Network Working Group	K. Wierenga
Internet-Draft	Cisco Systems
Intended status: Informational	S. Winter
Expires: January 16, 2014	RESTENA
	T. Wolniewicz
	Nicolaus Copernicus University
	July 15, 2013

The eduroam architecture for network roaming

draft-wierenga-ietf-eduroam-01.txt

Abstract

This document describes the architecture of the eduroam service for federated (wireless) network access in academia. The combination of 802.1X, EAP and RADIUS that is used in eduroam provides a secure, scalable and deployable service for roaming network access. The successful deployment of eduroam over the last decade in the educational sector may serve as an example for other sectors, hence this document. In particular the initial architectural and standards choices and the changes that were prompted by operational experience are highlighted.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on January 16, 2014.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction

In 2002 the European Research and Education community set out to create a network roaming service for students and employees in academia [eduroam-start]. Now over 10 years later this service has grown to more than 5000 service locations, serving millions of users on all continents with the exception of Antarctica.

This memo serves to explain the considerations for the design of eduroam as well as to document operational experience and resulting changes that led to IETF standardization effort like RADIUS over TCP [RFC6613] and RADIUS with TLS [RFC6614] and that promoted alternative uses of RADIUS like in ABFAB [I-D.ietf-abfab-arch]. Whereas the eduroam service is limited to academia, the eduroam architecture can easily be reused in other environments.

First this memo describes the original architecture of eduroam. Then a number of operational problems are presented that surfaced when eduroam gained wide-scale deployment. Lastly, enhancements to the eduroam architecture that mitigate the aforementioned issues are discussed.

1.1. Terminology

This document uses identity management and privacy terminology from [I-D.iab-privacy-considerations]. In particular, this document uses the terms Identity Provider, Service Provider and identity management.

1.2. Notational Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

Note: Also the policy that eduroam participants subscribe to, expresses the requirements for participation in RFC 2119 language.

1.3. Design Goals

The guiding design considerations of eduroam were as follows:

- Unique identification of users at the edge of the network

The access service provider (SP) needs to be able to determine whether a user is authorized to use the network resources. Furthermore, in case of abuse of the resources, there is a requirement to be able to identify the user uniquely (with the cooperation of the user’s IdP operator).

- Enable (trusted) guest use:

In order to enable roaming it should be possible for users of participating institutions to get seamless access to the networks of other institutions.

Note: traffic separation between guest users and normal users is possible (for example through the use of VLANs), and indeed often desirable and widely used in eduroam.

- Scalable

The infrastructure that is created should scale to a large number of users and organizations without requiring a lot of coordination and other administrative procedures (possibly after initial set up). Specifically, it should not be necessary for a user that visits another organization to go through an administrative process.

- Easy to install and use

It should be easy for both organizations and users to participate in the roaming infrastructure as that may otherwise inhibit wide scale adoption. In particular, there should be no or easy client installation and only one-off configuration.

- Secure

An important design criterion has been that there needs to be a security association between the end-user and their home organization, eliminating the possibility of credentials theft. The minimal requirements for security are specified in the eduroam policy and subject to change over time. As an additional protection against user errors and negligence, it should be possible for participating organizations to set their own additional requirements for the quality of authentication of users without the need for the infrastructure as a whole to implement the same standard.

- Privacy preserving

The design of the system provides for user anonymization, i.e. it is possible to hide the user’s identity from any third parties, including visited institutions.

- Standards based

In an infrastructure in which many thousands of organizations participate it is obvious that it should be possible to use equipment from different vendors, therefore it is important to base the infrastructure on open standards.

1.4. Solutions that were considered

Three architectures were trialed: one based on the use of VPN-technology (deemed secure but not-scalable), one Web captive-portal based (scalable but not secure) and 802.1X-based, the latter being the basis of what is now the eduroam architecture.

The chosen architecture is based on:

- 802.1X ([dot1X-standard])as port based authentication framework using
- EAP ([RFC3748]) for integrity and confidentially protected transport of credentials and a
- RADIUS ([RFC2865]) hierarchy as trust fabric.

2. Classic Architecture

Federations, like eduroam, implement essentially two types of direct trust relations (and one indirect). The trust relation between an end-user and the Identity Provider (IdP, operated by the home organization of the user) and between the IdP and the Service Provider (SP, in eduroam the operator of the network at the visited location). In eduroam the trust relation between user and IdP is through mutual authentication. IdPs and SP establish trust through the use of a RADIUS hierarchy.

These two forms of trust relations in turn provide the transitive trust relation that makes the SP trust the user to use its network resources.

2.1. Authentication

Authentication in eduroam is achieved by using a combination of IEEE 802.1X [dot1X-standard] and EAP [RFC4372] (the latter carried over RADIUS, see below).

2.1.1. 802.1X

By using the 802.1X [dot1X-standard] framework for port-based network authentication, organizations that offer network access (SPs) for visiting (and local) eduroam users can make sure that only authorized users get access. The user (or rather the user's supplicant) sends an access request to the authenticator (wireless access point or switch) at the SP, the authenticator forwards the access request to the authentication server of the SP which in turn proxies the request through the RADIUS hierarchy to the authentication server of the user's home organization (the IdP, see below).

Note: The security of the connections between local wireless infrastructure and local RADIUS servers is a part of the local network of each SP, therefore it is out of scope of the document. For completeness it should be stated that security between access points and their controllers is vendor specific, security between controllers (or standalone access points) and local RADIUS servers is based on the typical RADIUS shared secret mechanism.

In order for users to be aware of the availability of the eduroam service, an SP that offers wireless network access MUST broadcast the SSID 'eduroam', unless that conflicts with the SSID of another eduroam SP, in which case an SSID starting with "eduroam-" MAY be used. The downside of the latter is that clients will not automatically connect to that SSID, thus losing the seamless connection experience.

Note: A direct implication of the common eduroam SSID is that the users cannot distinguish between a connection to a home network and a guest network at another eduroam institution (IEEE802.11-2012 does have the so-called “Interworking” extensions to make that distinction, but these are not widely implemented yet). Therefore, users should be made aware that they should not assume data confidentiality in the eduroam infrastructure.

To protect over-the-air user data confidentiality IEEE 802.11 wireless networks of eduroam SP's MUST deploy WPA2+AES, and MAY additionally support WPA/TKIP as a courtesy to users of legacy hardware.

2.1.2. EAP

The use of the Extensible Authentication Protocol (EAP) [RFC4372] serves 2 purposes. In the first place a properly chosen EAP-method allows for integrity and confidentiality protected transport of the user credentials to the home organization. Secondly, by having all RADIUS servers transparently proxy access requests regardless of the EAP-method inside the RADIUS packet, the choice of EAP-method is between the 'home' organization of the user and the user, in other words, in principle every authentication form that can be carried inside EAP can be used in eduroam, as long as they adhere to minimal requirements as set forth in the eduroam policy.

```
                         +-----+
                        /       \
                       /         \
                      /           \
                     /             \
    ,----------\    |               |   ,---------\
    |    SP     |    |    eduroam    |   |    IdP  |
    |          +----+  trust fabric +---+         |
    `------+---'    |               |   '-----+---'
           |        |               |         |
           |         \             /          |
           |          \           /           |
           |           \         /            |
           |            \       /             |
      +----+             +-----+              +----+
      |                                            |
      |                                            |
  +---+--+                                      +--+---+
  |      |                                      |      |
+-+------+-+    ___________________________     |      |
|          |   O__________________________ )    +------+
+----------+
Host (supplicant)      EAP tunnel       Authentication server
```

Figure 1: Tunneled EAP

Proxying of access requests is based on the outer identity in the EAP-message. Those outer identities MUST be of the form something@realm, where the realm part is the domain name of the domain that the IdP belongs to. In order to preserve credentials protection, participating organizations MUST deploy EAP-methods that provide mutual authentication. For EAP methods that support outer identity, anonymous outer identities are recommended. Most commonly used in eduroam are the so-called tunneled EAP-methods that first create a server authenticated TLS tunnel through which the user credentials are transmitted. As depicted in Figure 1, the use of a tunneled EAP-method creates a direct logical connection between the supplicant and the authentication server, even though the actual traffic flows through the RADIUS-hierarchy.

2.2. Federation Trust Fabric

The eduroam federation trust fabric is based on RADIUS. RADIUS trust is based on shared secrets between RADIUS peers. In eduroam any RADIUS message originating from a trusted peer is implicitly assumed to originate from a member of the roaming consortium.

2.2.1. RADIUS

The eduroam trust fabric consists of a proxy hierarchy of RADIUS servers (organizational, national, global), loosely based on the DNS hierarchy. That is, typically an organizational RADIUS server agrees on a shared secret with a national server and the national server agrees on a shared secret with the root server. Access requests are routed through a chain of RADIUS proxies towards the home organization of the user, and the access accept (or reject) follows the same path back.

Note: In some circumstances there are more levels of RADIUS servers, like for example regional or continental servers, but that doesn't change the general model. Also the packet exchange that is described below requires in reality several round-trips.

```
                               +-------+
                               |       |
                               |   .   |
                               |       |
                               +---+---+
                                 / | \
               +----------------/  |  \---------------------+
               |                   |                        |
               |                   |                        |
               |                   |                        |
            +--+---+            +--+--+                +----+---+
            |      |            |     |                |        |
            | .edu |    . . .   | .nl |      . . .     | .ac.uk |
            |      |            |     |                |        |
            +--+---+            +--+--+                +----+---+
             / | \                 | \                      |
            /  |  \                |  \                     |
           /   |   \               |   \                    |
    +-----+    |    +-----+        |    +------+            |
    |          |          |        |           |            |
    |          |          |        |           |            |
+---+---+ +----+---+ +----+---+ +--+---+ +-----+----+ +-----+-----+
|       | |        | |        | |      | |          | |           |
|utk.edu| |utah.edu| |case.edu| |hva.nl| |surfnet.nl| |soton.ac.uk|
|       | |        | |        | |      | |          | |           |
+----+--+ +--------+ +--------+ +------+ +----+-----+ +-----------+
     |                                        |
     |                                        |
  +--+--+                                  +--+--+
  |     |                                  |     |
+-+-----+-+                                |     |
|         |                                +-----+
+---------+
user: paul@surfnet.nl             surfnet.nl Authentication server
```

Figure 2: eduroam RADIUS hierarchy

Routing of access requests to the home IdP is done based on the realm part of the outer identity. For example (see: Figure 2), when user paul@surfnet.nl of SURFnet (surfnet.nl) tries to gain wireless network access at the University of Tennessee at Knoxville (utk.edu) the following happens:

- Paul's supplicant transmits an EAP access request to the Access Point (Authenticator) at UTK with outer identity say anonymous@surfnet.nl
- The Access Point forwards the EAP message to its Authentication Server (the UTK RADIUS server)
- The UTK RADIUS server checks the realm to see if it is a local realm, since it isn't the request is proxied to the .edu RADIUS server
- The .edu RADIUS server verifies the realm, and since it is not a in a .edu subdomain it proxies the request to the root server
- The root RADIUS server proxies the request to the .nl RADIUS server
- The .nl RADIUS server proxies the request to the surfnet.nl server
- The surfnet.nl RADIUS server decapsulates the EAP message and verifies the user credentials
- The surfnet.nl RADIUS server informs the utk.edu server of the outcome of the authentication request (accept or deny) by proxying the outcome through the RADIUS hierarchy in reverse order.
- The UTK RADIUS server instructs the UTK Access Point to either accept or deny access based on the outcome of the authentication.

Note: The depiction of the root RADIUS server is a simplification of reality. In reality the root server is distributed over 3 continents and each maintains a list of top level realms a specific root server is responsible for. So in reality, for intercontinental roaming there is an extra proxy step from one root server to the other involved.

3. Issues with initial Trust Fabric

While the hierarchical RADIUS architecture described in the previous section has served as the basis for eduroam operations for an entire decade, the exponential growth of authentications is expected to lead to, and has in fact in some cases already lead to, performance and operations bottlenecks on the aggregation proxies. The following sections describe some of the shortcomings, and the resulting remedies.

3.1. Server Failure Handling

In eduroam, authentication requests for roaming users are statically routed through pre-configured proxies. The number of proxies varies: in a national roaming case, the number of proxies is typically 1 or 2 (some countries deploy regional proxies, which are in turn aggregated by a national proxy); in international roaming, 3 or 4 proxy servers are typically involved (the number may be higher along some routes).

RFC2865 [RFC2865] does not define a failover algorithm. In particular, the failure of a server needs to be deduced from the absence of a reply. Operational experience has shown that this has detrimental effects on the infrastructure and end user experience:

1. Authentication failure: the first user whose authentication path is along a newly-failed server will experience a long delay and possibly timeout
2. Wrongly deduced states: since the proxy chain is longer than 1 hop, a failure further along in the authentication path is indistinguishable from a failure in the next hop.
3. Inability to determine recovery of a server: only a "live" authentication request sent to a server which is believed inoperable can lead to the discovery that the server is in working order again. This issue has been resolved with RFC5997 [RFC5997].

The second point can have significant impact on the operational state of the system in a worst-case scenario: Imagine one realm's home server being inoperable. A user from that realm is trying to roam internationally and tries to authenticate. The RADIUS server on the hotspot location will assume its own national proxy is down, because it does not reply. That national server, being perfectly alive, in turn will assume that the international aggregation proxy is down; which in turn will believe the home country proxy national server is down. None of these assumptions are true. Worse yet: should any of these servers trigger a failover to a redundant backup RADIUS server, it will still not receive a reply, because the request will still be routed to the same defunct home server. Within a short time, all redundant aggregation proxies might be considered defunct by their preceding hop.

In the absence of proper next-hop state derivation, some interesting concepts have been introduced by eduroam participants; the most noteworthy being a failover logic which considers up/down states not per next-hop RADIUS peer, but instead per realm (See [dead-realm] for details). As of recent, RFC5997 [RFC5997] implementations and cautious failover parameters make such a worst-case scenario very unlikely to happen, but are still an important issue to consider.

3.2. No error condition signalling

The RADIUS protocol lacks signalling of error conditions, and the IEEE 802.1X protocol does not allow to convey extended failure reasons to the end-user's device. For eduroam, this creates issues in a twofold way:

- The home server may have an operational problem, for example if its authentication decisions depend on an external data source such as ActiveDirectory or an SQL server, and if these external dependencies are out of order. If the RADIUS interface is still functional, there are two options how to reply to an Access-Request which can't be serviced due to such error conditions:
    1. Do Not Reply: the inability to reach a conclusion can be treated by not replying to the request. The upside of this approach is that the end-user's software doesn't come to wrong conclusions and won't give unhelpful hints such as "maybe your password is wrong". The downside is that intermediate proxies may come to wrong conclusions because their downstream RADIUS server isn't responding.
    2. Reply with Reject: in this option, the inability to reach a conclusion is treated like an authentication failure. The upside of this approach is that intermediate proxies maintain a correct view on the reachability state of their RADIUS peer. The downside is that EAP supplicants on end-user devices often react with either false advice ("your password is wrong") or even trigger permanent configuration changes (e.g. the Windows built-in supplicant will delete the credential set from its registry, prompting the user for their password on the next connection attempt). The latter case of Windows is a source of significant helpdesk activity; users may have forgotten their password after initially storing it, but are suddenly prompted again.

There have been epic discussions in the eduroam community which of the two approaches is more appropriate; but they were not conclusive.

- Similar considerations as above apply when an intermediate proxy does not receive a reply from a downstream RADIUS server. The proxy may either choose not to reply to the original request, leading to retries and its upstream peers coming to wrong conclusions about its own availability; or it may decide to reply with Access-Reject to indicate its own liveliness, but again with implications for the end user.

The ability to send Status-Server watchdog requests is only of use after the fact, in case a downstream server doesn't reply (or hasn’t been contacted in a long while, so that it’s previous working state is stale). The active link-state monitoring of the TCP connection with e.g. RADIUS/TLS (see below) gives a clearer indication whether there is an alive RADIUS peer, but does not solve the defunct backend problem. An explicit ability to send Error-Replies, on the RADIUS (for other RADIUS peer information) and EAP level (for end-user supplicant information), would alleviate these problems but is currently not available.

3.3. Routing table complexity

The aggregation of RADIUS requests based on the structure of the user's realm implies that realms ending with the same top-level domain are routed to the same server; i.e. to a common administrative domain. While this is true for country code Top Level Domains (ccTLDs), which map into national eduroam federations, it is not true for realms residing in generic Top Level Domains (gTLDs). Realms in gTLDs were historically discouraged because the automatic mapping "realm ending" -> "eduroam federation's server" could not be applied. However, with growing demand from eduroam realm administrators, it became necessary to create exception entries in the forwarding rules; such realms need to be mapped on a realm-by-realm basis to their eduroam federations. Example: "kit.edu" needs to be routed to the German federation server; "iu.edu" neeeds to be routed to the U.S.A. federation server.

While the ccTLDs occupy only approx. 50 routing entries in total (and have a upper bound of approx. 200), the potential size of the routing table becomes virtually unlimited if it needs to accomodate all individual entries in .edu, .org, etc.

In addition to that, all these routes need to be synchronised between three international root servers, and the updates need to be applied manually to RADIUS server configuration files. The frequency of the required updates makes this approach fragile and error-prone as the number of entries grows.

3.4. UDP Issues

RADIUS is based on UDP, which was a reasonable choice when its main use was with simple PAP requests which required only exactly one packet exchange in each direction.

When transporting EAP over RADIUS, the EAP conversations requires multiple round-trips; depending on the total payload size, 8-10 round-trips are not uncommon. The loss of a single UDP packet will lead to user-visible delays and might result in servers being marked as dead due to the absence of a reply. The proxy path in eduroam consists of several proxies, all of which introduce a very small packet loss probability; i.e. the more proxies are needed, the higher the failure rate is going to be.

For some EAP types, depending on the exact payload size they carry, RADIUS servers and/or supplicants may choose to fill as much EAP data into a single RADIUS packet as the supplicant's layer 2 medium allows for, typically 1500 Bytes. In that case, the RADIUS encapsulation around the EAP-Message will itself also exceed 1500 Byte size which in turn means the UDP datagram which carries the RADIUS packet will need to be fragmented on the IP layer. While this is not a problem in theory, practice has shown evidence of misbehaving firewalls which erroneously discard non-first UDP fragments, which ultimately leads to a denial of service for users with such EAP types and that specific configuration.

One EAP type proved to be particularly problematic: EAP-TLS. While it is possible to configure the EAP server to send smaller chunks of EAP payload to the supplicant (e.g. 1200 Bytes, to allow for another 300 Bytes of RADIUS overhead without fragmentation), very often the supplicants which send the client certificate do not expose such a configuration detail to the user. Consequently, when the client certificate is beyond 1500 Bytes in size, the EAP-Message will always make use of the maximum possible layer-2 chunk size, which introduces the fragmentation on the path from EAP peer to EAP server.

Both of the previously mentioned sources of errors (packet loss, fragment discard) lead to significant frustration for the affected users. Operational experience of eduroam shows that such cases are hard to debug since they require coordinated cooperation of all eduroam administrators on the authentication path. For that reason the eduroam community is developing monitoring tools that help to locate fragmentation problems.

3.5. Insufficient payload encryption and EAP server validation

The RADIUS protocol's design foresaw only the encryption of select RADIUS attributes, most notably User-Password. With EAP methods conforming to the requirements of RFC4017, the user's credential is not transmitted using the User-Password attribute, and stronger encryption than the one for RADIUS' User-Password is in use (typically TLS).

Still, the use of EAP does not encrypt all personally identifiable details of the user session. In particular, the user's device can be identified by inspecting the Calling-Station-ID attribute; and the user's location may be derived from observing NAS-IP-Address, NAS-Identifier or Operator-Name attributes. Since these attributes are not encrypted, even IP-layer third parties can harvest the corresponding data. In a worst-case scenario, this enables the creation of mobility profiles.

These profiles are not necessarily linkable to an actual user because EAP allows for the use of anonymous outer identities and protected credential exchanges. However, practical experience has shown that many users neglect to configure their supplicants in a privacy-preserving way or their supplicant doesn't support that. In particular, for EAP-TLS users, the use of EAP-TLS identity protection is not usually implemented and cannot be used. In eduroam, concerned individuals and IdPs which use EAP-TLS are using pseudonymous client certificates to provide for better privacy.

One way out, at least for EAP types involving a username, is to pursue the creation and deployment of pre-configured supplicant configurations which makes all the required settings in user devices prior to their first connection attempt; this depends heavily on the remote configuration possibilities of the supplicants though.

A further threat involves the verification of the EAP server's identity. Even though the cryptographic foundation, TLS tunnels, is sound, there is a weakness in the supplicant configuration: many users do not understand or are willing to invest time into the inspection of server certificates or the installation of a trusted CA. As a result, users may easily be tricked into connecting to an unauthorized EAP server, ultimately leading to a leak of their credentials to that unauthorized third party.

Again, one way out of this particular threat is to pursue the creation and deployment of pre-configured supplicant configurations which makes all the required settings in user devices prior to their first connection attempt.

Note: there are many different and vendor-proprietary ways to pre-configure a device with the necessary EAP parameters (examples include Apple, Inc's "mobileconfig" and Microsoft's "EAPHost" XML schema). Some manufacturers even completely lack any means to distribute EAP configuration data. We believe there is value in defining a common EAP configuration metadata format which could be used across manufacturers, ideally leading to a situation where IEEE 802.1X network end-users merely needs to apply this configuration file to configure any of their devices securely with the required connection properties.

Another possible threat involves transport of user-specific attributes in a Reply-Message. If, for example, a RADIUS server sends back a hypothetical RADIUS Vendor-Specific-Attribute "User-Role = Student of Computer Science" (e.g. for consumption of a SP RADIUS server and subsequent assignment into a "student" VLAN), this information would also be visible for third parties and could be added to the mobility profile.

The only way out to mitigate all information leakage to third parties is by protecting the entire RADIUS packet payload so that IP-layer third parties can not extract privacy-relevant information. RFC2865 RADIUS does not offer this possibility though.

4. New Trust Fabric

The operational difficulties with an ever increasing number of participants as documented in the previous section have led to a number of changes to the eduroam architecture that in turn have, as mentioned in the introduction, led to standardization effort.

Note: The enhanced architecture components are fully backwards compatible with the existing installed base, and are in fact gradually replacing those parts of it where problems may arise.

Whereas the user authentication using 802.1X and EAP has remained unchanged (i.e. no need for end-users to change any configurations), the issues as reported above have resulted in a major overhaul of the way EAP messages are transported from the RADIUS server of the SP to that of the IdP and back. The two fundamental changes are the use of TCP instead of UDP and reliance on TLS instead of shared secrets between RADIUS peers.

4.1. RADIUS with TLS

The deficiencies of RADIUS over UDP as described in Section 3.4 warranted a search for a replacement of RFC2865 [RFC2865] for the transport of EAP. By the time this need was understood, the designated successor protocol to RADIUS, Diameter [RFC3588], was already specified by the IETF. However, within the operational constraints of eduroam:

- reasonably cheap to deploy on many administrative domains
- supporting NASREQ Application
- supporting EAP Application
- supporting Diameter Redirect
- supporting validation of authentication requests of the most popular EAP types (EAP-TTLS, PEAP, and EAP-TLS)
- possibility to retrieve these credentials from popular backends such as Microsoft ActiveDirectory, MySQL

no single implementation could be found. In addition to that, no Wireless Access Points at the disposal of eduroam participants supported Diameter, nor did any of the manufacturers have a roadmap towards Diameter support. This led to the open question of lossless translation from RADIUS to Diameter and vice versa; a question not satisfactorily answered by NASREQ.

After monitoring the Diameter implementation landscape for a while, it became clear that a solution with better compatibility and a plausible upgrade path from the existing RADIUS hierarchy was needed. The eduroam community actively engaged in the IETF towards the specification of several enhancements to RADIUS to overcome the limitations mentioned in Section 3. The outcome of this process was [RFC6614] and [I-D.ietf-radext-dynamic-discovery].

With its use of TCP instead of UDP, and with its full packet encryption, while maintaining full packet format compatibility with RADIUS/UDP, RADIUS/TLS [RFC6614] allows to upgrade any given RADIUS link in eduroam without the need of a "flag day".

In a first upgrade phase, the classic eduroam hierarchy (forwarding decision taken by inspecting the realm) remains intact. That way, RADIUS/TLS merely enhances the underlying transport of the RADIUS datagrams. But this already provides some key advantages:

- explicit peer reachability detection using long-lived TCP sessions
- protection of user credentials and all privacy-relevant RADIUS attributes

RADIUS/TLS connections for the static hierarchy could be realised with the TLS-PSK operation mode (which effectively provides a 1:1 replacement for RADIUS/UDP's "shared secrets"), but since this operation mode is not widely supported as of yet, all RADIUS/TLS links in eduroam are secured by TLS with X.509 certificates from a set of accredited CAs.

This first deployment phase does not yet solve the routing table complexity problem (see (Section 3.3); this aspect is covered by introducing dynamic discovery for RADIUS/TLS servers.

4.2. Dynamic Discovery

When introducing peer discovery, two separate issues had to be addressed:

1. How to find the network address of a responsible RADIUS server for a given realm?
2. How to verify that this realm is an authorised eduroam participant?

4.2.1. Discovery of responsible server

Issue 1 can relatively simply be addressed by putting eduroam-specific service discovery information into the global DNS tree. eduroam does so by using Network Authority Pointer (NAPTR) records as per the S-NAPTR specification [RFC3958] with a private-use NAPTR service tag ("x-eduroam:radius.tls"). The usage profile of that NAPTR resource record is that exclusively "S" type delegations are allowed, and that no regular expressions are allowed.

A subsequent lookup of the resulting SRV records will eventually yield hostnames and IP addresses of the authoritative server(s) of a given realm.

Example (wrapped for readability):

```
> dig -t naptr education.example.

;; ANSWER SECTION:
education.example.            43200   IN      NAPTR   100 10 "s"
                                  "x-eduroam:radius.tls" ""
                                  _radsec._tcp.eduroam.example.


> dig -t srv _radsec._tcp.eduroam.example.

;; ANSWER SECTION:
_radsec._tcp.eduroam.example. 43200  IN      SRV     0 0 2083
                                             tld1.eduroam.example.

> dig -t aaaa tld1.eduroam.example.

;; ANSWER SECTION:
tld1.eduroam.example.         21751  IN      AAAA    2001:db8:1::2
```

Figure 3: SRV record lookup

From the operational experience with this mode of operation, eduroam is pursuing standardisation of this approach for generic AAA use cases. The current radext working group document for this is [I-D.ietf-radext-dynamic-discovery].

4.2.2. Verifying server authorisation

Any organisation can put "x-eduroam" NAPTR entries into their Domain Name Server, pretending to be eduroam Identity Provider for the corresponding realm. Since eduroam is a service for a heterogeneous, but closed, user group, additional sources of information need to be consulted to verify that a realm with its discovered server is actually an eduroam participant.

eduroam has chosen to deploy a separate PKI infrastructure which issues certificates only to authorised eduroam Identity Providers and eduroam Service Providers. Since certificates are needed for RADIUS/TLS anyway, this was a straightforward solution. The PKI fabric allows multiple CAs as trust roots (overseen by a Policy Management Authority), and requires that certificates which were issued to verified eduroam participants are marked with corresponding "X509v3 Policy OID" fields; eduroam RADIUS servers and clients need to verify the existence of these OIDs in the incoming certificates.

The policies and OIDs can be retrieved from the "eduPKI Trust Profile for eduroam Certificates" ([edupki]).

4.2.3. Operational Experience

The discovery model as described above is currently deployed in approx. 10 countries that participate in eduroam, making more than 100 realms discoverable via their NAPTR records. Experience has shown that the model works and scales as expected; the only drawback being that the additional burden of operating a PKI which is not local to the national eduroam administrators creates significant administrative complexities. Also, the presence of multiple CAs and regular updates of Certificate Revocation Lists makes the operation of RADIUS servers more complex.

4.2.4. Possible Alternatives

There are two alternatives to the above approach which are monitored by the eduroam community:

1. DNSSEC + DANE TLSA records
2. ABFAB Trust Router

For DNSSEC+DANE TLSA, its most promising plus is that the certificate data itself can be stored in the DNS - possibly obsoleting the PKI infrastructure *if* a new place for the server authorisation checks can be found. Its most significant downside is that the DANE specifications only include client-to-server certificate checks, while RADIUS/TLS requires also server-to-client verification.

For the ABFAB Trust Router, the most promising plus is that it would work without certificates altogether (by negotiating TLS-PSK keys ad-hoc). The current downside is that it is not formally specified and not as thoroughly understood as any of the other solutions.

5. Abuse prevention and incident handling

Since the eduroam service is a confederation of autonomous networks, there is little justification for transferring accounting information from the visited site to any other in general, or in particular to the home organization of the user. Accounting in eduroam is therefore considered to be a local matter of the visited site. The eduroam compliance statement ([eduroam-compliance]) in fact specifies that accounting traffic SHOULD NOT be forwarded.

The static routing infrastructure of eduroam acts as a filtering system blocking accounting traffic from misconfigured local RADIUS servers. Proxy servers are configured to terminate accounting request traffic by answering to Accounting-Requests with an Accounting-Response in order to prevent the retransmission of orphaned Accounting-Request messages.

Roaming creates accounting problems, as identified by [RFC4372] (Chargeable User Identity). Since the NAS can only see the (likely anonymous) outer identity of the user, it is impossible to correlate usage with a specific user (who may use multiple devices). A NAS that supports this can request the Chargeable-User-Identity and, if supplied by the authenticating RADIUS server in the Access-Accept message, add this value to corresponding Access-Request packets. While eduroam does not have any charging mechanisms, it may still be desirable to identify traffic originating from one particular user. One of the reasons is to prevent abuse of guest access by users living nearby university campuses. Chargeable User Identity (see below) supplies the perfect answer to this problem, however at the moment of writing, to our knowledge only one hardware vendor (Meru Networks) implements RFC4372 on their Access Points. For all other vendors, requesting the Chargeable-User-Identity attribute needs to happen on the RADIUS server to which the Access Point is connected to. Currently, the RADIUS servers FreeRADIUS and Radiator can be retrofitted with the ability to do this.

5.1. Incident Handling

10 years of experience with eduroam have not exposed any serious incident. This may be taken as evidence for proper security design as well as suggest that awareness of users that they are identifiable, acts as an effective deterrent. It could of course also mean that eduroam operations lack the proper tools or insight into the actual use and potential abuse of the service. In any case, many of the attack vectors that exist in open networks or networks where access control is based on shared secrets are not present, arguably leading to a much more secure system.

The European eduroam policy [eduroam-policy], as an example, describes incident scenarios and actions to be taken, in this document we present the relevant technical issues.

The first action in the case of an incident is to block the user's access to eduroam at the visited site. Since the roaming user's true identity is likely hidden behind an anonymous/fake outer identity, the visited site can only rely on the realm of the user. Without cooperation from the user's home institution, the SP's options are limited to blocking authentications from the entire realm, which may be considered as too harsh. On the other hand, the home institution has only the possibility of blocking the user's authentication entirely, thus blocking this user from accessing eduroam in all sites. With eduroam becoming more and more global it can be expected that differences of opinions in interpreting user's actions may arise between SPs and IdPs. It is obviously the right of an SP to provide guest access only under certain conditions. When these conditions are violated by the user, the network access may be blocked at the current site.

However there may be situations where such a restriction should only apply at a given SP and not eduroam as a whole. The initial implementation has been lacking a tool for an SP to make it's own decision or for an IdP to introduce a conditional rule applying only to a given SP. The introduction of support for Operator-Name and Chargeable-User-Identity (see below) to eduroam makes both of these scenarios possible.

5.2. Operator Name

The Operator-Name attribute is defined in [RFC5580] as a means of unique identification of the access site.

The Proxy infrastructure of eduroam makes it impossible for home sites to tell where their users roam to. While this may be seen as a positive aspect enhancing user's privacy, it also makes user support, roaming statistics and blocking offending user's access to eduroam significantly harder.

Sites participating in eduroam are encouraged to add the Operator-Name attribute using the REALM namespace, i.e. sending a realm name under control of the given site.

The introduction of Operator-Name in eduroam has identified one operational problem - the identifier 126 assigned to this attribute has been previously used by some vendors for their specific purposes and has been included in attribute dictionaries of several RADIUS server distributions. Since the syntax of this hijacked attribute had been set to Integer, this introduces a syntax clash with the the RFC definition (OctetString). Operational tests in eduroam have shown that servers using the Integer syntax for attribute 126 may either truncate the value to 4 octets or even drop the entire RADIUS packet (thus making authentication impossible). The eduroam monitoring and eduroam test tools try to locate problematic sites.

When a visited site sends its Operator-Name value, it creates a possibility for the home sites to set up conditional blocking rules, depriving certain users of access to selected sites. Such action will cause much less concern than blocking users from all of eduroam.

In eduroam the Operator Name is also used for the generation of Chargeable User Identity values.

The addition of Operator-Name is a straightforward configuration of the RADIUS server and may be easily introduced on a large scale.

5.3. Chargeable User Identity

The Chargeable-User-Identity (CUI) attribute is defined by RFC4372 [RFC4372] as an answer to accounting problems caused by the use of anonymous identity in some EAP methods. In eduroam the primary use of CUI is in incident handling, but it can also enhance local accounting.

The eduroam policy requires that a given user's CUI generated for requests originating from different sites should be different (to prevent collusion attacks). The eduroam policy thus mandates that a CUI request be accompanied by the Operator-Name attribute, which is used as one of the inputs for the CUI generation algorithm. The Operator-Name requirement is considered to be the "business requirement" described in Section 2.1 of RFC4372 [RFC4372] and hence conforms to the RFC.

When eduroam started considering using CUI, there were no NAS implementations, therefore the only solution was moving all CUI support to the RADIUS server.

CUI request generation requires only the addition of NUL CUI attributes to outgoing Access-Requests, however the real strength of CUI comes with accounting. Implementation of CUI based accounting in the server requires that the authentication and accounting RADIUS servers used directly by the NAS are actually the same or at least have access to a common source of information. Upon processing of an Access-Accept the authenticating RADIUS server must store the received CUI value together with the device's Calling-Station-Id in a temporary database. Upon receipt of an Accounting-Request, the server needs to update the packet with the CUI value read from the database.

A wide introduction of CUI support in eduroam will significantly simplify incident handling at visited sites. Introducing local, per-user access restriction will be possible. Visited sites will also be able to notify the home site about the introduction of such a restriction, pointing to the CUI value an thus making it possible for the home site to identify the user. When the user reports the problem at his home support, the reason will be already known.

6. Privacy Considerations

The eduroam architecture has been designed with protection of user credentials in mind as may be clear from the discussion above. However, operational experience has revealed some more subtle points with regards to privacy.

6.1. Collusion of Service Providers

If users use anonymous outer identities, Service Providers can not easily collute by linking outer identities to users that are visiting their campus. This poses however problems with remediation of abuse of misconfiguration. It is impossible to find the user that exhibits unwanted behaviour or whose system has been compromised.

For that reason the Chargeable-User-Identity has been introduced in eduroam, constructed so that only the IdP of the user can uniquely identify the user. In order to prevent collusion attacks that CUI is required to be unique per user per Service Provider.

6.2. Exposing user credentials

Through the use of EAP, user credentials are not visible to anyone but the IdP of the user. That is, if a sufficiently secure EAP-method is chosen.

There is one privacy sensitive user attribute that is necessarily exposed to third parties and that is the realm the user belongs to. Routing in eduroam is based on the realm part of the user identifier, so even though the outer identity in a tunneled EAP-method may be set to an anonymous identifier it MUST contain the realm of the user, and may thus lead to identifying the user. This is considered a reasonable trade-of between user privacy and usability.

6.3. Track location of users

Due to the fact that access requests (potentially) travel through a number of proxy RADIUS servers, the home IdP of the user typically can not tell where a user roams to.

The introduction of Operator-Name and dynamic lookups (i.e. direct connections between IdP and SP) however, give the home IdP insight into the location of the user.

7. Security Considerations

This section addresses only security considerations associated with the use of eduroam. For considerations relating to 802.1X, RADIUS and EAP in general, the reader is referred to the respective specification and to other literature.

7.1. Man in the middle and Tunneling Attacks

The security of user credentials in eduroam ultimately lies within the EAP server verification during the EAP conversation. Therefore, the eduroam policy mandates that only EAP types capable of mutual authentication are allowed in the infrastructure, and requires that Identity Providers publish all information that is required to uniquely identify the server (i.e. usually the EAP server's CA certificate and its Common Name or subjectAltName:dNSName).

While this in principle makes Man-in-the-middle attacks impossible, practice has shown that several attack vectors exist nonetheless. Most of these deficiencies are due to implementation shortcomings in EAP supplicants. Examples:

7.1.1. Verification of Server Name not supported

Some supplicants only allow to specify which CA issues the EAP server certificate; it's name is not checked. As a result, any entity who is able to get a server certificate from the same CA can create its own EAP server and trick the end user to submit his credentials to that fake server.

As a mitigation to that problem, eduroam Operations suggests the use of a private CA which exclusively issues certificates to the organisation's EAP servers. In that case, no other entity will get a certificate from the CA and the above supplicant shortcoming does not present a security threat any more.

7.1.2. Neither Specification of CA nor Server Name checks during bootstrap

Some supplicants allow for insecure bootstrapping in that they allow to simply select a network and accept the incoming server certificate, identified by its fingerprint. The certificate is then saved as trusted for later re-connection attempts. If users are near a fake hotspot during initial provisioning, they may be tricked to submit their credentials to a fake server; and furthermore will be branded to trust only that fake server in the future.

eduroam Identity Providers are advised to provide their users with complete documentation for setup of their supplicants without the shortcut of insecure bootstrapping. In addition, eduroam Operations has created a tool which makes correct, complete and secure settings on many supplicants: eduroam CAT ([eduroam-cat]).

7.1.3. User does not configure CA or Server Name checks

Unless automatic provisioning tools such as eduroam CAT are used, it is cumbersome for users to initially configure an EAP supplicant securely. User Inferfaces of supplicants often invite the users to take shortcuts ("Don't check server certificate") which are easier to setup or hide important security settings in badly accessible sub-menus. Such shortcuts or security parameter ommissions make the user subject to man-in-the-middle attacks.

eduroam Identity Providers are advised to educate their users regarding the necessary steps towards a secure setup. eduroam Research and Development is in touch with supplicant developers to improve their User Interfaces.

7.1.4. Tunneling authentication traffic to obfuscate user origin

There is no link between the EAP outer ("anonymous") identity and the EAP inner ("real") identity. In particular, they can both contain a realm name, and the realms need not be identical. It is possible to craft packets with an outer identity of user@RealmB, and an inner identity of user@realmA. With the eduroam request routing, a Service Provider would assume that the user is from realmB and send the request there. The server at realm B inspects the inner user name, and if proxying is not explicitly disabled for tunneled request content, may decide to send the tunneled EAP payload to realmA, where the user authenticates. A CUI value would likely be generated by the server at realmB, even though this is not its user.

Users can craft such packets to make their identification harder; usually, the eduroam SP would assume the troublesome user to originate from realmB and demand there that the user be blocked. The operator of realmB however has no control over the user, and can only trace back the user to his real origin if logging of proxied requests is also enabled for EAP tunnel data.

eduroam Identity Providers are advised to explicitly disable proxying on the parts of their RADIUS server configuration which processes EAP tunnel data.

7.2. Denial of Service Attacks

Since eduroam's roaming infrastructure is based on IP and RADIUS, it suffers from the usual DoS attack vectors that apply to these protocols.

The eduroam hotspots are susceptible to typical attacks on consumer edge networks, such as rogue RA, rogue DHCP servers, and others. Notably, eduroam hotspots are more robust against malign users' DHCP pool exhaustion than typical open or "captive portal" hotspots, because a DHCP address is only leased after a successful authentication, which reduces the pool of possible attackers to eduroam account holders (as opposed to the general public). Furthermore, attacks involving ARP spoofing or ARP flooding are also reduced to authenticated users, because an attacker needs to be in possession of a valid WPA2 session key to be able to send traffic on the network.

This section does not discuss standard threats to consumer edge networks and IP networks in general. The following sections describe attack vectors specific to eduroam.

7.2.1. Intentional DoS by malign individuals

The eduroam infrastructure is more robust against Distributed DoS attacks than typical services which are reachable on the internet because triggering authentication traffic can only be done when physically being in proximity of an eduroam hotspot (be it a wired IEEE 802.1X enabled socket or a Wi-Fi Access Point).

However, when being in the vicinity, it is easy to craft authentication attempts that traverse the entire international eduroam infrastructure; an attacker merely needs to choose a realm from another world region than his physical location to trigger Access-Requests which need to be processed by the SP, then SP-side national, then world region, then target world region, then target national, then target IdP server. So long as the realm actually exists, this will be followed by an entire EAP conversation on that path. Not having actual credentials, the request will ultimately be rejected; but it consumed processing power and bandwidth across the entire infrastructure, possibly affecting all international authentication traffic.

EAP is a lock-step protocol. A single attacker at an eduroam hotspot can only execute one EAP conversation after another, and is thus rate-limited by round-trip times of the RADIUS chain.

Currently eduroam processes several hundred thousands of successful international roaming authentications per day (and, incidentally, approximately 1.5 times as many Access-Rejects). With the requirement of physical proximity, and the rate-limiting induced by EAP's lock-step nature, it requires a significant amount of attackers and a time-coordinated attack to produce significant load. So far eduroam Operations has not yet observed critical load conditions which could reasonably be attributed to such an attack.

The introduction of dynamic discovery further eases this problem, as authentications will then not traverse all infrastructure servers, removing the world-region aggregation servers as obvious bottlenecks. Any attack would then be limited between an SP and IdP directly.

7.2.2. DoS as a side-effect of expired credentials

In eduroam Operations it is observed that a significant portion of (failed) eduroam authentications is due to user accounts which were once valid, but have in the meantime been de-provisioned (e.g. if a student has left the university after graduation). Configured eduroam accounts are often retained on the user devices, and when in the vicinity of an eduroam hotspot, the user device's operating system will attempt to connect to this network.

As operation of eduroam continues, the amount of devices with left-over configurations is growing, effectively creating a pool of devices which produce unwanted network traffic whenever they can.

Up until recently, this problem did not emerge with much prominence, because there is also a natural shrinking of that pool of devices due to users finally de-commissioning their old computing hardware.

As of recent, particularly smartphones are programmed to make use of cloud storage and online backup mechanisms which save most, or all, configuration details of the device with a third-party. When renewing their personal computing hardware, users can restore the old settings onto the new device. It has been observed that expired eduroam accounts can survive perpetually on user devices that way. If this trend continues, it can be pictured that an always-growing pool of devices will clog up eduroam infrastructure with doomed-to-fail authentication requests.

There is not currently a useful remedy to this problem, other than instructing users to manually delete their configuration in due time. Possible approaches to this problem are:

- Creating a culture of device provisioning where the provisioning profile contains a "ValidUntil" property, after which the configuration needs to be re-validated or disabled. This requires a data format for provisioning as well as implementation support.
- Improvements to supplicant software so that it maintains state over failed authentications. E.g. if a previously known-working configuration failed to authenticate consistently for 30 calendar days, it should be considered stale and be disabled.

8. IANA Considerations

There are no IANA Considerations

9. References

9.1. Normative References

[I-D.iab-privacy-considerations] Cooper, A., Tschofenig, H., Aboba, B., Peterson, J., Morris, J., Hansen, M. and R. Smith, "Privacy Considerations for Internet Protocols", Internet-Draft draft-iab-privacy-considerations-03, July 2012.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2865] Rigney, C., Willens, S., Rubens, A. and W. Simpson, "Remote Authentication Dial In User Service (RADIUS)", RFC 2865, June 2000.

[RFC2866] Rigney, C., "RADIUS Accounting", RFC 2866, June 2000.

[RFC3748] Aboba, B., Blunk, L., Vollbrecht, J., Carlson, J. and H. Levkowetz, "Extensible Authentication Protocol (EAP)", RFC 3748, June 2004.

[RFC4279] Eronen, P. and H. Tschofenig, "Pre-Shared Key Ciphersuites for Transport Layer Security (TLS)", RFC 4279, December 2005.

[RFC4372] Adrangi, F., Lior, A., Korhonen, J. and J. Loughney, "Chargeable User Identity", RFC 4372, January 2006.

[RFC5176] Chiba, M., Dommety, G., Eklund, M., Mitton, D. and B. Aboba, "Dynamic Authorization Extensions to Remote Authentication Dial In User Service (RADIUS)", RFC 5176, January 2008.

[RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, August 2008.

[RFC5247] Aboba, B., Simon, D. and P. Eronen, "Extensible Authentication Protocol (EAP) Key Management Framework", RFC 5247, August 2008.

[RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R. and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, May 2008.

[RFC5580] Tschofenig, H., Adrangi, F., Jones, M., Lior, A. and B. Aboba, "Carrying Location Objects in RADIUS and Diameter", RFC 5580, August 2009.

[RFC5997] DeKok, A., "Use of Status-Server Packets in the Remote Authentication Dial In User Service (RADIUS) Protocol", RFC 5997, August 2010.

[RFC6066] Eastlake, D., "Transport Layer Security (TLS) Extensions: Extension Definitions", RFC 6066, January 2011.

[RFC6613] DeKok, A., "RADIUS over TCP", RFC 6613, May 2012.

[RFC6614] Winter, S., McCauley, M., Venaas, S. and K. Wierenga, "Transport Layer Security (TLS) Encryption for RADIUS", RFC 6614, May 2012.

9.2. Informative References

[I-D.ietf-abfab-arch] Howlett, J., Hartman, S., Tschofenig, H., Lear, E. and J. Schaad, "Application Bridging for Federated Access Beyond Web (ABFAB) Architecture", Internet-Draft draft-ietf-abfab-arch-06, April 2013.

[I-D.ietf-radext-dtls] DeKok, A., "DTLS as a Transport Layer for RADIUS", Internet-Draft draft-ietf-radext-dtls-05, April 2013.

[I-D.ietf-radext-dynamic-discovery] Winter, S. and M. McCauley, "NAI-based Dynamic Peer Discovery for RADIUS/TLS and RADIUS/DTLS", Internet-Draft draft-ietf-radext-dynamic-discovery-07, July 2013.

[MD5-attacks] Black, J., Cochran, M. and T. Highland, "A Study of the MD5 Attacks: Insights and Improvements", October 2006.

[RFC3539] Aboba, B. and J. Wood, "Authentication, Authorization and Accounting (AAA) Transport Profile", RFC 3539, June 2003.

[RFC3588] Calhoun, P., Loughney, J., Guttman, E., Zorn, G. and J. Arkko, "Diameter Base Protocol", RFC 3588, September 2003.

[RFC4107] Bellovin, S. and R. Housley, "Guidelines for Cryptographic Key Management", BCP 107, RFC 4107, June 2005.

[RFC4346] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.1", RFC 4346, April 2006.

[RFC4953] Touch, J., "Defending TCP Against Spoofing Attacks", RFC 4953, July 2007.

[RFC6125] Saint-Andre, P. and J. Hodges, "Representation and Verification of Domain-Based Application Service Identity within Internet Public Key Infrastructure Using X.509 (PKIX) Certificates in the Context of Transport Layer Security (TLS)", RFC 6125, March 2011.

[RFC6421] Nelson, D., "Crypto-Agility Requirements for Remote Authentication Dial-In User Service (RADIUS)", RFC 6421, November 2011.

[dead-realm] Tomášek, J., "Dead-realm marking feature for Radiator RADIUS servers", 2006.

[dot1X-standard] IEEE, "IEEE std 802.1X-2010", February 2010.

[edupki] Delivery of Advanced Network Technology to Europe , "eduPKI", 2012.

[eduroam-cat] Delivery of Advanced Network Technology to Europe , "European CAT", 2012.

[eduroam-compliance] Trans-European Research and Education Networking Association , "eduroam compliance statement", 2011.

[eduroam-homepage] Trans-European Research and Education Networking Association , "eduroam Homepage", 2007.

[eduroam-policy] Trans-European Research and Education Networking Association , "European eduroam policy", 2011.

[eduroam-start] Wierenga, K., "Initial proposal for what is now called eduroam", 2002.

[geant2] Delivery of Advanced Network Technology to Europe , "European Commission Information Society and Media: GEANT2", 2008.

[radsec-whitepaper] Open System Consultants, "RadSec - a secure, reliable RADIUS Protocol", May 2005.

[radsecproxy-impl] Venaas, S., "radsecproxy Project Homepage", 2007.

[terena] TERENA, "Trans-European Research and Education

Networking Association", 2008.

Appendix A. Acknowledgments

The authors would like to thank the participants in the TERENA Task Force on Mobility and Network Middleware as well as the Geant project for their reviews and contributions. Special thanks go to Jim Schaad for doing an excellent review of the first version.

The eduroam trademark is registered by TERENA.

Appendix B. Changes

This section to be removed prior to publication.

- 00 Initial Revision
- 01 Added Dynamic Discovery, addressed review comments

Authors' Addresses

Klaas Wierenga
Cisco Systems
Haarlerbergweg 13-19
Amsterdam, 1101 CH
The Netherlands
Phone: +31 20 357 1752
EMail: klaas@cisco.com

Stefan Winter
Fondation RESTENA
6, rue Richard Coudenhove-Kalergi
Luxembourg, 1359
Luxembourg
Phone: +352 424409 1
Fax: +352 422473
EMail: stefan.winter@restena.lu
URI: http://www.restena.lu.

Tomasz Wolniewicz
Nicolaus Copernicus University
pl. Rapackiego 1
Torun,
Poland
Phone: +48-56-611-2750
Fax: +48-56-622-1850
EMail: twoln@umk.pl
URI: http://www.home.umk.pl/~twoln/