Mobility Task Force

Deliverable D

Inventory of 802.1X-based solutions for inter-NRENs roaming

Version 1.2

Authors: Erik Dobbelsteijn erik.dobbelsteijn@surfnet.nl

Contributions: Klaas Wierenga (SURFnet bv) klaas.wierenga@surfnet.nl, Paul Dekkers (SURFnet bv) paul.dekkers@surfnet.nl, Henny Bekker (SURFnet bv) henny.bekker@surfnet.nl, James Sankar (UKERNA) j.sankar@ukerna.ac.uk, Tim Chown (University of Southampton) tjc@ecs.soton.ac.uk, Sami Keski-Kasari Tampere (University of Technology, Finland) <samikk@cs.tut.fi>

Abstract

This document describes how 802.1X can be used to enable roaming between NRENs. See 'Glossary[1]' for terms and definitions.

1. Introduction

When a network is IEEE 802.1X enabled, a user initially has to perform a couple of actions to be able to logon to the network. The OS should support 802.1X or software has to be installed that adds this functionality, and the user account has to be set up.
After the initial setup, the user can move freely from one network to another, while his terminal logs onto the 802.1X enabled networks without additional efforts. The networks that the user can use can be either fixed (LAN) networks or Wireless LAN networks.
The IEEE 802.1X standard for port-based authentication is a layer 2 solution between client and the Access Control Device (either a wireless Access Point or a switch). In the 802.1X framework authentication information is carried over the Extensible Authentication Protocol (EAP) that enables the use of various authentication methods. Access control devices communicate with a RADIUS backend for user verification, which is both secure and scalable.
After authentication, the communication between client and Wireless Access Point is encrypted using dynamic keys.

2. Architecture

The 802.1X framework adds functionality to existing components in a network. Therefore, no additional components are necessary.
In a fixed network, the terminal (a PC or laptop for instance) has to have a network card (NIC), and the operating system should have so called 802.1X supplicant functionality on board.
The port to which the terminal will connect resides on a switch that is 802.1X enabled. The switch is called the Authenticator within the framework. Based on 802.1X commands, the switch can

[1] The glossary can be found as appendix in the deliverable B

open and close a connection on the port. The third component of the architecture is the authentication server. In general, a switch will interrogate a RADIUS server to check if the user is allowed to use a port, and to which VLAN the traffic must go.
When 802.1X is applied to a wireless network, a wireless access control device replaces the switch as the Authenticator. It is not relevant which wireless transport protocol (802.11b or upcoming protocols like 802.11g) is used.

The complete infrastructure looks as follows:

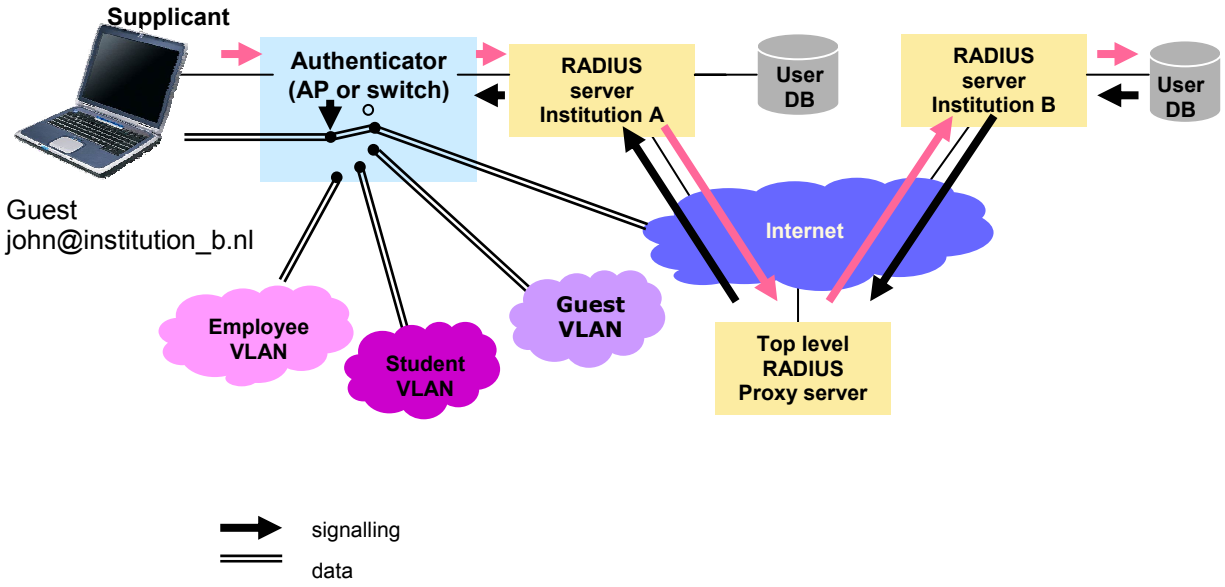

Figure 1: Authentication infrastructure for cross-domain authentication using 802.1X

When a user connects to the network he provides his credentials to the authenticator (the access control device) that verifies this using the RADIUS backend. The credentials should always include a username and a realm which results in a credential that looks like an e-mailaddress (user@realm.topleveldomain).
If a visitor uses the network, the local RADIUS server will notice that the realm of the user is not the realm that it serves itself. That is where the RADIUS proxying mechanism comes in and makes sure that the EAP encapsulated credentials get transported towards the home RADIUS server. In fact, the RADIUS server only has to forward the request to a higher-level RADIUS proxy server. This proxy server knows all other RADIUS servers in the roaming constellation and forwards the request to the server that it knows can handle the realm. The home RADIUS server is installed at the home network of the visitor, either in the same country or abroad, where the user gets authenticated against a local user database. The local RADIUS server only has to know to which proxy unknown user requests should be sent. When a new network enters this roaming agreement, only the proxy has to be updated.

Figure 2 (below) shows the protocol stack of the 802.1X framework. In the 802.1X framework authentication information is carried over the Extensible Authentication Protocol (EAP, RFC 2284), a protocol that enables the use any authentication method, like username/password,

certificates, OTP (One Time Password, f.i. via SMS) or credentials on a mobile operators' SIM-card. These mechanisms are implemented in the EAP types MD5, TLS, TTLS, MS-CHAPv2, PEAP, Mob@c and EAP-SIM.
Both the supplicant and the home RADIUS server should use the same EAP type. The Access control device, switch or proxying RADIUS servers do not have to be aware of the EAP type.
Currently, TLS (Transport Layer Security), TTLS (Tunneled Transport Layer Security) and PEAP (Protected EAP) are the most serious candidates for immediate implementation. Additional tests were done with authentication based on one time passwords sent by SMS.
TLS, TTLS and PEAP set up a TLS connection between client and access control device based on a RADIUS server certificate. This mutual authentication mechanism can prevent Man in the Middle attacks. TLS then uses a client certificate to authenticate the user, while TTLS is generally used to transport username/password. Since both TTLS and PEAP are tunneling protocols, any other protocol can be used on top of them. MOBAC is an example of this, implementing One Time Password via SMS.

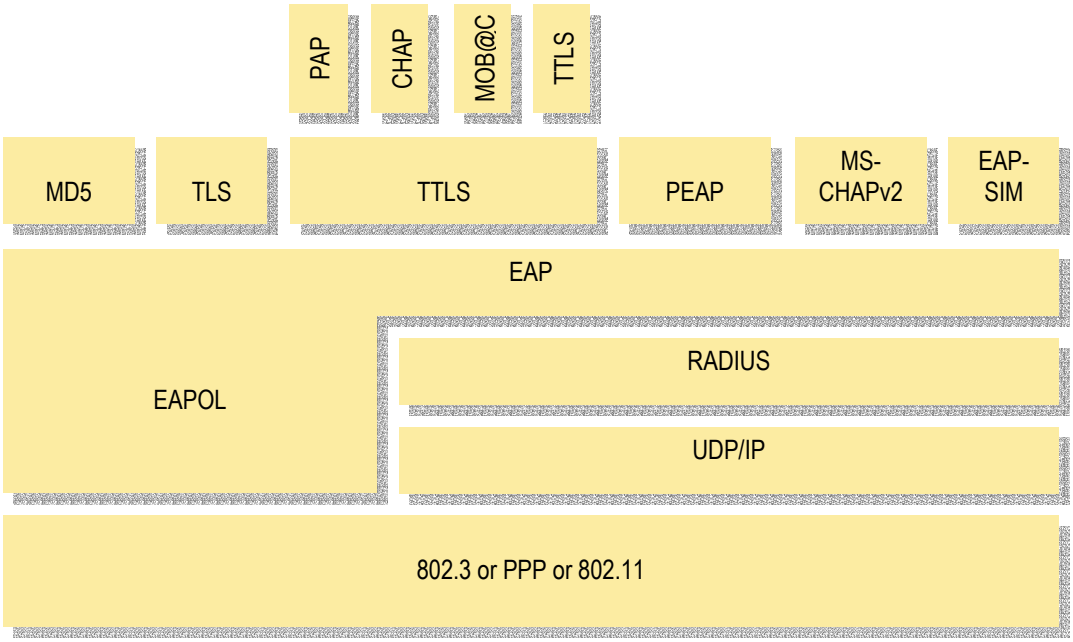

Figure 2: EAP can support various forms of authentication mechanisms

If the user is properly verified against the home authentication backend (which can be LDAP, for instance) he will be authenticated and the home RADIUS server passes an acknowledgement to the access control device. When a user is in his home network, the RADIUS server can tell the Authenticator in which VLAN the users' traffic must reside. The access control device then passes the user traffic onto this VLAN until de-authentication. VLAN switching is based on the 802.1Q standard. A visitor will be put into a guest-VLAN determined by the visitor network RADIUS server.
In this stage of the process, Ethernet connectivity is provided, after which the usual mechanisms for obtaining IP connectivity can play their role, like providing the client with an IP address through DHCP. In fact, anything is possible on layer 3 after the authentication process: not only the IP protocol but any layer 3 protocol can be transported (IPv6, IPSEC, IPX, PPPoE etcetera)

and any layer 3 mechanism (VPN, Multicast, NAT etcetera) finds a transparent layer two transportation layer.

When the user pulls out the cable or leaves the coverage area of a wireless access control device, the access control device detects the disruption of the connection and the port will be closed. More and more supplicants also have a built-in possibility to gracefully disconnect from a network, which enables them to reconnect using different credentials to access other VLANs.

3. Existing implementations

In the first quarter of 2003, two institutions in the Netherlands are using 802.1X with TTLS and PAP as the authentication mechanism for their wireless networks. Both institutions installed Cisco AP-1200 Access Points and RADIATOR 3.5 as the RADIUS server. The functionality will be extended to the fixed networks, on both Cisco as well as HP switches.
In the second quarter of 2003, two other institutions will implement respectively TLS and PEAP. Here, also Cisco AP-350 Access Points are used. Another institution intends to use 802.1X for both its fixed and wireless networks. When 802.1X is successfully implemented at a couple of student dormitories, two other institutions plan on implementing it for their wireless networks as well.

SURFnet provides the RADIUS proxy infrastructure for this roaming platform, based on RADIATOR 3.5. An institution can connect to it by creating a RADIUS relationship (exchange server names and shared secrets) with SURFnet and provision a VLAN for visitors that directly connects to SURFnet.

SURFnet furthermore aims at extending this functionality towards public hotspots that are under construction by Wireless ISPs in the Netherlands. By connecting the RADIUS server of the WISP to the SURFnet RADIUS server, users of the roaming platform mentioned earlier can instantly make use of the public hotspots as well.

4. Scalability

As mentioned before, the local RADIUS server only has to know to which proxy unknown user requests should be sent. When a new network enters this roaming agreement, only the proxy has to be updated.
To extend this roaming infrastructure to a European scale, a RADIUS proxy on an international level is the only component that has to be added, as depicted in figure 3.

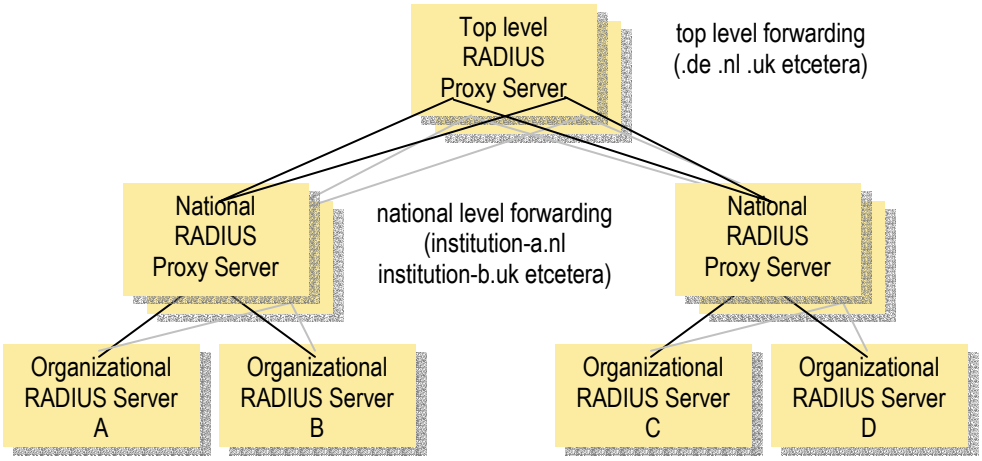

Figure 3: International roaming architecture

When a new institution enters the constellation, its realm only has to be entered into the National RADIUS Proxy Server, not into the servers of other institutions. The same holds for the addition of a group of institutions in a country that enters the constellation: the Top level RADIUS Proxy Server must be updated with the new high-level domain (for instance ".nl"), after which the forwarding mechanism works for each institution in the constellation. It is always possible to realize bilateral relationships between servers that exchange much traffic, or traffic that is only relevant locally.
The use of RADIUS also makes it easy to connect the existing roaming infrastructure to a mobile operators' network (WIFI, GPRS or UMTS).
The RADIUS infrastructure as it is presented here can introduce loops in the message flow, which can lead to failing RADIUS servers. To prevent this, each RADIUS server can be obliged not to forward messages destined for the realm that it handles itself. Also, the proxy can filter these events, and observe hop counts in messages.
Resilience is possible on all levels in the architecture. Access control devices can be installed in pairs, though this is not usually done because of the high costs. Furthermore, each Access control device can be configured to interrogate two (or more) RADIUS servers. When one RADIUS server fails, the other can take over. The same mechanism can be used between RADIUS servers in the proxy infrastructure.
Since the average RADIUS server software does not consume much hardware resources, an average contemporary PC could already serve dozens of authentication requests or even hundreds of forwarding requests per second. Authentication is only needed in the beginning of a user session and when a user roams between access control devices, therefore a RADIUS server on a national proxy level can serve potentially thousands of user sessions simultaneously.

Scalability in terms of throughput performance is implicitly achieved by the fact that each access control device handles encryption of data on layer two at wirespeed.

5. Security

As long as a strong EAP capable protocol like TLS is used, 802.1X provides a framework that gives a sufficient level of security for the intended purpose, i.e. access control to the network. Tunneling protocols such as PEAP and TTLS can be configured to prevent currently known types of Man in the Middle attacks.
For data integrity or privacy issues a number of wireless security extensions (WPA, TKIP, 802.11i etc.) have been proposed, that also build on the 802.1X framework. However, the currently usable mechanism of refreshing WEP keys delivers a very high level of encryption when the keys are refreshed regularly (typically every 20 minutes or less when using 64 bit keys). It is recommended to use 64 bit keys, because of backward compatibility with old network adapters.
Security in the RADIUS infrastructure is provided by the use of shared keys between RADIUS servers and by installing these servers in carefully designed secure parts of the network. Every RADIUS peering must have its own shared key. However, some messages might be altered along the way, so in addition the paths between RADIUS servers can be protected by setting up IPSEC tunnels. This has been successfully done both in at least the Netherlands and Finland.
Here is a short overview of security and abuse issues with corresponding mechanisms provided by 802.1X to tackle them:

Identity-takeover

It is hard to detect abuse in any authentication process when an abuser has the set of credentials of someone else. RADIUS enables detailed logging of user sessions, so complaints of a victim in whose name abuse was conducted can be related to the actual sessions that the abuser has initiated.

Loss of credentials

As soon as a user reports its loss of credentials to the help desk, the help desk can disable the account (in either the user database or the organizational RADIUS server). No abuse is possible from that moment on.

Bandwidth abuse

Detection and prevention of abuse of bandwidth on layer 2 is a problem in any network. Action can be taken to shape the traffic based on VLAN parameters or limit up/downlink capacity in VPN concentrators or Web-gateways, but this does not prevent users from flooding the air with a big bunch of packets in case of a wireless network. Using 802.1X, the sender can be backtracked in any case.
On Layer 1, the network administrator is helpless. Jamming the 2.4 GHz with a microwave simply blows everyone's data out.

Content abuse

Any abuse can be backtracked to a user account at the moment that guest use logging is related to IP-address assignment in the visitor network. This requires logging of the relationship between the user ID and the assigned IP address for a certain session.

Any suspicious accounts or even entire realms can be blocked on any level in the architecture, thus preventing the suspicious user or institution to log on to the network at all.

6. Clients

For username/password authentication, EAP-TTLS is tested extensively. It is easy to setup because it doesn't require a PKI with end-user certificates. Only the RADIUS server has to have a certificate. When a PKI is in place, EAP-TLS can be used as a strong way of authenticating.

In order for the client (supplicant in 802.1X terminology) to use 802.1X based authentication, either the client OS needs to support EAP and the required authentication method natively or a piece of software has to be installed.
Currently the MS Windows XP and Windows 2000 operating system support EAP and TLS. For earlier windows versions EAP support is announced. SURFnet distributes a TTLS module for these platforms.
Apart from this module, commercial clients exist that include EAP with TTLS and TLS. Both Funk and Meetinghouse clients have been tested. For Apple systems commercial clients exist and for various Unix flavors both a public domain implementation (open1X) as well as a commercial product is available. The number of implementations grows rapidly. Including them all in this document would make this information very static. An up-to-date overview is available at http://www.surfnet.nl/innovatie/wlan.

7. Access control devices

Most modern Access control devices support 802.1X authentication or have announced this. In the pilot products from Cisco (350 and 1200) and Orinoco (AP2000) have been tested successfully. Manufacturers currently handle the mapping of SSIDs to VLANs somewhat differently. Cisco currently supports the most VLANs on one Access control device, while it only broadcasts one SSID. The Orinoco product can broadcast two SSIDs, each mapped onto a corresponding VLAN.

Switches

VLAN assignment on HP, 3COM and Cisco switches based on the provided credentials was successfully demonstrated.

8. RADIUS servers

The home RADIUS server must support the requested EAP type. The intermediate RADIUS proxying servers must be able to forward EAP messages.
Radiator is tested in many scenarios. Other RADIUS servers that support TTLS and TLS include Meetinghouse, FUNK Steel-Belted and FreeRADIUS (the latter only supports TLS).
Accounting is a built-in feature in RADIUS that enables logging of authentication requests. Combining these logs with the registration of IP-address assignment, it is easily possible to track malicious access attempts or network abuse. The accounting feature may become especially relevant when connecting commercial access providers to the roaming platform. Accounting messages can easily be forwarded over the same proxy infrastructure.

9. Usability

One of the design criteria was the ease of use for the end-users. The pilot has shown that, once the user has installed the 802.1X client, use of the wireless network is handled seamlessly and transparently. In fact, it was found that a user would appreciate to get stronger visual clues as to the network he is entering and the level of security of the connection.

Some issues arise when combining 802.1X with configuring multiple networks. In Windows XP, the integration with the Windows domain logon is premature and incomplete.
The only disadvantage of 802.1X is the relative novelty and the fact that (currently) client software is necessary on most platforms. It is very likely that 802.1X functionality will be incorporated further into operating systems, thus becoming as common as DHCP.
The decision to use TTLS on the short term is a result of the higher availability of TTLS supplicants than PEAP supplicants. If PEAP gains momentum, it is easy to migrate from TTLS to PEAP.
Much work is done on these issues, and the usability is improved especially in the area of integrating 802.1X with other authentication processes (Windows domain logon, multiple network profiles).

10. Interoperability

The 802.1X mechanism provides network access on layer 2 of the OSI model. Open Ethernet and IP connectivity will only come up after successful authentication. This ensures compatibility with all IP-enabled applications. It is transparently possible to set up a VPN connection towards a concentrator that gives access to protected resources, or logon to a web-based gateway. Note that a user can impossibly logon to a foreign 802.1X network if he has no supplicant software and is provided with valid credentials. To make the network friendlier to non-802.1X enabled visitors, a default VLAN for these visitors with restricted connectivity can be offered, on which a temporary logon page or supplicant download facilities are provided.

The 802.1X framework is fully standardized, as well as EAP and EAP over LAN (EAPOL). The same holds for EAP-MD5 and EAP-TLS. Other subcomponents are in some cases still in development. TTLS and PEAP are both in draft within the IETF standardization process, as well as EAP-SIM. TTLS is available in many software implementations, PEAP in two (and expanding).
The use of WEP-keys within 802.1X is also standardized. First WPA and TKIP will follow up WEP. WPA is standardized but not widely available yet. Eventually, 802.1X including TKIP and AES will form 802.11i, a dedicated standard for security on WLAN.

11. Implementation guidelines

To give a good indication of the effort that is necessary to setup a proxy infrastructure, the configuration items of each element in the architecture is depicted below.

Organizational RADIUS Proxy Servers (ORPS) must:

- handle requests for its own realm
- forward requests for other realms to the National RADIUS Proxy Server
- all RADIUS attributes must be forwarded transparently to ensure EAP-transparency, but VLAN-assignment must be stripped preferably
- accept requests coming from National RADIUS Proxy Servers. Therefore, the IP addresses of the RADIUS servers must be exchanged and a RADIUS Secret must be determined to be used between each Organizational RADIUS Proxy Server and the National Top level RADIUS Proxy Server (one for each direction). The port number will be 1812
- forward accounting messages transparently on port 1813
- prevent looping by not forwarding requests to the server where they came from
- be implemented in pairs (a primary and a secondary), each Organizational RADIUS Proxy Server sending requests to a secondary server when the primary is down. After a timeout, it should try to reach the primary again. Timeouts and fall-back rules must be in

line with the optimal performance parameters of the underlying access infrastructure (the Access Control Device: Access Points and switches)
- log at least time, date, username+realm, accept/deny of each request (which can be related to IP address that was assigned to the user after logon)
- (Optional) the communication with a National RADIUS Proxy Server can be encrypted with SSL or IPSEC for additional security
- may strip optional RADIUS attributes of incoming Accept-Accept messages that are only relevant in the context of the home RADIUS domain of the visitor (f.i. VLAN assignment)
- a test account within the realm of the ORPS is necessary for testing purposes

The EAP handling within the organizational RADIUS infrastructure can be carried out by the same RADIUS servers that may be in use for other purposes (f.i. dial-in services). The guidelines as described here, are only relevant for the EAP-handling and forwarding.
It is very well possible to directly connect certain Organizational RADIUS Proxy servers when they are already closely related and they are likely to exchange a lot of requests. This can be the case when a couple of organizations reside on one campus, with many of their employees visiting each others buildings.

The National RADIUS Proxy Servers (NRPS) must:
- forward requests based on second-level realm (utwente.nl, ukerna.ac.uk etcetera)
- all RADIUS attributes must be forwarded transparently to ensure EAP-transparency
- accept requests coming from trusted top level RADIUS Proxy Servers and Organizational RADIUS Proxy Servers. Therefore, the IP addresses of the RADIUS servers must be exchanged and a RADIUS Secret must be determined to be used between each National RADIUS Proxy Server and the European Top level RADIUS Proxy Server (one for each direction). The port number will be 1812
- forward accounting messages transparently on port 1813
- prevent looping by not forwarding requests to the server where they came from
- be implemented in pairs (a primary and a secondary), each Organizational RADIUS Proxy Server sending requests to a secondary server when the primary is down. Timeouts, retries and fall-back to the primary when it is up again must be tuned to achieve optimal performance in combination with the values of these parameters at organizational level.
- log at least time, date, username+realm, accept/deny of each request
- (Optional) the communication to and from the National level RADIUS Proxy Server can be encrypted with SSL or IPSEC for additional security
- a test account within the realm of the NRPS is necessary for testing purposes

A choice can be made to let the National RADIUS Proxy Server handle subrealms as well, for instance the .uk NRPS can also process ac.uk and ed.uk subrealms. It is also possible to insert any number of sublevels with corresponding RADIUS servers for that level in a country (for instance state-level servers or even regional level servers).
It is very well possible that parallel proxy-infrastructures will arise for the same national level domain, for instance a commercial RADIUS clearing house. The exchange of subrealms that each proxy handles can be done in an administrative process.

The Top level RADIUS Proxy Server (TRPS) must:
- forward requests based on top-level realm (.nl, .ch, ac.uk, .be etcetera)
- all RADIUS attributes must be forwarded transparently to ensure EAP-transparency
- accept requests coming from National RADIUS Proxy Servers Therefore, the IP addresses of the RADIUS servers must be exchanged and a RADIUS Secret must be

determined to be used between each National RADIUS Proxy Server and the (European) Top level RADIUS Proxy Server (one for each direction). The port number will be 1812
- forward accounting messages transparently on port 1813
- prevent looping by not forwarding requests to the server where they came from
- be implemented in pairs (a primary and a secondary), each National RADIUS Proxy Server sending requests to a secondary server when the primary is down. Timeouts and retries must be tuned.
- log at least time, date, username+realm, accept/deny of each request
- (Optional) The communication between a National RADIUS Proxy Server and the European Top level RADIUS Proxy Server can be encrypted with SSL or IPSEC for additional security

12. Conclusion

The solution was tested in various pilots and proved to be both scalable and seamless. With the proper EAP authentication protocol the solution is also secure for the intended purpose. SURFnet will require use of 802.1X authentication in the nation-wide wireless test bed that will be established in the Freeband project, thus stimulating the use of this technology instead of less secure (web-based) or less scalable (VPN) solutions.
Apart from providing institutions with a sufficiently secure and easy to deploy solution the large added benefit of this solution is the ease in deploying a nationally and even internationally solution for inter-institutional roaming.

More information on implementation of this framework can be found at:
http://www.surfnet.nl/innovatie/wlan