eap-types

choices

The decision which EAP type(s) to deploy on your eduroam IdP depends on several factors:

- Capabilities of your Identity management backend
- Types of devices you want to support

Choices depending on the Identity Management System

Regarding the identity management backend, the most fundamental differentiation between EAP types is the type of credential they support.

- Does your identity management backend support X.509 Client Certificates? Then you can use EAP-TLS.
- Does your identity management backend use username/password combinations?
  - Does it store the passwords as either clear text - or - encrypted as NT-Hash? Then you can use EAP-TTLS, PEAP, EAP-FAST, EAP-PWD and more.
  - Does it store the passwords in a different crypt format? Then you can use EAP-TTLS only.

As you see, the decision is largely dependent on your identity management system; so your choices may be limited. As a more concrete advice for some IdM backends:

- Microsoft ActiveDirectory: stores passwords as NT-Hashes.

Anonymous outer identities

Almost all EAP types support the use of anonymous outer identities. The primary use of anonymous outer identities is for better preservation of privacy for your users; a properly configured supplicant will then not even reveal the real username of the user to the visited eduroam SP; instead, the username is replaced with a dummy value.

This feature needs protocol support by the EAP type in question; the basic idea is that there have to be two stages of communicating the client identity:

- one identity, the outer identity, is used to route the user's login request from the eduroam SP via the eduroam RADIUS path to the eduroam IdP
- the second, "inner" identity, is only revealed inside a cryptographically protected tunnel to the IdP

Since the outer identity is only needed for routing purposes towards the IdP, the local username part does not have to be accurate and can be obfuscated. The IETF-suggested way of obfuscating the username is to leave it empty; but it can just as well be replaced with "anonymous", "anon" or similar. However, the realm part (i.e. behind the @ sign) always needs to be accurate because it contains the routing information.

The inner identity always needs to be fully accurate, because it is used to authenticate the user. It does not necessarily have to contain an @ sign at all, because that username is local to the IdP and is only seen and evaluated there.

Example:

- Outer identity: anonymous@restena.lu
- Inner identity: stefan.winter

For eduroam request routing, the part @restena.lu of the outer identity is used to route the request to the restena.lu realm and to establish a secure tunnel; while the real username inside this tunnel which is looked up in a user database is "stefan.winter".

Here is a break-down of anonymous outer identity support for some popular EAP types:

EAP-Type	Support for anonymous outer identites
EAP-TTLS	yes
PEAP	yes
EAP-FAST	yes
EAP-TLS	support in protocol, but not typically available in supplicants
EAP-PWD	no

If the EAP type allows for the use of outer identities, it is a client device configuration option to either make use of them or not; there is little you as an IdP can do to force the use of anonymous outer identities (except for providing and encouraging the use of pre-configured installers which will then make all the necessary settings on the client device automatically).

Choices depending on the envisaged devices

The landscape of wireless-enabled devices is rather heterogenous, and support for EAP types varies. Ideally, you should survey which types of devices you should come to expect among your user base, check the capabilities of these devices, and make an informed decision regarding the EAP type of choice.

However, the EAP protocol is flexible enough to handle multiple EAP types: if your IdM backend can support the use of multiple EAP types, then you can configure all the supported EAP types. In that case, you have to select a "default" EAP type - it should be set to the EAP type with the broadest support in your client base.

Now, assuming you have the option of configuring a range of EAP types *and* your clients support that same range, which of these types should you prefer?

- We suggest the use of PEAP over EAP-TTLS for it does a mild amount of protection of the user password inside the secure tunnel.
- If you cannot support PEAP, consider to allow TTLS-PAP **and** the more unusual variant TTLS-GTC (initially Generic Token Card; also used for passwords which are not savable on the client device). Some older devices (certain Symbian OS builds) support TTLS, but not PAP inside. Enabling TTLS-GTC will allow these devices to connect.