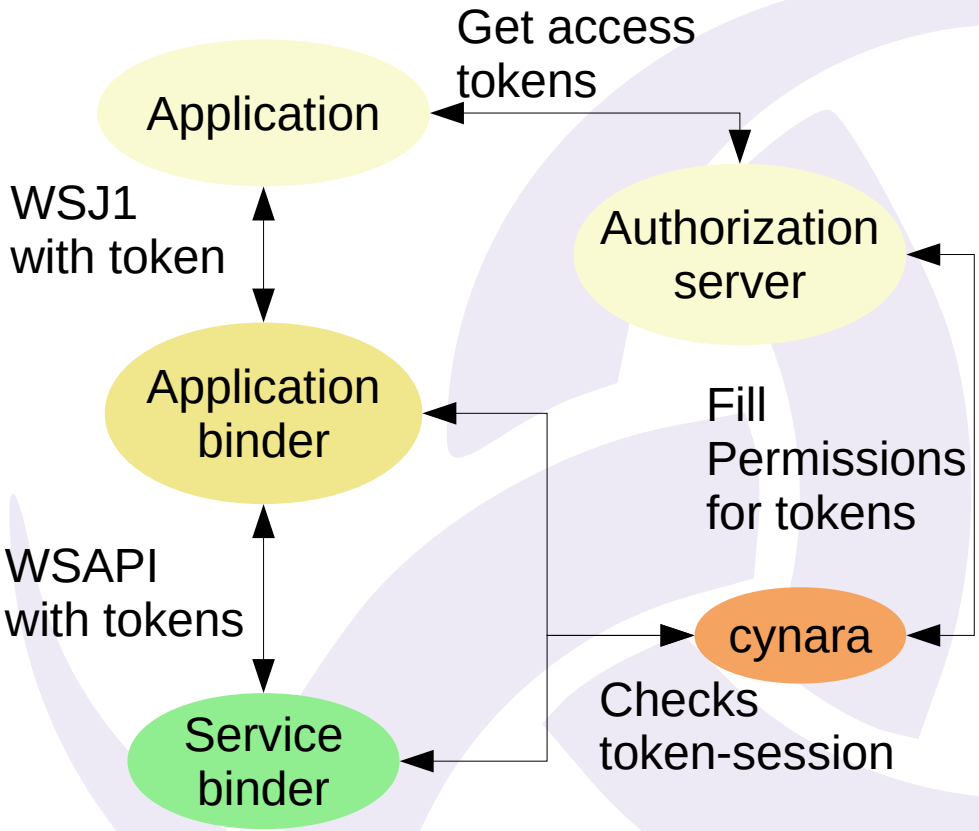
Current state

- Bindings can require a token to be checked verb by verb. This allows fine resource control: a resource can expose public and protected API.
- Binders require token for WSJ1 connections.
- Tokens are created by binders, each its own. Only the binder knows its initial secret token.
- WSAPI connections do not transmit tokens but relies on Smack and cynara to check its client rights.

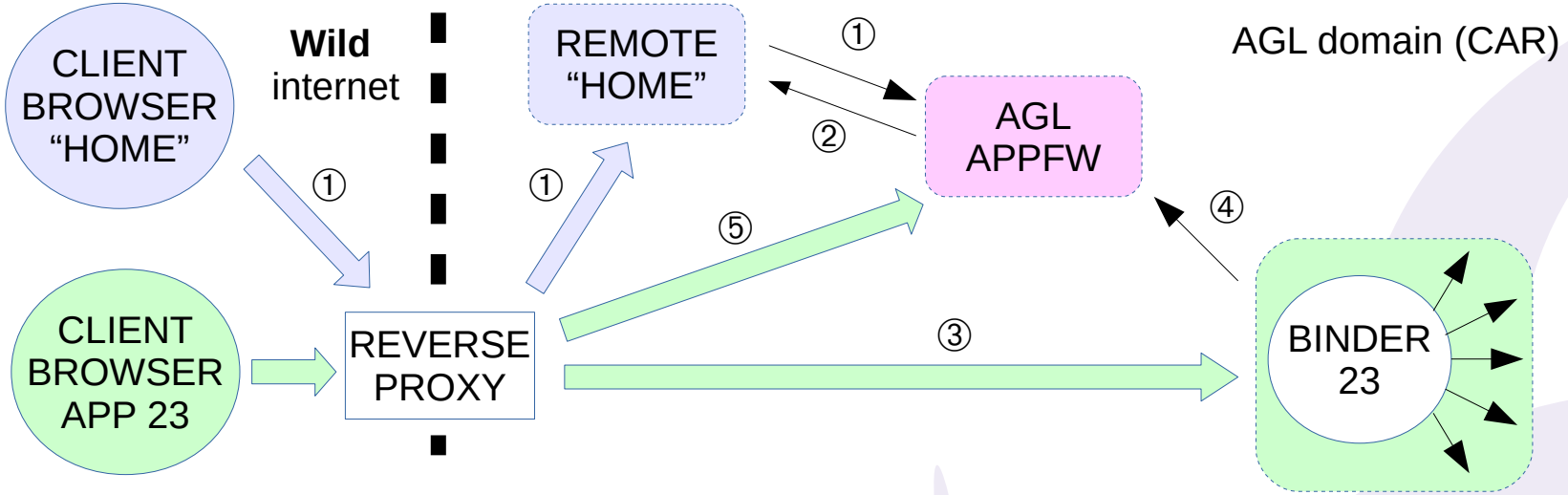


Expected state

- An authorization server forge access tokens.
- Applications and services gain tokens for accessing resources.
- Resources and binders check the access token of their clients.
- The authorization server uses cynara as backend to check access tokens.
- Smack labels can still be used when meaningful.

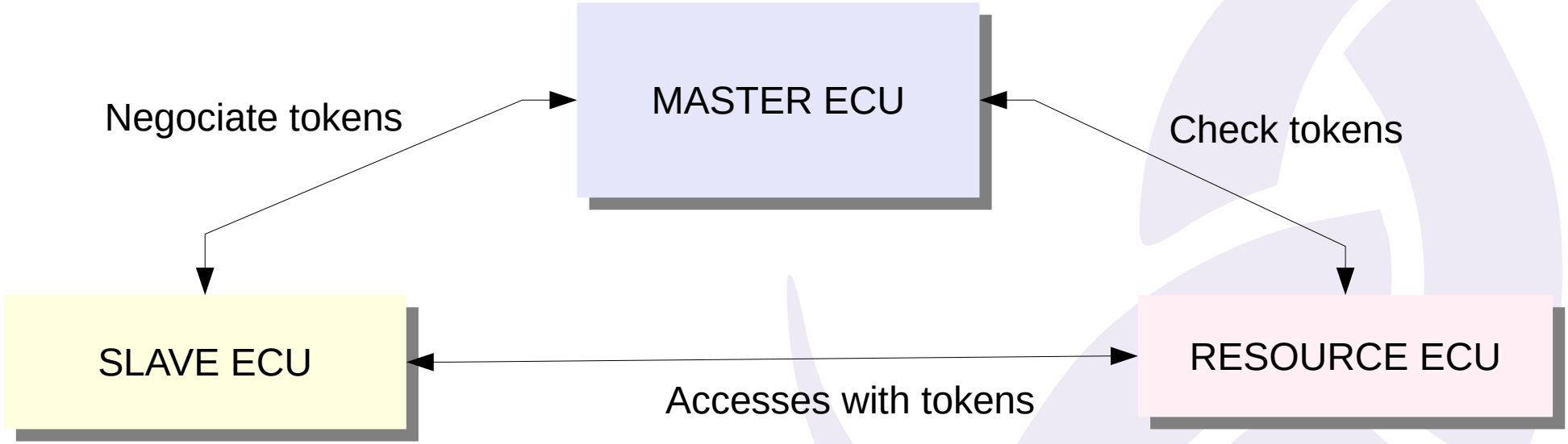


Use case: External App



(1) Ask to launch APP 23
(2) Returns the URL of the BINDER for APP 23
(3) Connect to the binder with its token
(4) Check token validity
(5) Negociate a token

Use case : Multi ECU



Limitation of token logic

- System resources are not bound to token logic today. It means that token logic can not protect accesses to files and processes. It implies that the token logic has to be completed by usual access control mechanism: DAC, discretionary access control (User, Group, ACL, …) and MAC, mandatory access control (Smack, SELinux, …)
- Connections have to be secured, implies cryptographic overload
- Asking resource owner might sometime be difficult

Work to be done

- Update the binder to the new token logic
- Shift to new cynara
- Make use of TLS
- Missing component
  - The authorization server and its user interface
  - The reverse proxy
- Split applications in two autonomous services: the app and the binder
- ...

Q&A

Gulf of Morbihan, south of Brittany, France

Links

- IoT.bzh:
  - Website: https://iot.bzh/
  - Publications: https://iot.bzh/en/publications
  - Github: https://github.com/iotbzh
- AGL:
  - Website: https://www.automotivelinux.org/
  - Documentation: http://docs.automotivelinux.org/
  - Sources: https://git.automotivelinux.org/