Network Working Group	K. Wierenga
Internet-Draft	Cisco Systems
Intended status: Informational	S. Winter
Expires: April 18, 2013	RESTENA
	T. Wolniewicz
	Nicolaus Copernicus University
	October 15, 2012

TOC

The eduroam architecture for network roaming
draft-wierenga-ietf-eduroam-00.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

Copyright Notice

Table of Contents

TOC

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 in 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 for example RADIUS over TCP **[RFC6613]** and RADIUS with TLS **[RFC6614]** and that promoted alternative use 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.

TOC

1.1. Terminology

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

TOC

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].

1.3. Design Goals

TOC

The guiding design considerations of eduroam were as follows:

- Unique identification of users at the edge of the network

In order to determine whether a user has the right to use the network resources the user needs to be identified. Furthermore, in case of abuse of the resources, there is a requirement to be able to identify the user. Lastly, it should not be possible for a person to impersonate someone else or take over their identity.

- 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 that participate in the service.

- 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 to go through an administrative process when a user visits another organization.

- Easy to install and use

It should not be very complicated to participate in the roaming infrastructure as that may inhibit wide scale adoption. In particular, there should be no or easy client installation and one-off configuration.

- Secure and privacy preserving

Whereas it is impossible to create a secure system in the absolute sense, it is important to have a system that strikes a good balance between ease of use and security. One important design criteria has been that there needs to be a security association between the end-user and their home organization, so no exposure of credentials to a third party. In particular, it should be possible for participating organizations to set their own requirements for the quality of authentication of users without the need for the infrastructure as a whole to implement the same standard.

- 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.

These considerations have led to an architecture 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

TOC

Federations, like eduroam, implement essentially two types of trust. 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 establishment of 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 in turn provide the transitive trust that makes the SP allow the use of its network resources.

TOC

TOC

2.1. Authentication

Authentication in eduroam is achieved by using a combination of IEEE 802.1X **[dot1X-standard]** and EAP **[RFC4372]**.

TOC

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).

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. To protect user data confidentiality eduroam SPs IEEE 802.11 wireless networks MUST support WPA2+AES, and MAY additionally support WPA/TKIP as a courtesy to users of legacy hardware.

TOC

2.1.2. EAP

The use of the Extensible Authentication Protocol (EAP) **[RFC4372]** serves 2 purposes. In the first place a proper 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 the policy with regards to security properties.

```
                         +------+
                        /        \
                       /          \
                      /            \
                     /              \
    ,----------\    |                |   ,---------\
    |    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 privacy, 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.

2.2. Federation Trust Fabric

TOC

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 romaing consortium.

2.2.1. RADIUS

TOC

The eduroam trust fabric is based on a proxy hierarchy of RADIUS servers, loosely based on the DNS hierarchy. That is, the organizational RADIUS servers agree on a shared secret with the national servers and the national servers agree 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.

```
                               +-------+
                               |       |
                               |   .   |
                               |       |
                               +---+---+
                                 / | \
               +----------------/  |  \---------------------+
               |                   |                        |
               |                   |                        |
               |                   |                        |
            +--+---+            +--+--+                +----+---+
            |      |            |     |                |        |
            | .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, 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.

TOC

3. Issues with initial Trust Fabric

While the hierarchical RADIUS architecture 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 performance and operations bottlenecks on the aggregation proxies. The following sections describe some of the shortcomings, and the resulting conclusions.

TOC

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 deducted 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 deducted states: since the proxy chain is longer than 1 hop, a failure further down 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 [http://wiki.eduroam.cz/dead-realm/docs/dead-realm.html] 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.

TOC

3.2. No error condition signalling

The RADIUS protocol lacks signalling of error conditions, and the IEEE 802.1X protocol does not allows 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 reactively if a downstream server doesn't reply. The active link-state monitoring of the TCP connection with e.g. RADIUS/TLS 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.

TOC

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 ccTLDs, which map into national eduroam federations, it is not true for realms residing in 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 exceptional 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 occupied only approx. 50 routing entries in total (and has 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 needed to be applied manually to RADIUS server configuration files. The frequency of the required updates made this approach fragile and error-prone as the number of entries grew.

TOC

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 tiny 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 EAP peer -> EAP server.

The operational experience regarding EAP-TLS leads to the following RECOMMENDATION: EAP supplicants should either make the maximum EAP chunk size configurable OR use cautious values regarding the EAP chunk size (e.g. max. 1200 Bytes per chunk, even if the layer 2 medium provides foresaw more space).

Both of the previously mentioned sources of errors (packet loss, fragment discard) are hard to diagnose and can lead to significant user frustration for the affected users.

TOC

3.5. Insufficient payload encryption

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 computing 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. Worse, 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 configuration 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 configuration 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 meta data format which could be used across manufacturers; ideally leading to a situation where any IEEE 802.1X network end-user merely needs to apply this configuration file to configure any of his 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.

Note: This operational experience of eduroam could be taken as a guideline for supplicant implementers to leave sufficient space in transmitted packets.

TOC

4. Enhanced Architecture

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 is in fact gradually replacing those parts of it where problems may arise.

4.1. Federation Trust Fabric

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.

4.1.1. 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.

4.1.2. Dynamic Discovery

XXX

TOC

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 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 Chargeable User Identity can request additional information - Chargeable-User-Identity and if this is supplied by the authenticating RADIS server in the Access-Accept message, this value will then be added 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 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.

TOC

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 and awareness of users that they are identifiable, acts as an effective deterrent.

For example the European eduroam policy **[eduroam-policy]** 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 visited institution'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 blacking the user's authentication entirely, thus blocking this user from accessing eduroam in all sites. This may also be seen as a too harsh an action, especially since visited and home sites could differ in interpreting the user's actions. Introduction of support for Operator-Name and Chargeable-User-Identity (see below) to eduroam can significantly improve the situation.

TOC

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 then 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.

TOC

5.3. Chargeable User Identifier

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 form 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.

TOC

6. Privacy Considerations

XXX

TOC

6.1. Collusion of RPs

XXX

TOC

6.2. Exposing user credentials

XXX

TOC

6.3. Track location of users

XXX

TOC

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.

TOC

7.1. Man in the middle and Tunneling Attacks

XXX

TOC

7.2. Denial of Service Attacks

XXX

TOC

8. IANA Considerations

There are no IANA Considerations

TOC

9. References

TOC

9.1. Normative References

[I-D.hansen-privacy-terminology] Hansen, M., Tschofenig, H., and R. Smith, “**Privacy Terminology**,” draft-hansen-privacy-terminology-03 (work in progress), October 2011 (**TXT**).

[RFC2119] **Bradner, S.**, “**Key words for use in RFCs to Indicate Requirement Levels**,” BCP 14, RFC 2119, March 1997 (**TXT**, **HTML**, **XML**).

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

[RFC2866] Rigney, C., “**RADIUS Accounting**,” RFC 2866, June 2000 (**TXT**).

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

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

[RFC4372] Adrangi, F., Lior, A., Korhonen, J., and J. Loughney, “**Chargeable User Identity**,” RFC 4372, January 2006 (**TXT**).

[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 (TXT).

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

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

[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 (TXT).

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

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

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

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

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

TOC

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," draft-ietf-abfab-arch-03 (work in progress), July 2012 (TXT).

[I-D.ietf-radext-dtls] DeKok, A., "DTLS as a Transport Layer for RADIUS," draft-ietf-radext-dtls-02 (work in progress), July 2012 (TXT).

[I-D.ietf-radext-dynamic-discovery] Winter, S. and M. McCauley, "NAI-based Dynamic Peer Discovery for RADIUS/TLS and RADIUS/DTLS," draft-ietf-radext-dynamic-discovery-04 (work in progress), June 2012 (TXT).

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

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

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

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

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

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

[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 (TXT).

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

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

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

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

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

[eduroam-start] Wierenga, K., "Initial proposal for (now) eduroam," 2002 (PDF).

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

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

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

[terena] TERENA, "Trans-European Research and Education Networking Association," 2008 (TXT).

TOC

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.

The eduroam trademark is registered by TERENA.

Appendix B. Changes

TOC

This section to be removed prior to publication.

- 00 Initial Revision.

Authors' Addresses

TOC

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/