13-01-12

# Deliverable DJ3.1.2,2 Roaming Developments, Second Edition

**Deliverable DJ3.1.2,2**

| | |
|---|---|
| Contractual Date: | 30-09-2011 |
| Actual Date: | 13-01-2012 |
| Grant Agreement No.: | 238875 |
| Activity: | JRA3 |
| Task Item: | T1 |
| Nature of Deliverable: | R (Report) |
| Dissemination Level: | PU (Public) |
| Lead Partner: | RESTENA |
| Document Code: | GN3-11-354 |
| Authors: | Stefan Winter (RESTENA, ed.), Zbigniew Ołtuszyk, (PIONIER),Tomasz Wolniewicz (PIONIER), Gunnar Bøe (UNINETT) and Gurvinder Singh (UNINETT) |



**Abstract**

This deliverable provides an update to DJ3.1.2,1, and to research and development activities undertaken by JRA3 T1 to support new and emerging eduroam services supplying roaming access to wireless networks. The task includes monitoring developments in the network industry, producing software to improve eduroam operations, and contributing to standards bodies.

# Table of Contents

# Table of Figures

# Executive Summary

This deliverable provides a summary update of research and development activities of Roaming Developments, GÉANT Joint Research Activity 3, Task 1 (JRA3 T1). It covers tasks that support new and emerging eduroam services. JRA3 T1 also supports eduroam with Service Activity 3, Task 2 (SA3 T2), in which the core business is the supply of roaming access to wireless data networks. Activities include monitoring developments in the network access industry, contributing to standards bodies, supporting and seeking new possible services for eduroam operations, and liaising with other GÉANT activities.

This document describes an update to the activities undertaken by JRA3 T1 for this task since the publication of the first edition of this document (*Deliverable DJ3.1.2,1: Roaming Developments* [DJ3.1.2,1]) in October 2010.

Advances in wireless and wired network services provide new challenges and opportunities by way of changes, developments, threats and problem resolution. JRA3 T1 is tasked with producing and implementing software to support eduroam operations in such developments. Additional features such as CUI support, RADIUS over TLS, F-Ticks to collect usage statistics, debugging tools and the eduroam CAT tool are included in this task.

JRA3 T1 influences the network access industry's evolution by participating and contributing to technology standards bodies such as IEEE and IETF. It continues to provide input into key areas in standardisation and internationalisation of protocols.

JRA3 T1 works to improve authentication. It monitors developments in eduroam core authentication protocol (EAP) to determine the benefit to eduroam operations, tests EAP methods such as EAP FAST, reviews speed and reliable performance. It monitors and investigates interesting technology developments such as mobile access to spread network coverage beyond the boundary of an institution's campus.

# 1 Introduction

This deliverable provides a summary of the efforts undertaken in Roaming Developments (JRA3 Task 1) from October 2010 to October 2011 (M19–M30). It is intended to provide an update into the research and development activities undertaken to support eduroam operations (SA3 Task 2).

eduroam has been an operational service since the launch of GÉANT2 (September 2004). Its core focus is on the supply of roaming access to wireless networks, but it has also been deployed on other media, such as IEEE 802.3 wired networks. It is vital that eduroam is in step with the latest advances in the wireless LAN industry, and it should also strive to influence the development of the industry in order to evolve the service, and provide the best possible user experience to customers in the education and research community.

Task 1 of JRA3 assumes the role of research and development for eduroam in Europe, and its aim is to increase the security and usability of eduroam's architecture. The task also seeks to investigate and propose a solution to provide seamless authentication and authorisation in the current eduroam infrastructure. The activities pursued in JRA3 T1 include:

- Providing the development work necessary to introduce RADIUS over TLS (RadSec) into the eduroam infrastructure. This work will ensure that the underlying concepts of TLS security for RADIUS and dynamic discovery of nodes are tested and migrated into SA3 T2 (eduroam).

- Testing new digital certificates for the eduroam infrastructure, in collaboration with SA3 T1 (eduPKI). Task 1 will provide the development support concerning eduroam server certificates for the current eduroam infrastructure, as well as for the future infrastructure, based on dynamic discovery. Upon satisfactory tests, migration into eduroam service will be planned in conjunction with SA3.

- Adding authorisation and auditing elements to the current roaming infrastructure. Currently, eduroam does not allow for fine-grained authorisation.

- Extending eduroam to cover wired access.

- Enhancement of the eduroam infrastructure, according to the requests from SA3 T2.

The document provides a 12-month update on network industry developments, new service elements and software to support and link eduroam to other GÉANT activities. It is structured as follows:

- Section 2 contains a summary of interactions with standards bodies (IEEE and IETF), and provides an update on watching briefs set out in the previous version of this deliverable (DJ3.1.2,1), as well as details of active contributions to standards development.

- Section 3 describes specific improvements to the eduroam infrastructure, including an update on RADIUS/TLS, a tool for dynamic server discovery, considerations for IPv6 and multiple VLANs, and the eduroam Configuration Assistant Tool (CAT).

- Section 4 looks at the coordination of eduroam with other activities, including: consultancy on WPA discontinuation (SA3 T2), traffic path debugging in eduroam Norway, and details of the eduroam community worldwide.

- Section 5 provides a summary of the conclusions of the update.

Please note that some of JRA3 T1's efforts have already been set out in detail as part of the previous version of this document (DJ3.1.2,1), and have not been repeated here.

The following mapping details the status of 'watching brief' issues and other information updated since the publication of DJ3.1.2,1.

| | |
|---|---|
| • Ongoing RADIUS/TLS (RadSec) standardisation:<br>○ Failure scenarios if a server supports only one of the three packet types.<br>○ DTLS and TLS approach. Both transports should do application-layer demultiplexing or none (whereby DTLS gets a new port assignment). | Section 2.2.1 |
| • Possible consolidation of DTLS/TLS. | Section 2.1.3 |
| • F-Ticks service update. | Section 3.1 |
| • Status of RADIUS/TLS, poss. including reference to libradsecproxy (standalone library) to enable transport for arbitrary applications. | Section 3.2 |
| • Operational status of eduPKI. | Section 3.2.1 |
| • Watching briefs on IETF/IEEE technologies. | Section 2.1.1 |
| • Transition of RADIUS/TLS with dynamic discovery of service. | Section 3.2 |
| • Field test of iEEE 802.1x 2010 Section 2.1.2 and new EAP methods. | Section 2.1.1, 2.1.5 and 3.4.2 |
| • Useful leveraging of NREN eduroam support tools on a European scale. | Section 4.2 |

# 2 Standard Development and eduroam Contribution

## 2.1 Watching Briefs: Update

### 2.1.1 IETF: NEA/Federated TNC

The Network Endpoint Assessment Working Group (NEA WG) has made considerable advances since the first edition of this deliverable (DJ3.1.2,1), and a large part of the TNC architecture has now been defined:

- Posture Attribute (PA): defines the syntax of endpoint attributes (e.g. “Endpoint OS: Microsoft Windows“).
- Posture Broker (PB): defines how to collate one or more PA attributes into an assessment message; consists of a header and a list of PA attributes.
- Posture Transport (PT): defines how to transmit a PB message from the client device to a posture assessment server.

The third point, PT, has not yet been completely specified, due to friction between major vendors. There is agreement, however, that two distinct ways of transporting the assessment data are needed: one for network deployments that use the Extensible Authentication Protocol (EAP) protocol, an authentication framework providing for the transport and usage of keying material and parameters (generated by EAP methods [EAP] (used in eduroam), and one for networks that define security on the IP level.

One major point to keep in mind, however, is that the NEA WG still considers federated authentication, such as that used in eduroam, as being outside its scope.

The working group has achieved consensus for IP-level networks. Transport of PB messages is to be carried out over a TLS-secured channel between the client device and the assessment server. The exact details can be found online [PB message transport].

As of beginning of November 2011, the NEA WG has failed to achieve consensus on how to transport PB messages over EAP-based networks so far. The basic difference between the two major approaches to transporting PB messages may be found within the EAP encapsulation within:

- The PT-EAP approach (led by Juniper Networks) defines a new EAP type, which shall run in sequence after authentication of EAP methods.
- The EAP-TLV approach (led by Cisco Systems) adds new attributes to existing EAP types, i.e. it piggybacks on the PB message while the EAP authentication is carried out.

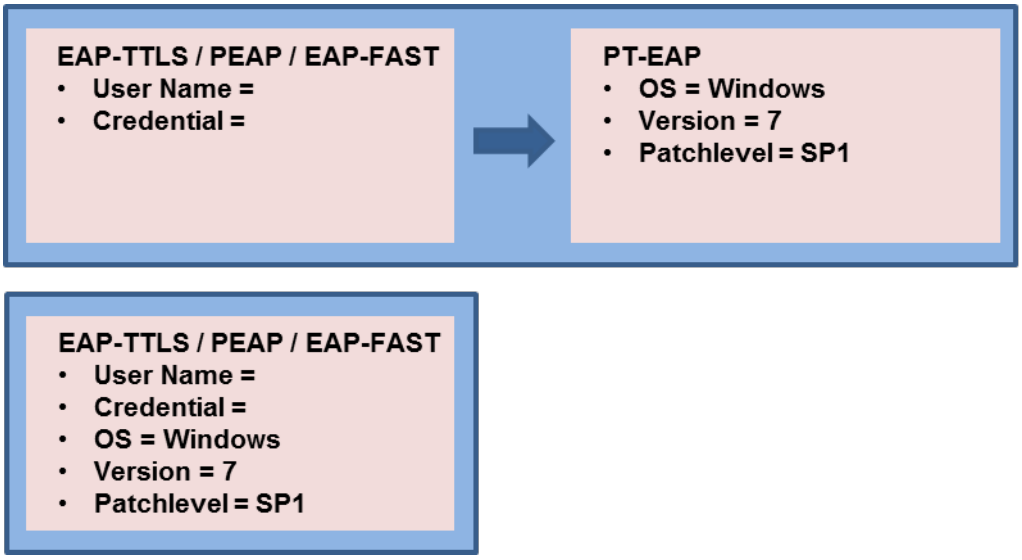


Figure 2.1: Two approaches to the transport of PB messages

The differences between both approaches do not appear to be very significant, and therefore, have not been included within this document. It remains to be said that even after a very thorough exchange of arguments, both approaches have been found to have unique advantages and disadvantages, and the working group could not settle on any one approach.

As a result, the decision was escalated to the IETF Security Area Directorate, which had to settle the competition by unilateral decision. The decision was made in favour of PT-EAP; with the provision that some of the shortcomings of PT-EAP could be rectified later as part of standardisation.

It remains to be seen how this decision will transpire into actual, deployed software, and therefore, the extent of its potential impact on the eduroam infrastructure is unknown.

### 2.1.2 IEEE 802.1X-2010

The IEEE has now released the final version of IEEE 802.1X-2010 as a free download [IEEE 802.1X standards].

Out of the many features discussed in the first edition of this deliverable, only a few have actually been implemented in commercial hardware products. Most notably, there is no hardware known to support named networks ("beaconing") on wired networks (see Section 2.1.2.1 Changes of Behaviour on Wired Networks, of DJ3.1.2,1). Consequently, the possibility of using out-of-band authentication on such named networks has also yet to be implemented (see Section 2.1.2.2 Introduction of Out-of-Band Authentication of DJ3.1.2,1).

Major hardware vendors have started implementing encryption on the medium (see Section 2.1.2.3 Encryption on the Medium of DJ3.1.2,1). Unfortunately, wired eduroam would need this technology to be deployed at the point where the user plugs in his/her device (i.e. the very first switch behind the socket in the wall), and support for this feature at present is limited to top-of-the-line switch devices. The commercial-use case for encryption on the wire appears to be in the area of switch-to-switch communication to secure links within an infrastructure; in that same-use case, the need for beaconing or user-interactive authentication does not arise.

JRA3 T1 staff are not aware of IEEE 802.1X supplicants that would support the new features of IEEE 802.1X-2010 at present, which further limits the usefulness of this standard in real life.

JRA3 T1 will continue to monitor the situation and test both authenticator hardware and supplicant implementations as soon as they become available. It may be necessary to procure appropriate hardware in a hardware testbed, in case the hardware does not become a commodity in a timely manner.

### 2.1.3 IETF: Alternative RADIUS Transports

The original "RadSec" Internet Draft defined both a new underlying transport, Transmission Control Protocol (TCP), which communicates at a level between the application and Internet Protocol (IP), and a new security model, Transport Layer Security (TLS), a protocol that allows secure client/server application communication across a network. In the course of discussion at the IETF, there was an interest to separate transport and security functions into two separate protocols. Some deployments may have a need for more reliable transport (such as TCP), but do not need or want a new security model; for example, if they use a VPN or IPSec to encrypt their server-to-server communication. In this case, the additional TLS encryption would be superfluous.

Consequently, the IETF specification was split into two: the definition of RADIUS/TCP in order to address handling issues related to RADIUS/TLS, and security on the transport layer of RADIUS/TCP. The document that defines the use of RADIUS over TCP, "draft-ietf-radext-tcp-transport" (the draft), can be found online [draft-ietf-radext-tcp-transport]. Work on the draft has passed all stages of peer review, and is awaiting publication as an IETF Request for Comments (RFC). The other half of the RadSec specification, "draft-ietf-radext-radsec" (online at [draft-ietf-radext-radsec]) is a normative reference, meaning it will not be published until the other half is also finished. For further details on the TLS encryption document, see Section 2.2.1.

As reported in the first edition of this deliverable, the second alternative transport, RADIUS/DTLS (Datagram TLS), remains a pending work item. There has yet to be any subsequent revision of the draft specification, and the draft has since expired. The author is known to still be supportive of this work, so new revisions can be expected in the future. JRA3 T1 will keep monitoring RADIUS/DTLS.

### 2.1.4 IETF: Internationalisation Challenges

The internationalisation problem present in the RADIUS and Extensible Authentication Protocol (EAP) area has gained more attention in the IETF. Investigations on the main specification RFC4282 have been carried out, with the chastening result that the document is seriously flawed. Luckily, the rules set forth in the specification regarding internationalisation have been widely disregarded, however, and are not typically implemented.

As a consequence, there is no proper standard encoding of usernames. To rectify this situation, a revision to RFC4282 is planned as a first step, but has not been issued yet. It remains a mid- to long-term goal for the IETF to tackle this problem.

JRA3 T1 will continue to monitor the state of this internationalisation update.

### 2.1.5 IETF: EAP Types

The watching brief on the EAP Method Update Working Group (EMU WG) currently focuses on two issues: the definition of a common EAP tunnelling (secure connection) method, which will replace the subtly different PEAP and EAP-TTLS methods at some point in the future, and watching new EAP methods, particularly EAP-EKE, as they arise, as reported in DJ3.1.2,1.

#### 2.1.5.1 *EAP-EKE / EAP-PWD*

EAP-EKE has been published as an Informational RFC (RFC6124). The patents which hampered a possible implementation so far are scheduled to expire in November 2011.

A contender to EAP-EKE is EAP-PWD ("EAP Authentication using only a password" RFC 5931, [EAP-PWD]). It has also been issued as an Informational RFC, in October 2010. EAP-PWD shares many properties with EAP-EKE, in particular, secure mutual authentication of supplicant and authentication server, without ever revealing the actual password.

EAP-PWD has the upside of not using patent-protected cryptography or algorithms; making it potentially much easier to implement. Unfortunately, however, there is also a downside. The cryptographic functions of EAP-PWD, based on either elliptic curve cryptography or finite field cryptography, are relatively untested. The RFC is lacking review from renowned cryptographers to assert the strength of security behind it.

Over time, implementations of EAP-PWD are becoming available. A UNIX supplicant (*wpa_supplicant module*), a Microsoft Windows supplicant, and a FreeRADIUS server module are thought to be in progress. These three implementations have the potential to significant improve field deployment. JRA3 T1 will monitor the situation and test upcoming implementations.

#### 2.1.5.2 *EAP Tunnelling Method*

The EMU WG has now settled on the criteria to be used to assess the utility of a suggested tunnelling method.

There have also been several proposals that claim to fulfil these criteria: EAP-TEAM (a derivate of PEAP), and EAP-FASTv2 (unsurprisingly, a derivate of EAP-FAST). One method, EAP-TTLS, which was popular in eduroam deployments, was abandoned by the standardisation body; the actors which previously developed and supported EAP-TTLS are now with EAP-FASTv2. Efforts to port the same functionality to one of the two other contenders were undertaken, resulting in an on-par feature set within EAP-FASTv2. This is important to the eduroam community because some of the features of EAP-TTLS (plaintext authentication inside, i.e. TTLS-PAP) were previously unmatched in EAP-FAST.

The two remaining protocols, EAP-TEAM and EAP-FASTv2 have very similar approaches to tunnelling authentication data within a TLS tunnel. The differences are mostly subtle, with one major exception, which is in the handling of plaintext passwords. Plaintext passwords are a common sight in eduroam deployments, so special care should be taken that plaintext authentication will continue to work after TTLS-PAP has been abandoned.

The EAP-TEAM protocol does not support basic password authentication within the tunnel in the same way as TTLS-PAP. It supports the authentication in principle, but only when wrapping the password authentication within an inner-EAP method inside the tunnel.

Two such inner-EAP methods were defined for EAP-TEAM:

- EAP-TEAM-EAP-MSCHAPv2 (which is largely equivalent to PEAP-MSCHAPv2, and requires passwords to be stored by the identity management back-end, either as Microsoft NT-Hash or as clear-text).

- EAP-TEAM-EAP-GTC, for identity management back-ends, which use a different encryption scheme for their users' passwords.

Thanks to the work carried out by SRCE earlier as part of JRA3 T1, which configured and tested EAP-GTC as an inner-EAP method (see Section 4.4 EAP-TTLS-GTC: Circumventing Supplicant Restrictions in Nokia Phones of DJ3.1.2,1), it is known that supplicants (devices such as laptops or wireless devices that require authentication) do not treat generic token card (GTC) credentials in the same way as they do passwords. Typically, they do not offer the user the option to store the ‘password’ (rightly so, because the GTC is, by definition, a one-time password only, so storing it as a permanently valid password defeats its purpose). In other words, the EAP-TEAM usage of GTC breaks the semantics of this pre-existing inner-EAP method.

Using EAP-TEAM would mean that many eduroam end-users would probably lose the ability to store their passwords on their machines. This would result in substantial inconvenience for the end user, and it is believed that it could also seriously hamper uptake of eduroam for these end users.

Alternatively, EAP-FASTv2 was retrofitted with TTLS-PAP-style plain password authentication, and is believed to behave in the same way that TTLS-PAP does, i.e. it will allow users to store passwords.

Further to this, other, earlier work in JRA3 T1 showed that EAP-FAST authentication takes significantly fewer round-trips than PEAP, and with it, EAP-TEAM authentication.

These issues were voiced by eduroam personnel and included in a final vote which took place at the IETF80 meeting in Prague, Czech Republic, on 30 March, 2010, when the working group triggered a final decision between the two contenders: EAP-TEAM and EAP-FASTv2.

The outcome of the vote was largely in favour of EAP-FASTv2, and a "rough consensus" was declared. EAP-FASTv2 will become the new IETF way of conducting TLS-tunnelled user authentication.

It should be noted that this decision has a downside: Cisco holds several patents and other intellectual property rights on parts of the EAP-FASTv2 technology, particularly in a part of the specification around the “Protected Access Credential“; which is not detailed in this deliverable. These IPR limitations may limit the native availability of this EAP type in operating systems (see also Section 4.2 EAP-FAST of DJ3.1.2,1).

The licensing terms of EAP-FASTv2 are very open (usage is for free, reciprocity required), and the IPR encumbrance can be seen as the lesser of two evils.

The course of this decision is a success story for JRA3 T1 work. Many parts of the work that was carried out for DJ3.1.2,1 (EAP-FAST speed tests, EAP-GTC testing, and others), was highly relevant, and it was important that the knowledge gained included lessons learned, in order to make an informed decision on the best way forward for the entire eduroam network.

Even better, these results were brought to the wider audience of the corresponding IETF working group, and contributed to the discussion and favourable decision for eduroam’s needs.

## 2.2 Active Contributions

### 2.2.1 Ongoing RADIUS/TLS (RadSec) Standardisation

This section is an update to Section 2.2.1 in the previous version of this deliverable (DJ3.1.2,1), and will not be repeated in full here.

Since the issue date of the first edition, the remaining open questions regarding the draft specification have been resolved.

- The question about port usage was resolved. The specification remains unchanged, and all traffic is to be sent to port TCP/2083. The differentiation of packet types is to be made by packet-type number. This solution ensures a minimum loss of information. It is also no longer possible for a server to indicate unwillingness to process a certain packet type by closing a port (and subsequently, sending an “ICMP Port Unreachable” message). Roaming partners, however, will need to manually negotiate their destinations for Accounting and/or DynAuth packet types, so the occurrence of an incoming, unwanted, wrong-packet type is the result of an administrative misconfiguration. It is best practice to solve such misconfigurations out-of-band.

- The question about identification of a client when using X.509 certificates was not able to be fully resolved. Numerous trust models exist around the X.509 certificates currently deployed, and it appeared unwarranted to prescribe a specific one for this specification without good reason. Since the

draft is intended for 'experimental' use, the question of how exactly an implementation identifies its clients, based on the presented certificate is left open. Should there be a pattern of common certificate usage, then this pattern will be incorporated to a definitive revision of the RADIUS/DTLS specification on the IETF Standards Track.

The draft specification was updated to convey the above information and has passed the "Working Group Last Call" without further comments. Consequently, working group consensus was declared.

The next steps are for a senior IETF member to be declared 'document shepherd' to create a summary write-up, and guide the draft specification document through the subsequent steps of the review process, namely "IETF Last Call" and publication.

The current state of the document as it passes formal document review can be viewed online in real-time as part of the "IETF Tracker" [IETF Tracker], which also shows the latest version (09) of the draft specification.

### 2.2.2 Ongoing Specification of "Dynamic Peer Discovery" for RADIUS/TLS

The dynamic discovery of RADIUS/TLS servers builds on the RADIUS/TLS base specification, and therefore, cannot be finalised before that base specification is published.

Work has continued on the dynamic discovery specification. Compared to earlier versions, the usage of NAPTR records to find a target server had to be modified so that IETF formal assignment rules for NAPTR labels were honoured. The draft specification now follows the "Straightforward NAPTR" (S-NAPTR) assignment model. The IETF draft specifies three S-NAPTR service tags:

- aaa+auth: Authentication
- aaa+acct: Accounting
- aaa+dynauth: Dynamic authorisation change.

Usage of this schema comes from a joint effort with the developers of the IETF's Diameter specification, who have aligned their Diameter 'application' discovery to a similar scheme.

The draft further defines two S-NAPTR protocol tags for the two transport models that may make use of dynamic discovery:

- radius.tls
- radius.dtls.

It is worth noting that these name tags are still in flux because the specification has not been finalised. For the existing eduroam deployment, the tag *x-eduroam:radius.tls* has been chosen (eduroam only carries out authentication, and is only specified for RADIUS/TLS). Since there can be more than one NAPTR DNS entry, it is possible to adapt to an upcoming IETF standard at a later date. The latest revision of the dynamic discovery draft specification can be found online [dynamic discovery draft]

### 2.2.3 RADIUS and SAML ("Moonshot")

It has been a long-term aim to combine network authentication (eduroam) with application level authentication/authorisation (web federations based on SAML). Due to the significant technological differences between these approaches and their use cases, several attempts to achieve such a combination have been tried, but have not been successful.

During GÉANT3, a novel approach to the network and application level authentication problem has been taken. The "Moonshot" project, which is in part hosted as a part of JRA3 T2, transports SAML assertions over the RADIUS transport channel, which is also used by eduroam. That way, the same infrastructure that is also used for eduroam (i.e. a RADIUS infrastructure), can be used to authenticate end-user devices to applications in the same way that a supplicant would authenticate them to an eduroam network. This enables re-use of already deployed technology. Administratively, however, the two uses of the infrastructure are independent, and require different policies.

While being an elegant approach, SAML over RADIUS requires a considerable amount of adjustment to existing protocols and implementations to make them flexible enough to work in this unprecedented context. As it is outside the scope of this document to detail Moonshot's architecture, please see Moonshot's dedicated website for further information [Moonshot].

One aspect of the architectural updates does fall within the scope of GN3 JRA3, however, and deserves to be mentioned in this document. eduroam uses the EAP protocol for user authentication, specifically it uses EAP over LAN/WLAN to transport EAP authentication data from a client device (supplicant) via an access point or switch (authenticator) to a RADIUS server (authentication server). This is the traditional use of EAP, and there is a document stating that EAP is to be used for exactly such a purpose: RFC3748, and in particular Section 1.3: Applicability. The applicability statement discourages the use of EAP except for *network authentication, in situations where IP transport may not be available* (paraphrased from RFC).

Moonshot is attempting to standardise the use of EAP over a different transport (an "EAP lower layer"), namely, the GSSAPI mechanism which runs on the IP protocol. It uses EAP for non-network-access purposes, namely application access. It also uses EAP for more than authentication, namely service authorisation. The aspects of Moonshot standardisation taking place in the IETF in the Application Bridging for Federated Authentication (abfab) working group has identified that the EAP applicability statement needs to be updated to accommodate this new use case. JRA3 T1 has dedicated some time to work on the EAP applicability statement update, and is working together with the Moonshot and IETF abfab groups to achieve this update.

The initial draft of a soon-to-be-RFC regarding the "Update to the EAP applicability statement" can be found online [EAP statement update]. Work to advance this draft from an individual contribution towards an IETF working group document is ongoing, and an update on its progress will be included in a later deliverable.

# 3 eduroam Infrastructure Enhancements

## 3.1 F-Ticks

As previously detailed in Section 3.5 Support Services for eduroam Next Generation Architecture: F-Ticks, of DJ3.1.2,1, the advanced statistics tool for eduroam's Federated Ticker system (F-Ticks), is now in production use in SA3 T2. As of November 2011, 26 out of 36 countries already report their authentication counts using this system. The previous statistics system, based on parsing of log files on top-level European RADIUS servers (ETLRs), is scheduled for deprecation once the remaining countries have joined F-Ticks.

F-Ticks data is available on the eduroam Operational Team's website: http://monitor.eduroam.org/f-ticks/.

The version of F-Ticks which is deployed on the eduroam OT's website has been modified for production use, in particular, support to enable the display of data on a Google Map was added to illustrate the flow of visitors and corresponding countries of roaming. Such graphics are expected to serve as an additional "selling point" for eduroam because it makes it easier to visualise the significant amount of international travel, and can demonstrate the value of being connected to eduroam for a given institution or country.

## 3.2 RADIUS/TLS

### 3.2.1 eduPKI Certificate Service

With the specification for RADIUS/TLS approaching its final stage (publication expected in 2012) and implementations becoming available (two are available now, and a third one is in progress), the third pillar required for moving RADIUS/TLS into actual production deployment needs to be defined. As reported in the previous edition of this document, eduroam is using the eduPKI service (SA3 T1) for eduroam IdP/SP vetting, and is issuing server certificates. This service is currently in production, and at the time of writing 37 certificates for 11 different participant countries have been issued.

It should be noted that the administrative overhead for issuing certificates is significant, due to the organisational and personal identity vetting, and cross-checking the authorisation of possession of a certificate with the eduroam database.

Ongoing research and standardisation work in the IETF may enable a different way of managing certificates, namely publishing valid certificates in the domain name system (DNS), as long as the DNS is secured with

DNSSEC. DANE is the most appropriate technology to use for certificate management and it is soon to be released as a product by the IETF. As the DANE specification approaches readiness, JRA3 T1 will look into possibilities to make use of DANE for eduroam IdP/SP authorisation checking. In the distant future, when DNSSEC is widely used, the use of Certification Authorities for this purpose may become obsolete.

### 3.2.2 Implementation Test: FreeRADIUS

FreeRADIUS, perhaps the most popular RADIUS server in federations and institutions, still does not have support for RADIUS/TLS. Ongoing contract negotiations with the main developer to implement these features are moving at a slow pace, but progress is being made. During the reporting timespan of this deliverable, support for RADIUS/TLS static connections has been added to the FreeRADIUS code base. JRA3 T1 personnel have verified the implementation and reported outstanding bugs to the developer.

The resulting code will be part of the 3.0 release of FreeRADIUS. At present, however, the code still has some drawbacks:

- The code cannot work with Status-Server watchdog packets – these packets are used to check whether a peer RADIUS server is still live, and the feature is marked as SHOULD in the RADIUS/TLS specification. When receiving a previously-sent Status-Server packet, the current development code of FreeRADIUS might crash.

- Dynamic discovery has not yet been implemented. In its current state, FreeRADIUS can only be used to connect to the European Top-Level servers (i.e. connect to a manually configured IP address).

JRA3 T1 will continue to oversee these remaining implementation components, and will report on the outcome and status of FreeRADIUS in the next edition of the deliverable.

## 3.3 Dynamic Discovery Test Tool

It has been noted by eduroam operations that the introduction of dynamic discovery adds extra complexity and introduces possible misconfiguration opportunities for federations and eduroam SPs / IdPs.

Since eduroam IdPs are going to be responsible for their own DNS entries, which enable dynamic discovery, it is important to give the IdPs a simple method of verification. JRA3 T1 has been working on a dynamic discovery test tool, using only the realm name as the input, the tool will perform DNS lookups to determine whether the realm is properly configured for dynamic discovery. After that, it will try to establish a RADIUS/TLS connection with the discovered hosts, and will check whether a proper server certificate is presented and that RADIUS traffic may be processed over that connection.

This tool is finished, but will not be released as a stand-alone service. Instead, it will be made available as a module to the eduroam Configuration Assistant Tool (CAT) (see Section 3.5). The screenshot in Figure 3.1 shows an example of successful dynamic discovery configuration verification (NAPTR resolution, subsequent SRV and A/AAAA resolution, and an example of a connection attempt to these IP addresses). Institutions that want to use this testing tool, but do not need any other features of the eduroam CAT, can register for the

eduroam CAT and disable the other features. It should be noted though that in order to thoroughly perform all the tests, many of the EAP-specific details of the eduroam CAT are needed, so a minimum amount of IdP configuration within the CAT tool is required in any case.

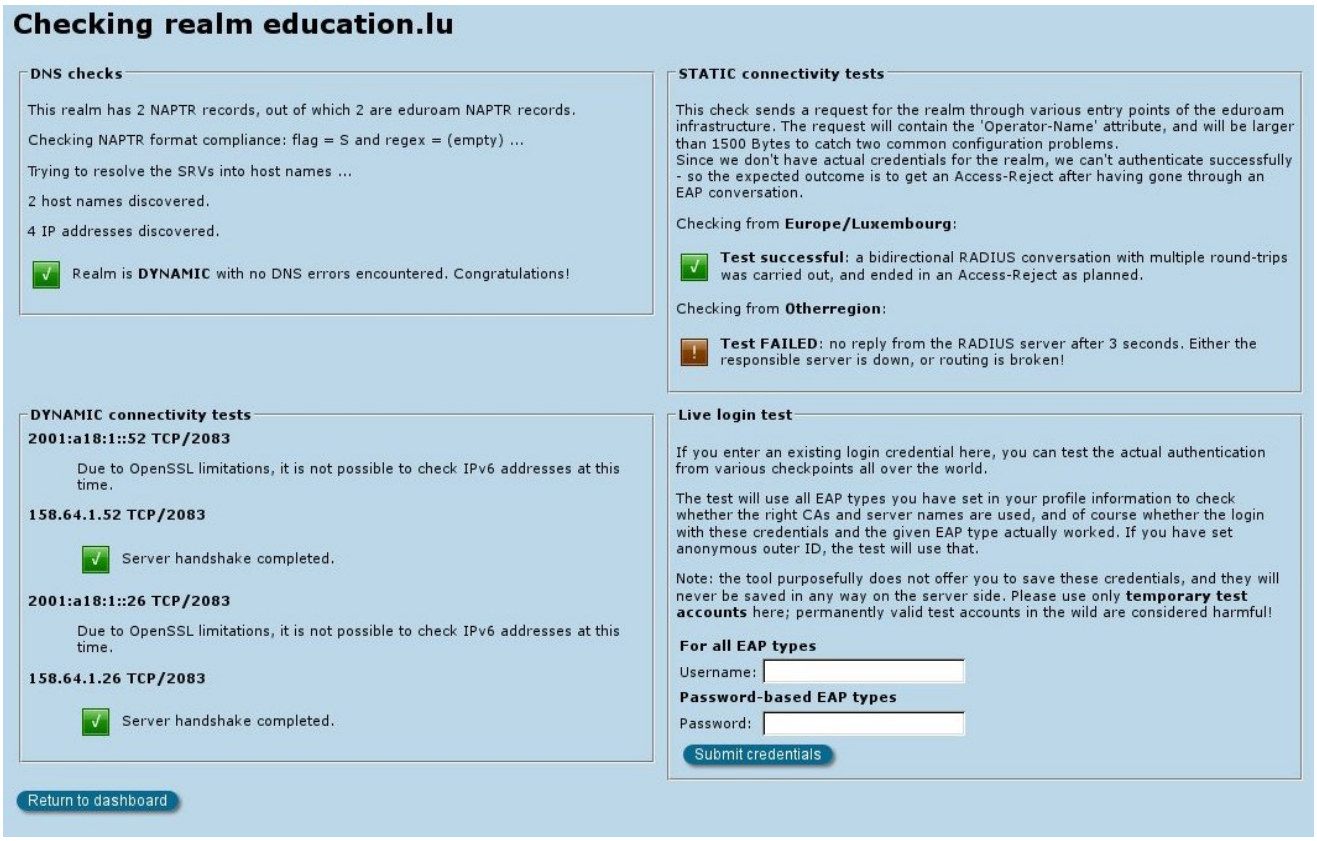


Figure 3.1: Illustration of dynamic discovery configuration verification

## 3.4 Considerations for IPv6 and Multiple VLANs

As is widely known, the uptake of IPv6 in end systems and access networks is happening at a slow pace. Many NREN networks are in the comfortable position of having deployed IPv6 throughout their network backbone and can offer IPv6 to their constituency. That makes eduroam SPs prime candidates for an early rollout of IPv6 support.

When taking a look at the IP stack from an ISO/OSI perspective, no special considerations seem to apply when deploying IPv6 connectivity to end users in a wireless network: the wireless network is operating on ISO/OSI layer 2 (MAC layer connectivity), while IPv4 and IPv6 are deployed on top of that layer, namely ISO/OSI layer 3.

Upon a closer look, some problematic differences can be observed, however, which need to be taken into account when deploying IPv6 on a wireless access network. Wireless networks are not typically deployed in strict ISO/OSI separated layers, and are often part of an integrated solution, together with access point controllers, access control, intrusion detection and similar features.

Intrusion detection systems and access control mechanisms are IP-version specific; e.g. a network firewall for the access network needs to understand and parse IPv6 packets in order to appropriately filter them. An intrusion detection system for IPv4 could, for example, scan for Rogue DHCP servers (this concept has been detailed in Section 2.2.4.1 Problem Description, of the first edition of this deliverable). In order to provide a similar level of security for IPv6, it needs to support equivalent mechanisms such as Rogue Router Advertisement detection.

A second concern, that has an impact on basic network operation, is exhibited when an eduroam SP uses dynamic VLAN assignment for different user groups.

The IEEE 802.11 wireless LAN standard does not have the notion of VLANs, which require modifications of packets on the wire and need to be prefixed with an integer. (This is part of the standard on wired LANS, whereas this is not the case on wireless LANs, as “VLAN assignment” on wireless LANs does not modify packets on the medium). Access points can only encapsulate traffic in a VLAN frame on their wired backhaul interface. On the wireless side, VLAN membership is a simulated property. Since every user has his/her own session key, traffic for this user cannot be decoded by any other station within reach of the same access point. Also, the access point can keep track of the station's MAC address and identify the VLAN (currently) belonging to this MAC. This internal tracking of VLAN membership enables it to only forward the VLAN traffic to the other stations it knows are in the same VLAN, thus effectively separating the user groups from each other.

Unfortunately, this approach is only appropriate for unicast traffic, which is directed to a specific, connected station. Broadcast traffic on a wireless LAN is encrypted with the same encryption key for the entire broadcast domain, and its destination MAC address is the broadcast address. For IPv4, this is not very significant, because all non-trivial traffic is either unicast traffic or can be identified by the receiving station as ‘not of interest’. For example, an address lease with DHCP requires a two-way communication: a request from the user device to the DHCP server, and vice versa. This communication happens with the unicast MAC address of the user device and DHCP server, so there is no danger of misinterpretation.

For IPv6, stateless autoconfiguration using Router Advertisements can replace DHCP without the need for two-way, unicast traffic. The router for a subnet simply multicasts its own address and the subnet it serves to all stations in the same LAN. The multicast traffic uses the above-mentioned broadcast domain key, which can reach every station configured for this key. The problem with multiple dynamic VLANs is that there will be a (different) IPv6 router for each of the configured VLANs. When all of the routers send their announcements to all stations, all stations will be presented with multiple routers and multiple subnets to use. However, when they start sending actual (unicast) payload traffic, only one of the routers will provide service (the one which is mapped to the same VLAN the client is in), and the others will appear unreachable, which leads to delays in communication until the user device notices that the router cannot be reached and tries a different router.

This situation can render an IPv6 configuration inoperational. The only solution to the problem is to define different broadcast domain keys for every VLAN. The access point / wireless controller needs to provision distinct broadcast keys for every configured VLAN, even if these are in the same WLAN SSID. These keys cannot be taken from the EAP keying material exchange, because the EAP exchange does not take VLAN memberships into consideration. That way, multicast and broadcast traffic can be encrypted with the VLAN-specific broadcast key. All stations will receive the raw packet over-the-air (the destination MAC address is still the broadcast address which reaches every station on the WLAN), but only those stations that are on the corresponding VLAN will be able to decrypt the packet and see its contents; other stations will report a (non-fatal) decryption failure.

Initial investigations of JRA3 T1 participants have shown mixed support for these per-VLAN-broadcast-keys. As an example, the early Cisco autonomous Access Points (configured via IOS) could be configured for per-VLAN-broadcast-keys; in fact, the first edition of the eduroam 'cookbook' [eduroam Cookbook] advises this. Later generations of Cisco, controller-based solutions are NOT capable of such per-VLAN-broadcast-keys. Therefore, at present, it is not possible (or at least not advisable) to deploy IPv6 in combination with dynamic VLAN assignment on Cisco controller solutions. JRA3 T1 has decided that making operations aware of the problem is advisable, but the actual tracking of vendors that support the feature is more an operational, not a research, concern. As a consequence, JRA3 T1 suggests that the eduroam HOWTO Wiki [eduroam Wiki] (see also Section 4.3.3) should be updated with documentation on IPv6 capabilities by operators that deploy the equipment in question.

## 3.5 eduroam Configuration Assistant Tool (CAT)

eduroam's Configuration Assistant Tool (CAT) is a user-oriented system that configures various wireless devices for eduroam use, so that end users do not need to worry about entering configuration details.

The CAT database contains information provided by local eduroam Identity Provider administrators, such as supported EAP methods, trusted RADIUS server names, trusted server certificates, etc.

CAT installers are device-dependant entities (Windows installers, XML profiles, etc.), which carry all institution-dependent information. An eduroam CAT installer is created when a user selects his/her institution, a user group, and one of the supported devices from a series of screen prompts.

The design of the CAT is the result of significant feedback and experience supporting eduroam users from different institutions using a number of possible local configurations. The CAT's main purpose is to make eduroam easier to install and safer to use, both for administrators (to provide them with a means of verification whether their setup is consistent, complete and functional (see also subsection 3, above) and for end users (to provide them with a ready-made custom installer that appropriately sets all security parameters).

There are a number of key components in the eduroam CAT, which are discussed in turn, below.

### 3.5.1 Installer

A CAT installer includes a device/system dependent code instance, which is capable of setting up the device for eduroam access. An installer should configure the wireless profiles on the device, either by requesting the user's login credentials or by leaving this step to be completed at the time of the user's first eduroam access.

### 3.5.2 Institution

An active eduroam identity provider (authenticating institution (idP) is the entity wishing to provide eduroam installers via the eduroam CAT system. Eduroam idPs are listed in the eduroam database (the result of SA3 T2).

The eduroam CAT will make use of the eduroam database to establish a link of known IdPs to their installer details.

### 3.5.3 EAP Type

The EAP type results from the combination of inner and outer EAP methods supported at a given institution or by a given device (examples include PEAP-MSCHAPv2, TTLS-PAP, and TLS). There is no universal set of EAP methods, supported by home institutions or by current wireless devices, therefore, the lists of available EAP types must be defined both for institutions (i.e. which EAP types are supported by an institution's RADIUS authentication server support) and devices (i.e. which EAP types can be configured on a given device). The common subset of IdP-supported EAP types and device-supported EAP types may be empty. In that case, the device cannot be used for eduroam at this institution.

### 3.5.4 User Group (Profile)

A user group is determined by the set of users sharing the same eduroam configuration settings (realm, trusted certification authority, trusted RADIUS servers, support contacts, supported EAP types, etc.). If all users of a given institution are placed within the same group, the group still needs to be defined, but its name will never be visible to the users.

The name of the user group should be easily understandable by users of a given institution (for instance: 'staff', 'students') and should be given in all official languages specific to a given institution as well as in English. The local Identity Provider administrator needs to decide the default name used when the installer name is different from the one in which the profile names had been specified.

### 3.5.5 System Design

The following four, main areas can be easily identified within the eduroam CAT and are described, in turn, below.

- Database
- Core services
- Device modules
- Web front-end.

#### 3.5.5.1 *Database*

At present, the eduroam CAT uses a dedicated, self-sufficient database, although consolidation with the main eduroam database is planned in the future. Fortunately, the overlap between the two is minimal, and this task can be easily left until the eduroam CAT is ready for production.

### 3.5.5.2 *Core Services*

The core services area is responsible for all preparatory work before the control is passed to the device module.

The settings for a given user group define the supported EAP types for the home institution.

An institution may support several EAP methods (for example, FAST, TLS, PEAP) that carry a user's credentials from his/her device to the authentication server. The Identity Provider administrator defines the order of preference for these EAP methods. When a user picks up a device, it is likely that the EAP methods supported by this device will overlap with EAP methods supported by the institution, but not all three methods will be supported. The device module announces the methods it can support, and it becomes the role of core services to pick the best possible EAP method and tell the device module to use it. Similarly, if the device module needs a certificate in a specific format or one of the certificate fingerprints, this will be delivered by core services, without the need for conversion at the local level.

### 3.5.5.3 *Device Modules*

Device modules make use of information available in the CAT database via the CAT application programming interface (API) and produce the installer files. The device modules are the most vital part of the system, and should be written by those with expertise configuring such devices. The interaction between the device module and the rest of the system is limited to, at most, API calls. All institution and user group parameters are collected into a single attributes array. In cases where multiple choices are available, the eduroam CAT system configures the most optimal combination, so that the administrator's preferences for device configuration are matched as closely as possible. As a result, the device module programmer receives simple instructions, and does not have to know anything about the eduroam CAT core system itself.

The module MUST configure the device to access the list of SSIDs passed to it by the eduroam CAT module API. The module publishes the set of EAP methods it can support, then the eduroam CAT API compares this to the prioritised list of EAP types supported by a given user profile, and returns the most appropriate one. The device module creates an installer for this EAP method. If the device requires separate configuration for WPA2/AES and WPA/TKIP, then it should use the information passed by the API, specifying which encryptions must be supported. As a rule, if WPA/TKIP is specified, then WPA2/AES must also be configured for this SSID.

### 3.5.5.4 *Web Front-end*

The eduroam CAT makes use of two front ends to help users access the system:

- One interface is for the eduroam IdP administrator, where he can define all the properties of his/her IdP such as uploading the infrastructure certificate of the CA which issued the server certificate, defining which EAP methods his/her RADIUS server supports, and whether s/he wants to enable anonymous outer identities, etc.

- One interface for end users, where a user can select his/her institution and user group, and subsequently download installers.

Both interfaces need to be fully translated into all the eduroam participant languages to encourage uptake in non-English speaking European countries.

Since the two web interfaces are the most visible parts of the system, they are described in more detail below.

#### 3.5.5.5 *IdP Administrators' interface*

There are five major steps in defining an eduroam IdP:

1. Enrol initial Identity Provider (IdP) (institution country and name).
2. Define general (non-EAP) properties of the institution (logos, acceptable use policies).
3. Define EAP profiles (CA certificate, server names).
4. Perform sanity checks (check if all the necessary data has been uploaded).
5. Test installer download area (administrator testing before release).

The following screenshots show the layout of each of these steps, as viewed by the user.


Figure 3.2: Identity provider (IdP) enrolment screen

This simple enrolment page acts as a placeholder until the configuration assistant tool is connected to SA3 T2's eduroam database, which contains the names and country affiliations of eduroam IdPs. IdP Administrators will no longer need to manually enter this data. While the tool is still in development, the current enrolment page provides a simple entry point. It contains a sample of countries and a free-text entry for the institution's name.

In order to facilitate complete internationalisation, an institution will need to be given the opportunity to be known/identified in all languages it cares about. This is especially the case for multilingual countries, such as Belgium, where there is more than one entity for “the name”. Multilingual naming is scheduled for further development.

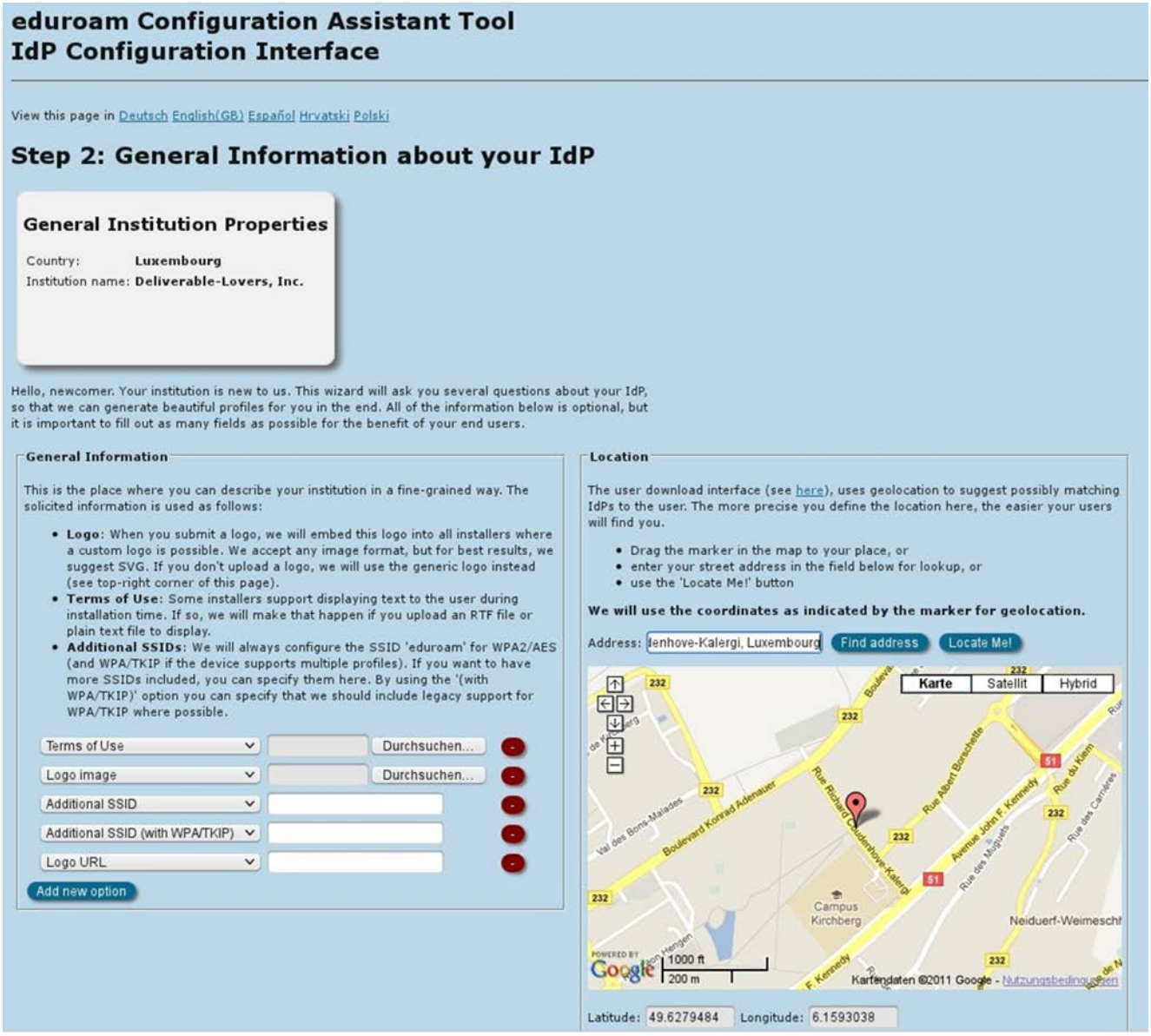


Figure 3.3: General properties required for IdP configuration

There are numerous attributes to an eduroam IdP that are not directly related to its EAP settings, but are important. An initial survey about the eduroam CAT toolset's requirements has shown that branding of eduroam installers to meet local needs is considered to be very important. The main areas of local branding have already been addressed:

- **Logo**: Where technically feasible, the generated installers will contain a logo of the institution, if the institution has uploaded this to the system.
- **Terms of Use**: Where technically feasible, the installers will also display the terms of use of the institution, if this has been uploaded.
- **SSIDs**: If the institution deploys another service set identifier (SSID) besides eduroam, it should be included in the installers.
- **Helpdesk contact details**: These will be included in the installers and on the eduroam CAT user download page so that users know where to seek help if something does not work as planned.

Another general property of the IdP is its physical location(s). The user download interface will use these, along with the physical location of the user accessing the download page, to estimate which IdP might be the correct one for the user (see next section).

Finally, there might be EAP details that are common throughout the organisation, e.g. the CA that issues the server certificates is always the same. For these cases (this is expected to the default case for small organisations), it is practical to upload common EAP immediately at the general properties page (as opposed to defining them in the subsequent step on a per-profile level).

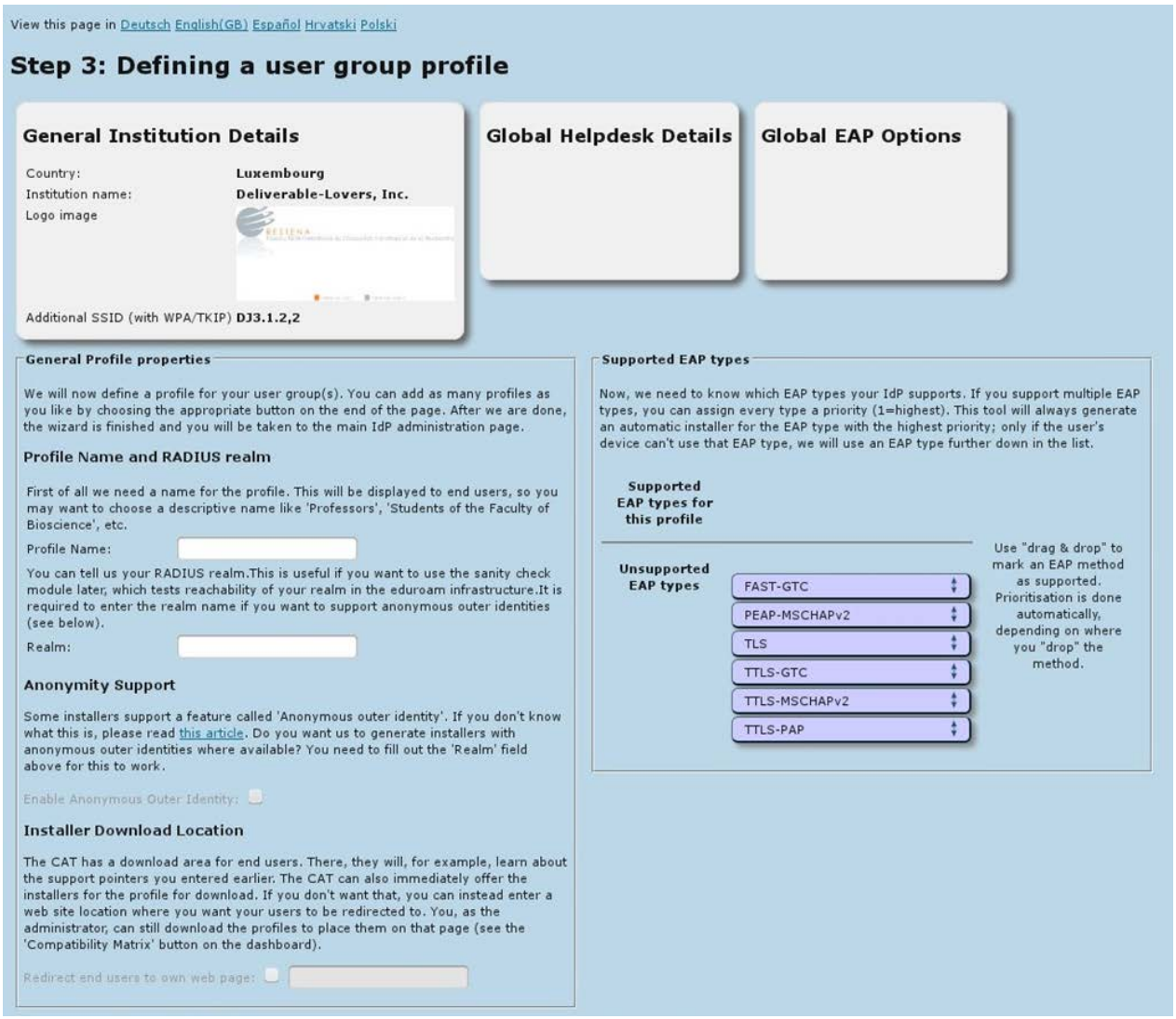


Figure 3.4: Identity-defining EAP profiles

The initial requirements analysis also revealed that an organisation may serve different types of user groups, e.g. Students, educational staff, other personnel. Technically, these may be managed separately, and require different treatment for EAP authentications.

Therefore, the eduroam CAT allows IdP administrators to define multiple EAP profiles within their institution. The screenshot above shows the setup step for one such a profile, which allows the IdP administrator to define the following properties:

- **Descriptive profile name**: this description is important to ensure users are able to recognise which installer is appropriate for them. This name is also envisaged to be multilingual in future updates.

- **RADIUS realm**: this information is optional (end users typically enter it as part of their username when connecting) but is useful for connectivity tests and for configuring anonymous outer identities, if the IdP administrator has signalled to support these (see next bullet).

- **Anonymous outer identities**: support for user privacy.

- **Supported EAP types**: This information is essential for preparing the installers for the various devices. The eduroam CAT strives to support all of the common EAP types available. This is an ongoing R&D effort, as it is not always obvious how to pre-generate an installer on every kind of device API.

This necessary information can be supplemented with the aforementioned helpdesk details and EAP properties. If such details are specified at the profile level, they will override any general input that might have already been entered on a previous page.

The idea behind the override is that certain user groups may have dedicated helpdesks that are able to provide a higher level of service than the standard ones (e.g. a 'staff' helpdesk might provide personal consultancy to professors, while the 'student' helpdesk only provides baseline email support). In most cases, the profile-level overrides can be left empty (i.e. if the general properties for the institution are valid and unchanged for the profile).

After the user defines a user group profile (and fills out the form shown in Figure 3.4), the necessary information is collected and the eduroam IdP administrator is sent to an overview page (the "dashboard").

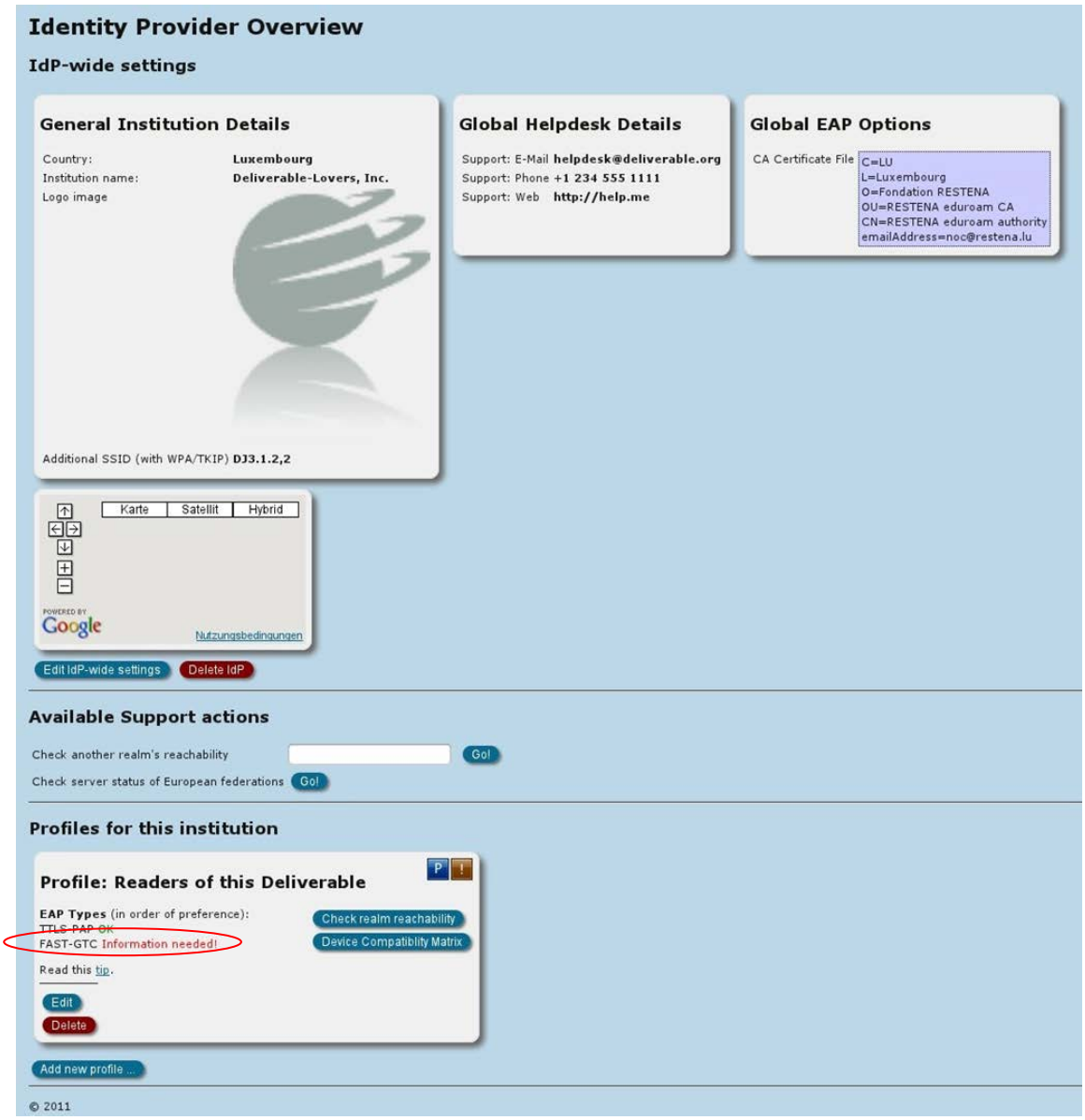


Figure 3.5: Identity Provider overview and sanity check screen

The dashboard contains all the submitted information and contains a preview of logos and certificate information. It provides a central control point for the administrator. From here, the administrator can:

- Check if the submitted settings make sense (if not, an exclamation mark and the notice “Information needed!” are displayed, as shown by the red text circled in Figure 3.5, above).
- Trigger connectivity checks for the user’s RADIUS realm (if the user supplied that information).
- Trigger connectivity checks for arbitrary other realms (e.g. If the administrator wants to help a roaming user who reported connectivity problems).
- Go to the download page for the user’s institution's installers.

Please note the development of further support modules for eduroam IdPs is ongoing; this deliverable captures only a snapshot of the developments made to date.

Of particular interest for the administrator will be the download area for device/EAP-Type combinations, which are accessible via the “Compatibility matrix” button in the profiles. The next screenshot shows a sample matrix.

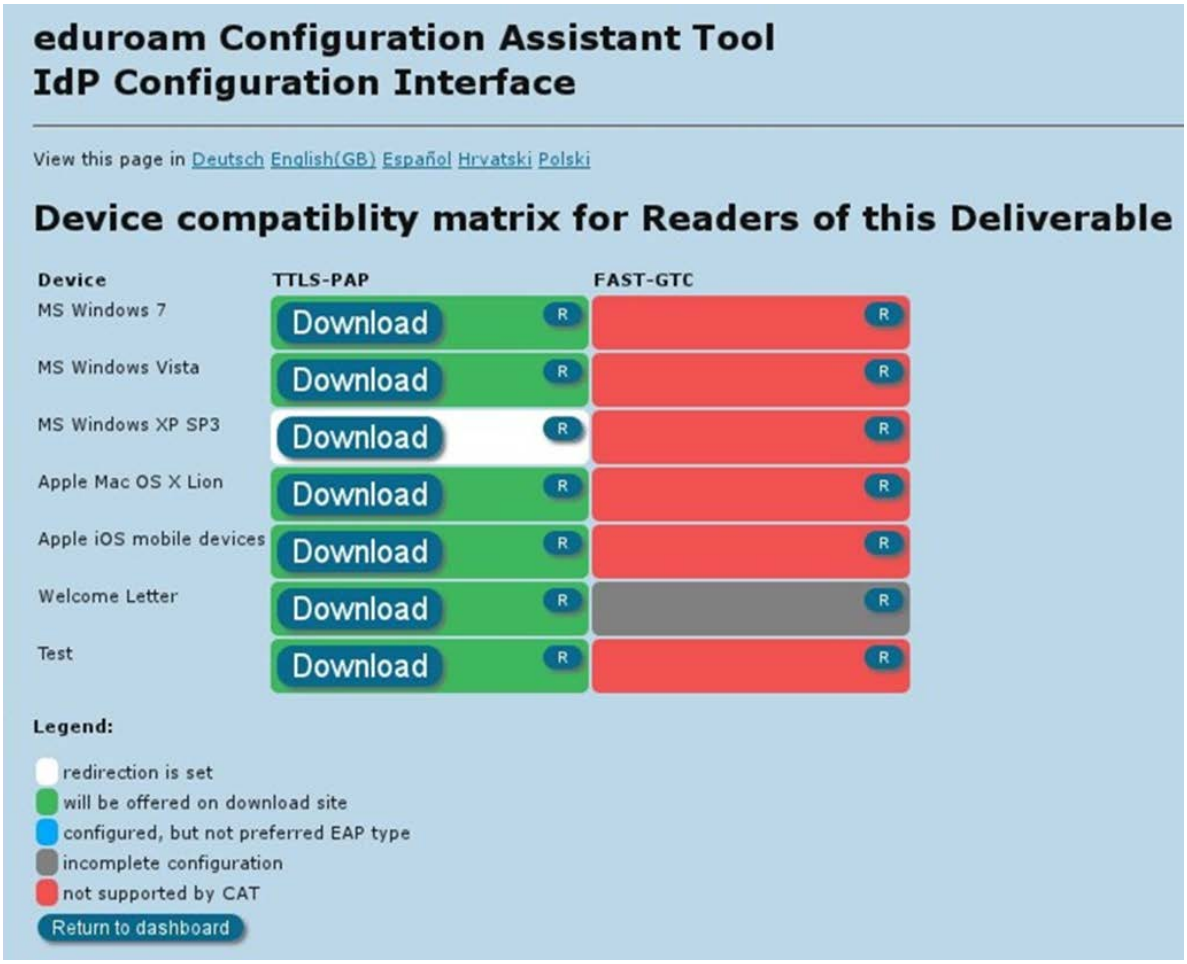


Figure 3.6: Device compatibility and installer download

The matrix shown in Figure 3.6 provides an eduroam IdP administrator with an overview of the devices which the eduroam CAT can support, cross-correlated with the EAP types configured by the administrator. The status of the combinations is colour-coded, as follows:

- Green is used if the eduroam CAT can generate an installer for a given combination, and the resulting installer can be downloaded by clicking on the appropriate button.

- If the profile supports multiple EAP types, only the installer for the most preferable EAP type will be in green, and generated; all other EAP types for the device will be marked in blue as ‘stand-by’ combinations.

- If a certain combination is not supported by eduroam CAT, this will be indicated by red.

- Grey will be used if a combination could be supported by the eduroam CAT, but the administrator has supplied insufficient information.

An administrator can determine if a certain combination should NOT be made available for download by end users on the eduroam CAT end-user interface. If this is configured, the fill colour of the box will be white. This option would be necessary if an administrator has already purchased dedicated support software for a certain device, and prefers that support over the one generated by the eduroam CAT.

#### 3.5.5.6 *End-user Interfaces*

It has been assumed that the components that are visible during a typical user interaction with eduroam CAT must be reduced to an absolute minimum, which is reflected in the hierarchy of selecting the home institution first, possibly selecting a user group, then selecting the device. It should also be possible for the user to have access to an installer via a direct link. The installer interface should not ask the user for any personal information.

User interfaces must work on a large variety of devices, so the interface should be both technically and graphically attractive, however, the eduroam CAT also needs to work on simple devices with small screens.

As previously stated, the installers should also be available for download via a direct link, so that eduroam institutions may point to the CAT from their local support pages.

#### 3.5.5.7 *Privacy Considerations*

The eduroam CAT relies exclusively on public information, which eduroam IdPs are supposed to publish on their local eduroam support pages. By submitting the same information in a structured way via the eduroam CAT interface, no sensitive information is revealed by the eduroam IdP administrator.

When a user downloads an installer, only these publicly available pieces of information are used and installed on the user's device. In particular, the eduroam CAT website will not ask a user for his credentials. All credential handling occurs exclusively on the end user's device.

#### 3.5.5.8 *Tool Limitations*

The eduroam CAT is only able to support devices that allow some degree of automated configuration. In addition, the CAT can only produce installers if the EAP methods supported by the institution and by the device have at least one EAP method in common. Finally, the eduroam CAT may ‘refuse’ to create installers in cases where it would require users to pass on some private information (a private key for EAP-TLS is one example, which is discussed further in Section 3.5.9.1 EAP-TLS.

Some very special limitations may appear, for instance, certificate chains do not seem to be well supported for Android devices at present, which would exclude CAT support for certain devices from a number of institutions.

#### 3.5.5.9 *The CAT and eduroam Safety*

eduroam relies on the secure transmission of credentials between the end user's device and his/her authentication server (this is exclusive, no other party will ever be able to retrieve and process private user credentials).

For this to work, end user devices need to be correctly configured, especially with regard to verification of the server certificate and certification authority.

Although it is documented best practice to give users the complete set of information and instructions needed for this server-side verification to work, it can be observed that a small number of eduroam IdPs are negligent, and exhibit some or all of the following bad practices, often due to reasons related to convenience (i.e. because it results in an 'easier' supplicant setup):

- Advise users to completely turn off server validation.
- Advise users to set up only CA trust, not server name validation.
- Do not communicate the server name or CA to their users.

The eduroam CAT can drastically improve this situation because it will always generate installers with complete information (without needing to display all of this information to the user), which keeps the installation process simple. It is hoped that the availability of pre-built, completely configured installers will significantly reduce the occurrence of the bad practices outlined above.

A second aspect of the eduroam CAT can also improve eduroam safety. The installers will be digitally signed by eduroam Operations, which are likely to be recognised as a trusted CA path by most operating systems (TERENA TCS Certificates).

This secures the distribution path for the installers, and is a significant improvement. Typically, even institutions which generate pre-configured installers on their own would not sign the resulting executable/config file, or use their own CA to sign the installers. Such behaviour makes it difficult for the end user to verify whether his/her installer download is genuine.

### 3.5.6 Implementation

The eduroam CAT has been implemented in PHP and uses a mySQL database. JavaScript is used for its user interfaces, also through the jQuery library.

Device modules can be quite independent, in particular, Windows installers are prepared in PHP, but the final installer is then produced with NSIS (Nullsoft Scriptable Install System).

### 3.5.7 End-user Interfaces

It has been assumed that the user interaction with the eduroam CAT must be reduced to an absolute minimum, namely to select the home institution, possibly a user profile, and a device. It should also be possible to access an installer via a direct ('deep') link so that the installer can be referenced from a third-party web page (e.g. the institution's help page).

The installer interface should not prompt the user for any personal information.

The interface leads the user through a number of choices. When the user selects the appropriate user group, support information from his/her home institution will be displayed. Thus, if the user finds problems with either the installers or with the CAT interface itself, the local support contact should be the first (and probably the only) choice for help.

CAT interfaces are multilingual, starting with the preferred language from the browser settings, but may also be selected from the interface.

#### 3.5.7.1 *Default Interface*

We expect that the default CAT interface will be accessed by most users. It has been written with JavaScript and AJAX via the jQuery API. Home institution selection is done with DiscoJuice and its geo-location API.

If the user accesses the eduroam CAT while being geographically close to his/her home institution, the geolocation capabilities of DiscoJuice will narrow the choice of institution to a few cases at most (in most instances, the first one on the list will be correct). As shown in Figure 3.7, if the user is away from home, then an autocomplete search or simple browsing will be equally convenient methods of finding the institution.

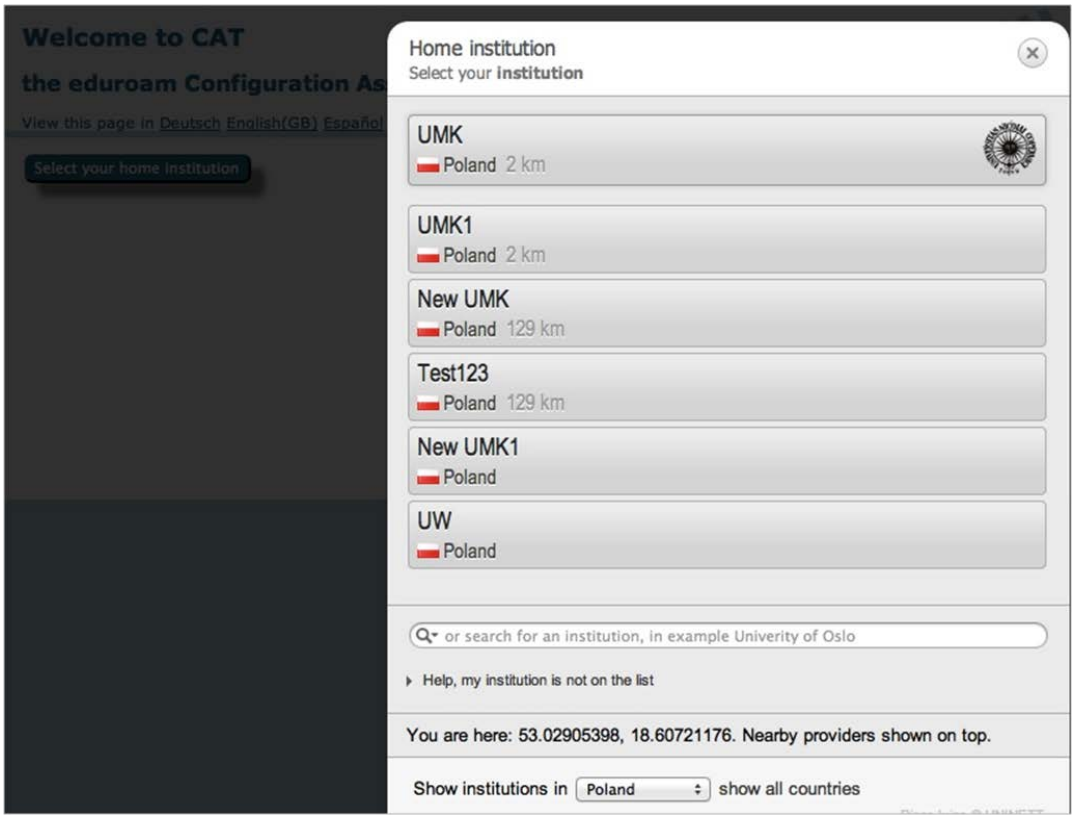


Figure 3.7: Example of DiscoJuice geolocation

Next, as shown in Figure 3.8, the user will be shown names of user groups to choose from (this step will be omitted if the institution has only defined one user group).


Figure 3.8: Selection of user group

After choosing the user group, the user is presented with a list of available installers (Figure 3.9). This screen also displays contact information for the user's home institution.

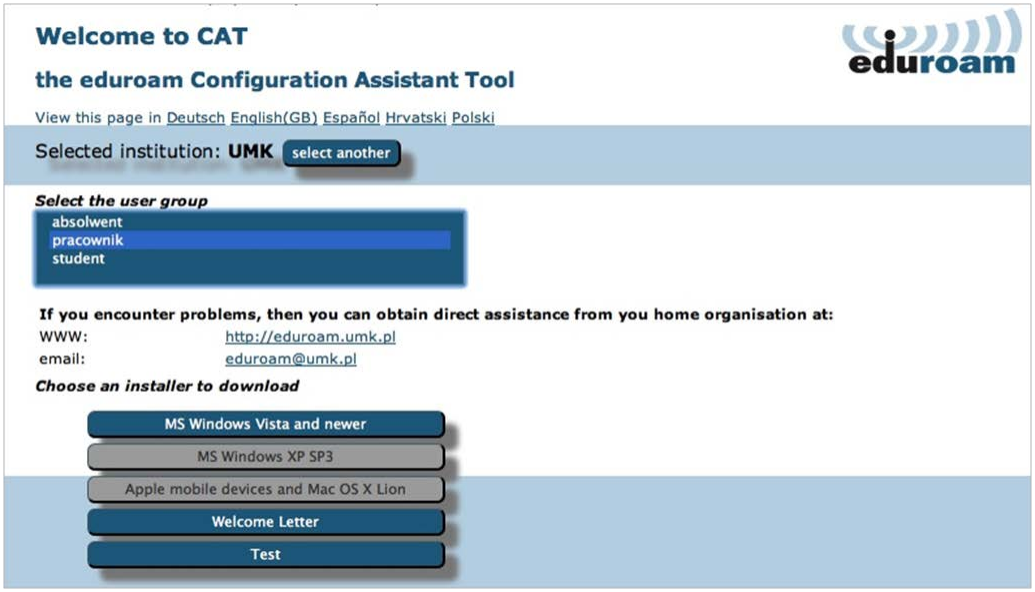


Figure 3.9: List of available installers

On the resulting screen, two installer options are greyed out, which means that they are not available with the settings that the institution has defined and the current CAT capabilities.

### 3.5.7.2 *Simple Interface*

The simple interface takes the user through the same steps as the default interface, but does not use any advanced browser features. While its design is basic, it has the advantage of working on the most basic browser, and does not require JavaScript support. Automated features, such as geolocation, are also not available. Any screen of this interface can be directly addressed, so it is possible to create links from the support pages of the home institution. In Figure 3.10 we present the page shown as it would appear in the simple interface.

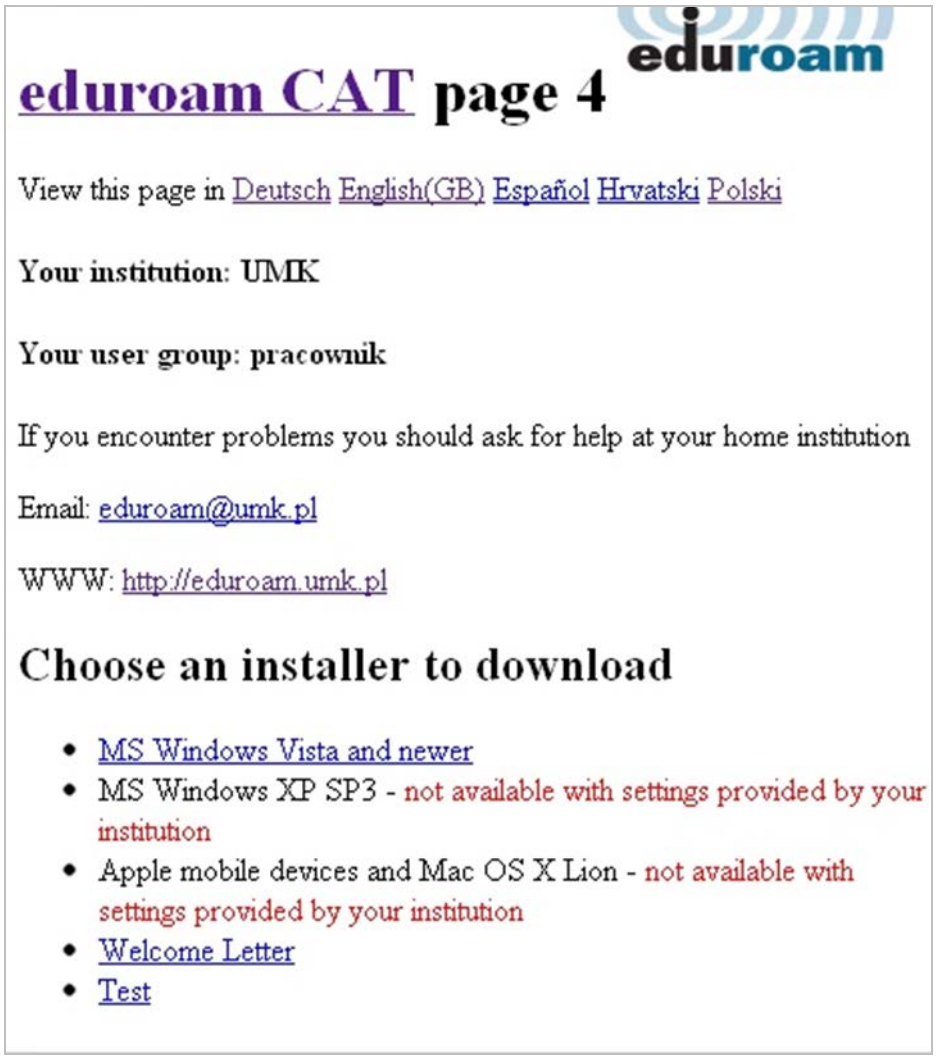


Figure 3.10: Example page of the simple interface

### 3.5.7.3 *Download Interface*

It is also possible to download an installer with a direct link from a third-party web page. This feature is expected to be used via the home institution's support pages, which will lead the user through all steps and then point the installer download to the CAT. This approach may be particularly useful if an institution already has a system of support pages and needs to include additional information (for instance, how to obtain a personal user certificate).

### 3.5.7.4 *Escapes*

Escapes are special cases when, for some reason, an institution prefers to provide its own installer support for a given device. It may be that the institution has a commercial contract and prefers to use it. In such cases, the administrator can mark the device with an escape URL, to which the user will be redirected instead of being presented with an installer file.

### 3.5.8 Device Modules

#### 3.5.8.1 *What does the CAT API do for device modules?*

Before passing control to the device module, the CAT system prepares a few things for the module to use. These include:

- Interfacing with the user.
- Obtaining all the configuration parameters from the CAT database.
- Creating a temporary working directory.
- Preparing certificate files and making them ready for storing in the working directory.
- Storing information files in the working directory (possibly changing the character set).
- Preparing graphic files (such as an institution's logo, for installer customisation).

After the device module produces the installer file, it leaves this in its working directory and returns the name of the installer file to the CAT API. The file is then delivered via the web interface to the user.

It is important to understand how the device module fits into the whole setup. An external caller (for instance, *GUI::generateInstaller()*) creates the module device instance and prepares its environment for a given user profile by calling *DeviceConfig::setup()* method. Finally, the module *DeviceConfig::writeInstaller* is called and the returned path name is used for user download.

#### 3.5.8.2 *Directory Structure and Naming*

All device modules reside in the devices directory. Each device module has its own subdirectory. If a device module requires additional files that will need to be copied to the working directory, then these files should be placed in the Files subdirectory of the module directory.

The name of the module directory may be arbitrary, but the name of the module file and the name of the device class must be synchronised. For instance, if the name of the module is TestModule, then its source file should be called TestModule.php and the name of the class must be Device_TestModule.

Naming is defined in the devices.php file, which is a configuration feature outside of the module.

#### 3.5.8.3 *Device Driver Code*

In order to produce an installer, one typically needs to know:

- The certificate of the CA that has signed the RADIUS server certificate.
- The names of the trusted RADIUS servers.
- The EAP method to be used.
- Which SSIDs to connect.

The installer will only work properly for users from one institution and, possibly one user group, so it is wise to display an appropriate warning. If an institution has only one user group, then it is not necessary to make users aware that this concept exists, and device modules should probably try to hide this. The *internal:profile_count* attribute can be used to check the number of profiles held by a given institution.

In some environments, it is possible to display additional text information, such as a list of usage terms. It may also be possible to customise the installer graphics by adding the institution logo. If an IdP administrator has configured this information in the web interface, the CAT API makes it available to the device module and the module can make use of it. When writing a new device module, the following restrictions for the module code apply:

- The device module class must extend the DeviceConfig class, thus obtaining access to all methods and properties provided by this class.
- The module class must define a constructor, and this constructor MUST set *$this->supportedEapMethods* to an array listing EAP methods supported by this particular device.
- The module must define a *writeInstaller()* method, which is to produce the actual installer file. All useful profile properties are provided within the device's attributes property (see DeviceConfig) and set by the setup(*Profile $profile*) method when a device module is being prepared to be called.
- The *writeInstaller()* method must create the installer in the form of a single file in the module.

#### 3.5.8.4 *Example Device Module Implementations*

The current alpha version of the eduroam CAT tool contains modules for Windows 7, Vista, XP, Mac OS 10.6, 10.7. Work on additional modules (e.g. for several popular Linux/Android supplicants) is ongoing.

#### 3.5.8.5 *Test Module*

The test module can show how easy it is for a module to interface with the eduroam CAT. The module contains one static file, which it will copy to its working directory, initially preserving the name and the second time, changing the name. Then it will dump all profile attributes into a file called *profile_attributes* and create a zip file of the whole working directory. Since the trusted CA certificates and logo files are automatically copied during module bootstrap, they will also appear in the zip file. The complete code listing is shown below.

```
<?php
require_once('Device.php');
class Device_TestModule extends DeviceConfig {
   /**
    * Constructs a Device object.
    *
    * It is CRUTCIAL that the constructor sets $this->supportedEapMethods to an
array of methods
    * available for the particular device.
```

```
     */
     final public function __construct() {
       $this->supportedEapMethods  = array(EAP::$TLS, EAP::$PEAP_MSCHAP2,
EAP::$TTLS_PAP);
     }
   /**
    * prepare a zip archive containing files and settings which normally would
be used inside the module to produce an installer
    */
    public function writeInstaller() {
    // create certificate files and save their names in $CA_files array
      $CA_files = $this->saveCertificateFiles('der');

     // copy a fixed file from the module Files directory
        if(! $this->copyFile('Module.howto'))
           debug(2, "copying of Module.howto failed\n");

     // copy a fixed file from the module Files directory and save under a
different name
        if( ! $this->copyFile('test_file','copied_test_file'))
           debug(2, "copying of Module.howto to copied_test_file failed\n");
        $this->dumpAttributes('profile_attributes');
        $installer_path =  $this->zipInstaller($this->attributes);
        return($installer_path);
    }
/**
   * zip files and return the archive name
   */
   private function zipInstaller($attr) {
     $e = preg_replace('/ +/','_',$attr['internal:inst_name'][0]).'.zip';
     $o = system('zip -q '.$e.' *');
     return $e;
    }

}
?>
```

### 3.5.8.6 *MS Windows*

The MS Windows installer has been designed for use with Windows Vista and newer Windows operating systems, since these are the only systems that support a multiple wireless profile needed to support two different types of encryption. The Windows module supports EAP-PEAP-MSCAPv2, EAP-TLS and EAP-TTLS-PAP (using a GNU licensed version of SecureW2). Figure 3.11 presents example screenshots of a running installer. Customised elements, such as the logo, institution name, local support contacts and the user group can be seen.

If required, this installer can also display a license agreement, where the user is required to consent to conditions of use.

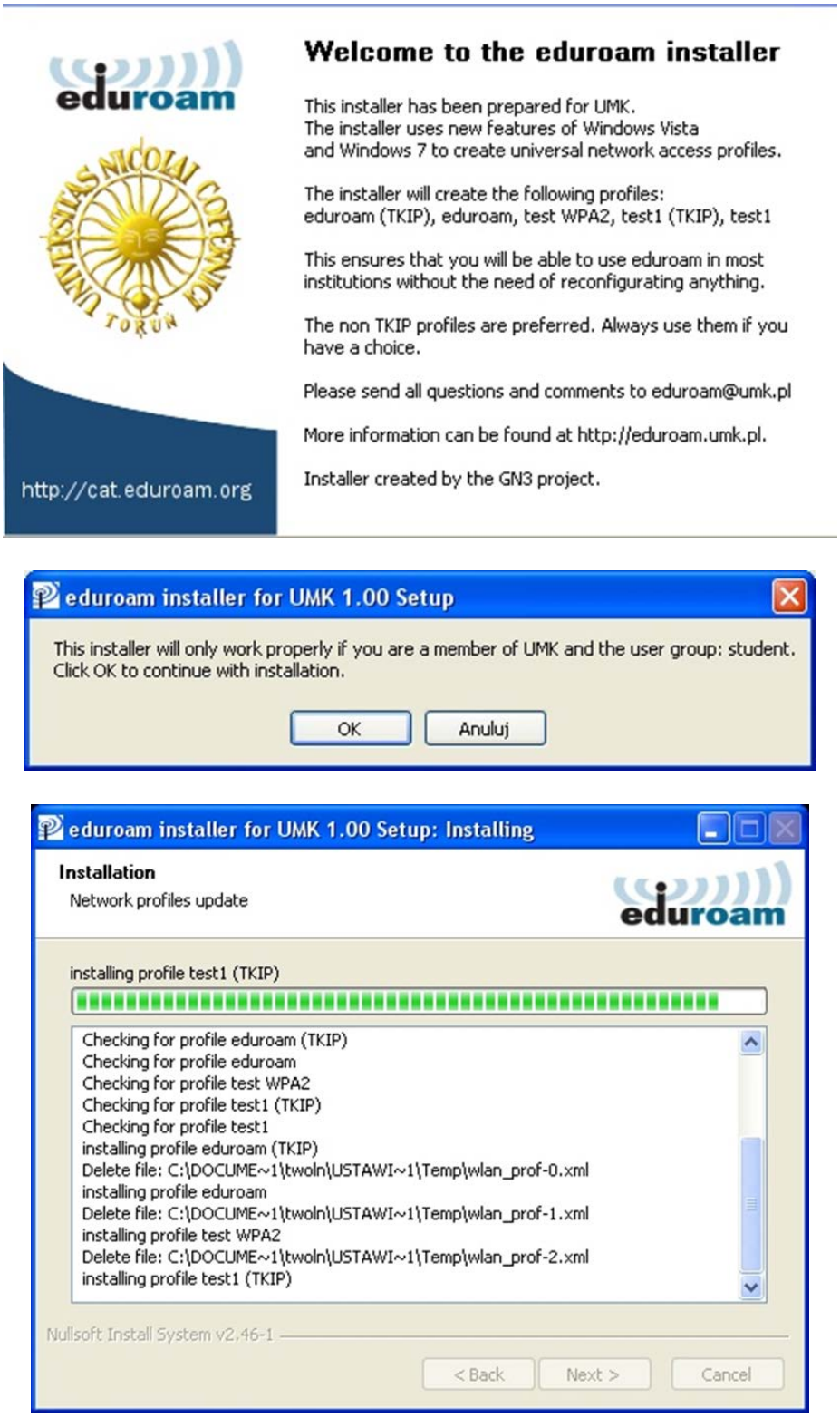


Figure 3.11: Example installer screens, as viewed in a Windows environment

### 3.5.9 Open Issues and Future Work

#### 3.5.9.1 *EAP-TLS*

As previously mentioned, EAP-TLS support may cause problems if an installer is required to contain a user's private key. Even if the key is password protected, a brute-force attack can break it, therefore, telling the user to trust an external system with the private key would be a bad security practice. On the other hand, not supporting EAP-TLS in such cases would be a painful limitation. Decisions on what can be done for this scenario will need to be taken by the eduroam CAT development team before the public software release of Version 1.0.

#### 3.5.9.2 *Installer Caching*

The eduroam CAT API currently generates fresh installers whenever a user clicks the 'Download' button. This is a waste of computing resources, because the installer code will seldom change, and the code could be stored for future downloads. Therefore, a caching system is planned. The system is designed in such a way that the presentation layer may easily check for a cached file and immediately serve it if present, or call the installer generator to produce a new one (which is then saved to the cache). The device module code will not have to be touched.

Implementing a caching system as part of the eduroam CAT install will require it to check for database updates that may invalidate the cache, but this is fairly straightforward work, and will certainly be done before version 1.0 is released.

#### 3.5.9.3 *User Instructions*

Users of the CAT may require written instructions on how to use the installers, therefore, multilingual instructions will need to be written and be presented to the users before the download commences.

Alternatively, some eduroam institutions may prefer to provide instructions locally, in which case, it is possible that the CAT should simply link to them. The best approach will become clear after some real usage and user experience is collected.

Similarly, instructions that are used in place of installers, for use by devices that cannot support an installer, could be another useful service, but again, local instructions may prove to be more appropriate.

# 4 Coordination with Other Activities

## 4.1 SA3 T2: WPA2 Discontinuation Consultancy

Deploying eduroam as an eduroam SP offers various degrees of freedom in equipment configuration. One of these degrees of freedom is the choice of encryption levels. The eduroam policy currently requires either WPA/TKIP or WPA2/AES, with a preference for WPA2/AES. While this was a good recommendation at the time of writing the original eduroam policy, the industry has since moved on to near-ubiquitous equipment support for WPA2/AES.

At the same time, known attacks on WPA/TKIP are surfacing, which make the use of this encryption setting increasingly less useful.

Another downside of the freedom of choice is that some supplicants will refuse or require manual user reconfiguration if they encounter an encryption level that differs from their usual settings. Thus, an eduroam user who roams to a different eduroam SP might encounter difficulties in his/her device configuration. It would be more user-friendly if all eduroam SPs would offer a known, working-baseline encryption that works ubiquitously, without the need to reconfigure an end-user device.

Further to that, industry specifications for new chipsets no longer foresee the possibility of operating on a WPA/TKIP-only network. As shown on the last page of the Wi-Fi Alliance security roadmap [Wi-Fi Alliance roadmap], newly produced client devices (STAtions in their terminology) are no longer required to be able operate on a pure WPA/TKIP network (WPA-only networks (STA)“not tested” from January 2010 onwards). In addition, there is a plan signalled by the Wi-Fi Alliance to disallow operation on WPA/TKIP networks for new chipsets from 2012. Although this timeline may be a bit too stringent, it is clear that the expected lifetime of WPA/TKIP is nearing its end.

As a result, eduroam Operations is striving to establish a common baseline of encryption levels. With WPA/TKIP already being attackable and on an industry certification discontinuation path, it became apparent that the only encryption level to be pursued would be WPA2/AES.

JRA3 T1 provided input to the change process, which resulted in a new draft version of the eduroam policy (both European policy and the Global Compliance Document, see Section 4.3.1 Participation in the Global eduroam Governance Committee). In addition, JRA3 T1 also plans to issue an advanced warning advisory with corresponding text to eduroam SP administrators.

## 4.2 eduroam Norway: Traffic Path Debugging

Project partner NORDUnet, specifically UNINETT, has created ‘eduDbg’, an eduroam traffic path debugging tool. eduDbg is described in this section.

The purpose of eduDbg is to collect traces of eduroam authentication traffic as that traffic passes through the eduroam RADIUS server hierarchy. In the case of authentication problems, this tool can be used to examine how far a certain login attempt can be progressed before failing. It can reveal infrastructure or misconfiguration problems on all levels of the infrastructure.

### 4.2.1 Introduction

eduDbg is a toolset that reads the national proxy's syslog files which contains information regarding the authentication request by a visitor at a visiting institution. It stores this information in the database, so that it is easily accessible at a later stage from a built-in, user-friendly web front end.

### 4.2.2 Usage

The front-end (the eduDbg web service) reads the stored information in the database and allows the administrator/user to search for the required information. The information contains the received authentication information at the national proxy level from the request performed by a visitor when visiting a remote institution. It allows the administrator/user to search the following fields to find the required information.

- Date From: The date from which the administrator/user would like to search for information.
- Date To: The date *up to* which administrator/user would like to search for the information.
- Username: The username is userID/outer identity provided by the user while making the request.
- Realm: The realm name provided in the request.
- National Host: As there can be more than one national server, it is possible to search for a specific national server to process the request.
- Status: The status field enables a search for the specific status message received from the home institution server (e.g. Access-accept, Access-reject etc.).
- Home Server: This field allows a search for users from a specific institution, as this field will be as the home server in the request.
- Remote Server: This field allows users to conduct a search of users visiting a specific institution, as this field can be the name of server from which the request is sent for authentication to home institution.

- Station ID:This field allows the administrator/user to search for the specific access station from where the visiting user has tried to connect to the eduroam network.

The results provided by the above search details the information that can be used to debug the problem at hand. It can be used to handle the connection failure problem or can also be used for proactive maintenance undertaken by the network administrator.

### 4.2.3 Use Cases

Two major use cases of the eduDbg tool are:

- Connection Failure: A user is not able to connect to the network through his/her eduroam account while visiting another institution.

- Proactive Maintenance: In this use case, an administrator can search for rejections or other anomalies and can contact the users to make the correction in their client's configuration; or contact an institution to correct its RADIUS configuration (loops, etc.).

### 4.2.4 Authentication and Authorisation

The front-end can provide authentication and authorisation to limit access to relevant realm(s) for the user. This is done by using plug-in modules. The Norwegian version uses the federated identity system, Feide, for authentication and a separate plug-in to obtain authorisation data from the customer management system. Other identity federations can replace the Feide module by a federated identity system of their choice. Due to the private nature of the collected data, it is not recommended to use this tool without a fine-grained authentication and authorisation infrastructure in front of it.

### 4.2.5 Deployment

eduDbg creates a wealth of data, which is very useful for debugging. Such a large amount of data may be a downside, however, when scaled to European level or even beyond. It is unclear if the tool can be scaled for European or worldwide operation.

JRA3 T1 encourages NRENs to participate in a pilot rollout in their federations. Further investigations on the scalability of eduDbg within these federations, and the corresponding large-scale dataset, need to be carried out. The tool will soon be made available as a beta version at http://downloads.geant.net.

## 4.3 Worldwide eduroam Community

### 4.3.1 Participation in the Global eduroam Governance Committee

eduroam was started as a TERENA "Mobility" Task Force research initiative, and was subsequently trademarked by TERENA in many parts of the world. Obviously, interest in the concept of educational roaming is not geographically limited to Europe. Nevertheless, eduroam has spread much more rapidly in Europe than elsewhere, and was consequently adopted on a more formal level, through the European eduroam policy and service definition, by GÉANT2 (GN2), and later GÉANT3 (GN3).

Because the resulting documentation that was produced was part of GÉANT, and was signed only by GÉANT participants, these documents do not inherently carry meaning beyond GÉANT's service borders. Still, the eduroam concept is defined worldwide by TERENA, leaving a gap of definitions on the governance structure of eduroam beyond GÉANT's service borders.

To fill this gap, TERENA has formed the Global eduroam Governance Committee (GeGC). It consists of representatives from all world regions with eduroam deployments (currently North America, Europe, and the Asia-Pacific region). Two European representatives come from SA3 T2, and one representative comes from JRA3 T1.

The group is working on a global harmonisation of eduroam across the regions of the world. Its first major milestone is to draw up the "Global eduroam Compliance Statement", which defines eduroam's foundation and future harmonisation, and is to be signed by all participating countries worldwide (for the GÉANT service region, a signature from the consortium replaces the individual country signatures).

### 4.3.2 Operator-Name Advisory

As reported in Section 3.3.1 Operator-Name of DJ3.1.2,1, the introduction of the new IETF attribute "Operator-Name" is, even if useful, problematic on some RADIUS server implementations. JRA3 T1 has worked together with the national eduroam operator in the United Kingdom, JANET, and in May 2011, has issued an advisory for eduroam IdP administrators which use Microsoft RADIUS servers (MS IAS or MS NPS) [eduroam advisory].

### 4.3.3 Transformation of GN2-DJ5.1.5,3 (The eduroam 'Cookbook')

The eduroam 'cookbook' is an early piece of work from GÉANT2. It describes best practice in equipment configuration, and provides a basic understanding of how eduroam works. It is targeted at eduroam administrators (not end users) and aims to provide a good introduction to the setup of eduroam at a hotspot or as an eduroam IdP.

The first edition of the cookbook (GN2 DJ5.1.5) was issued in February 2007, and the third edition (DJ5.1.5,3) was issued in October 2008. [eduroam cookbook]

Naturally, a document with in-depth technical content is bound to become outdated and in need of constant maintenance.

In GN3, JRA3 T1 has inherited maintenance of this GN2 deliverable. It quickly turned out that the GÉANT content delivery structure of creating PDF deliverables and long inter-version delays is less than ideal for a technical support resource.

As a consequence of this finding, the deliverable's content has been transferred into the TERENA Wiki [eduroam TERENA Wiki] and is open for reading to the general public. Write access is restricted to known contributors in order to maintain high-quality content. The Wiki has become a lively resource and is regularly updated with new versions of equipment, manufacturer information and “HOWTOs” as the technology and best practices evolve.

Note that the TERENA Wiki is the authoritative source for eduroam setup information, and there is no plan to continue publication of the cookbook in its previous format.

# References


| | |
|---|---|
| **DJ3.1.2,1** | http://www.geant.net/Media_Centre/Media_Library/Media%20Library/GN3-10-304%20DJ3.1.2,1%20%20Roaming%20Developments%2024FEB11.pdf |
| **draft-ietf-radext-radsec** | http://tools.ietf.org/html/draft-ietf-radext-radsec-09 |
| **draft-ietf-radext-tcp-transport** | http://tools.ietf.org/html/draf-ietf-radtext-tcp-transport-09 |
| **dynamic discovery draft** | http://www.ietf.org/internet-drafts/draft-ietf-radext-dynamic-discovery-03.txt |
| **EAP** | http://en.wikipedia.org/wiki/Key_(cryptography) |
| **EAP statement update** | http://tools.ietf.org/html/draft-winter-abfab-eapapplicability-00 |
| **EAP-PWD** | http://tools.ietf.org/html/rfc5931 |
| **eduroam advisory** | http://www.eduroam.org/downloads/docs/advisory/eduroamOT-admin-advisory-004.pdf |
| **eduroam cookbook** | http://www.eduroam.org/downloads/docs/GN2-08-230-DJ5.1.5.3-eduroamCookbook.pdf |
| **eduroam HOWTO Wiki** | http://www.eduroam.org/index.php?p=europe&s=docs |
| **eduroam TERENA Wiki** | https://confluence.terena.org/display/H2eduroam/ |
| **F-Ticks** | http://monitor.eduroam.org/f-ticks/. |
| **IEEE 802.1X standards** | http://standards.ieee.org/getieee802/download/802.1X-2010.pdf |
| **IETF Tracker** | https://datatracker.ietf.org/doc/draft-ietf-radext-radsec/. |
| **Moonshot** | http://www.ja.net/moonshot |
| **PB message transport** | http://www.ietf.org/proceedings/74/IDs/draft-ietf-nea-pb-tnc-04.txt |
| **Wi-Fi Alliance roadmap** | http://downloads.geant.net/repository/public/Roaming/Documents/201005_wifi_certified_security_roadmap_public.pdf |

# Glossary

| | |
|---|---|
| **802.11** | A set of standards for implementing wireless local area network (WLAN) computer communication in the 2.4, 3.6 and 5GHz frequency bands |
| **API** | Application Programming Interface |
| **CA** | Certification Authority |
| **CAT** | Configuration Assistant Tool |
| **DANE** | DNS-based Authentication of Named Entities |
| **DHCP** | Dynamic Host Configuration Protocol |
| **DNS** | Domain Name System |
| **DNSSEC** | DNS Security Extensions |
| **DTLS** | Datagram Transport Layer Security |
| **EAP** | Extensible Authentication Protocol |
| **EAP-EKE** | EAP Encrypted Key Exchange |
| **EAP-FAST** | EAP Flexible Authentication via Secure Tunnelling. |
| **EAP-FASTv2** | EAP Flexible Authentication via Secure Tunnelling version 2 a derivate of EAP-FAST |
| **EAP-GTC** | EAP Generic Token Card |
| **EAP-PWD** | EAP Authentication using only a password |
| **EAP-TEAM** | EAP a derivative of EAP-PEAP |
| **EAP-TLS** | EAP Transport Layer Security |
| **EAP-TLV** | EAP Type-Length-Value |
| **EAP-TTLS** | EAP Tunnelled Transport Layer Security |
| **eduDbg** | Traffic path debugging tool for eduroam |
| **eduroam** | Roaming confederation aiming to provide mutual roaming network access to its members – users from the education and research sector worldwide |
| **eduPKI** | Education Public Key Infrastructure |
| **EMU WG** | EAP Method Update Working Group |
| **F-Ticks** | Federated Ticker system (eduroam's advanced statistics tool) |
| **Feide** | Norwegian Federated Identity system |
| **FreeRADIUS** | Open Source version of RADIUS |
| **GÉANT2 (GN2)** | Gigabit European Advanced Network Technology 2 is the main European network for research and education purposes, successor to the pan-European multi-gigabit research network GÉANT |
| **GÉANT3 (GN3)** | The GN3 Project (3rd GÉANT project year, successor to GÉANT 2) |
| **GeGC** | Global eduroam Governance Committee |
| **GNU** | GNU's Not Unix – a Unix-like computer operating system developed by the GNU project |
| **GTC** | Generic Token Card |
| **GUI** | Graphical User Interface |
| **IdP** | Identity Provider |
| **IEEE** | Institute of Electrical and Electronics Engineers |
| **IETF** | Internet Engineering Task Force |
| **IP** | Internet Protocol |
| **IPR** | Intellectual Property Rights |
| **IPSec** | Internet Protocol Security |
| **IPv4** | Internet Protocol Version 4 |
| **IPv6** | Internet Protocol Version 6 |

| | |
|---|---|
| **ISO/OSI** | International Organization for Standardization/Open System Interconnection Reference Model |
| **JANET** | UK National Education and Research Network |
| **JRA3** | GN3 Joint Research Activity 3 (Multi-Domain User Application Research) |
| **LAN** | Local Area Network |
| **MAC** | Media Access Control address |
| **Moonshot** | Project Moonshot is a JANET(UK)-led initiative, in partnership with GN3 and others, to develop a single unifying technology for extending the benefits of federated identity to a broad range of non-Web services, including Cloud infrastructures, High Performance Computing & Grid infrastructures and other commonly deployed services including mail, file store, remote access and instant messaging. |
| **MS IAS** | Microsoft RADIUS servers |
| **MS NPS** | Microsoft RADIUS servers |
| **NAPTR** | Name Authority Pointer |
| **NEA WG** | Network Endpoint Assessment Working Group |
| **NORDUnet** | Nordic Infrastructure for Research & Education |
| **NREN** | National Research and Education Networks |
| **NSIS** | Nullsoft Scriptable Install System |
| **OT** | Operations Team |
| **PA** | Posture Attribute |
| **PB** | Posture Broker |
| **PEAP** | Protected Extensible Authentication Protocol |
| **PHP** | Personal Home Page |
| **PT** | Posture Transport |
| **PT-EAP** | Posture Transport (PT) Protocol For EAP Tunnel Methods |
| **RADIUS** | Remote Authentication Dial In User Service |
| **RadSec** | Protocol for transporting RADIUS datagrams over TCP and TLS. |
| **RFC** | Request for Comments |
| **S-NAPTR** | Straightforward Name Authority Pointer |
| **SA3** | GN3 Service Activity 3 (Multi-Domain User Applications) |
| **SAML** | Security Assertion Mark-up Language |
| **SecureW2** | EAP-TTLS client for Windows platforms |
| **SP** | Service Provider |
| **SQL** | Structured Query Language |
| **SRCE** | University Computing Centre, University of Zagreb |
| **SSID** | Service Set Identifier |
| **TCP** | Transmission Control Protocol |
| **TERENA** | Trans-European Research and Education Networking Association |
| **TKIP** | Temporal Key Integrity Protocol |
| **TLS** | Transport Layer Security |
| **TTLS-PAP** | Tunnelled Transport Layer Security with Password Authentication Protocol |
| **TNC** | Trusted Network Connect |
| **UNINETT** | Norwegian National Research and Education Network |
| **UNIX** | A Multitasking, Multi-user Computer Operating System |
| **URL** | Uniform Resource Locator |
| **VLAN** | Virtual Local Area Network |
| **VPN** | Virtual Private Network |
| **Wi-Fi** | A trademark of the Wi-Fi Alliance and the brand name for products using the IEEE 802.11 family of standards |

| | |
|---|---|
| **WLAN** | Wireless Local Area Network |
| **WPA** | Wi-Fi Protected Access - security protocols and security certification programs developed by the Wi-Fi Alliance to secure wireless computer networks |
| **WPA2** | Wi-Fi Protected Access II - security protocols and security certification programs developed by the Wi-Fi Alliance to secure wireless computer networks a derivate of WPA |
| **WPA2/AES** | Advanced Encryption Standard used by WPA2 |
| **WPA/TKIP** | Temporal Key Integrity Protocol used by WPA |
| **X.509** | Public Key Infrastructure Certificate and Certificate Revocation List |
| **XML** | Extensible Mark-up Language |