08.09.06

Deliverable DJ5.1.4: Inter-NREN Roaming Architecture: Description and Development Items

Deliverable DJ5.1.4

Contractual Date:	28/02/06
Actual Date:	08/09/06
Contract Number:	511082
Instrument type:	Integrated Infrastructure Initiative (I3)
Activity:	JRA5
Work Item:	1 – Roaming
Nature of Deliverable:	R (Report)
Dissemination Level	PU (Public)
Lead Partner	SURFnet
Document Code	GN2-06-137v5

Authors: K. Wierenga (SURFnet, main editor), S. Winter (RESTENA, main editor), R. Arends (Telematica Instituut), R. Castro (RedIRIS), P. Dekkers (SURFnet), H. Eertink (Telematica Instituut), L. Guido (FCCN), J. Leira (UNINETT), M. Linden (CSC), M. Milinovic (SRCE), R. Papez (ARNES), A. Peddemors (Telematica Instituut), R. Poortinga (Telematica Instituut), J. Rauschenbach (DFN) , D. Simonsen (UNI-C), M. Sova (CESNET), Manuela Stanica (DFN), and with contributions from other GN2 JRA5 group members

Abstract: This document describes the current eduroam architecture, the motivation for technical changes and the changes that are feasible, using state-of-the-art software, today.

Table of Contents

Table of Figures

0 Executive Summary

The JRA5 Roaming deliverables include so far the Glossary of Terms (DJ5.1.1), the Roaming Requirements Specification (DJ5.1.2), the legislation overview and policy document (DJ5.1.3,1 and DJ5.1.3,2). This deliverable is the first comprehensive technical document covering the architecture and further development items, based on the experimental work done in year 2. Two other deliverables are under development in year 2: the roaming infrastructure and service support description (DJ5.1.5) and the Transition to Service document (DJ5.0.1).

GÉANT2 JRA5 has adopted the current eduroam pilot infrastructure as its experimental platform, so that JRA5 participants and other interested parties could join easily. In parallel to the experimental usage the infrastructure provides the opportunity to test new developments under production conditions. Both the collected practical experience and the analysis of the current infrastructure show that the architecture in place provides a usable model, but could and should be improved in several points. This deliverable explains in the first part the technical model of the operational eduroam pilot, followed by a list of potentially possible alternatives, the evaluation of these alternatives and the result of the technical analysis.

One of the characteristic parts of the current eduroam is the strict tree-like architecture introduced by the usage of the RADIUS [RFC2865] hierarchy. The main drawbacks of this approach are a decrease of security and a suboptimal communication and transport model, which have been the main drivers in the process of searching for a better technical solution.

A number of alternatives have been investigated. As this document outlines, none of the available alternatives provide a complete and fully satisfying solution. The only way forward based on the results of this deliverable is to start with the stepwise introduction of new features, without breaking the usability of the pilot. In respect to IETF standards documents, Diameter would be best suited to the roaming requirements, but the implementations available are functionally limited or not stable enough. The technology tested and evaluated as the most feasible for now was developed by the vendor “Open Systems Consultants” [OSC] for their “Radiator” product, called RadSec [RadSec]. While it supports a more secure and reliable architecture, it is not yet standardised and not yet available on all RADIUS platforms used in eduroam. A next step of significant importance for eduroam will thus be issuing an Internet Draft to support the standardisation process of this technology. On the practical side the downward compatibility will be ensured, and by keeping the hierarchical solution RadSec can be used in parallel mostly on the confederation level, where a relatively high number of Radiator installations are present.

1 Introduction

During the project life time a number of JRA5 partners have joined eduroam. A few of them are still preparing the set-up of (at least) a small national infrastructure as a pre-condition for participating in the project (e.g. HEAnet, SUNET), while other NRENs that were not partners of JRA5 have already joined. One of the obstacles so far has been the lack of a confederation policy, that is now available in the deliverable [DJ5.1.3,2] “Roaming policy and legal framework document Part 2: Policy document”. As can be seen on the eduroam web site www.eduroam.org, by now 23 countries with about 500 connected institutions are already included in eduroam. Three important objectives of JRA5 in the next phase are: 1) getting every JRA5 partner on board, 2) improving the coverage in the connected NRENs and 3) extending eduroam to a big as possible subset of all NRENs in Europe. On the global level interoperations with other eduroam confederations will be studied.

The functioning of the pilot infrastructure is pretty stable. Establishing the policy will support the pilot and pave the way for an eduroam service, to be prepared during a one year test period. In this test period practical feedback about the usability of the policy will be collected and fed into a revision of the policy document.

Apart from the introduction of the confederation policy, improvements in the technical area have also been discussed. The majority of the problems discovered are related to the strict hierarchy in the RADIUS infrastructure or to other disadvantages of the RADIUS protocol. This deliverable outlines these problems in more detail, describing and evaluating the investigated alternatives.

Chapter 2 deals with the current architecture of eduroam, its building blocks and main components. Chapter 3 covers the technologies investigated and provides a comparison against a list of criteria. The detailed evaluation of RadSec and DNSRoam is provided in chapter 4. Chapter 5 outlines the current *eduroam-ng* (eduroam-next generation) architecture. Configuration diagrams and use cases are added in chapters 6 and 7 respectively. Finally, security and privacy considerations and conclusions are provided.

2 Architectural Overview of the Current eduroam Infrastructure

The operation of a roaming confederation is quite complex. It is a challenging task to understand fully how all the entities interact with each other to achieve the ultimate goal of a seamless roaming experience. This chapter describes the eduroam architecture in its current state. This description examines the system from various angles to provide the best possible insight.

Section 2.1, “Components and protocols”, describes the main functional components as well as the network protocols that are used in eduroam. These include standards from both the IEEE and the IETF. Section 2.2, “Trust Management”, describes the impact of the protocols involved with respect to security and trust relationships. The third section presents eduroam from a more service-oriented perspective, in line with the service-oriented architecture view of the eduGAIN sister project. Finally, the last section analyses the current eduroam architecture, highlighting those areas which are non-optimal and provides a plan to further develop the architecture towards a more robust infrastructure with the internal name “eduroam-ng”. As soon as the results of the development work are mature enough, they are supposed to be integrated into the pilot infrastructure eduroam.

2.1 Components and protocols

eduroam uses the following main components to realise its roaming service:

- Network Access Server (NAS) – a switch or wireless access point (AP) that provides clients with access to the local network.
- Client/ Supplicant – the end-user’s device that handles user authentication to the network. The Supplicant is a software being part of the operational system or not, developed for network access.
- Authentication Servers (AS) – for authentication and authorisation of clients and dynamic configuration of network access servers. The AS has a user database backend containing users' credentials (such as passwords or certificates) that are used to authenticate the client. Eduroam uses RADIUS Authentication Servers.

- IEEE 802.1X [1X] – Standard for Port Based Network Access Control.
- IEEE 802.1Q [1Q] – Standard for VLAN assignment.

2.1.1 RADIUS Authentication Servers

RADIUS is an acronym for Remote Authentication Dial In User Service and is defined by IETF RFC 2865 and RFC 2866. RADIUS is a protocol between a Network Access Server (NAS) and an Authentication Server (AS). It carries authentication, authorisation, accounting and configuration messages.

The basic principle is described in Figure 2.1.

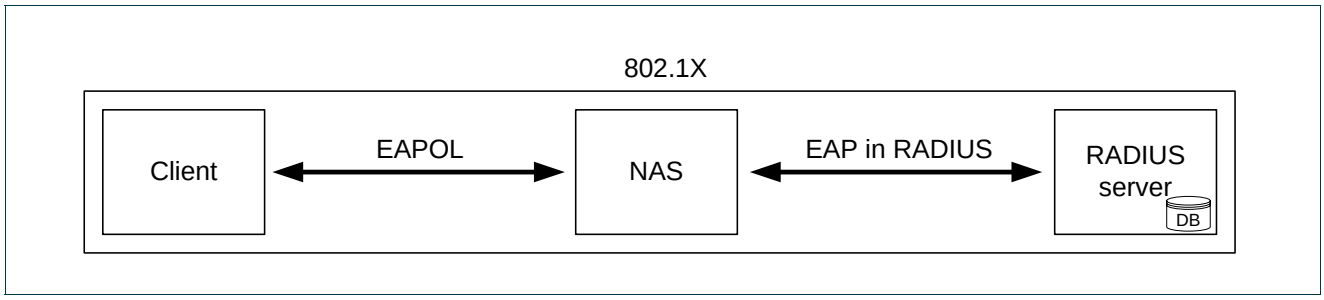


Figure 2.1: The IEEE 802.1X framework – interactions between components

The client uses a supplicant to connect to the NAS and to request access to the network. The NAS has the control on the network access. In IEEE 802.1X the protocol used is the Extensible Authentication Protocol (EAP over LAN between client and NAS). NAS encapsulates the EAP payload and transports the authentication messages to the RADIUS server. The RADIUS server performs the authentication and either accepts or rejects the access request. The NAS acts accordingly by denying or allowing the client access to the network.

The reply from the AS can also contain configuration elements that affect how the client can use the service. Several NAS devices can connect as clients to one AS. An AS can also cooperate with other AS through a backbone authentication network where one AS acts as a proxy server for another AS.

The AS may support a wide variety of methods of authentication, depending on the implementation. Unix login, text files, SQL databases, Certification Authorities and LDAP databases are examples of sources of credentials that the AS can check to validate a user's identity. It is also possible to use some methods in combination with other criteria such as a prefix or suffix to the username, the identity of the requesting NAS, etc. The fact that the AS supports such a wide variety of credential sources is important, as eduroam participants have the freedom to choose whatever type of credential source they like; a restriction of credential stores might force institutions to change their user management system which would in turn hinder the adoption of eduroam.

The above mentioned term “Authentication Server” is a generic term. Several protocols can be used to carry authentication credentials. The most well-known examples are TACACS+, RADIUS and Diameter. RADIUS servers have by far the largest deployment and lots of implementations available.

In the case of successful authentication, the local RADIUS server sends configuration options to the NAS in order to control which VLAN the client is assigned to. Different VLANs can have different permissions and can be connected to different parts of the campus. This is illustrated in Figure 2.2.

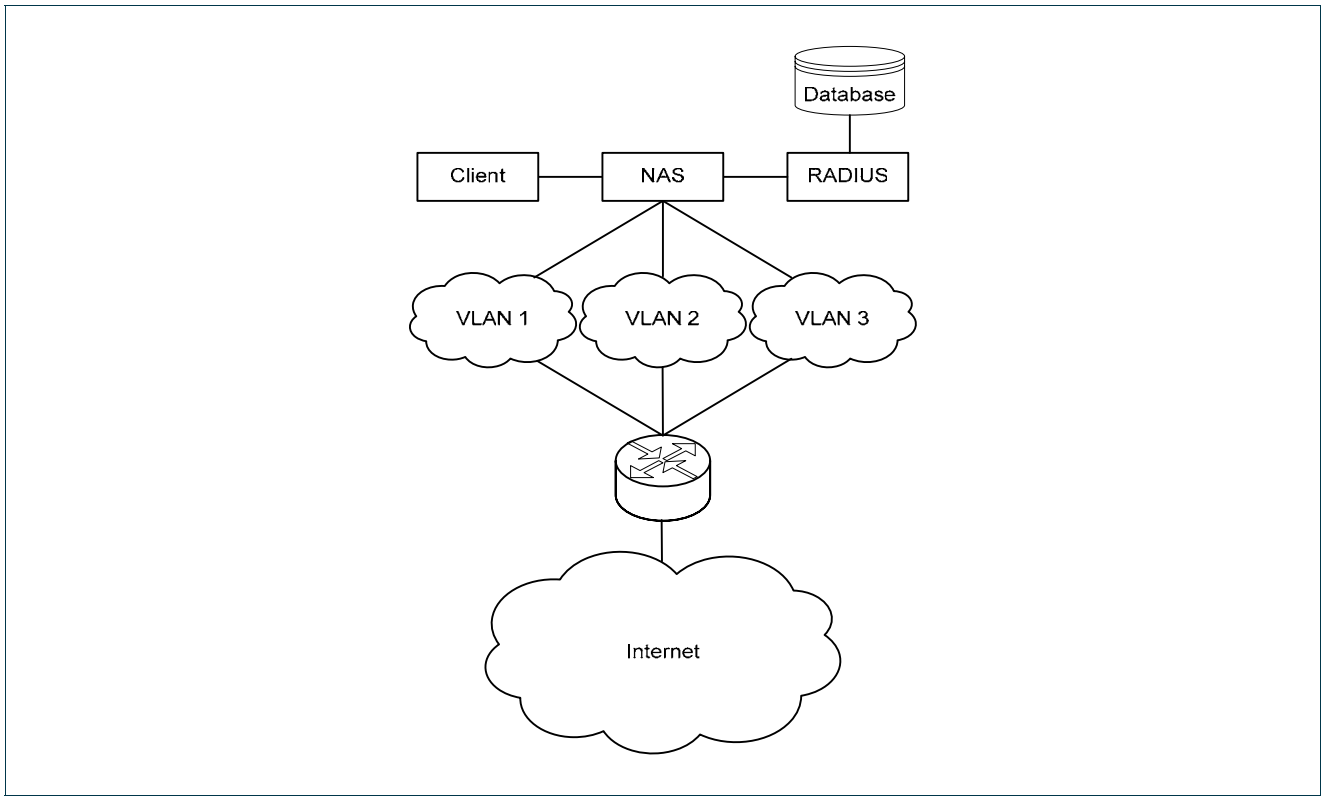


Figure 2.2: Assignment into different VLANs by the NAS device

This description applies to most institutions’ local networks using IEEE 802.1X. These are also the prerequisites for participating in the eduroam confederation.

In addition to this technological means of establishing an interconnection between institutions, eduroam also provides an agreement of cooperation (federation rules) between its participating institutions. The RADIUS servers are connected with each other through a multi-national RADIUS hierarchy (confederation). For wireless networks the SSID “eduroam” is made available for easy recognition, as an example for a federation rule.

RADIUS servers form thus the basics of the eduroam infrastructure. They play an important role in eduroam when being used as proxies for authentication requests. Every NREN participating in eduroam has one federation-level RADIUS server with at least one additional federation-level RADIUS server at another location for redundancy. These federation-level servers have a complete list of the participating eduroam institutions in

that federation (in most cases, the federation is a country), each of which is responsible for authenticating its local users. Each institution-level RADIUS server only needs to know their federation-level RADIUS servers. The federation-level RADIUS servers are also configured to connect as a RADIUS proxy client to the European eduroam confederation-level RADIUS servers (see Figure 7.2 in chapter 7).

Since RADIUS servers can act as a proxy for other servers, they enable a visiting user to authenticate using the same authentication configuration (user name, password, method) that is used at his home organization. This is possible because the local RADIUS server simply relays the authentication messages to the user's home organisation without needing to understand the messages; all that is required is that the local NAS accepts or rejects user access depending on the outcome of the authentication request at the home institution. This has possibly been subject to treatment by a number of proxies, which is transparent to the NAS.

When a user requests authentication, the user's realm determines where the request is routed to. The realm is the suffix of the user name, delimited with '@', and is derived from the organisation's DNS domain name. Realms within the same federation use the federation-level RADIUS server to forward the request to the home organisation. Realms from another federation use the federation-level first to connect to the European eduroam confederation-level, then to the home federation-level RADIUS server and finally then to the home organisation's RADIUS server. The format of the prefix and username is left to the user's home organisation and has no significance for the eduroam hierarchy. Omitting the realm in the outer identity will make roaming impossible for the user, since the request will not be routable while roaming outside of his institution. In this case authentication will still work at home (if no realm is given, the client is presumed local by most RADIUS implementations), but will lead to problems at a remote institution that are hard to trace. Administrators are advised to ensure that users include their realm also when using the wireless network at their home institution.

The eduroam RADIUS hierarchy enables a user at any eduroam-participating organisation to log in to an eduroam service within any other eduroam-participating organisation.

2.1.2 IEEE 802.1X

IEEE 802.1X is a standard for port-based network access control. An 802.1X enabled network access device is able to control its ports so that clients are only permitted to communicate through it if they have met authentication and authorisation criteria. Such a network access device could be a switch or a wireless access point.

The three components involved in an IEEE 802.1X authentication process as introduced in 2.1 are:

1. The **Supplicant** is the client software that requests access through a port.

2. The **Authenticator** is the network access device and is synonymous with a Network Access Server (NAS)

3. The **Authentication Server (AS),** which is in most cases a RADIUS server.

User authentication within 802.1X requires the use of the Extensible Authentication Protocol (EAP, [RFC3748]). EAP transports the authentication data over the LAN (EAPoL) and also within the RADIUS protocol ([RFC2869]).

The authenticator is only concerned with the status and VLAN membership of its ports and whether or not a supplicant is connected. The authentication and authorisation messages are simply passed through the authenticator, while the actual authentication is performed by the supplicant and the authentication server. The authentication server sends an Accept or Reject packet to the authenticator, indicating whether the user is allowed to connect or not. The authenticator opens the port or keeps it closed accordingly.

Another aspect of 802.1X that makes it versatile is the wide variety of different authentication methods that can be used within EAP. EAP-MD5, EAP with One-Time Password (EAP-OTP), Generic Token Card (EAP-GTC) and EAP-SIM are some examples. In order to address the security demands of wireless networks in particular, it is strongly recommended to use a method that offers mutual authentication. It is important for the supplicant to know it can trust the authentication server before it releases sensitive information such as a user name or a password. Examples of suitable authentication methods are EAP-TLS (certificate/certificate), EAP-TTLS (certificate/user name + password) and EAP-PEAP (certificate/user name + encrypted password). In each of these the RADIUS server first sends its certificate, containing its public key, to the client. The client can verify this certificate against an installed copy of the Certificate Authority (CA) public key and possibly an installed Certificate Revocation List (CRL) before continuing the authentication process.

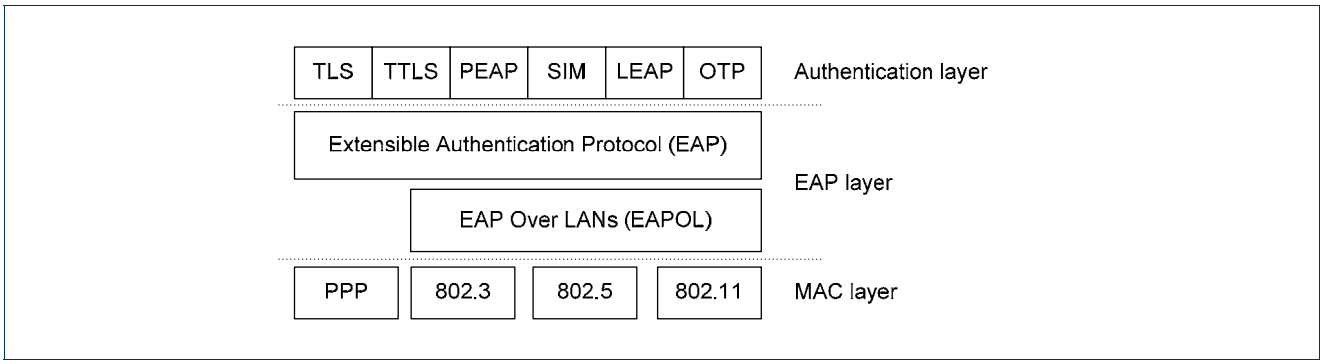


Figure 2.3: The layers of EAP authentication

It is important to note that encryption of the authentication process does not mean that data transmitted to/from the client will be similarly encrypted after the authentication process has been successfully completed. A separate encryption scheme is required to provide this feature. However, 802.1X in conjunction with a suitable authentication method can be used to distribute encryption keys that the client can use to encrypt its traffic. For wireless networks, methods of encryption are WEP (40 or 104 bit) with rotating keys, TKIP or AES.

The Wi-Fi Alliance made a “snapshot” of the work produced by the IEEE 802.11i working group during the development of the wireless encryption standard. The use of 802.1X authentication with TKIP encryption is branded as Wi-Fi Protected Access (WPA), while the use of 802.1X with AES encryption is branded as WPA2. WPA2 is broadly equivalent to the IEEE 802.11i standard.

There is a significant variation between organisations within the eduroam confederation with respect to the authentication and encryption schemes that each has chosen to deploy. The configurations depend on local preference, client compatibility and hardware/software limitations in the access points and/or in the RADIUS servers.

However, provided that there is an IEEE 802.1X capable system in place, the local method of authentication is irrelevant for the visiting client. The client's authentication process is forwarded by the RADIUS proxy to the client's home organisation and is processed as if the client was at home (see section 2.2.1, RADIUS).

The client may need to adapt the encryption scheme to the local configuration. Most organisations have access points that support one or more encryption methods; consequently it is possible that the client can use his home encryption method. Otherwise he has to select the method that is deployed at the visited institution.

2.1.3 Common EAP types

There are three EAP types that are commonly deployed within eduroam. They all provide an encrypted authentication process, mutual authentication of both supplicant and authentication server and can provide raw keying seeds that are suitable for generating encryption keys.

Not all RADIUS servers or supplicants support all three types. However, if the RADIUS server supports two or more types, they can all be used simultaneously according to the configuration of its supplicants.

EAP-TLS [RFC2716]

Transport Layer Security provides mutual authentication between two parties. It performs authentication between the server and the supplicant using certificates. The authentication server starts by sending its certificate, containing its public key, to the client. The client checks this key with the appropriate Certificate Authority (CA). If successful, the client sends its own certificate, also containing a public key, back to the authentication server. This key is checked by the authentication server in the same manner. If the key is valid, authentication is successful.

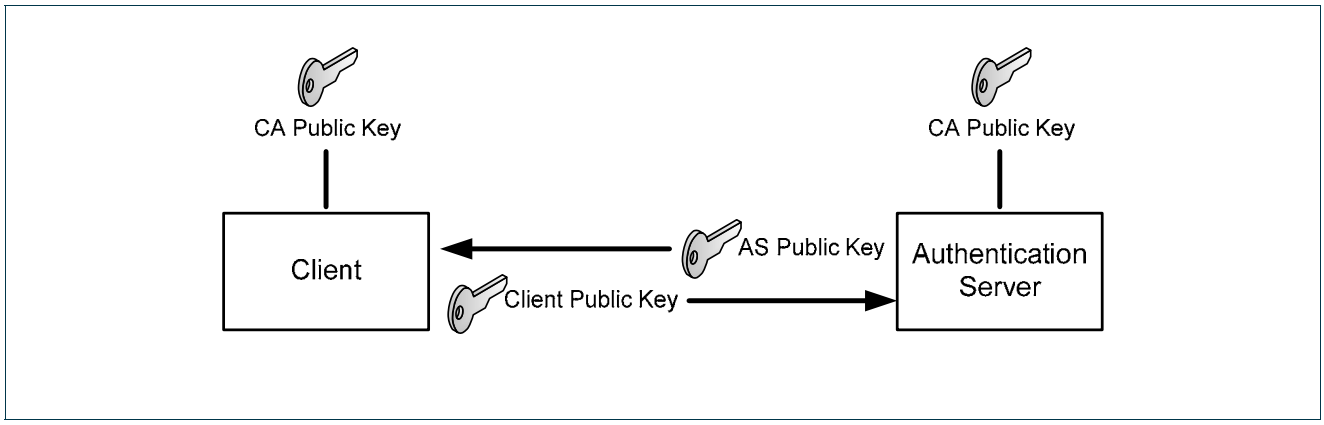


Figure 2.4: Usage of PKI in EAP-TLS

EAP-TTLS

The Tunnelled TLS (TTLS) protocol was created by Funk, Juniper and others. It is still an IETF draft; the protocol is currently at version 1.

Tunnelled TLS uses TLS to establish a secure tunnel between the authentication server and the client. As with EAP-TLS, the authentication server sends its public key to the client, who checks it. After this server authentication, the supplicant uses a password-based authentication protocol like PAP, CHAP or MS-CHAP. The corresponding credentials are transported within the TLS tunnel by using Diameter formatted attributes. Diameter is an authentication protocol similar to RADIUS and is discussed in more detail in chapter 3.1. The most commonly used protocol transported within TTLS is PAP. In any event, a valid username and password is required for the authentication to be successful.

EAP-PEAP

Protected Extensible Authentication Protocol (PEAP) is supported by Microsoft and Cisco amongst others. It is an IETF draft and the protocol is in version 2.

PEAP differs in some details to TTLS but the basic principle is the same. TLS is used to create a secure tunnel between client and authentication server using the authentication server's public key. Through this tunnel the client can securely use another password-based authentication. The only inner authentication protocol that is used in practice is EAP-MS-CHAPv2.

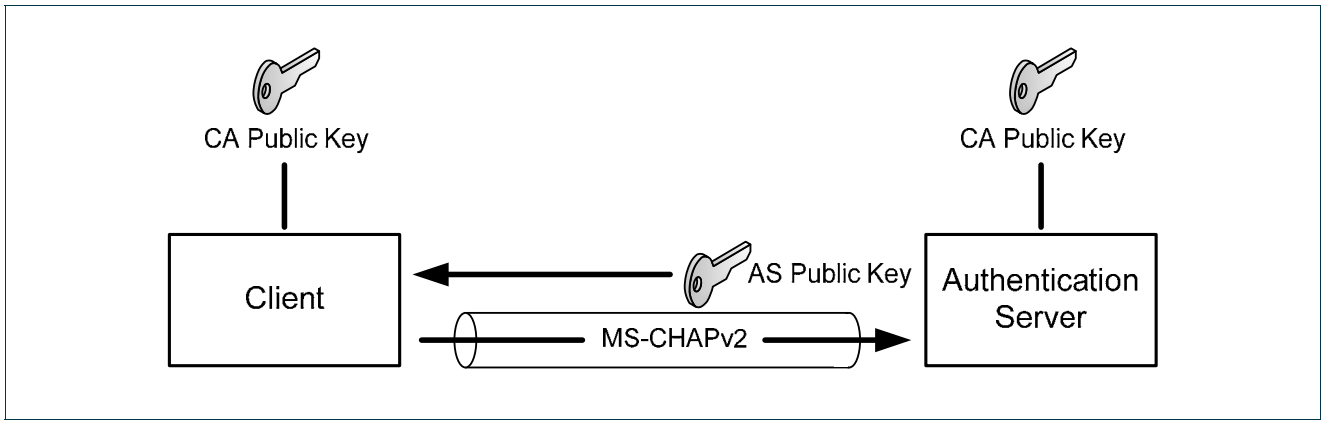


Figure 2.5: Usage of PKI and MS-CHAPv2 within PEAP

2.1.4 IEEE 802.1Q

IEEE 802.1Q is a standard for Virtual Bridged Local Area Networks (VLAN). It is a method allowing multiple bridged networks to share the same physical network link transparently. This link is often called a “trunk”. The traffic from all of the bridged networks is contained within the trunk; the Ethernet headers are modified with an additional 4-byte header to identify which VLAN the Ethernet frame is associated with. This header is a 12-bit VLAN ID (VID) allowing 4096 different VLANs, a 1-bit Canonical Format Indicator (CFI) for the MAC address format and a 3-bit user_priority field for IEEE 802.1p. In addition the EtherType is changed to signify the new frame format. It is also possible to define one of the VLANs as a Native VLAN; the Ethernet headers of this VLAN are not modified.

In the case of a switch that supports IEEE 802.1Q, it is possible to receive several VLANs in a trunk on one port and select a specific VLAN to attach to one or several of its other ports. In this way a switch can be used to connect clients to different VLANs. In a similar fashion a wireless access point with IEEE 802.1Q support can assign connected users to different VLANs.

In the eduroam environment it is common to have several VLANs available but reserved for different classes of users. For example faculty, students and guests could each have their own VLAN. They might have different levels of access to services: for example, faculty could have full access, students could have a more limited access and guests could have no access privileges except Internet connectivity.

RADIUS, together with IEEE 802.1X, is able to classify the users and assign them to the proper VLAN. This is achieved by means of the RADIUS server sending Attribute-Value pairs (AV-Pairs) as configuration directives back to the NAS after a successful authentication. The contents of these AV-Pairs can be dynamically configured to control VLANs (among other things).

All users can then use the same configuration profile (“eduroam”) to connect, but they will be attached to different VLANs depending on the user class they belong to. The same profile can be applied when users visit other eduroam-enabled organisations, but they are then automatically placed in the guest VLAN.

2.2 Trust management

The trust management within the RADIUS hierarchy is built implicitly into the AAA infrastructure. Every RADIUS server must know all of its potential peers in advance. This trust is installed via shared RADIUS secrets configured for each pair of servers and is exchanged using a secure out-of-band process (the shared secret is used by the RADIUS protocol to authenticate a peer). In the tree-like architecture of eduroam, one of the peers is always one level above the other one: an institutional level RADIUS server communicates only with its federation server; federation servers communicate with either institutional servers below or with their international confederation server. Thus the usual process to introduce a new roaming realm is that the administrator of the higher level server negotiates the shared secret with the administrator of the newly connecting realm.

RADIUS servers at the federation and confederation levels (and sometimes at the institutional level) are usually duplicated with a redundant server to improve the availability of the service. However this does not imply any changes from the trust management point of view since both servers belong to the same administrative domain. Their administrator just negotiates additional shared secrets for both servers with their peers.

From the point of view of the AS at the visited institution, the trust provided by the RADIUS hierarchy is transitive. The trust is not provided by the confederation in a top-down fashion; instead, it is managed by the RADIUS servers' administrators. Consequently, the federation has no technical means to influence the level of trust installed on nodes of the hierarchy.

From a user's point of view the trust is provided by protocols tunnelled through the hierarchy and connecting the user with its home institution's AS. This increases the importance of only using protocols providing mutual authentication of user and server.

2.3 Operational model

All interactions within eduroam are based on the components described in section 2.1, which gave an overview over the functioning of the relevant protocols.

This section examines the infrastructure from a service-oriented perspective. The protocols and procedures in use enable four distinct services, that can be mapped very well to the eduGAIN operational model (compare [GN2DJ5.2.2]:

- authenticating entities that want to use eduroam (Authentication Service)
- finding an Identity Provider (IdP) with authoritative information about the entity (Home Location Service)
- exchanging attributes that describe the entity (Attribute Exchange Service)

- determining what service level the entity is authorised to use (Authorisation Service)

2.3.1 Authentication

As described in section 2.1.2 the authentication service is based on EAP. This protocol consists of a set of request and response messages between the 802.1X supplicant on the user machine, the authenticator (access point or switch) and the authentication server (RADIUS server) of the organisation to which this user belongs. If the user is trying to authenticate from an organisation other than his home organisation, the request and response messages between the authenticator (of the visited organisation) and the authentication server (of the user's home organisation) are passed through a chain of intermediate RADIUS proxy servers (see next section, Home Location Service).

There are several different types of authentication mechanisms that can be carried within EAP, which can be classified into two main categories. The first category uses messages that are transported from the client to the home RADIUS server, via the chain of RADIUS servers, without encryption. This type of protocols are vulnerable to man-in-the-middle attacks, and so the credentials that a user sends to his home RADIUS server could get compromised at any point of the chain. The second category of EAP protocols is based on authentication over TLS connections.

For the eduroam confederation secure exchange of credentials is a necessity and therefore only the use of the latter category with mutual authentication is recommended, while the use of authentication mechanisms that expose the user's credentials unencrypted at intermediate servers is prohibited.

2.3.2 Home Location Service

For authentication and authorisation to proceed, it is necessary to locate the authentication or attribute release service of the organisation that the user belongs to.

In the case of eduroam, this service is based on two functions provided by the RADIUS servers that comprise eduroam's RADIUS hierarchy. The first is the proxy function that enables a RADIUS server to pass RADIUS messages to a different RADIUS server (including any protocol encapsulated in it, most notably the EAP protocol). The second function is to base this decision on the realm of the user name presented within the RADIUS packet. In the case of eduroam, the realm is based on the DNS domain of the user's identity. By leveraging these two functions, the eduroam RADIUS hierarchy permits the discovery of the RADIUS server of the home organisation, followed by the establishment of a virtual connection between this server and the client.

These two functions allow to discover the RADIUS server of the client's home organisation.

2.3.3 Attribute exchange

Since the current eduroam architecture relies on the RADIUS protocol and the messages it provides, the exchange of attributes is limited to the capabilities of RADIUS. This happens by means of well-defined message attributes. There are a number of attributes that are pre-defined in the various RADIUS RFCs of the IETF, such as the MAC address from which the user has attempted to authenticate (Calling-Station-Id). The number of pre-defined attributes is limited to 256 attributes altogether and the definition of new attributes within this space requires approval by the IANA. There is, however, an extension mechanism that allows RADIUS to transport arbitrary attributes. By using a generic attribute called "Vendor-Specific" (number 26), any entity can transport attributes that are outside of the normal attribute space. In order for RADIUS servers to interoperate when exchanging these custom attributes, a special dictionary must be defined that contains the required syntax.

Using RADIUS attributes, whether in the official attribute space or not, has some drawbacks. The most notable of these drawbacks is that there is no standard way of encrypting the content of the transmitted attributes end-to-end. Recently, some vendors have started to build proprietary extensions to RADIUS that allow end-to-end security for Vendor-Specific attributes, but since this mechanism is not standardized and has only very recently become available, to use it for eduroam is not an option. Another drawback is that the RADIUS protocol has very limited capabilities to negotiate which attributes should be transported. Therefore, it would be required to send all available user attributes "blind".

Because of the limitations of attribute handling, currently no attributes except those that are in the official attribute space of IANA are used for eduroam.

2.3.4 Authorisation

Authorisation typically means that a subset of the attributes associated to a user are matched against the requirements of the service that is to be accessed. The result of checking an individual's authorisation is either accepting or rejecting access, and possibly assigning the user to a specific service level. Checking authorisation in eduroam takes place in two phases: first, the desired service level is determined based on the attributes contained in the access request; and second, the appropriate service levels are set for the user's session.

Determining the desired service level

As outlined in the previous section about attribute exchange, the attributes that are transported over RADIUS are limited to the built-in attributes that are available in RADIUS. Therefore, authorisation for eduroam is limited to very few checks, which are performed at the RADIUS server of the institution where the user tries to gain access.

The most obvious check is whether the user is a guest or whether he belongs to the local institution. This can be easily determined by examining the realm of the user, since the realm part in the User-Name attribute of the RADIUS packet is always the correct one (as opposed to the local part of the user name, which may be

obscured in several EAP types). These realms can also be used to define specific sets of privileges that are granted to the users (this privilege level may be higher or lower compared to guests from other realms).

Another method for determining the service level is to examine the authentication protocol that the user is using to authenticate. The reason for such a check is to ensure that users have configured their supplicants correctly, so that they only transport credentials in a way that is considered secure. This can be achieved by inspecting the content of the authentication request. When using one of the RADIUS authentication types that do not use the EAP-Message attribute, the distinction between mechanisms is rather trivial, because the types of RADIUS attributes that are supplied differ between mechanisms. When using EAP-Message, the distinction is a little more difficult, but can also be done by looking into the first bytes of the EAP packet to find out about its EAP-Type. This is a feature that is available in most current RADIUS servers.

In a limited way, a user's authorisation level may also depend on the location he is currently visiting. Only a rough control over the user's location can be gained due to the nature of the RADIUS protocol. Current levels of location separation are:

- The user may only access the network at his home organisation (no roaming). Attempts to log in from an unauthorised location can be detected by checking if the authentication request is forwarded from the federation-level eduroam RADIUS server.

- The user may roam within the federation, but not abroad (national roaming). Attempts to log in from an unauthorised location can be detected by checking if the authentication request is forwarded from the European eduroam confederation RADIUS server. This setup can for example be used when user groups like pupils are to be restricted to national roaming access only.

- All locations are permitted for the user. In this case, no location checks are needed.

- A more fine-grained control over the user's location could be achieved if institution-level RADIUS servers were obliged to transport a location hint within the RADIUS Access-Request packet, for example with the IETF Geopriv working group RADIUS extensions [GeoPriv]. However, this RADIUS extension is currently unimplemented in most servers as it is still in an early stage of development and non-trivial to implement.

Remote organisations may evaluate the realm part of the user's login name to disable roaming access, as part of the eduroam escalation procedures, to realms whose users have shown undesired behaviour.

Another alternative commonly used as a base for authorisation decisions is the MAC address of the client, in order to ensure that a specific registered device is used to enter the service. This is not considered a good candidate for authorisation checks within eduroam because MAC addresses can very easily be forged and thus checking the MAC address of a user cannot ensure proper security.

Setting service levels

After the desired level of authorisation has been determined by one of the means mentioned above, the user can be assigned a service level that corresponds to his authorisation level. This is commonly achieved by defining a number of VLANs, where each VLAN corresponds to a user class. Most network equipments have the capability of assigning VLANs to individual users, so that even users that are simultaneously logged into the same access point and same SSID or plugged into the same switch can be in separate virtual LANs (VLANs).

A typical VLAN assignment might be that an institution's own users are assigned to a VLAN with the usual corporate permissions and access rights (e.g. access to printers), while guest users are assigned to a VLAN providing only those rights outlined in the eduroam policy document.

A second service level separation can be achieved by applying firewall rules to the IP address that is assigned to the user. This is done using a vendor-proprietary extension to RADIUS by transporting the firewall rules in a "Vendor-Specific" attribute extension. This method can be combined with VLAN assignments for achieving a more fine-grained control over the services offered to a user.

2.4 Limitations of the current eduroam architecture

The current setup of eduroam works remarkably well; in fact, its hierarchical structure has proven to be very powerful. At the moment, 23 countries are integrated with over 500 connected institutions. There are, however, some limitations associated with the current setup relating to trust establishment and the routing of authentication requests (the “Home Location Service”). The main weakness is the integration of routing decisions and trust establishment into the RADIUS hierarchy. This may have repercussions in the long run (especially bottlenecks in the root servers may become a problem) and therefore the introduction of new technologies into the existing eduroam architecture is being investigated.

The main points of attention are:

- The trust establishment between the RADIUS entities in these domains is accomplished using a static shared secret for each peer relationship.

Authentication requests are passed from one RADIUS server to another until the request reaches the home authentication server. This mechanism results in all traffic generated for authentication travelling through a chain of RADIUS proxies, while the authentication itself is only of interest to the RADIUS entities at the edges of the chain (the RADIUS server of the visited institution and the home institution of the user). Intermediate proxies may inspect the RADIUS payload, which places extra requirements on the type of authentication and raises privacy issues, therefore only EAP-TLS or tunnelled EAP types should be used. Furthermore, having a fixed chain of proxies is quite error-prone, as failure of one of the servers in the chain can easily result in denial of service to roaming users. Finally, a shared secret must be agreed upon and exchanged out-of-band for secure communication between RADIUS peers.

- The routing of authentication requests is based on a hierarchical system of (sub-)realms

The routing of authentication requests is based on the realm of the user. The choice of using the DNS domain system to reflect the origin of users (analogous to email addressing) means that the RADIUS hierarchy strongly resembles a DNS tree. The European eduroam confederation server resembles the root servers in DNS, the federation RADIUS proxies correspond to the country-specific top level domains etc. Trust is managed in the same way. Institutional membership of eduroam is decided upon by the NREN, and implemented by creating a sub-realm under the national top level domain (ccTLD). This results in problems when institutions have chosen not to use a country specific domain (but rather 'institution.net' for instance), or with supranational entities or even virtual organisations (like in the GRID environment), which cannot be uniquely assigned to a country.

Even though the above stated disadvantages may be only relevant to a relatively small part of the eduroam community, a number of possible alternative technologies have been considered for integration into the eduroam architecture. All of these alternatives have in common that they provide means to separate the establishment of trust from the actual authentication.

3 Possible Alternative Technologies for eduroam-ng

eduroam-ng is the next generation of the eduroam authentication infrastructure, a successor to the RADIUS infrastructure currently in place. Several possibilities exist for the new infrastructure, so a careful evaluation of the alternatives is needed. The following requirements are considered in the discussions concerning the alternatives for roaming infrastructure improvements, as taken from the roaming requirements document (DJ5.1.2).

General requirements:

- Reasonable security
- Data integrity
- Compliance with privacy regulations
- Verifiability

Standards compliance requirements:

- Openness
- Integration with existing eduroam configuration and usage of standards

Operational requirements:

- Scalability
- Ease of use
- Robustness

Using the listed requirements, the possible candidates for the implementation of eduroam-ng were examined in the sections 3.1 to 3.5; the conclusion in 3.6 presents the evaluation of these results and an outlook for the further steps.

3.1 Diameter

The Diameter protocol is positioned as the successor of RADIUS. Contrary to RADIUS, the Diameter standard has various features that explicitly support inter-domain roaming. Diameter can support different kinds of 'roaming models', the most relevant ones are discussed below.

3.1.1 Diameter roaming models

Diameter r*oaming with "Redirect Agents".* In this operation mode the step for peer discovery and the step that checks the roaming federation membership are implemented using the *redirect* mechanism. A Diameter server that receives an authentication request for a user from a foreign domain – a visitor – by default relays the request to a Diameter redirect agent that replies with a redirect message. This message indicates that the request must be relayed to another entity (possibly the Diameter server for the user's realm which takes care of the actual authentication or perhaps another redirect agent). Thus, redirect can be used to discover a peer. The redirector agent (or a group of redirector agents) has knowledge about the Diameter servers for all the realms that participate in the roaming federation.

The redirect mechanism is used in combination with a PKI: redirection is used for peer discovery and a PKI is used for checking the peer server's authorisation. When the peers are part of the same 'trusted' part of a PKI tree – e.g. their certificates are signed by the same roaming federation Certificate Authority (CA) – they know they both participate in the same roaming federation. A drawback of this model is the necessity to administer the roaming federation at two locations: configuration of the redirector and configuration of the PKI tree.

Diameter *roaming with DNS lookups*. Here, peer discovery is not implemented via a redirect service, but by using DNS NAPTR/SRV records (either using DNS or DNSSec for signed responses). The confirmation that the peer is part of the roaming federation is based on PKI, similar to the situation with Redirect Agents. This situation is depicted in Figure 3.1. This operation mode looks most promising for inter-domain authentication, and will be considered in the remainder of this document. The communication between Diameter peers may be based on IPSec or on TLS/TCP. IPSec is less suitable, because only a single certificate can be used for all applications running on the server. Furthermore, [RFC3588] recommends the use of TLS for inter-domain authentication.

When roaming with Diameter and DNS lookups, the following interactions take place (Fig. 3.1):

1. A Diameter client, such as an 802.11 WLAN access point, needs to authenticate a visiting user and sends the user credentials, including the user's realm/domain, to the local Diameter server

2. (and 3) The Diameter server in the visiting domain notices that this user is not part of the own domain and decides to setup a direct connection to the Diameter server that is capable of executing the authentication. It looks up, through regular DNS, the IP address and certificate of the RADIUS server of the home domain using NAPTR/SRV DNS records. Note that this lookup may return false information, for instance caused by DNS spoofing, but this is adequately detected during the next step. The Diameter server checks that the certificate holds information on the host- and domain name that matches with the user's home domain and the DNS query.

4. Using the IP address of the home Diameter server, the Diameter server of the visited institution establishes a TLS connection to the home Diameter server. The roaming infrastructure runs a Certificate Authority dedicated to checking the certificates of authenticating Diameter servers that are part of the roaming federation. So, this CA is not part of a multi-purpose PKI hierarchy: when a Diameter server's certificate is signed (ultimately) by this CA, you know that 1) the server participates in the roaming domain and 2) the server is authorised to authenticate for its own domain (home.org), as the host- and domain name are required to be part of the certificate. As part of the TLS connection setup, both the home server (4a, 4b) and the visiting server (4c, 4d) check that the certificate of their peer is signed by the trusted Certificate Authority of the roaming federation. If so, they know they are part of the same roaming federation and continue with their communication: the visiting server sends the authentication request to the home server. Note that the dashed arrows represent the PKI trust relation and out-of-band setup. No actual communication needs to take place to verify the validity of the certificates, except perhaps for the retrieval of Certificate Revocation Lists (CRLs).

5. The home server sends back the authentication result.

6. The server of the visited institution sends back the authentication result to the client.

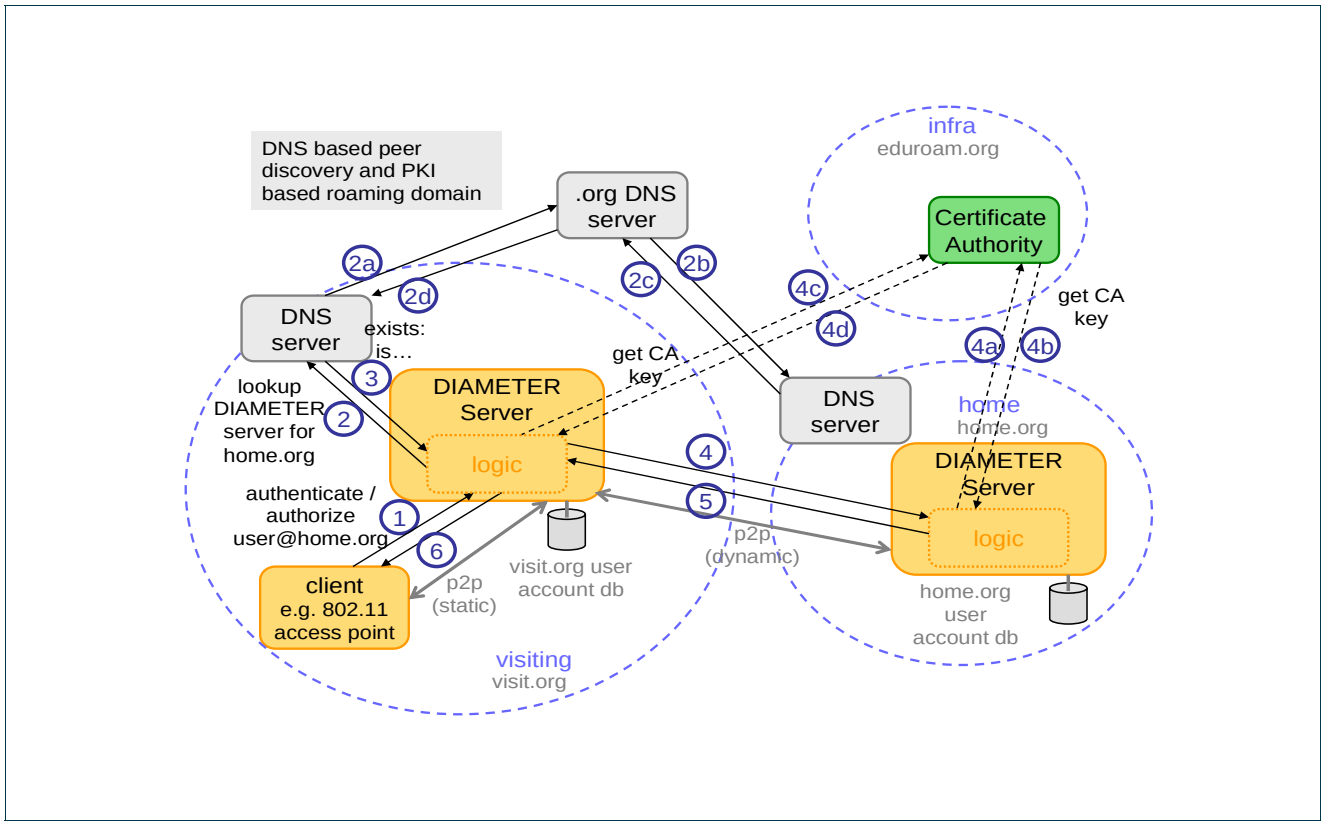


Figure 3.1: Diameter roaming with DNS

Note that the trust establishment for the interaction as described above is based on the availability of a dedicated roaming PKI. All the participating Diameter servers trust this PKI: all other trust during the interaction is derived from this. As this PKI is dedicated to a single roaming domain, the participants know that only certificates of Diameter servers that are actually part of this roaming domain will be signed.

3.1.2 Diameter and RADIUS legacy connections

The question that arises in the previous scenario is: how can a Diameter-based roaming infrastructure integrate with an existing RADIUS-based infrastructure? Two possible configurations are depicted in Figure 3.2 and Figure 3.3. We assume that an existing RADIUS hierarchy is in place for all current and future participants. This means that a participant can always relay an authentication request up into the RADIUS hierarchy, either because it simply chooses to deploy only RADIUS, or because the peer it wants to reach is only capable of handling RADIUS requests. Likewise, it means that all participants must be capable of receiving incoming RADIUS requests. Therefore, participants that wish to support Diameter must run a Diameter node that has multiple roles: 1) a translation agent to map RADIUS requests from local legacy RADIUS clients or RADIUS requests arriving from the hierarchy to Diameter requests, 2) a Diameter client for sending authentication requests to a Diameter peer, and 3) a Diameter server for local realm user authentication.

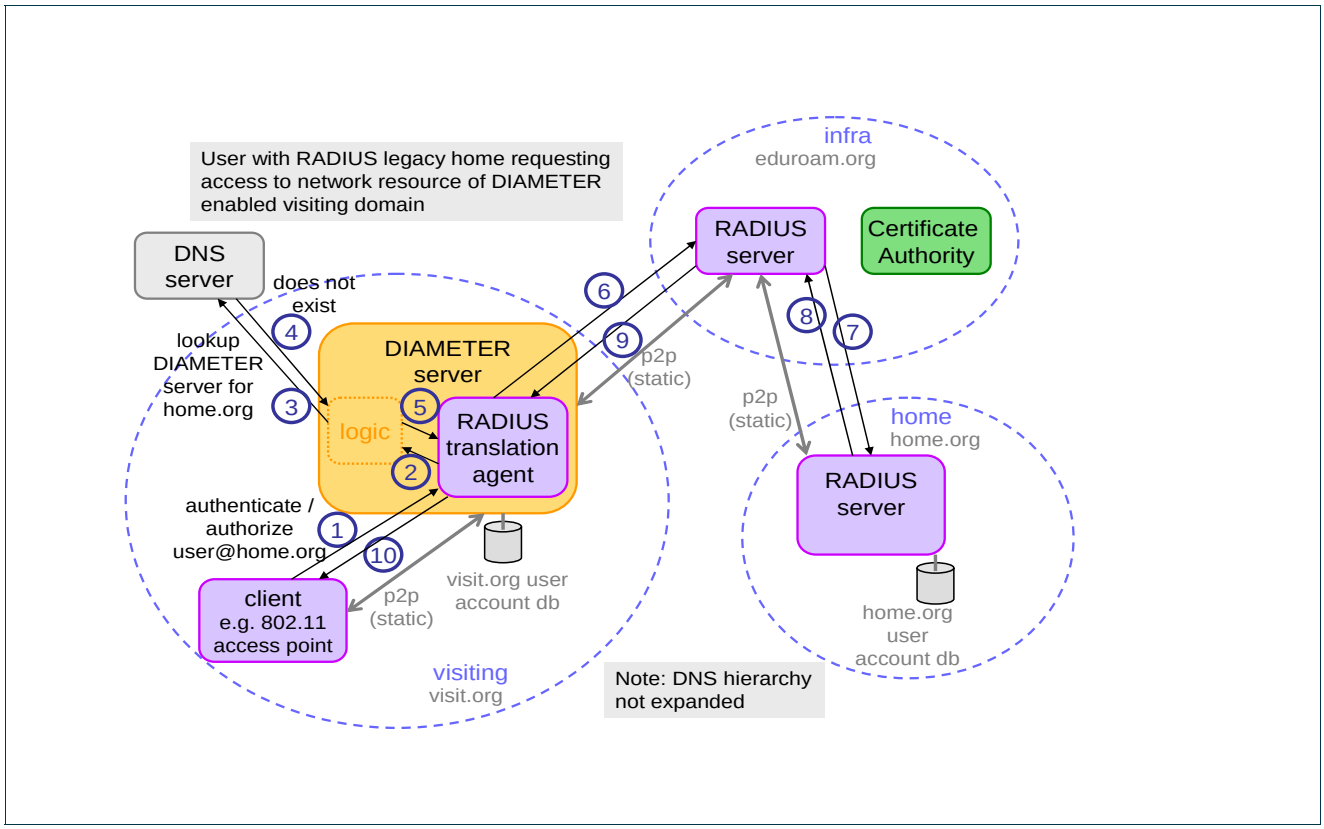


Figure 3.2: Diameter with RADIUS legacy: RADIUS-based peer communication

Let's look at a situation where a user visits a realm that is Diameter enabled but the user's home realm only speaks RADIUS (Figure 3.2). The following interaction takes place:

1. A legacy RADIUS client, such as an 802.11 WLAN access points, needs to authenticate the visiting user and sends the user credentials, including the user realm/domain, to the RADIUS translation agent of the local Diameter server.

2. The RADIUS translation agent translates the RADIUS request into a Diameter request

3. (and 4) The Diameter server logic tries to setup a Diameter peer connection to the server in the user's home realm. It initiates a server lookup through DNS, but finds out that the home domain does not have a Diameter server. The logic determines that, as a fall-through, the authentication request will be sent as a RADIUS request to the RADIUS hierarchy.

5. The logic hands over the request to the RADIUS translation agent to forward it into the legacy RADIUS hierarchy.

6. (and 7, 8, 9) The request passes through the RADIUS hierarchy and the results passes back

10. The result is provided to the RADIUS client.

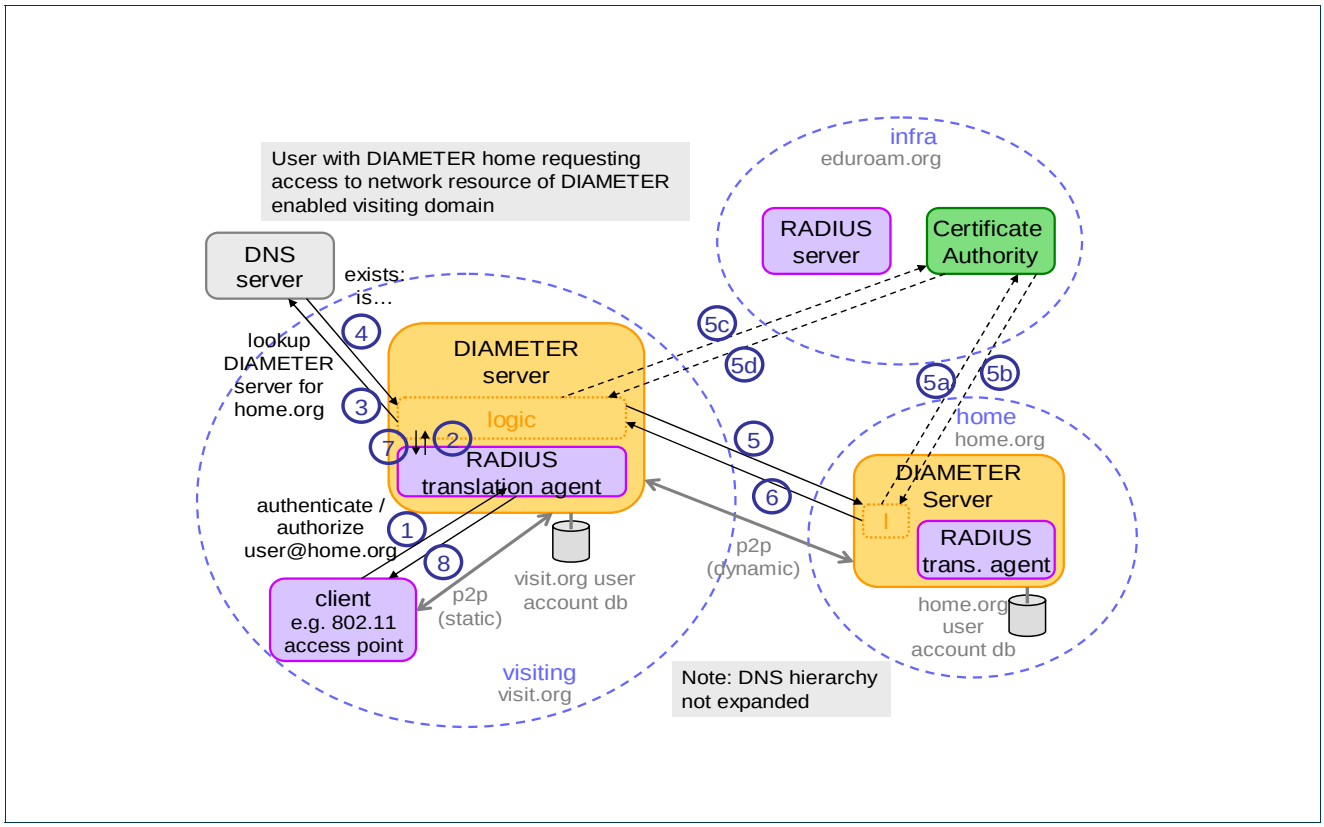


Figure 3.3: Diameter with RADIUS legacy: Diameter-based peer communication

In the same mixed Diameter/RADIUS environment, the interaction between two Diameter peers (Figure 3.3) could take place as follows:

1. A legacy RADIUS client, such as an 802.11 WLAN access points, needs to authenticate the visiting user and sends the user credentials, including the user realm/domain, to the RADIUS translation agent of the local Diameter server.

2. The RADIUS translation agent translates the RADIUS request into a Diameter request.

3. (and 4, 5, 6) The Diameter server handles the request as a regular Diameter request (the same as indicated in Figure 3.1).

7. (and 8) The Diameter server logic hands back the result to the RADIUS translation agent, which sends it to the RADIUS client.

One advantage of the mixed configuration as indicated above is that it builds on high probability that many sites have to deal with a mixed RADIUS/Diameter configuration anyway because of the usage of RADIUS by network access equipment. Another advantage is that NRENs and TLDs such as eduroam.org have an unaltered, simple and proven configuration based on RADIUS. No need for translation back and forth between RADIUS and Diameter inside the infrastructure. Additionally, the above configuration allows for a gradual

transition from RADIUS to Diameter. A drawback is the requirement to have RADIUS in place for all participants, even if almost all participants have migrated to Diameter.

Note that only a few possible configurations are listed; many more mixed RADIUS/Diameter environments are feasible. In principle, any server that has both Diameter and RADIUS functionality may choose (typically through DNS – SRV records) whether it authenticates via RADIUS or Diameter. And any organisation may list a proxy-authentication server (e.g. the national RADIUS or Diameter relay server) as their authentication server in DNS, and rely on any authentication transport between the proxy-authentication server and the authentication server of the organisation.

Figure 3.4 below shows an example of such a configuration, where static RADIUS peers are used in the lower parts of the authentication hierarchy and dynamic Diameter peers in the higher parts of the hierarchy.

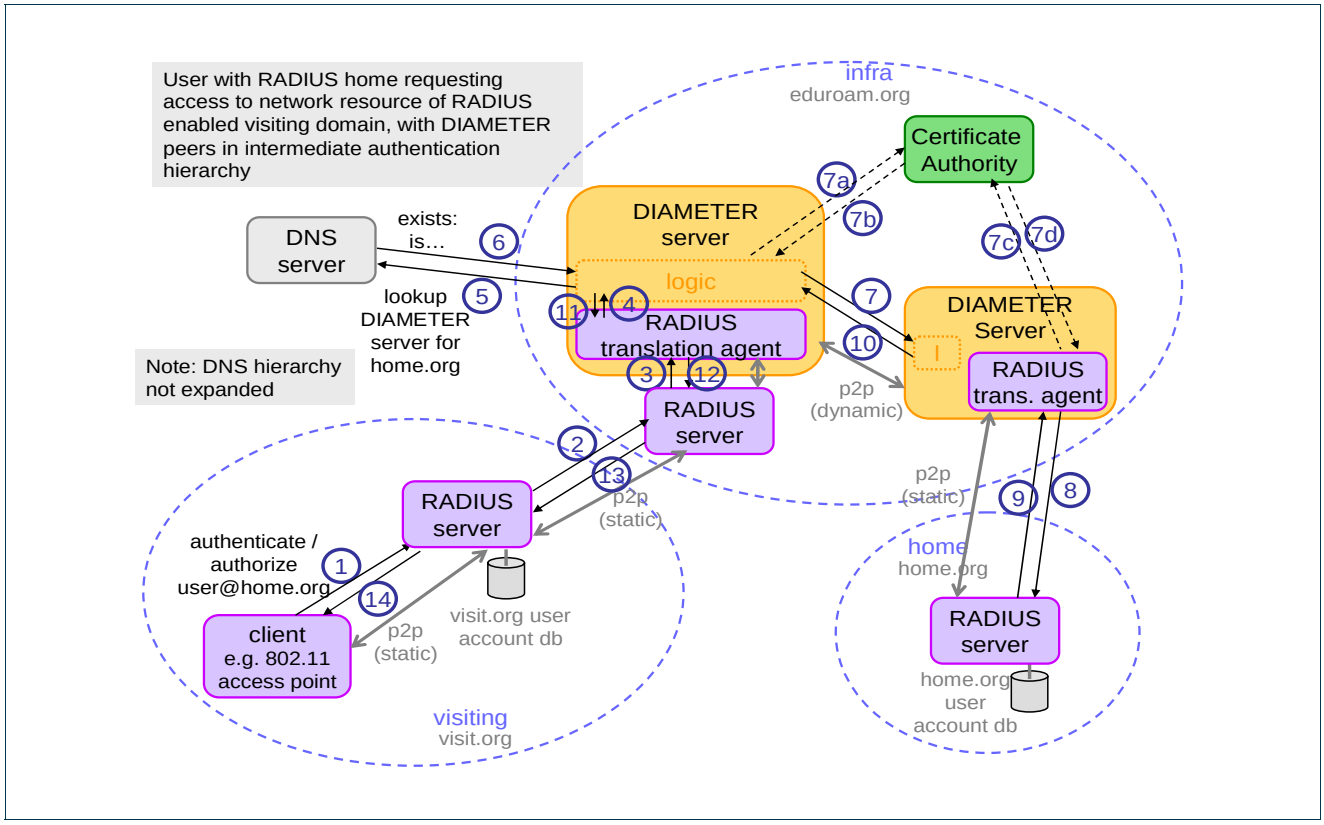


Figure 3.4: Mixed RADIUS/Diameter, RADIUS lower in hierarchy, Diameter higher up

For the lower parts in this authentication hierarchy, the chained RADIUS approach is followed. At a certain point in the chain, however, a Diameter server is encountered which handles the request by initiating a dynamic peer to peer connection. The most interesting question in this setup is how the Diameter server discovers the Diameter peer associated with a user's home domain. This could be realized by requiring that all participating organisations have a DNS SRV record such as _roamingdomainname._roam.home.org for the home.org

participant, which points to the associating Diameter server higher up in the hierarchy, used here during steps 5 and 6. Such a setup would allow an organisation to participate in multiple roaming domains at the same time.

The security considerations for dealing with RADIUS legacy connections are similar as with Diameter roaming with DNS lookups (see section 3.1.1). One difference is that the Diameter server that looks up another Diameter peer for a participating organisation cannot immediately check that the peer is actually serving for the participating organisation (because it serves the organisation only indirectly through its underlying RADIUS chain). A drawback of this approach is the requirement on a specific DNS configuration for participants: the usage of RADIUS in the lower parts of the hierarchy hides the usage of dynamic Diameter peers higher in the hierarchy, allowing for an easy transition, but additional DNS configuration will undo parts of this advantage. Additionally, changes in the higher parts of the hierarchy will likely need DNS reconfiguration in lower parts.

3.2 RadSec

RadSec, originally described in [RadSec], is a modification of the traditional RADIUS protocol. It preserves the RADIUS packet format and thus provides good backward compatibility with pure RADIUS clients and servers in mixed environments. The differences between RADIUS and RadSec lie in the transport mechanisms used for packet delivery, peer authentication and peer discovery. The following sections provide an in-depth explanation of the differences between the two protocols and give an overview of the current implementation status.

3.2.1 Operational differences to plain RADIUS

There are several types of extensions in RadSec that can be used depending on the needs of the concrete authentication infrastructure.

Transport Protocol

The first – and most basic – extension is that the transport protocol used is TCP instead of UDP. The RADIUS protocol uses UDP as a transport protocol, but specifies additional mechanisms that duplicate several features of TCP in a custom-built manner. For example, the RADIUS protocol defines positive acknowledgements for the accounting messages in RADIUS (*Accounting-Response)* to reduce the probability of information loss (which is an inherent problem on unreliable transports). Also, periodic re-sending of packets is specified for the cases where no reply is received. This functionality, along with a lot of other improvements (like, for example, the three-way handshake to detect if the server is up and running) comes built-in in TCP. Thus, using TCP actually makes packet handling for RadSec servers simpler because a lot of the specific details of packet handling within RADIUS are handled automatically by the TCP stack of the operating system.

Packet Encryption and Peer Authentication

The second extension in RadSec, which is deployed on top of the TCP extension, is the use of TLS tunnels for communication between servers. Using TLS provides a much more elegant and secure way to prove the authenticity of the communicating entities. First, authentication is no more bound to the IP address of a node,

but instead to the certificate presented while establishing the connection. Relevant parts are the CA and Common Name or subjectAltName fields of the X.509 certificate presented by the peers. Furthermore, the entire RADIUS packet is transported within the TLS tunnel so that no information about an ongoing user authentication is revealed to intermediate IP hops. As a side-effect of the strong encryption that protects the entire packet and the peer authenticity verification with certificates, the RADIUS way of protecting user passwords and peer authenticity with the shared secret becomes obsolete.

Making the shared secret obsolete and becoming independent of IP addresses has also another positive side-effect: even RadSec servers that need to constantly change IP addresses (for example because of being behind a DSL dial-up connection) can be integrated into the infrastructure without problems.

3.2.2 Current implementation status

Radiator currently includes RadSec support, implementing the extensions for reliable transport (TCP), packet encryption and peer authentication (TLS). There is also an implementation of dynamic peer discovery that can be considered beta quality.

Other implementations, like FreeRADIUS, have shown interest in RadSec since it solves the most basic shortcomings of RADIUS in a simple way and without adding the complexities of Diameter. However, not having an officially proposed standard for RadSec to refer to (only a whitepaper from Open System Consultants) could keep back some implementers.

3.3 RADIUS with DNSSec

RADIUS-DNSSEC is based on using DNS both for peer-discovery and for trust establishment.

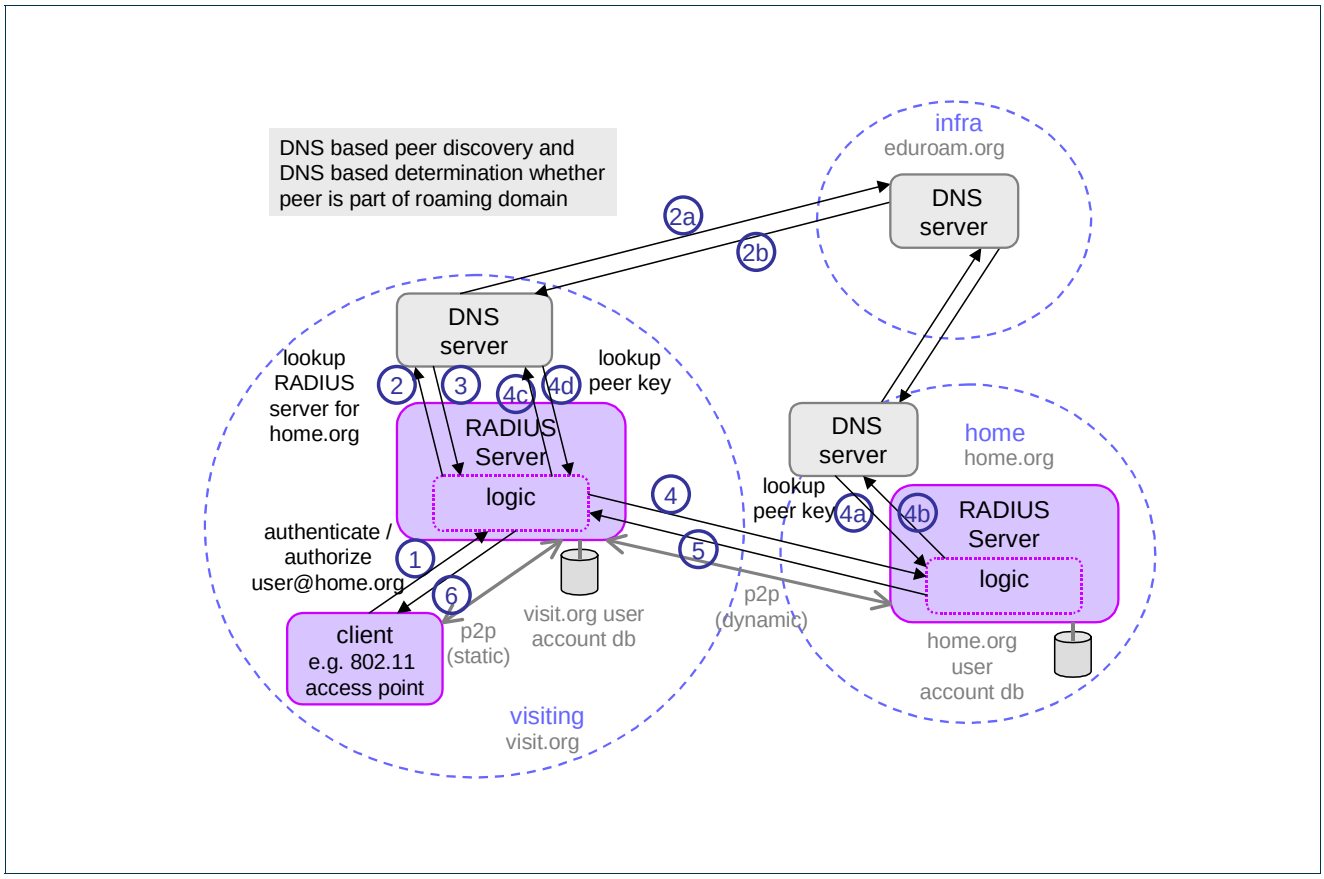


Figure 3.5: RADIUS-DNSSEC roaming model

The following actions take place in this scenario:

1. A RADIUS client, such as an 802.11 WLAN access point, needs to authenticate a user and sends the user credentials, including the user realm/domain, to the local RADIUS server.

2. The RADIUS server in the visited domain notices that this user is not part of its own domain and decides to setup a direct connection to the RADIUS server that is capable of authenticating the user. It looks up, through Secure DNS, the IP address and certificate of the RADIUS server of the home domain *as part of the roaming domain DNS tree*. (2) The roaming federation administrator manages this tree, effectively deciding which parties cooperate in a roaming agreement. Secure DNS makes sure that the answers are correct and trustworthy. Every RADIUS server has its own public and private key pair and has its public key published in the form of a certificate in the roaming domain DNS tree. These certificates can be self-signed and do not need to be part of a Public Key Infrastructure (PKI).

3. see 2.

4. Using the IP address and the certificate of the home RADIUS server, the RADIUS server of the visited institution establishes a TLS connection to the home RADIUS server to negotiate a shared secret. A protocol has been developed for that by Telematica Instituut, the so-called RADIUS Key Exchange (RKE)

protocol. The server in the home domain executes similar checks as the proxy in the visiting domain, e.g. checks – through DNS – that the incoming request for key exchange is from a party that is part of the roaming domain (checks 4(a) – 4(d)).

5. The shared secret is now used to setup a normal RADIUS connection between the peers in order to perform a straightforward user authentication over RADIUS.

Note that the trust establishment for the interaction as described above is based on the availability of a secure DNS tree dedicated to the roaming domain. All the participating servers trust this tree: all other trust during the interaction is derived from this. This means that trust establishment is essentially the same as with the PKI based solutions discussed before. As this DNS tree is dedicated to a single roaming domain, the participants know that only certificates of servers belonging to this domain will be part of the tree.

The advantage of this approach is that it uses native RADIUS, in combination with DNS for peer-discovery. It results in dynamic peer-to-peer connectivity. The disadvantage is that it requires DNSSEC deployment for the authentication infrastructure, and that it requires an additional protocol for establishing RADIUS shared keys.

3.4 RadSec with DNSSec

When using RadSec, it is also possible to have this server dynamically discover other peers. Enabling dynamic peer discovery makes hardwired configurations obsolete and allows for direct communication between the authentication servers. Provided that both endpoints are using the same techniques, the traditional hierarchy (with its root server and other potential points of failure) can be bypassed.

This solves both security and scalability issues: only the necessary authentication servers are used for a given request, and all user information is exposed only to the two servers that are dealing with the request.

The peer can be verified by the RadSec client using the previously described authentication mechanisms (TLS). Finding another peer requires a lookup-service that, in the current implementations, is based on DNS. When using DNS without DNSSec it is not possible to prove the authenticity of the server that was contacted, because no reliable source of information about the server's identity exists. Anyway, it is possible to prove that the server contacted is authorised to handle the request by checking the attributes of the certificate that is presented in the TLS challenge after the lookup is done. A variety of attributes can be used to check the authorisation, for example if the certificate is issued from a CA that is dedicated to issuing certificates for the roaming purpose, or a specific OID for certificate usage.

When blending this dynamic discovery in with an existing infrastructure, the technique can also be used to discover paths in only parts of the tree or leaves of the infrastructure. For instance an important part of the infrastructure with potential load, the root server, can be kept out if cross-federation requests are handled with dynamic discovery. While meshing only a smaller part of the infrastructure can be considered less desirable, it might be a feasible improvement to an existing infrastructure when requiring that all of the already implemented parts use a new technique is unrealistic.

Other lookup-services than DNS can be considered. Some lookup services provide reliable information (like LDAP or DNSSec) and thus offer a secure mechanism to prove a server's authenticity, or can even allow for exchanging more information than just providing an authentication-server's address. Examples are for instance fingerprints of certificates which are allowed to answer for a specific realm in a specific federation, or information about the CA / CRL data that can be used to validate the peer. If this complexity is added to a lookup service this also changes the view on the security of the setup: a lookup service that provides more information should have a higher level of trust, but it imposes less security constraints on the actual authentication servers, who no longer need to prove their membership to the federation themselves.

3.5 Web-based redirect combined with AAI

The web-redirect roaming model is fundamentally different from the technologies discussed before, since it does not use protocols designed for network AAA (such as RADIUS, Diameter and derivatives) but protocols designed for web application access control. Web redirection utilises features of HTTP (such as redirects and URL parameters), SOAP and HTML (such as forms). Nowadays, web redirection based network access is widely used in public places such as airports, hotels etc.

In a web redirection-based setup, once an IEEE 802.11 client has established a connection to a wireless access point, the client is provided with an IP address and unauthenticated layer 3 access to a specific docking network. The docking network is dedicated for wireless clients and separated from the rest of the network by a firewall called Network Access Controller. Initially, for new clients entering the docking network, the firewall blocks all the traffic in and out.

The Network Access Controller is bundled with a web server. Having entered the docking network, end users are expected to open a web browser in their client. Whatever URL the end user types in, the Network Access Controller captures the browser's initial HTTP request and redirects it to the on-board web server, which provides a welcome page to the user. The user uses the remote organisation's authentication procedure to ensure the web server of her authorisation to access the network. Once the web server has decided that the user is allowed to have access to the network, it changes the firewall rules so that the traffic for this client starts to flow in and out of the docking network.

What makes web redirection-based roaming access interesting, is that it brings network roaming close to application level Authentication and Authorisation Infrastructure (AAI) federations, which use technologies such as SAML, Shibboleth, Liberty Alliance and eduGAIN to deduce web users' authorisation to use a service on the web. In essence, it turns network access into just another service for which AAIs provide authorisation services.

Figure 3.6 illustrates how a Shibboleth-based AAI can be used for roaming network access. AAI protocols other than Shibboleth can be used as well; however, the setup in the figure has an implementation available [HUPNET].

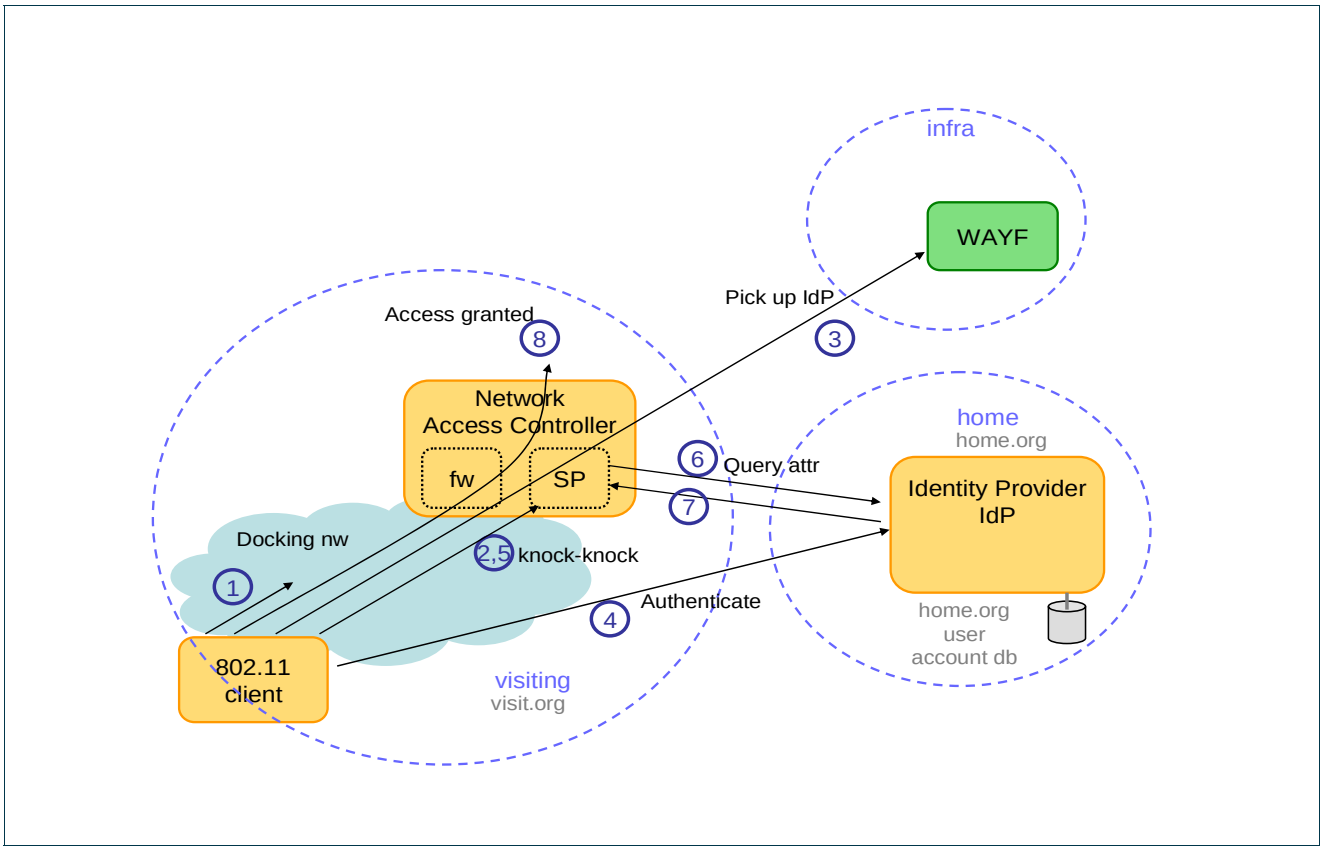


Figure 3.6: A roaming model based on web redirection and AAI

1. A user activates his 802.11 client, establishes a connection to an Access Point, and gets an IP address to the docking network via DHCP. A firewall in the Network Access Controller blocks the traffic out of the docking network.

2. The user opens his web browser. The Network Access Controller captures the initial HTTP request and redirects the browser to the on-board web server. The Shibboleth Service Provider (SP) component (typically, an Apache module) protecting the web server takes control of the HTTP request.

3. The Network Access Controller is a service registered in an AAI and, as any other service in the AAI, redirects the browser to the “Where Are You From” (WAYF) service provided by the AAI federation. WAYF presents to the user a drop-down list of Home Organisations in the AAI and the end user picks up one.

4. WAYF redirects the browser to the Shibboleth Identity Provider (IdP) of the end user’s Home Organisation. The Shibboleth Identity Provider uses a web dialogue to ask the user to enter username and password. The user enters the required credentials and the Shibboleth Identity Provider validates them against the local user database.

5. The Shibboleth Identity Provider redirects the browser back to the Shibboleth Service Provider, with an embedded SAML assertion called *a handle*.

6. The Shibboleth Service Provider uses the handle within a SAML/SOAP request to acquire the user's attributes from the Shibboleth Identity Provider

7. The Shibboleth Identity Provider responds with a SAML/SOAP response containing the attributes the end user has consented to release.

8. Based on the attributes, the Shibboleth Service Provider decides that the user is authorised to access the network. The Network Access Controller changes the firewall rules so that the traffic for this client may flow in and out of the docking network.

Web redirection-based roaming architecture combined with an AAI has the following advantages in an AAI available for roaming:

- end users' acceptance for web redirection-based roaming is high, as the popularity of the web has made users familiar with a web browser.

- fine-grained attribute-based authorisation, controlled by the visited institution.

- Identity Providers often provide single sign-on; one authentication is enough to make the services (both network access and web applications) available for the end user.

- if an AAI is able to serve network access as well, costs are saved as two overlapping federations and architectures need not be maintained.

However, using application layer mechanisms to protect access on the network level causes downsides, such as:

- the traffic is not authenticated by the access point, making it possible to hijack connections in the docking network.

- the traffic is not encrypted on the air, making it a subject for eavesdropping.

- in order to make the client's web browser communicate directly with the Identity Provider, the firewall must be dynamically opened (SSL port 443) for the user's Identity Provider and for the WAYF to let the user authenticate directly in the home organisation.

- AAI architectures are typically not compatible with RADIUS, making smooth transition problematic.

3.6 Conclusion

Several mechanisms can be used for dynamic establishment of peer-to-peer authentication channels. The table below summarizes the key elements and differences of these solutions.

Element	Diameter	RadSec	RADIUS-DNSSEC	RadSec + DNS	Web redirection + AAI
Trust-establishment	PKI, one root-CA per roaming agreement	PKI, one root-CA per roaming agreement	Self-signed certificates in secured DNS zone (one per roaming federation)	Certificates in participant's DNS zone	PKI and AAI federation metadata
Peer discovery	DNS	static	DNSSec	DNS	WAYF
Support for multiple roaming agreements	Select proper certificate	Select proper certificate	Select proper DNSSec subtree	Select proper certificate	Yes, user picks one
Authentication protocol	Diameter over TLS	RADIUS over TLS/TCP	RADIUS over UDP, prop. Protocol for key establishment	RADIUS over TLS/TCP	HTTP/TLS/TCP
Insight in membership of roaming organisation	No	Yes	Yes (DNS)	Yes	Yes (WAYF)
Deployment issues	Requires PKI, DNS configuration	Requires PKI; non-standardised transport layer	Requires DNSSec, proprietary key establishment	non-standardised transport layer (RadSec)	Requires PKI and AAI federation. Requires browser access to IdP before general internet connectivity

Table 3.1: Main properties of authentication models

The mapping to the JRA5 requirements is as follows:

Requirement	Diameter	RadSec	RADIUS-DNSSEC	RadSec + DNS	Web redirection + AAI
Reasonable security	Covered, all trust derived from dedicated PKI	Covered, all trust derived from dedicated PKI	Covered, all trust derived from dedicated secure DNS tree	Covered, all trust derived from PKI	Low. After authentication: In the docking network traffic is unauthenticated and unencrypted, the identity proving process is as secure as the AAI is, but unauthenticated access is relatively easy.
Data integrity	Good, TLS encryption	Good, TLS encryption	Good, dynamic shared secret encryption	Good, TLS encryption	Moderate. In the docking network traffic is unauthenticated, the rest is as secure as the AAI is.
Compliance with privacy regulations	P2P, no auth. payload disclosed	payload disclosed to intermediate RadSec nodes	P2P, no auth. payload disclosed	P2P, no auth. payload disclosed	P2P, no auth. payload disclosed. Authorisation not based on identity but on controlled release of attributes (e.g. role).
Peer identity Verification	Verification of peer identity based on roaming federation hierarchy	Verification of peer identity based on roaming federation hierarchy	Verification of peer identity based on roaming federation hierarchy	Verification of peer identity based on roaming federation hierarchy	Verification of peer identity based on AAI federation metadata.

Requirement	Diameter	RadSec	RADIUS-DNSSEC	RadSec + DNS	Web redirection + AAI
Openness	Very good, fully standards based	Medium: not yet based on standards, but work started	Medium: based on non-standard combination of standard + proprietary key exchange	Medium: not yet based on standards, but work started	Good: AAIs based on open standards and open source implementations.
eduroam integration	Good possibilities, but no good implementations	Good possibilities	Good possibilities	Good possibilities	Low. AAI not available, co-op 802.1X and Web-based for further study
Scalable	Yes, due to P2P nature	like RADIUS	Yes, due to P2P nature	Yes, due to P2P nature	Moderate. Transactions are P2P, but the AAI needs to distribute metadata on each home organisation.
Ease of use	Medium, new standard	Good, close to existing RADIUS configuration	Medium, close to existing RADIUS configuration but DNSSEC is new	Good, close to existing RADIUS configuration	Very good. Users familiar with web UI. Administrators need to maintain only one architecture.
Robust	Yes, due to P2P nature	Yes, due to reliable transport protocol	Yes, due to P2P nature	Yes, due to P2P nature	Yes, due to P2P nature.

Table 3.2: Authentication models and their mapping to JRA5 requirements

Naturally, in a configuration with mixed solutions, such as Diameter with RADIUS legacy, the mapping of requirements will also be mixed. All the concepts that were presented in this chapter show a good promise. However, for deployment in the next generation of eduroam, the availability of implementations is a major factor

in the deployment decision. Unfortunately, most of the concepts presented here have none or incomplete implementations available. There are only two candidates where implementations exist:

Diameter has one general-purpose implementation in the commercial product NavisAAA, but is considered to be in an early stage by its developers and is not compliant with the relevant RFCs. Unfortunately, Diameter is not at all available for NAS devices, so that a RADIUS to Diameter packet translation is required in any case. Radiator announced Diameter support recently, but this has not been tested yet.

The second candidate, RadSec, also has only one implementation available at the moment, from the vendor Open Systems Consultants. This implementation gives the user access to the source code after buying it, which makes bug fixing easier than in a completely closed-source product like NavisAAA. Also, implementing the RadSec extensions into a plain RADIUS server is a rather easy task, so that more implementations, even in NAS devices are likely to appear as soon as the protocol is formally specified. A specification effort is underway, and the FreeRADIUS team is currently drafting a second implementation.

This situation made RadSec the most promising solution. Therefore, intensive tests were carried out with the available RadSec implementation. These tests are described in the next chapter.

4 Evaluation of RadSec + DNSRoam

Out of the many choices that were described in chapter 3, using RadSec potentially in combination with DNS-based peer discovery (called “DNSRoam” by the vendor) was selected as the most promising alternative. At the same time RadSec seems to be the only feasible solution for now.

In order to test these new technologies available within the Radiator implementation and to see how they would fit in the current eduroam infrastructure, a testing group was created to investigate the new protocol features and the implementation stability. Three possible scenarios that could improve the traditional hierarchy were sketched and tested.

In the tests three levels of authentication servers were introduced, replicating the traditional eduroam hierarchy as described in chapter 2.

The tests were conducted in three subsequent phases, each eliminating bigger parts of the hierarchy and using more of the new Radiator features RadSec and DNSRoam. The first test was a duplication of the current hierarchy, where the RADIUS servers were replaced with RadSec servers. In the second phase, the federation-level servers used dynamic peer discovery for the other federation-level servers, which eliminated the need for a central confederation root server. In phase three, institutional servers were set up to detect the home server directly, which also made the federation-level servers obsolete. The last two phases also included a fallback behaviour in case a server could not be dynamically discovered (as would happen if using a non-RadSec capable server). The following sections describe in detail the three test phases:

- Duplicating the current hierarchy with RadSec
- Peer discovery for the federation-level servers
- Peer discovery for all servers

4.1 Description of the evaluation

4.1.1 Phase 1: Duplication of the RADIUS hierarchy with RadSec

The initial test was to replace the RADIUS (UDP) protocol with RadSec (using TCP and TLS). The intention was to verify that all functionality in use with RADIUS is also available with RadSec and to create a setup similar to the current situation with RADIUS.

Peers were statically configured and unknown realms were forwarded to servers higher in the hierarchy using static definitions, up to the point where the realm or a part of would be known.

In order to keep the administrative and trust domains the same as in the traditional setup, a node has to trust the adjacent nodes in the hierarchy. For a federation (operated in eduroam most often by an NREN), this implies having two trust domains: one between the federation and the eduroam confederation root servers, and one between the federation's server and its institutions. In order to ensure trust between entities, NRENs often have their own PKI in place. It makes sense to use such a PKI setup in the trust domain between the institutions and the NREN. However, the existing Intra-NREN-PKIs have no direct existing relationship to other NRENs and institutions. Therefore the PKIs were not disseminated to other NREN servers, but instead a small root-level CA was created for the tests in order to not only authenticate the NREN servers with the top level server, but also to authorise them (and with them their connected institutions) as members of the confederation.

There are two ways to split these PKI domains in the RadSec implementation. Either multiple instances of the RadSec server can be started on different ports (each with its own TLS certificate settings), or the RadSec server process on the NREN servers can be configured to allow multiple CAs (i.e. its own intra-NREN CA and the root-level CA). The latter alternative has a significant drawback: when the server is contacted by another server, it is unclear which of its server certificates to present to the other server. This could be worked around by selecting the certificate based on the IP address of the incoming request, but this is considered a suboptimal solution.

Apart from these design choices, there were no significant technical difficulties during the first test scenario. The overall impression of the RadSec testing group was that Radiator's RadSec implementation is mature enough to fulfil the requirements for replacing the current RADIUS hierarchy, albeit only on federation servers where local monitoring tools permit a close monitoring of the new protocol, in case more bugs show up that could bring the server down.

4.1.2 Phase 2: Dynamic discovery of TLD servers

The second step in the experimental setup was to use only federation-level servers for determining the next hop in the authentication chain, thus eliminating the root server. Appropriate TLD entries were created in a centralized DNS zone, and the DNSRoam component of Radiator was configured to lookup the TLD part of the realm under <tld>.test.eduroam.org. The tests were conducted using plain DNS, without DNSSec validation,

because the RadSec implementation in Radiator was not yet prepared to do DNSSec query validation. Since the peer validation does not rely on secure DNS lookups, this was not an issue.

Using a sub-domain of test.eduroam.org implied that the realms of incoming authentication requests had to be rewritten to match the lookup scheme. For example, stefan.winter@restena.lu was rewritten to stefan.winter@restena.lu.test.eduroam.org and this realm was then looked up in the DNS, with the outcome of establishing a connection to the corresponding server (based on an entry restena.lu.test.eduroam.org in the DNS).

The trust issues mentioned in phase 1 related to the use of PKIs remained the same (different PKIs used between the NRENs and between a NREN and the institutions). However the issue of peer certificate attribute validation was even more important compared to the first scenario, since there was no room within Radiator for configuring this on a per realm basis.

Requests from NRENs not yet configured for DNSRoam or to NRENs with no configured DNS entry were proxied through the top level server just as in phase 1. This showed that backward compatibility with NRENs not using DNSRoam was working.

A number of bugs showed up with DNSRoam. Especially at the time when one of the NREN servers was unavailable for different reasons, the other authentication servers started to show instabilities. Besides the required techniques within the authentication servers (realm rewriting), additional dependencies are introduced with a lookup service like DNS. During the tests DNS-specific problems showed up, that where not foreseen and not related to the authentication-servers. These were due to incorrect zone transfers of SRV records on some server implementations, and could be sorted out by using properly working nameservers. Even though these DNS issues are not a problem of RadSec or of the DNSRoam component in the authentication servers, they still add a point of failure that may lead to service degradation in production use.

During the second phase multiple issues regarding DNSRoam were reported to Radiator's vendor, Open System Consultants, and the problems found were swiftly patched. Yet, the frequency of the appearance of bugs suggests that there are more of them to be expected in the current implementation.

The overall impression of the testing group was that the DNSRoam implementation is still too problematic to be considered for production use. The concept itself shows a great promise though and should be observed thoroughly.

4.1.3 Phase 3: Dynamic discovery for all peers

In the third phase six NRENs participated in the tests and 18 institutional servers were available for testing the completely meshed setup. In this scenario the goal was to have direct connections from one authentication server to another, without ever unintentionally proxying the request through a third server. Instead of using a dedicated zone for finding a peer, the realm was looked up directly in the DNS zone of the institution that was to be contacted. If no records for the realm were found, the request was forwarded to a default route -- which in

the test setup was the top-level server still configured as in the previous scenario, thus again providing a fallback mechanism for non RadSec+DNSRoam enabled servers.

One of the difficulties in this completely meshed setup was the use of PKIs. Since every NREN had its own CAs and each server from one NREN could contact each server from the other NRENs, for every participating server a list of all involved root-certificates had to be created. Updating the list (along with the associated Certificate Revocation Lists, CRLs) requires some distribution. The use of CRL certificates proved to be rather complicated since in the Radiator implementation CRL validation was only possible as long as CRL files for *all* involved CAs were available. Both CRLs and a proper distribution mechanism are of course required for daily use of the tested scenario.

Questions were raised regarding the use of NREN CAs in this setup. For an NREN it might be hard to keep multiple CAs running, not only because of the costs involved. A general-purpose CA certificate however does not contain sufficient information about the federation to whom a server belongs, so there should be other means for providing this information. This information could for instance be published in DNS too – if that is properly secured with DNSSec – or put into a different lookup service. Another way is to put some attributes (OIDs) in the certificate while signing. This allows the CA server to sign a certificate specifically for a federation (e.g. eduroam) but it requires additional changes in the software (along with a definition of these extensions) in order to do this. Furthermore, the NREN CA's policies would probably need to be modified to allow inclusion of the new extensions into certificates.

A further, non-technical, hurdle is that some system administrators are (likely to be) reluctant to having their authentication server's ports wide open for the world, without any firewall or filtering mechanism in place. While static and hierarchical configurations only communicate with a limited amount of hosts for which specific firewall rules can be set up, dynamic setups have to send and receive traffic to and from unpredictable addresses. This is not necessarily a problem, given that peers are validated "at the doorstep" using TLS sessions. Still, the hosts are exposed to the general internet which allows a new set of possible attacks, for example Distributed Denial of Service (DDoS) attacks.

The testing group agreed that it is not an option to pursue a completely dynamic mesh setup in the near future.

4.2 Conclusions

The tests showed that moving to a new protocol is not an easy task. Even small protocol enhancements like using the RadSec extensions already require a set of design decisions and introduce new possible points of failure. Moving further on to more advanced techniques, i.e. DNSSec peer discovery, brings in a whole new level of complexity, both conceptually and implementation-wise.

The only extension to the current eduroam hierarchy that could be agreed upon was to deploy the RadSec extensions in a production environment on a limited set of hosts, under close monitoring to ensure the stability of the service and backward compatibility. The first servers to be considered for RadSec deployment are those that already use the Radiator implementation, since in this case no additional costs for software licenses are required. The availability of an open-source solution that also offers the RadSec features is desired and actually

currently in development. The popular (and widely deployed in eduroam) open-source RADIUS server FreeRADIUS is currently being extended with RadSec features by its main developer.

With two independent implementations of RadSec, a logical next step to bring the RadSec idea forward is the development of an open standard that describes RadSec and its interoperability requirements between implementations in a renowned format. So, in parallel to the FreeRADIUS developments a standardisation effort is underway in JRA5, with support of the TF Mobility, that might eventually lead to an IETF RFC.

A very detailed description of the tests that were conducted as well as a survey of the participants of the tests can be found under [Radiate].

The exact conditions for an initial RadSec deployment in the eduroam infrastructure are described in the following chapter.

5 Architectural Overview of eduroam-ng

The result of the evaluation of the various possible protocol enhancements in chapter 3 and of the conducted tests, described in chapter 4, resulted in choosing the RadSec protocol. RadSec offers a smooth transition from plain RADIUS servers and is also fully backward compatible with existing NAS devices. The currently available implementation in the Radiator product has proven to be stable enough to allow prototypical deployment in a production environment. As a further advantage, vendor support has proven to be excellent and fast, so that possibly arising problems can be expected to be fixed quickly.

RadSec will initially be deployed without the additional DNSRoam feature because it is not considered stable enough yet and because there may be other means for dynamic peer discovery that are better suited for the roaming purposes. This means that eduroam-ng in its first phase will still be a static hierarchy just as the RADIUS hierarchy. Moving to dynamic peer discovery will be evaluated at a later stage. A first possible evaluation scenario for DNSRoam is described in chapter 5.2.4.

Since not all participants of eduroam are willing to use Radiator and no other RADIUS server implementations provide RadSec support yet, it is not possible (nor advisable) to force a move of the complete infrastructure to RadSec. Therefore, establishment of a mixed RADIUS/RadSec environment is envisaged.

The coexistence of RADIUS and RadSec has to be ensured on all layers of the hierarchy. The various components of the mixed hierarchy are detailed in the section below. As a general rule that applies to all servers in the eduroam hierarchy, the following recommendation is given:

- RADIUS servers running the Radiator implementation SHOULD become RadSec nodes according to the following requirements, if the local infrastructure permits close monitoring and testing of the new technology.

- Other servers MAY become RadSec nodes.

All RadSec nodes, independent of the layer they are deployed at, must follow the following requirements:

- They MUST be reachable by TCP for RadSec communication.

- When communicating with another RadSec peer, the communication MUST be encrypted via TLS.

- They MUST also be reachable for plain RADIUS connections on UDP unless all of the authorised communication partners are able to speak RadSec, in which case RADIUS support is optional.

- RadSec nodes MUST negotiate the protocol to be used by first trying to establish a RadSec connection and only if this is not possible fall back to RADIUS.

- They MUST be configured in a way that the RadSec server component accepts at least the same peers as the RADIUS server component.

- the shared secret for RadSec connections to be configured as "eduroam-ng" (without the quotes) is obsolete, but has to be present to calculate the RADIUS payload authentication.

These requirements ensure that all RadSec servers have a common ground to communicate on, that backwards compatibility to existing RADIUS peers exists, that communication is tried in order of protocol preference and that a later move of a plain RADIUS peer to RadSec does not require administrative changes on the already RadSec-enabled server.

5.1 Components

The hierarchy consists of several layers, where each layer has different properties with regards to number of connected peers, willingness to move towards RadSec, and the level of administrative control over the connected peers.

These components are denominated as the "confederation level" (the connection between the TLD servers and the root server), the "federation level" (connections between TLD servers and all connections to all the connected servers below) and the "edge level" (connections between NAS devices and the first server they are connected to).

5.1.1 Confederation Level

On the confederation level, a relatively high number of Radiator installations is already present, which makes a move towards RadSec less cumbersome. Additionally, the number of servers involved is small, which makes the TLS certificate handling simple. The following procedure is recommended for the move towards eduroam-ng:

The interconnection and trust establishment on the confederation level is as follows:

- The certification authority and hierarchy for eduGAIN (the “eduGAIN CA”) is issuing the server certificates for eduroam federation-level servers. NRENs, which are connected to the eduGAIN CA with their own CA issue the certificates for their RadSec servers themselves. NRENs without an own CA receive certificates from the eduGAIN CA directly.

- As a distinction between eduroam servers and eduGAIN servers for other purposes, the eduroam server certificates contain an URN namespace that identifies the servers as part of the eduroam confederation. For the naming scheme of the URN the eduGAIN CA rules apply.

- The root servers are configured to accept RadSec connections from every peer presenting a valid certificate from the eduGAIN CA with the correct URN (the URN identifies a server as federation-level server, other server types are not allowed on this level).

- eduroam server administrators are responsible for keeping an up-to-date list of revoked certificates of the eduGAIN CA and all connected CAs in the certification hierarchy.

- The federation-level servers only accept RadSec connections from servers:

  a) that have a certificate from the eduGAIN CA with a URN from the eduroam namespace, b) whose URN in the certificate identifies them as a root server. If a TLD server also serves as RadSec node towards its federation servers, additional servers may be accepted. In case a root server's certificate gets compromised and its certificate is added to the eduGAIN CA's CRL, this CRL gets published as soon as possible and the TLD server's administrators get notified of the update so that they can take appropriate action and the compromised certificate gets void on the entire hierarchy as fast as possible.

These operational procedures allow for an easy integration of new RadSec nodes: it is only necessary for the eduroam working group to issue a new server certificate to the node and configure the new realm that is to be proxied.

5.1.2 Federation Level

On the federation level, it can be expected that a wide variety of different RADIUS server implementations is in use. Federations possibly interconnect a large number of connected institutions, which in turn may have an arbitrary number of servers underneath. This makes it difficult to give specific recommendations about deploying RadSec. It is clear that a certificate and CRL distribution service is required. The exact details are left to the federation, but the following restrictions apply:

- Revoked certificates MUST be put in a CRL and the CRL must be distributed in a timely manner.

- There MUST be a means to differentiate between authorised eduroam-ng server certificates and other server certificates that the federation's CA might possibly also issue.

- Since the TLD server has to handle connections from both the eduGAIN CA and its own federation servers, it is required to either listen on two separate network ports, each port negotiating connections with one server certificate, or use a certificate from the eduGAIN CA for their federation servers.

Using the eduGAIN CA as described in section 4.1.1 is recommended. If no NREN-level CA to connect to the eduGAIN CA exists, it is recommended to make use of the eduGAIN CA directly to get the needed certificates.

5.1.3 Edge Level

NAS devices are usually closed, proprietary products that don't offer an easy path towards new, experimental features. As a consequence, NAS support for RadSec is unlikely in the near future. The impact of this is that eduroam-ng will very likely still employ RADIUS at the edges. Experiments with open NAS products (for example, Open Source Software based Access Points) are planned for the future.

5.2 Routing non-country-bound domains in the eduroam hierarchy

Static routing of requests through the eduroam hierarchy relies on the weak assumption that the roaming realms match the organisation's country-code domain name in the DNS system (the realm ends in a country-code top-level domain, or ccTLD). The current routing decisions in the eduroam hierarchy don't provide a scalable solution to a problem where different organisations that are connected to different top-level RADIUS servers have the same top-level domain name component.

For example: a Slovenian university of Maribor which is connected to the Slovenian federation-level RADIUS server might have a domain name "uni-maribor.edu" and a Spanish university of Malaga which is connected to the Spanish federation-level RADIUS server might have a domain name "uni-malaga.edu". A request to authenticate an internationally roaming user coming from such organisations goes up the tree hierarchy and when it reaches the confederation root server, the root server must decide to which federation to proxy the request. The decision can obviously not be based on the TLD ".edu" because this TLD does not contain a country-code. For example: should the root RADIUS server proxy request for user "john.doe@uni-maribor.edu" to the Spanish or to the Slovenian federation?

There are the following possible solutions:

5.2.1 Using the country's TLD as realm

Organisations register and use a domain name in a domain namespace which matches their national RADIUS tree hierarchy namespace. In the example above, uni-malaga.edu gives up using the .edu domain (at least for eduroam purposes) and uses a realm name ending in ".es" instead.

For some smaller organisations this might be a practical solution and no change of the current eduroam system is required. However most of the large universities have a public image in which the domain name plays a big role. Adding or changing a domain name is out of the question for them.

5.2.2 Add all required information to the root servers

Confederation root servers are proxies for authentication and accounting requests based on the full domain name for cases when the realm name cannot be connected to a ccTLD uniquely.

This way organisations could use whichever domain they have. However for every organisation in the world that would connect to eduroam with a non-ccTLD realm, a reconfiguration of the root RADIUS servers would be required. This solution doesn't scale well enough and introduces complexity at the systems most critical points of failure. It however requires little change in the overall eduroam system.

5.2.3 Introducing a TLD server for non-ccTLD realms

A special RADIUS server is set up for every top-level domain that is not linked to a country (a so-called gTLD) for handling of domain namespaces that don't match existing RADIUS hierarchy tree namespace.

This solution would require setup and running of another TLD RADIUS server for organisations from different federations and with variations of eduroam technical specifications. Since some NRENs filter incoming and outgoing RADIUS attributes during the authentication process, this solution introduces a new problem: it is very hard to do country-specific filtering of sent attributes and thus enforcements of certain federation policies on the country level. Administratively it would be very hard to operate this server since every connected institution might require different filtering rules, however the rest of the eduroam system would be unaffected.

5.2.4 Dynamic discovery of the routing path with DNSRoam

The current eduroam architecture is extended to include a mechanism for dynamic peer discovery for institutions whose realm ends in a gTLD.

Currently only the DNSRoam technology is available for dynamic peer discovery as is described in chapter 4: “Evaluation of RadSec and DNSRoam”. Since the tests have shown that this technology isn't mature enough for a broad deployment, a mixed approach is proposed to solve this problem: DNSRoam is to be used only by the root RADIUS servers to locate the matching TLD server for a given domain name, and only for those cases where the usual ccTLD forwarding rules can not be applied.

An organisation with a roaming domain outside their NREN’s ccTLD namespace would still connect to their national TLD server. This server would have the domain name configured and the roaming inside the country would work with no change. However for the international roaming, the root RADIUS servers would use the DNSRoam mechanism to locate the correct federation-level server and deliver the international roaming requests to the correct federation-level server.

This way DNSRoam would need to be configured at the root RADIUS servers for roaming domains that don't match their NREN’s namespace. DNS records from for the participating institutions would need to be set up by the institution. The actual authentication payload could still travel either via RADIUS or RadSec, depending on

the NREN server RadSec-readiness. The impact on national eduroam hierarchy would be minimal, but the complexity of using dynamic discovery would be added at the system's most critical points, the root servers.

In order to keep stability of the production root servers, this complexity could be off-loaded to another (third) root server for use by gTLDs only, and the DNSRoam mechanism would be running only on that server.

5.3 Trust management

The eduroam-ng architecture described in this chapter introduces two new elements influencing trust management:

1) PKI used for authenticating RadSec connections and

2) possible dynamic routing path discovery via DNSRoam.

The PKI can be used both for statically configured connections within the RADIUS/RadSec hierarchy and for authenticating dynamically discovered peers. In the former case the PKI just replaces the shared secrets installed by RADIUS servers' administrators.

Replacing shared secrets with PKI effectively enables the dynamic routing path discovery. Because DNS, the only available global information system capable of storing eduroam routing information, lacks data authentication (time schedule of the DNSSec deployment is far behind the needs of the eduroam project), the PKI serves as the trust-providing service for authenticating peers of dynamically discovered connections. This means that although the route discovery service itself is vulnerable to DNS-targeted attacks, the peers are able to verify each other's identity and discover possible forged routes.

The federation manages the trust by several activities including:

- managing the URN namespace designated to identify eduroam realms authentication servers,

- defining requirements and rules for certificate authorities,

- accrediting certificate authorities for acceptance by eduroam nodes.

The eduroam-ng architecture assumes co-existence of RadSec (PKI secured) and RADIUS (shared-secret secured) connections. This implies that until all connections outside of an institution are secured by PKI the perception of the trust provided by the infrastructure by visited institutions as well as by users remains the same as in the current RADIUS-hierarchy based configuration, i.e. the trust is still transitive and the number of transitions as well as the participating parties are not directly known.

5.4 Policy

The eduroam ambition is to establish a full fletched service that people and organisations use and rely on. A formal policy governing eduroam is necessary to ensure that the already existing trust in the system remains stable and can be further developed.

Different contractual forms exist on the federation level. NRENs may include the eduroam service into their usual service portfolio or base it on a special purpose contract with the participating institutions. On the confederation level the best method is a policy document signed by all participating NRENs. By signing such a document, a NREN commits to providing a certain level on security and to the establishment of appropriate operational procedures to obey the needed obligations and responsibilities inside of the federation.

The regional policy for Europe [DJ5.1.3,2] has been developed in GEANT2 Joint Research Activity 5 (JRA5) and will provide the framework into which the various national eduroam policies will have to fit. The JRA5 policy deliverable also provides guidelines for a national eduroam policy as an appendix. This will hopefully ease the job of writing national policies, while being at the same time an attempt to render the policy-landscape as homogeneous as possible.

5.5 Operational model

The interactions in eduroam-ng are very similar to those described in section 2.4. In the general case, a supplicant communicates with the authenticator element of the user's home organisation for authentication and attribute exchange. To perform these operations, the messages pass through a series of authenticators that establish a virtual communication channel between the 802.1X supplicant of the user and its home authentication server.

There is a small technical difference between eduroam and eduroam-ng: the authenticator elements are – at least partially – RadSec servers instead of RADIUS servers. The main consequence is the way of managing trust in the communications between authentication server peers. In the current eduroam architecture, the communication between two servers is based on UDP and security is established using a shared secret key that is negotiated in a static way between the server administrators. In the case of RadSec, the communication between two servers is based on TLS tunnels over TCP, so security is guaranteed with the traditional use of PKI in TLS.

The eduroam-ng interactions for authentication, attribute exchange and authorisation are equivalent to the ones in eduroam described in section 2.4 of this document. The interactions between authentication servers depend on whether the communication takes place between two RADIUS nodes, a RADIUS and a RadSec node or two RadSec nodes. Therefore, the interactions in the Home Location Service are different from chapter 2.4 and described in the following paragraph.

Home Location service

The home location service has to be able to locate the authentication and attribute release services of the user's home organisation for the different phases. The information that is taken into account in this task is the user's realm. In the eduroam-ng architecture, there are three possibilities for a server to determine the home location:

- Static route solution: It is a solution equivalent to eduroam (as described in point 2.4.2). The communication is established through chains of RADIUS or RadSec servers, organised in a hierarchical way, that proxy EAP messages to the next server in the chain based on static routes stored in the configuration of the servers. This solution can be used both with RADIUS and RadSec connections.

- Dynamic discovery solution: Based on dynamic discovery protocols (as DNSRoam), the RadSec server of the home organisation is located in a dynamic way (using DNS entries that point to the appropriate server). The actual authentication communication with the discovered server will only begin after the discovered server has proven its identity and its authorisation with a valid eduroam certificate. This solution is only available for connections where RadSec servers are in use on both ends.

- Mixed solution: For discovering the RadSec server of the home organisation a chain of servers is used (as in the static route solution), but at some levels dynamic discovery solutions can and probably will be applied (for example between TLD –Top Level Domain - servers). This solution makes it possible to integrate legacy RADIUS servers into eduroam-ng, because between any two hops of the communication one of the two rules mentioned above can be applied.

6 Configuration Diagrams

This section presents three typical configurations depicting the eduroam architecture.

6.1 No direct interaction between authentication servers

The currently used eduroam configuration is shown in Figure 6.1. The service provider's (e.g. visited organisation) and the home organisation's RADIUS servers are connected to the respective national RADIUS hierarchies through the RADIUS proxy servers at the federation level. Federation RADIUS proxy servers are connected to the European top level proxy servers in eduroam. Each and every authentication request passes through all the elements of the infrastructure and there is no direct interaction between authentication servers. Authorisation is performed locally, by the service provider, based on the available information and the local AAA system.

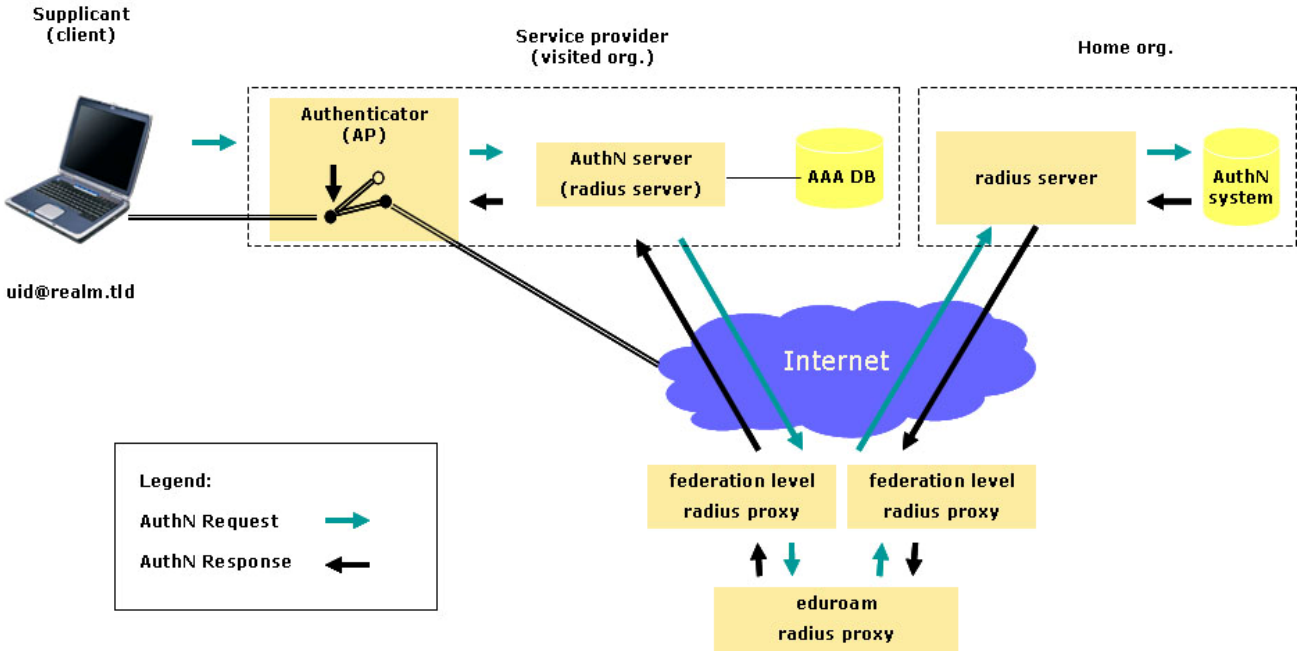


Figure 6.1: eduroam configuration: no direct interaction between authentication servers

The following steps are performed:

1. The user requests network access through the service provider's (SP) AP, presenting his credentials (user identification (uid@realm.tld) and respective password).

2. The authentication request reaches SP's RADIUS server where realm.tld as part of the userid is analysed. Based on the result, the request is transferred to the SP's corresponding national RADIUS proxy server.

3. The national RADIUS proxy server, based on the analysis of the realm.tld part of the userid forwards the request to the eduroam proxy server (confederation top level).

4. The eduroam proxy server forwards the request to the corresponding national RADIUS proxy server based on the analysis of the realm.tld part of the userid.

5. The national RADIUS proxy server discovers the correct home organisation by analysing the realm.tld part of the userid and forwards the request to the corresponding RADIUS server.

6. The RADIUS server at the home organisation performs the authentication step and sends back the authentication response.

7. The authentication response travels back through the RADIUS hierarchy until it reaches the SP's RADIUS server.

8. In case of positive authentication response, the SP's RADIUS server performs it's authorisation step and grants the network access to the user.

6.2 Direct interaction between authentication servers

The possible configuration in the case of direct access between authentication servers is shown in Figure 6.2. The service provider's authentication server is capable of sending the home location service request to the corresponding higher level servers. Once it discovers the location of the home organisation's RADIUS server, the SP's RADIUS server sends the authentication request directly to the corresponding address. Direct interaction between authentication servers is established. The authorisation step is still performed locally by the service provider, based on the available information and local AAA system.

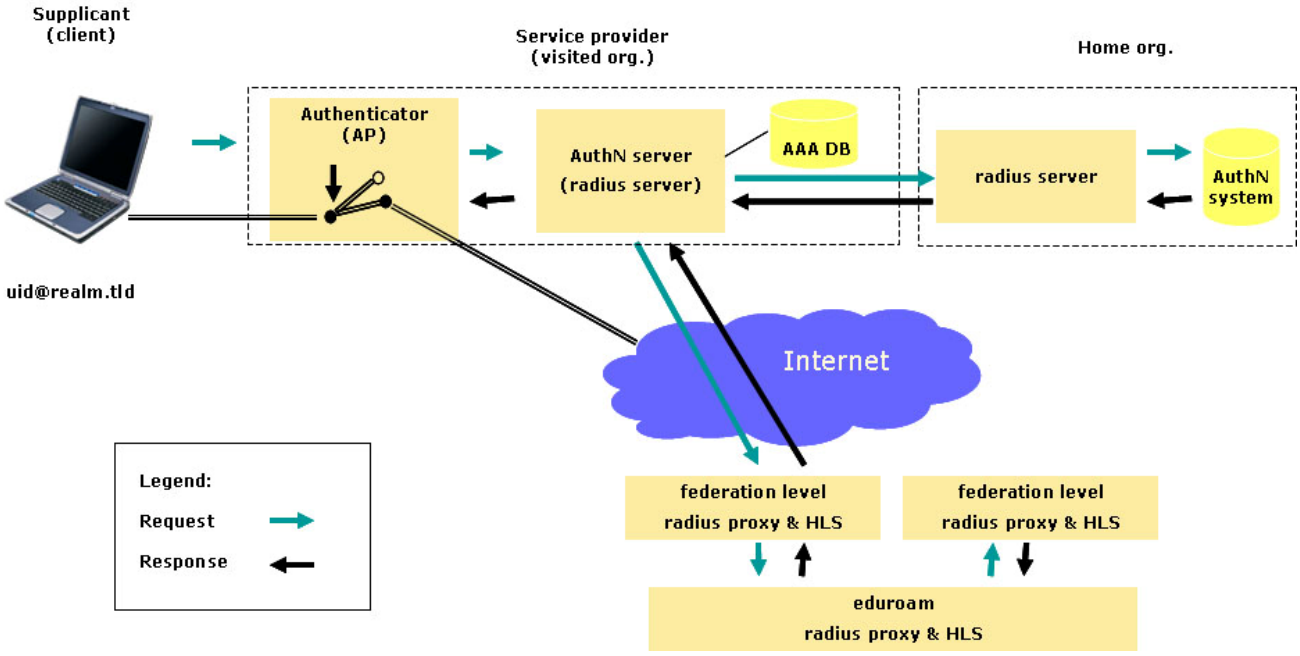


Figure 6.2: eduroam configuration: direct interaction between authentication servers

The following steps are performed:

1. The user requests the network access through the service provider's (SP) AP, presenting his credentials (userid (uid@realm.tld) and the respective password).

2. The authentication request reaches the SP's RADIUS server where the realm.tld part of the userid is analysed. Based on the result, the request for the location of the corresponding home organisation's RADIUS server is transferred to the SP's corresponding national RADIUS proxy and HLS service.

3. The national RADIUS proxy server, based on the received information, forwards the request to the eduroam proxy and HLS service.

4. The eduroam proxy and HLS service forward the request to the corresponding national RADIUS proxy and HLS service based on the received information.

5. The national RADIUS proxy and HLS service respond with the address of the RADIUS server corresponding to the request. The response is forwarded back to the SP's RADIUS server.

6. The SP's RADIUS server sends the authentication request directly to the home organisation's RADIUS server.

7. The home organisation's RADIUS server receives the request, performs the authentication step and sends back the authentication response.

8. In case of positive authentication response, the SP's RADIUS server performs the authorisation step and grants the network access to the user.

6.3 Additional services (attribute exchange, authorisation)

The configuration with the additional services such as attribute exchange and authorisation service is shown in Figure 6.3. This configuration is based on the previous one, presented in 6.1 but. without the direct interaction between SP's and home organisation's services.

Attribute exchange or authorisation services are used to enhance the AA process. Therefore the authorisation decision at SP's side could be based on the additional attributes retrieved form the home organisation's IdP system or on the response of the corresponding AuthZ service. The SP's and home organisation's servers must be able to perform the additional services. The eduGAIN infrastructure will be used to fulfil that task.

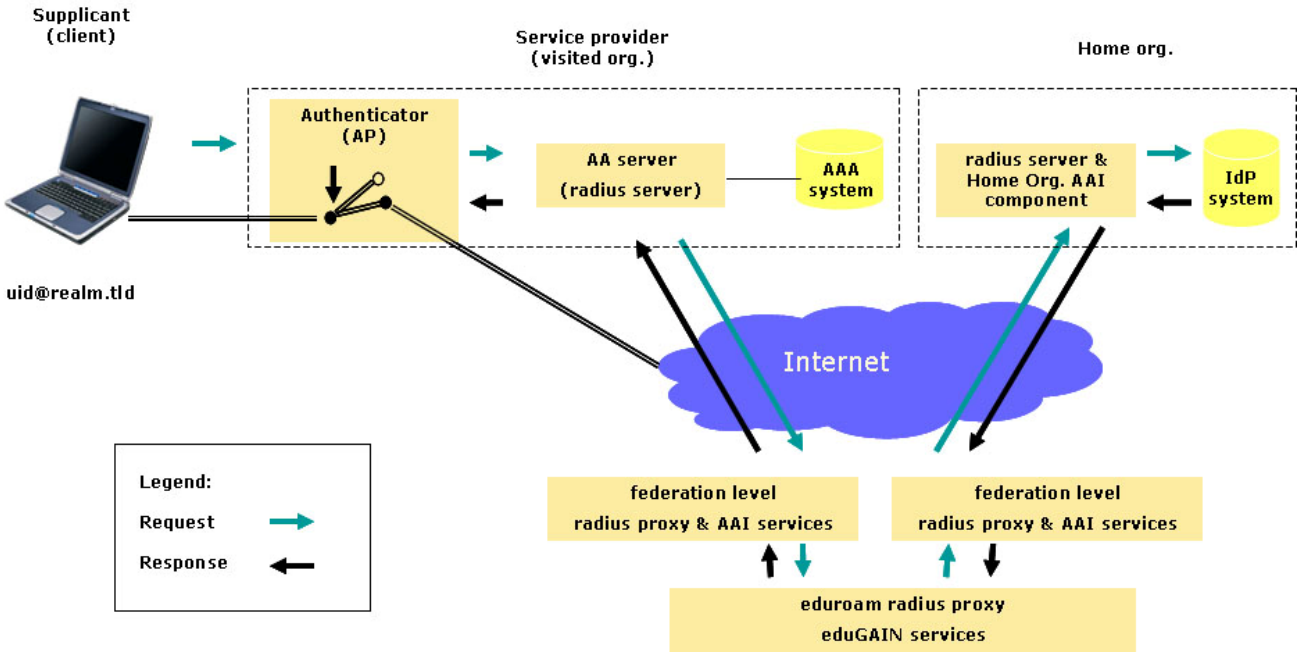


Figure 6.3: eduroam configuration with additional services

The first steps performed for the authentication process correspond to the steps 1-6 listed in 6.1. After the authentication process has been completed, the SP's AA service issues the attribute or authorisation request following a similar path but now using the eduGAIN-based services.

7 Use Cases

7.1 The Generic Use Case

The user named Tom studies anthropology at Institution-1. He always carries his laptop with him as the wireless network has been deployed almost all over the campus. Every time he wants to use the network, Tom has to authenticate himself using his personal credentials (email address and password).

7.1.1 The logon procedure, general description (EAP-TTLS): using eduroam authentication at the identity provider (home institution)

Tom finds a spot where the wireless network called 'eduroam' is visible. The local wireless access point acts as a gatekeeper, asking for Tom's credentials on behalf of the network administrator.

The login process at Institution-1 is based on the 802.1X-protocol (eduroam standard) and the secure TTLS authentication protocol (open standard). The authentication server at Institution-1 (containing all users at the campus) is contacted by the access point on behalf of Tom. If he is authenticated he will be allowed to access the wireless network by the wireless access point. Before that he doesn't have an IP address and can therefore not interfere with network traffic in any way.

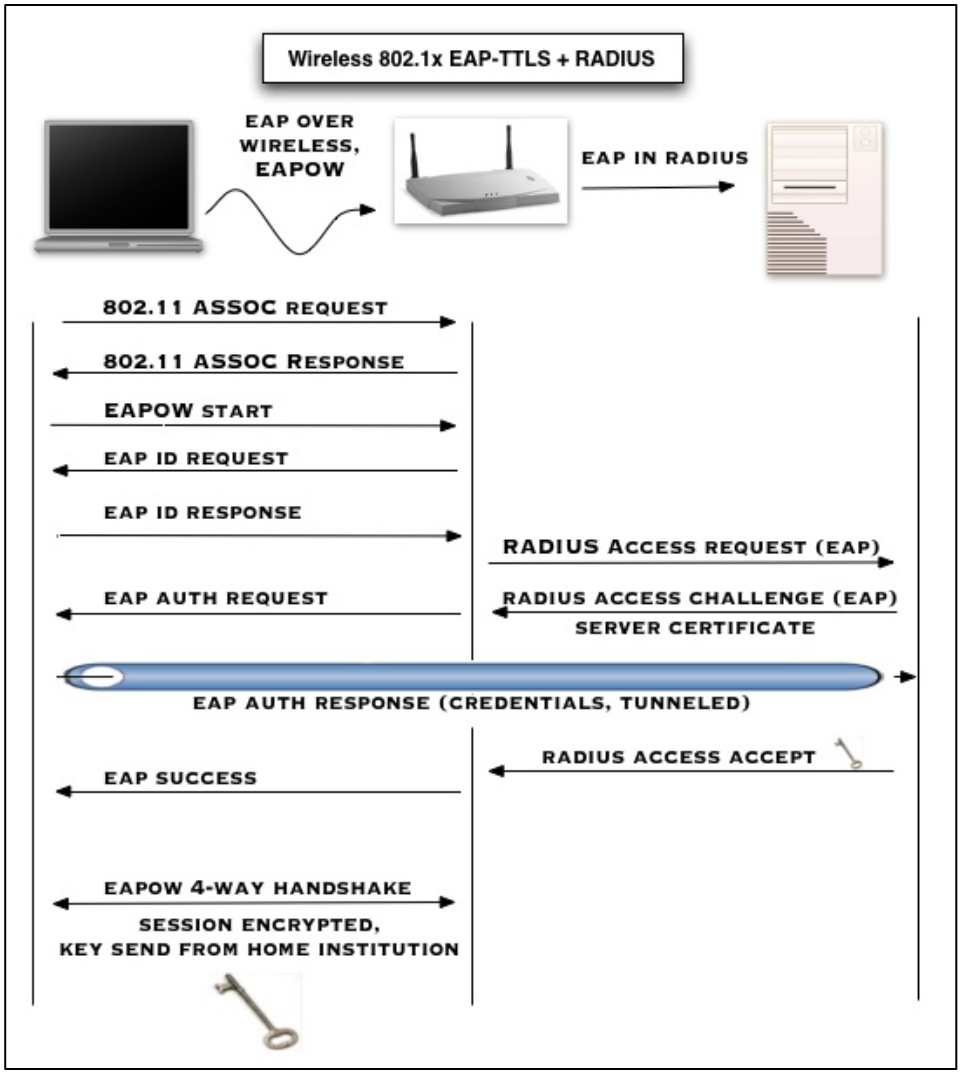


Figure 7.1: Wireless user authentication using 802.1X, EAP, TTLS and RADIUS

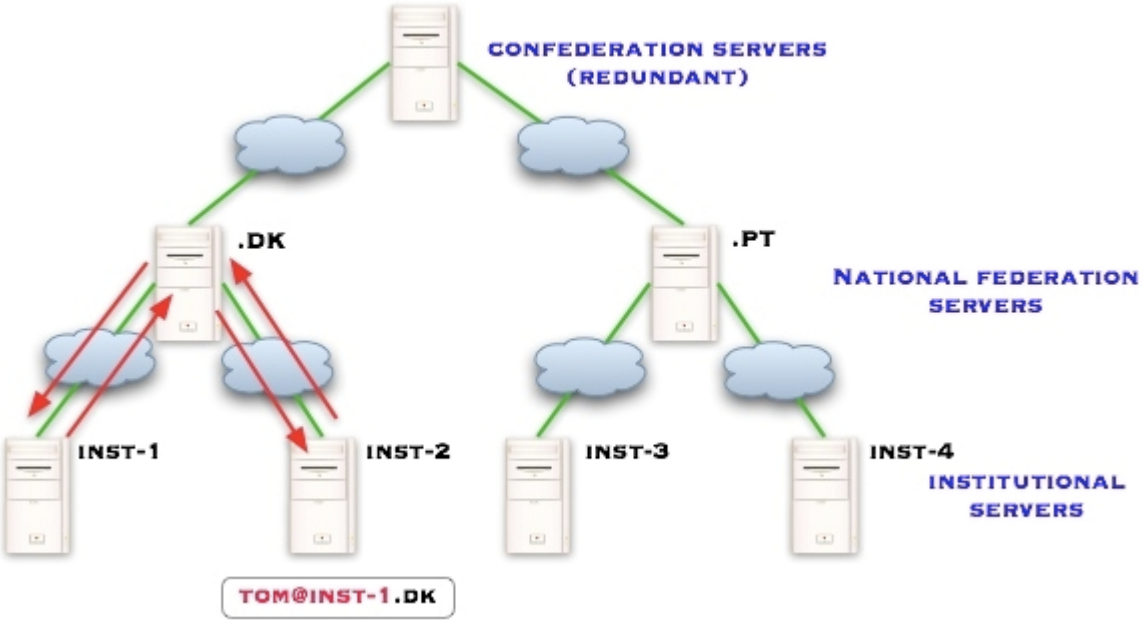
When the wireless access point associates with Tom's machine, it asks for his 'outer identity' (anonymous@inst-1.dk) which includes the realm of Institution-1. A message is routed via the RADIUS protocol to the authentication server, based on Tom's 'outer identity'. The authentication server sends back to Tom its server certificate, which can be used as public key for encryption of Tom's personal credentials. The encrypted message, containing Tom's username and password, can *only* be read by the authentication server. It cannot be eavesdropped, not even by the wireless access point or by any intermediate RADIUS server. When Tom's credentials have been accepted as valid, a message is sent (via RADIUS) from the authentication server to the wireless access point, which in turn allows Tom to access the network. In addition, keying material is sent to establish an encrypted and secure connection between Tom's machine and the access point. The connection is secured for as long as Tom is connected to that wireless access point.

7.2 Specific Use Cases

7.2.1 Using eduroam at another institution, same country (EAP-TTLS)

Tom has decided to listen to a series of talks at a neighbouring Danish university, Institution-2. He will go there every week for two months but does not intend to sign up - he just wants to hear the talks. Insitution-2 has of course a wireless network and does also use eduroam for authentication.

One difference, though, is that he will not be able to print or use any special services being a guest at Insitution-2. This is because Institution-2 has decided that these services are only available to people with direct affiliation with the institution. But Tom can use the web, check email and use his VPN-client to login at Institution-1.



eduroam message flow using the national RADIUS server hierarchy. Three levels of RADIUS servers exist: institutional, national and international. Red arrows indicate the authentication message path. The red part of Tom's email address represents the minimal information needed to authenticate him at his identity provider. Green lines represent 'shared secret' relationships between RADIUS servers, each relationships' traffic is protected using symmetric encryption based on the shared secret only known by those two servers.

Figure 7.2: Use Case: National Roaming

To Tom, when getting network access at Institution-2, everything looks the same as when logging in at home. Technically the only difference is that the national Danish hierarchy of RADIUS servers comes into play. Since the authentication server at Institution-2 cannot handle the realm 'inst1.dk' it passes the authentication request to the national server, which keeps a list of participating institutions, including Institution-1. The message can therefore be routed back home, and the general scenario described in 7.1.1 (as in the case within the home organisation) is replayed - only this time via the national RADIUS hierarchy. Again, when authenticated, Tom's wireless session remains encrypted, based on the cryptographic keying material provided by Institution-1.

7.2.2 Using eduroam at another institution, abroad (EAP-TTLS)

Tom has been asked to present his thesis in a poster session at a conference at Institution-3 to be held in Portugal.

Both Portugal and, in turn, Institution-3 have joined the European eduroam and therefore he can connect as usual - still being authenticated by Institution-1 back home. This of course gives the organisers of the conference (and the local network administrators) much less to think about as they know that all guests using eduroam will be authenticated users.

As if he was at home, Tom's laptop connects to the wireless network called 'eduroam'. Actually everything seems to work exactly the same way: he types in his credentials and is allowed access to the network.

What he doesn't see is that his authentication request is first sent to the local authentication server at Institution-3, then to the national eduroam-server in Portugal. As Institution-1 is unknown to the national server (which only knows about the .pt top level domain), the request is passed on to the European top level RADIUS server which keeps a list of participating countries. It knows which country to send the request to.

The Danish national eduroam server knows where Insitution-1 is and finally Tom gets authenticated. The acknowledgement message is sent back to the wireless access point at Institution-3, following the same route.

As always the traffic between the wireless access point and Tom's machine will remain encrypted during the entire session.

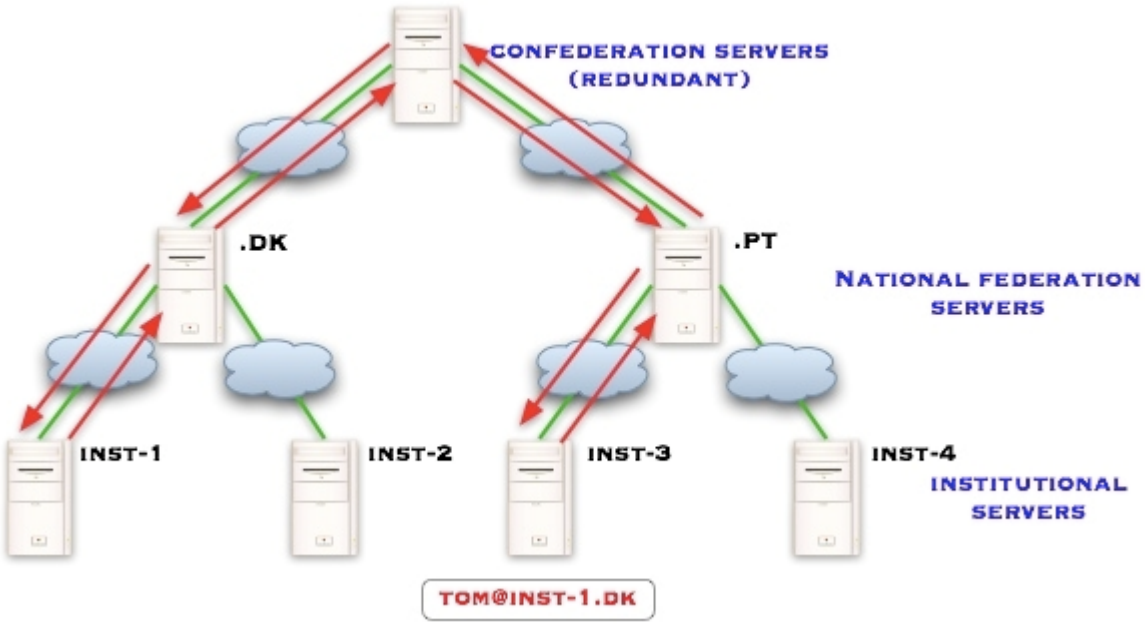


Figure 7.3: Use Case: International Roaming

8 Security and Privacy Considerations

8.1 General Considerations

DJ5.1.2 (Roaming Requirements) describes the required security and privacy as follows:

Reasonable security: The GÉANT2 roaming infrastructure eduroam-ng must provide a sufficient level of trust to all participating partners (NRENs, institutions). The resources involved are the corresponding networks (backbones, campus networks, department networks) together with the docking network (based on wireless or wired technology) as part of the network infrastructure that shall be protected primarily. Granting network access must be available for authorised users only and shall not put unpredictable risks on the network provider.

Data integrity: eduroam-ng must ensure the integrity of data transferred and processed in the entire infrastructure (user data, federated control information, payload). Providing confidence in the data integrity (users and administrators) is an essential element for establishing a fabric of trust in a distributed (and expandable) roaming infrastructure. To ensure the data integrity, a certain level of local maintenance is needed, in order to guarantee that only valid digital identities are used. A revocation procedure must be in place to handle cases of abuse by disabling rogue users.

Compliance with privacy regulations: When dealing with authentication and authorisation mechanisms, privacy becomes an extremely important area, both because of general public concern about this issue (specifically in the research and academic community), and because of the strict European and national regulations and general guidelines on privacy preservation (a dedicated documentation will be available later in the project). The infrastructure must avoid undesirable data leakage when performing AA interactions, and provide users with the ultimate control over what information about them is exchanged for what purposes. However, it may be helpful to point the roaming user to the local Acceptable Usage Policy (AUP) of the visited institution, if available.

Verifiability: The very nature of authentication and authorisation requires to keep a clear and end-to-end record of whom, when, what and why a given service was granted. The infrastructure shall be able to comply with any legal requirements concerning AA-actions provided by the infrastructure. To which degree this might involve data protection issues is discussed in the policy documents (DJ5.1.3, part 1 and 2).

8.2 Rules and Policies of Federations

The rather abstract requirements from the roaming requirements document [DJ5.1.2] have been made explicit in [DJ5.1.3,2] (Roaming Policy and legal framework) as:

- The security of the user credentials must be preserved and privacy regulations must be observed.
- *eduroam* MUST always provide trustworthy and secure transport of all messages traversing the eduroam infrastructure.
- User credentials MUST stay securely encrypted end-to-end between the personal device and the identity provider (home institution) when traversing the eduroam infrastructure. This ensures that they will only be used by the user and his identity provider.
- Confederation members (NRENs) and federation participants (institutions) taking part in eduroam MUST ensure that eduroam servers and services are maintained according to server build, configuration and security best practices to maintain a generally high level of security, and thereby trust, in the European eduroam confederation .
- The confederation members MUST ensure that the participating institutions are aware of their responsibility to establish an appropriate level of security.

These requirements contribute to the necessary security and privacy, together with the following operational requirements from the European eduroam service level agreement specified in the same document:

- Each confederation member joining eduroam MUST establish the necessary infrastructure to support eduroam services and to ensure that it is maintained according to server build, configuration and security best practices.
- Confederation members MUST ensure that their federation participants observe the security requirements of the European eduroam confederation policy.
- The federation participants are responsible for proper user management and the authentication and authorisation of eligible users only.
- eduroam resource providers MUST keep sufficient logging information to be able to correlate between a client's layer 2 (MAC) address and the layer 3 (IP) address that was issued after login, all relevant logs MUST be created with synchronization to a reliable time source.
- eduroam resource providers MUST deploy NASes that support IEEE 802.1X and symmetric keying using keys provided within RADIUS Access-Accept packets, in accordance with section 3.16 of RFC3580.

- eduroam resource providers MUST assign a single user per NAS port.
- eduroam resource providers MUST deploy NASes that include the supplicant's MAC address within the Calling-Station-ID RADIUS attribute within Access-Request packets.
- eduroam identity providers MUST select an EAP-type, or types, for which their EAP server will generate symmetric keying material for encryption ciphers, and configure their RADIUS authentication server to encapsulate the keys, in accordance with section 3.16 of RFC3580 (IEEE 802.1X RADIUS Usage Guidelines), within RADIUS Access-Accept packets.
- eduroam identity providers MUST log all authentication attempts; the following information MUST be recorded:
    - The authentication result returned by the authentication database
    - The reason given if the authentication was denied or failed

Two fundamentally different approaches to the security and privacy requirements exist:

1. Protect all elements involved in the eduroam authentication request and by a combination of technology, auditing and contracts guarantees that the required security levels are upheld.
2. Consider the structure as a whole as inherently unsafe and instead protect data end to end between the provider and consumer of that data as much as possible.

Possibility 1 can in practise only be achieved in a controlled environment with a limited number of elements to be controlled. In a highly dynamic environment with thousands of elements controlled by hundreds of administrative entities this is not a practical and scalable approach.

The main design principle of the eduroam-ng (and eduroam for that matter) architecture is therefore:

The roaming infrastructure is inherently unsafe and privacy sensitive data should therefore be only visible to the end-user and his home institution

This ensures that a breach of security at one of the participating institutions will not result in a downgrade of the security level of the system as a whole.

8.3 End-to-End Security

Protection of privacy-sensitive data

The key ingredient in providing the required protection of user data, especially credentials like passwords, is the use of the Extensible Authentication Protocol (EAP). As described in previous chapters, by selecting the

appropriate EAP-types it is impossible to eavesdrop on the EAP-communication between supplicant and authentication server. Furthermore authentication of the AS to the supplicant, in addition to the authentication of the supplicant to the AS (so-called 'mutual authentication') prevents rogue authentication servers tricking the user into providing credentials to a false server.

Authentication of the AS is typically done by having it present a X.509-certificate to the user. This does introduce a security risk, as users may be inclined to just trust whatever server and thus gain access to the network. Even though this may be considered as an end-user problem, it is advisable to provide means of preventing users from providing their credentials to untrustworthy servers, for instance by configuring fixed servers in the supplicant software, but that is beyond the scope of this document.

Another point of concern is that apart from the core information that is used to authenticate the user at the home authentication server, the RADIUS messages typically contain information that can be regarded as an unnecessary revelation of user data. For instance, additional RADIUS attributes may reveal data like the username, the type of user (university, k12 etc.) or the type of service the user requests (fixed, wireless etc.). Therefore ongoing research is being done in enabling direct peer to peer connections between the servers of the identity provider and resource provider.

Security of the roaming infrastructure

The roaming infrastructure is implemented by a hierarchical system of RADIUS servers. In eduroam, and also eduroam-ng for now (until peer to peer connections are introduced) this hierarchy is implemented by a chain of connections between servers, thus creating a transitive trust. This means that the connection between 2 servers provides the elementary piece in constructing the membership of eduroam, i.e. having a connection with one of the RADIUS servers that takes part in eduroam means taking part in eduroam as well.

Currently the connection between two RADIUS servers (and thus the membership of eduroam) is implemented by the use of shared secrets between the two servers. In eduroam-ng a movement towards security based on X.509-certificates has started; with the use of RadSec, membership is no longer based on a shared secret, and with the use of a PKI peer to peer connections are in principle no longer needed.

9 Conclusions

This document describes the current eduroam infrastructure and the alternatives/extensions that were evaluated after looking at the current state-of-the-art of the software development in the roaming area. The result is not yet convincing in all points. The simple message is that more work is needed, both on the improvement of the selected technology (RadSec, DNSRoam) and on the further evaluation of non-selected protocols (for instance, Diameter). Some very recent lab tests showed more inherent limitations of the RADIUS protocol which are not easy to overcome.

JRA5 will proceed in the following directions:

- continue and extend the test work based on the selected solution
- check continuously for the implementation progress of standardised protocols like Diameter, and integrate these solutions into the test scenarios as soon as they become generally available and stable.

The already started JRA5 subproject DAMe (Deployment of Authorisation Mechanisms for federated services in the eduroam architecture) will investigate how the roaming infrastructure can be functionally extended to also transport SAML-based attribute and security information, thus leading to better integration with the eduGAIN AA infrastructure. Deliverable DJ5.1.5,2 (cookbook v2) will summarise the progress and the results from this process in project year 3.

JRA5 will continue to cooperate with the TERENA TF Mobility and, once established, with the eduroam service activity on all these objectives.

10 References

[1Q] http://standards.ieee.org./getieee802/download/802.1Q-2003.pdf

[1X] http://standards.ieee.org./getieee802/download/802.1X-2004.pdf

[DJ5.1.2] http://www.geant2.net/upload/pdf/GN2-05-71v6.pdf

[DJ5.1.3,1] http://intranet.geant2.net./upload/pdf/GN2-05-163v3.pdf

[DJ5.1.3,2] http://intranet.geant2.net/server/show/conMediaFile.4550

[DJ5.2.2] http://www.geant2.net/upload/pdf/GN2-05-192v6.pdf

[GeoPriv] http://www.ietf.org./html.charters/geopriv-charter.html

[HUPNET] http://www.terena.nl/publications/tnc2004-proceedings/papers/linden.pdf

[OSC] http://www.open.com.au./

[Radiate] http://aaa.surfnet.nl/info/attachment.db?133350

[RadSec] Open System Consultants Pty, Ltd. “RadSec. A secure, reliable RADIUS protocol” http://www.open.com.au./radiator/RadSec-whitepaper.pdf

[RFC2716] B. Aboba, D. Simon, “PPP EAP TLS Authentication Protocol”, http://www.ietf.org/rfc/rfc2716.txt

[RFC2865] C. Rigney et.al. “Remote Authentication Dial In User Service (RADIUS)”, http://www.ietf.org/rfc/rfc2865.txt

[RFC2869] C. Rigney et.al. “RADIUS Extensions”, http://www.ietf.org/rfc/rfc2869.txt

[RFC3588] P. Calhoun et.al. "Diameter Base Protocol", http://www.ietf.org/rfc/rfc3588.txt

[RFC3748] B. Aboba et.al. "Extensible Authentication Protocol (EAP)", http://www.ietf.org/rfc/rfc3748.txt

11 Acronyms and Glossary

(see also Roaming Glossary of Terms, DJ5.1.1, http://www.geant2.net/upload/pdf/GN2-04-111Final.pdf)

AS	Authentication Server
authenticator	piece of network equipment in the same LAN as the supplicant that forwards authentication information between the supplicant and an AS
AV-Pairs	Attribute-Value Pairs
CA	Certification Authority
ccTLD	country-code top-level domain
CFI	Canonical Format Indicator
CRL	Certificate Revocation List
DNSRoam	Domain Name System Roaming, name of the dynamic server discovery implementation in Radiator
EAP-MD5	Extensible Authentication Protocol with Message Digest Nr. 5 payload
EAP-OTP	Extensible Authentication Protocol with One Time Password payload
EAP-GTC	Extensible Authentication Protocol with Generic Token Card payload
EAP-SIM	Extensible Authentication Protocol with Subscriber Identification Module payload
EAPoL	EAP over LAN
eduGAIN	Géant2 Authorisation INfrastructure for the educational community

eduroam-ng	The second iteration of eduroam, using a mixture of RADIUS and RadSec as authentication protocol
gTLD	generic top-level domain
IEEE 802.1Q	IEEE specification of Virtual LANs
MS-CHAP	Microsoft Challenge/Handshake Authentication Protocol
NAPTR	Naming Authority PoinTeR, a DNS resource record
NAS	Network Access Server
OID	Object IDentifier
RadSec	modified RADIUS protocol, see also [RadSec]
realm	an authoritative domain for user authentication; an AS which is authoritative for a realm is able to prove the identity of a supplicant that tries to authenticate. eduroam realms are in the form of DNS domain names, delimited from the user name with an @ symbol
SRV	DNS resource record which specifies the location of the server(s) for a specific protocol and domain
SQL	Structured Query Language
supplicant	piece of software that initiates and performs network authentication on the client side (term from the IEEE 802.1X standard)
TLD	top-level domain
WAYF	“Where Are You From” – an interface to query a user’s realm
WPA2	Wi-Fi Protected Access, version 2.