WLAN Network Infrastructure

Best Practice Document

Produced by Funet led working group on wireless systems and mobility (MobileFunet) (WLAN infrastructure)

Author: Wenche Backman

Contributors: , Ville Mattila/CSC – IT Center for Science, Tanda Tuovinen/University of Helsinki, Matti Saarinen/University of Helsinki Juha Nisso/Tampere University of Technology, Siiri Sipilä/ Aalto University, Mikko Laiho/earlier University of Jyväskylä, currently University of Helsinki, Thomas Backa/Åbo Akademi University, Miika Räisänen/University of Oulu

May 2011

Document No: GN3-NA3-T4-wlan-infrastructure
Version / date: 13.05.2011
Original language: Finnish
Original title: "WLAN-verkon infrastruktuuri"
Original version / date: 1.0 of 13.05.2011
Contact: wenche.backman@csc.fi

Funet bears responsibility for the content of this document. The work has been carried out by a Funet led working group on wireless systems and mobility as part of a joint-venture project within the HE sector in Finland.

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 238875, relating to the project 'Multi-Gigabit European Research and Education Network and Associated Services (GN3)'.

Table of Contents

Executive Summary

The infrastructure of a WLAN network can be considered to include WLAN access points, the WLAN controller and software and services related to authentication, such as a RADIUS server and supplicants. Otherwise, the requirements of a WLAN network are the same as those for a fixed local-area network. In this document these parts of the WLAN network will be described, along with recommendations and configuration guidelines. Also monitoring the infrastructure is included in the document. The most detailed configuration guidelines can be found from the three appendices, in which Cisco controller configuration, HP controller configuration and FreeRADIUS configuration is described.

It is recommended that new networks should always be built to comprise a WLAN controller and controller-based access points. If the campus has more than one controller, these should, if possible, be connected to allow for seamless handover.

All Funet members should connect their wireless networks at least to the Funet roaming consortium, and if possible, to eduroam. Funet members should invest in providing information on roaming through their web site and as a first alternative to visitors arriving as well as to own users leaving for visits.

Regarding supplicant configuration, when using a public Certification Authority's (CA) signed certificate, it is important that the user's supplicant is configured to trust this certificate, and first and foremost, that the name of the authentication server to be trusted is defined in the supplicant.

As for monitoring the infrastructure, the organisation should monitor its own RADIUS server using a valid username, created for the purpose of monitoring. Monitoring should be handled at least using RADIUS authentication requests, but it is recommended to include also the chosen EAP methods in the monitoring.

1 Elements of the WLAN Network Infrastructure

The infrastructure of a WLAN network can be considered to include WLAN access points, the WLAN controller and software and services related to authentication, such as a RADIUS server and supplicants. Otherwise, the requirements of a WLAN network are the same as those for a fixed local-area network. If necessary, a separate local-area network can be set up between the access points and the controller, for example if you wish to use switches that support the Power over Ethernet (PoE) standard. With PoE, the power supply and network connection of access points can be handled with a single Ethernet cable, if the maximum power supplied by PoE is enough for the access points. If the WLAN network uses the same local-area network infrastructure than the other local-area network(s), a different VLAN can still be used for WLAN network traffic. This is done in several large organisations. Additionally, the traffic in different WLAN networks can be separated so that traffic from WLAN clients with different SSIDs is transferred in the fixed local-area network via a different VLAN.

There is also often a firewall between the WLAN network and the Internet. The rules of a WLAN network are usually either the same as those of other local-area networks, if the same firewall is used for all networks, or somewhat more lenient. If traffic in different WLAN networks is separated in the fixed local-area network using different VLANs, you can also define different firewall rules for the different networks. In this way, the rules for the most open networks can be stricter than those of the networks requiring user identification.

2 WLAN Network Controller Configuration

Previously, WLAN network coverage could only be achieved with stand-alone access points, but currently an increasing number of networks are built based on a controller. The controller will then control the access point functions such as message forwarding and signal level. A controller allows building a more uniform and clearer network. Maintenance is also easier.

With regard to configuration, all manufacturers have slightly different solutions, but the basic procedures are the same. Appenix 1 – Cisco Controller Configuration and Appendix 2 – HP controller configuration describe the configurations found out to be the best practices for each manufacturer.

The Cisco configuration instructions describe how to configure the controller so that traffic from access points always passes through the controller, and only then to the Internet via a router. Another alternative would be to direct the traffic from the access point directly to the nearest router. This can be achieved with the help of H-REAP functions (Hybrid Remote Edge Access Point). The H-REAP functions can be used to ensure that the WLAN clients are able to traffic even if the controller is down for a while. On the other hand, user identification, recommended in the Best Practice Document “WLAN Information Security” [1] only works through the controller even if traffic is directed from the access point to the nearest router. H-REAP functions can also be used to ensure that the controller never forms a bottleneck for the connections of the WLAN clients; on the other hand, the processing power and port speeds of the controller are in practice always high enough, at least today. Additionally, multicast between the controller and the access points to save network capacity cannot be implemented if traffic is not directed through the controller. For more information on implementing the H-REAP functions, see [2]. In short, directing traffic directly to the nearest router requires that both the access point and WLAN network (SSID) to which the user connects have been set into H-REAP mode.

In the HP controller, it is also possible to choose whether traffic always passes through the controller or whether it is directed straight from the access point to the nearest router. The instructions do not comment on how the user traffic is handled in a fixed LAN network. However, the HP instructions do mention that the network structure can be made simpler by passing all traffic through the controller.

2.1 Recommendations

With regard to the WLAN network infrastructure, we recommend the following:

- New networks should always be built to comprise a WLAN controller and controller-based access points.

With regard to controller configuration, we recommend the following:

- Define the same, basic configuration for access points located close to each other, for example in the same building or on the same campus, although the controller would support setting different parameters on different access points. This makes maintenance easier.

- Allow pinging the controller and/or access points from anywhere in the Access Control Lists.

- If possible, configure the access points to traffic in both 5 GHz and 2.4 GHz bands.

If more than one controller is in use on one campus, configure the controllers to communicate with each other. This allows users to switch between access points (handover) without disruption even if the access points are connected to different controllers. In Cisco controllers, mobility groups are defined for this purpose.

3 Authentication Methods and Roaming

While configuring the WLAN network controller, you also need to define the server connected to the organisation's user database, handling user authentication. The server is usually a RADIUS server, and the RADIUS servers of different organisations can be connected to a common root, if you wish to share the network with the other organisations. This method is called roaming.

With roaming, users can connect to the net quickly and easily while they are visiting different universities. Roaming is based on reciprocity between the organisations. This means that if organisations A and B are involved in roaming, the users of organisation A can use the public access network of organisation B, and vice versa.

Joining a roaming system is easy, but certain federations place certain requirements on the network. These should be taken into consideration at an early stage, even as early as during the network's planning stage.

The roaming systems available in the Funet network, ***eduroam*** and ***Funet roaming***, are based on a RADIUS hierarchy. The username and password are sent to the home organisation based on the username's realm. The organisation then compares them to the information stored in a database. Shibboleth roaming is the third roaming method, and it can be implemented in organisations that have joined the Haka infrastructure in Finland. Joining eduroam or Funet roaming requires that the organisation has to set up its own RADIUS server and authenticate its users using that RADIUS server. The network access points must also support RADIUS authentication.

Compared to Funet roaming, eduroam has better information security, because user authentication must be done in accordance with the 802.1x standard. This means that the network access points must support the standard in question. Additionally, traffic must be encrypted with WPA or WPA2; for more details, see [2]. In Funet roaming, Web authentication is allowed and traffic can be unencrypted. See Table 1 for the differences between eduroam and Funet roaming.

Table 1. The differences between eduroam and Funet roaming.

eduroam	**Funet roaming**
international	national
secure authentication (802.1x) and encryption	Web authentication allowed

(WPA/WPA2)	
a single SSID (eduroam)	uses the organisations' own SSIDs
only for higher-level educational and research institutions	commercial operators are allowed

3.1 RADIUS servers

Radiator and FreeRADIUS are the most common RADIUS servers, the former being commercial and the latter based on open source code. Other commercial RADIUS servers include Microsoft's IAS/NPS and Cisco's ACS.

RADIUS servers can be put in the following order of superiority based on testing by Funet members:

- **Radiator** is the easiest to use and has the most versatile feature set. However, the product is commercial.
- **FreeRADIUS** has gained a lot of popularity due to its open source code. Version 2 of the server is significantly easier to configure than version 1.
- **Microsoft IAS/NPS** server lacks some features such as realm stripping, which is a handy feature when you wish to relay only the username and password to the database.

You can find comprehensive RADIUS server configuration instructions in the eduroam cookbook [3].You can find detailed FreeRADIUS configuration instructions in Appendix 3.

Finnish organisations that have joined the roaming system prior to February 2010 should check that they have connected their RADIUS servers to both Finnish root servers, fltr.funet.fi and ftlr2.funet.fi. If you are using Radiator, we recommend using the Dead Realm Marking method [4].

3.2 IEEE 802.1x

In eduroam, authentication must be handled using the 802.1x standard. The IEEE 802.1x standard must be taken into consideration when configuring the WLAN controller and RADIUS server. Additionally, the standard must be taken into consideration when instructing the users, because the network is joined with a supplicant instead of a browser. IEEE 802.1x is used for port based authentication. 802.1x defines three parties: the supplicant or the customer (the user's terminal device/software), the authenticator (network connection point, for example an access point) and the authentication server.

In 802.1x based authentication, the EAP over LAN (EAPOL) protocol is used in the air interface. EAP (Extensible Authentication Protocol) messages are relayed inside RADIUS packets between access points and the authentication server. The actual authentication is done using Radius-Access-Request packets, where the user-provided information is relayed to the authentication server, and Radius-Access-Challenge packets sent by the authentication server as a reply, used for checking the user's identity. Access to the network is allowed

with a Radius-Access-Accept packet, after which the access point sends the required encryption keys to the supplicant. The required encryption keys are determined by the network's security level; see [1]. If you wish to change the encryption keys at certain intervals, the user must often be re-authenticated.

Unlike Web authentication, in 802.1x no IP address is assigned to the user before authentication, and packets are only sent when requested. No unrequested packets from unauthenticated users appear in the air interface.

3.3 EAP Methods and Authentication

802.1x is based on EAP, and rules on how authentication is handled is defined in the EAP method. The rules include how the user's identity is requested and sent, and how challenges related to authentication are handled. EAP includes both outer and inner methods. In practice, only the following outer EAP methods offer encryption of sufficient strength:

- EAP-TLS (Transport Layer Security), which is based on reciprocal authentication by the exchange of certificates. However, it requires the installation of a certificate on every terminal device.

- PEAP (Protected EAP) and TTLS (Tunneled TLS) are two-phased and very similar. In the first phase, a TLS tunnel is opened, through which the server's certificate is sent and accepted. Next, the TLS tunnel is used to authenticate the user using an inner authentication method. The most well-known inner authentication methods are:

    - MSCHAPv2 (Microsoft Challenge-Handshake Authentication Protocol version 2), where the password is sent in the hash format used by Windows.

    - PAP (Password Authentication Protocol), where the password is sent over the network in cleartext. Only works with TTLS, because TTLS does not require an EAP-type inner authentication method.

The RADIUS server used for authentication can be connected to any database, but the password storage format determines which EAP methods can be implemented in the infrastructure. If the RADIUS server is connected to AD, the password is stored in the hash format used by MSCHAPv2 (NTHASH), and all supplicants of the terminal devices, which usually support MSCHAPv2, can be used. MSCHAPv2 is supported by, for example, Linux, Windows, Nokia mobile phones and Macs. If the RADIUS server used and AD are incompatible, you will have to use a domain controller or other solutions. The RADIUS server can also be connected to a database comprising other hashes, for example MD5 hashes, but in this case, the operating systems' own supplicants cannot necessarily be utilised and a third-party supplicant has to be used. You can find a comprehensive list of the compatibility of different hash formats and authentication methods in [5].

The EAP methods used are defined in connection with the configuration of the home organisation's RADIUS server, and the defined methods are used to authenticate the user in both the home organisation's network and in the roaming organisation's network. Thus, you need to take into consideration that the EAP methods used are determined by the home organisation of each user. When roaming elsewhere, the same EAP methods as in the home organisation are always used.

3.4 Certificate

A certificate must absolutely be acquired for the RADIUS server handling the authentication of the WLAN network. The certificate can be either self-signed or confirmed by a public CA. When a certificate is acquired for the organisation's RADIUS server, the following things must be taken into consideration:

- If a commercial CA certificate is acquired, we recommend choosing a supplier that is as widely known as possible, with its root certificate pre-installed on most terminal devices.

- If you create a self-signed certificate on the server, you need to develop the user support process so that a reference certificate corresponding to the certificate can be securely installed on the terminal devices of the end users in your own organisation.

In order to achieve secure authentication with the certificate, close attention must be paid to the supplicant's settings. If the acquired certificate is signed by a public CA, it is not enough that a user defines the CA as a trusted supplier in its supplicant; the user must also define the name of the server for which the certificate has been signed, for example *auth.csc.fi*. If you are using a self-signed certificate, you must ensure that the certificate can be distributed to the users easily and securely. The certificate must be installed on the users' terminal devices before they connect to the network.

3.5 Recommendations

With regard to taking authentication and roaming into consideration in the network infrastructure, we recommend the following:

- all Funet members should connect their wireless networks at least to Funet roaming, and if possible, to eduroam;

- Funet members should invest in providing information on roaming at least in the following ways:

    - their website should include information on the possibilities of using Funet roaming and/or eduroam on member campuses;

    - when visitors arrive on campuses or when staff or students of your own organisation travel, eduroam and Funet roaming should be primarily considered. Only after this, an attempt to open a wireless network connection in another way would be made.

- Funet members should check that they have connected their RADIUS servers to both Finnish root servers, fltr.funet.fi and ftlr2.funet.fi. If you are using Radiator, we recommend using the Dead Realm Marking method for this.

- When using a public CA's signed certificate, Funet members should pay close attention to ensure that the CA used in the users' supplicants has been chosen as a trusted supplier and, very importantly, that the server name has also been defined.

4 User Connecting to the Network

When the WLAN network has been set up and the RADIUS server connected to the user database and possibly also to the roaming RADIUS hierarchy, users can connect to the network using the defined methods. In Funet roaming this means Web authentication, but in the case of eduroam, connection is made using the operating system's own supplicant or a third-party supplicant.

4.1 Web Authentication

In networks using Web authentication, including networks participating in Funet roaming, a login page opens for the user when he/she opens the browser. For authentication purposes, only a browser is required on the terminal device. Organisations involved in Funet roaming can configure their login page to include a list of all involved organisations so that the user can immediately know whether he/she has a possibility to connect to the network with his/her own user account. The username is entered in form username@organisation.fi. A list of involved organisations, maintained by Funet can be found from [6].

4.2 Supplicants

In networks applying the 802.1x standard, including eduroam, users connect to the network using the EAP method defined by their home organisation. The required supplicant is found in most operating systems and external network interface cards. Linux features a WPA supplicant, and Windows also has its own. Nokia's latest phones and the iPhone also include a 802.1x supplicant. It is also possible to use a third-party supplicant, of which SecureW2 is popular. Configuring the supplicant may be non-trivial, for which purpose a lot of guides have been created. Windows XP, Intel and SecureW2 supplicant configuration instructions for eduroam can be found from e.g. [7].

For Windows Vista and Windows 7, an installer can be created, generating an XML-file (eXtensible Markup Language) based on the WLAN network information. The file can be used to relay the settings to the supplicant, and the configuration is done. More information is currently only available in Finnish [8].

You can define both outer and inner identities for many supplicants. In the outer identity, the username can be anything, but the realm must be correct, for example anonymous@csc.fi. The inner identity must include both the correct username and domain, for example wbackman@csc.fi. By using an anonymous outer identity while roaming, you can avoid storing your identity in the logs of the visiting organisation. Only the home organisation

knows the inner identity. The supplicants of Linux, Nokia phones and Secure W2 support the definition of separate inner and outer identities, but the supplicant inbuilt in Windows does no support this.

4.3 Recommendations

With regard to supplicants, we recommend the following:

- Users should be instructed in using an anonymous outer identity whenever possible; if the authentication attempt fails for some reason, one should make at least one attempt with an easily remembered outer identity to help make troubleshooting easier.

5 Monitoring of Infrastructure

The roaming infrastructure should be monitored in order to detect and solve problems before the users become frustrated. Nagios is well suited to this purpose, and Funet monitors the Finnish root servers ftlr.funet.fi and ftlr2.funet.fi and its own RADIUS server by sending authentication requests and verifying that the authentication is successful. The monitoring is carried out through several routes:

- an @csc.fi username is sent to ftlr.funet.fi and ftlr2.funet.fi;
- the authentication of a username in the local database is tested at ftlr.funet.fi and ftlr2.funet.fi;
- the @csc.fi username is sent directly to CSC's RADIUS server;
- additionally, the authentication of a guest account is tested, and ping latency, packet loss and opening of SSH connections is monitored.

One of the most advanced monitoring methods is handling monitoring using messaging corresponding to that of the user's supplicant. Another option is to use just RADIUS authentication requests for this purpose, but in this case, certificate information etc. would be left unchecked. You can handle checking that the server functions from the perspective of the supplicant by using the eapol_test program [9] of the WPA supplicant. The eapol_test program can be connected at least to Nagios and Big Sister, the monitoring software used by the University of Helsinki. The program supports the implementation of monitoring for, for example, the PEAP-MSCHAPv2, TTLS-MSCHAPv2 and TTLS-PAP methods.

In order to track the real-time situation of roaming in the Funet network, monitoring based on the eapol_test program has been implemented, with its results displayed at the extranet (info.funet.fi). Depending on the methods used in the organisation, one or more of the following methods are being monitored: PEAP-MSCHAPv2, TTLS-MSCHAPv2 and/or TTLS-PAP. In order to connect more servers to the service, Funet will need a valid username and password and information on the server's certificate. When authentication implemented using all three methods is successful, the result is as depicted in Figure 1.

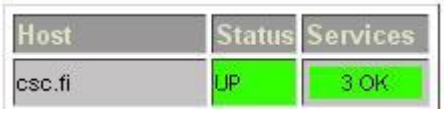

Figure 1. Monitoring roaming. The results from CSC's authentication server are displayed as an example.

Funet members are urged to monitor their own servers at least with RADIUS authentication requests, but preferably with the eapol_test program of the WPA supplicant. You can find installation instructions for eapol_test including the configuration information for (EAP) authentication methods from [9]. Two files must be given to the check_eapauth script as parameters, one of which is the .conf file found in the eapol_test installation instructions. For PEAP-MSCHAPv2 it can look like this:

```
network={
        ssid="example"
        key_mgmt=WPA-EAP
        eap=PEAP
        identity="bob@organisation.fi"
        anonymous_identity="anonymous@organisation.fi"
        password="hello"
        phase2="autheap=MSCHAPV2"

   #
   #  Uncomment the following to perform server certificate validation.
   ca_cert="/etc/raddb/certs/ca.der"
}
```

The other one is a file containing the IP address of the server used for roaming and the shared secret in the following format:

```
radius-server <the server's IP address>
radius-secret <the shared secret>
```

The monitoring server's IP address and shared secret must naturally also be added as a client of the server used for roaming.

5.1 Recommendations

With regard to the monitoring of roaming infrastructure, we recommend the following:

- the organisation monitors its own RADIUS server using a valid username, created for the purpose of monitoring. Monitoring should be handled at least using RADIUS authentication requests, but we recommend using the WPA supplicant's eapol_test program for monitoring so that the functionality of the EAP methods could also be monitored.

With regard to roaming infrastructure, MobileFunet points out the following:

- if practically possible, we recommend arranging a valid username for Funet for the purpose of real-time visualisation of roaming situation at the extranet (info.funet.fi). This would be for the benefit of both IT support and, ultimately, the end users.

6 A Few General Remarks

- In the PAP method, the password is sent unencrypted, but this does not prevent the user database from including the password in an encrypted form.
- FreeRADIUS does not work with the supplicants of some phones.
- In a large campus area, there may be several RADIUS servers, acquired for different purposes. Connecting them may make maintenance easier, but reaching an understanding may be difficult.

7 Authentication and Roaming Solutions of Funet members

The authentication and roaming solutions of the different Funet members are summarised as examples under this heading.

7.1 University of Helsinki

University of Helsinki uses PEAP-MSCHAPv2, TTLS-MSCHAPv2 and TTLS-PAP. The authentication request is directed to the correct database based on the realm with the help of the RADIUS hierarchy internal to the campuses. For example, usernames of the form @ad.helsinki.fi are directed to AD, where passwords are stored in the NTHASH format, and EAP methods PEAP-MSCHAPv2 and TTLS-MSCHAPv2 can be supported.

University of Helsinki uses Radiator RADIUS servers. They have poor experiences of FreeRADIUS v1, but FreeRADIUS v2 should work better.

There are no limitations on supplicants; all alternatives are supported.

7.2 Helsinki University of Technology / Aalto University

Aalto University currently has a FreeRADIUS v2 server connected to a Microsoft NPS server, which, in turn, is connected to AD. The FreeRADIUS v2 server is not directly connected to AD due to the current lacks in Samba. PEAP-MSCHAPv2 is supported of the EAP methods. Instead of aalto.fi, org.aalto.fi is used as the realm, because the domain controller has been configured for this realm, and authentication with another realm will not succeed. The NPS server does not feature realm stripping, and for this reason, just the username and password cannot be relayed to the database.

7.3 Åbo Akademi

Åbo Akademi uses FreeRADIUS v1 connected to an LDAP (Lightweight Directory Access Protocol) database in its Sparknet network. A Microsoft IAS server is used in Vaasa.

In addition, Åbo Akademi is currently undergoing a network upgrade, during which a FreeRADIUS v2 server connected to an LDAP database has been set up for eduroam, for example.

7.4 Tampere University of Technology

Tampere University of Technology uses Radiator connected to LDAP. The supported EAP methods include PEAP-MSCHAPv2, TTLS-MSCHAPv2 and TTLS-PAP. The passwords are stored in several different formats in the LDAP database, such as NTHASH and SHA-hash. However, the passwords are not stored in cleartext format.

Guidance and support is provided for the supplicants of Windows and mobile phones.

7.5 University of Jyväskylä

The University of Jyväskylä uses FreeRADIUS servers. PEAP-MSCHAPv2 and TTLS-PAP are the supported EAP methods. The passwords used during PEAP-MSCHAPv2 authentication must be changed every two weeks. SecureW2 is supported, and the licence required for pre-configuration has been acquired.

Shibboleth roaming is also used to some extent.

7.6 University of Oulu

The University of Oulu uses FreeRADIUSv2 servers, connected to two different Active Directories. FreeRADIUS is used for, e.g. realm rewriting and the execution of the ntlm_auth program with different parameters so that the users can use a more easily remembered form of username (~e-mail address).

In the access points of the University of Oulu, persons logging in using staff credentials are assigned a different VLAN than students and roaming organisations. This allows the elimination of a separate staff network SSID, and usage *should* be easier.

An eduroam installation package has been built for Windows XP, Vista and 7 users, installing the oulu.fi CA certificate on a workstation and creating an eduroam network profile. All the user needs to do is enter his/her username and password when the operating system prompts for them. The installation package has been implemented using NSIS.

PEAP-MSCHAPv2 is the EAP method supported, and of the supplicants, those included with the operating systems. There are no instructions for the use of the other supplicants.

Appendix 1 – Cisco Controller Configuration

This appendix describes the configuration of a Cisco WLAN controller. The screenshots are from a 4402 series controller, which means that the command windows of different models may look different.

Basic Settings and Defining the IP Address

In the first stage, the controller does not have an IP address, and configuration must be done using the Command Line Interface. Once the controller has been assigned an IP address, the rest of the configuration can be done using a browser over the Web interface.

First, open a connection to the controller either directly with a serial console, or connect the controller to a serial console server and use it. Perform the configuration as follows:

```
Welcome to the Cisco Wizard Configuration Tool
Use the '-' character to backup
System Name [Cisco_b2:e2:83]: <your_system_name>
Enter Administrative User Name (24 characters max): <your_username>
Enter Administrative Password (24 characters max): <your_password>
Re-enter Administrative Password                  : <your_password>

Service Interface IP Address Configuration [none][DHCP]: DHCP

Enable Link Aggregation (LAG) [yes][NO]: NO

Management Interface IP Address: e.g. xxx.yyy.zzz.1
Management Interface Netmask: <your_network_mask>
Management Interface Default Router: <your_router's_IP_address>
Management Interface VLAN Identifier (0 = untagged): <0 or 1>
Management Interface Port Num [1 to 2]: 1
Management Interface DHCP Server IP Address: e.g. xxx.yyy.zzz.2

AP Transport Mode [layer2][LAYER3]: <layer2 if the controller and the access
points are located in the same network, layer3 if there is a routed network in
between>

AP Manager Interface IP Address: e.g. xxx.yyy.zzz.3

AP-Manager is on Management subnet, using same values
```

```
AP Manager Interface DHCP Server (xxx.yyy.zzz.2):

Virtual Gateway IP Address: xxx.yyy.zzz.www
#NOTE: If you wish to define a Web-authenticated network, this IP must be
#from a different area than the IP addresses of the controller’s other
#interfaces. Otherwise, the certificate used for login will not work as
#desired.

Mobility/RF Group Name: <define a name if you wish to connect several controllers
together>

Enable Symmetric Mobility Tunneling [yes][NO]: NO

Network Name (SSID): <it would be good to define test_SSID or similar already at
this stage>

Allow Static IP Addresses [YES][no]: no

Configure a RADIUS Server now? [YES][no]: no #Done later
Warning! The default WLAN security policy requires a RADIUS server.
Please see documentation for more details.

Enter Country Code list (enter 'help' for a list of countries) [US]: FI

Enable 802.11b Network [YES][no]: no
Enable 802.11a Network [YES][no]: YES
Enable Auto-RF [YES][no]: YES

Configure a NTP server now? [YES][no]: no
Configure the system time now? [YES][no]: no

Warning! No AP will come up unless the time is set.
Please see documentation for more details.

Configuration correct? If yes, system will save it and reset. [yes][NO]: yes
#Once the system has restarted, define the NTP servers:
(Cisco Controller) >config time ntp server 1 xxx.yyy.z.www
(Cisco Controller) >config time ntp server 2 xxx.yyy.z.wwz
```

Software Update

At the next stage, we recommend performing a software update. You can download the latest software version from Cisco’s website, but you need a user account for this. However, we recommend first storing the current software version to a TFTP (Trivial File Transfer Protocol) server. You can do this as depicted in Figure 2.

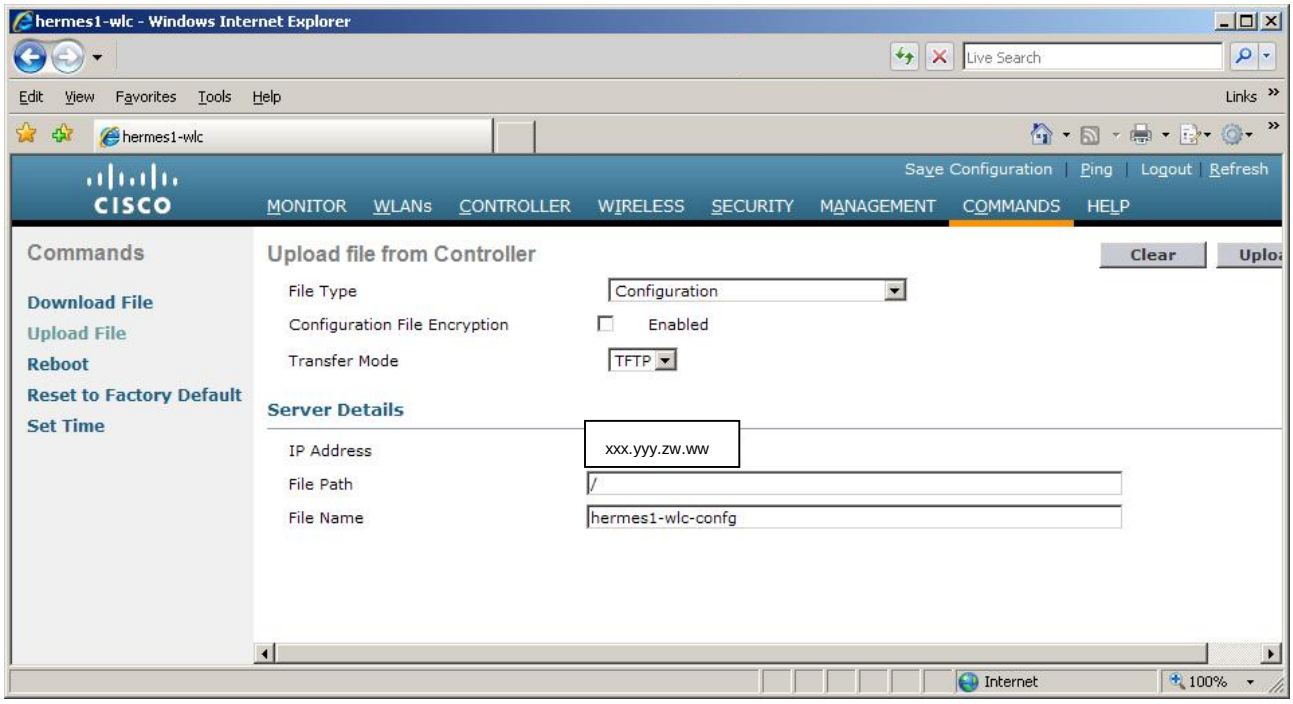

Figure 2. Storing the current software version to an TFTP server.

Next, download the latest software version from a TFTP server to the controller as depicted in Figure 3.

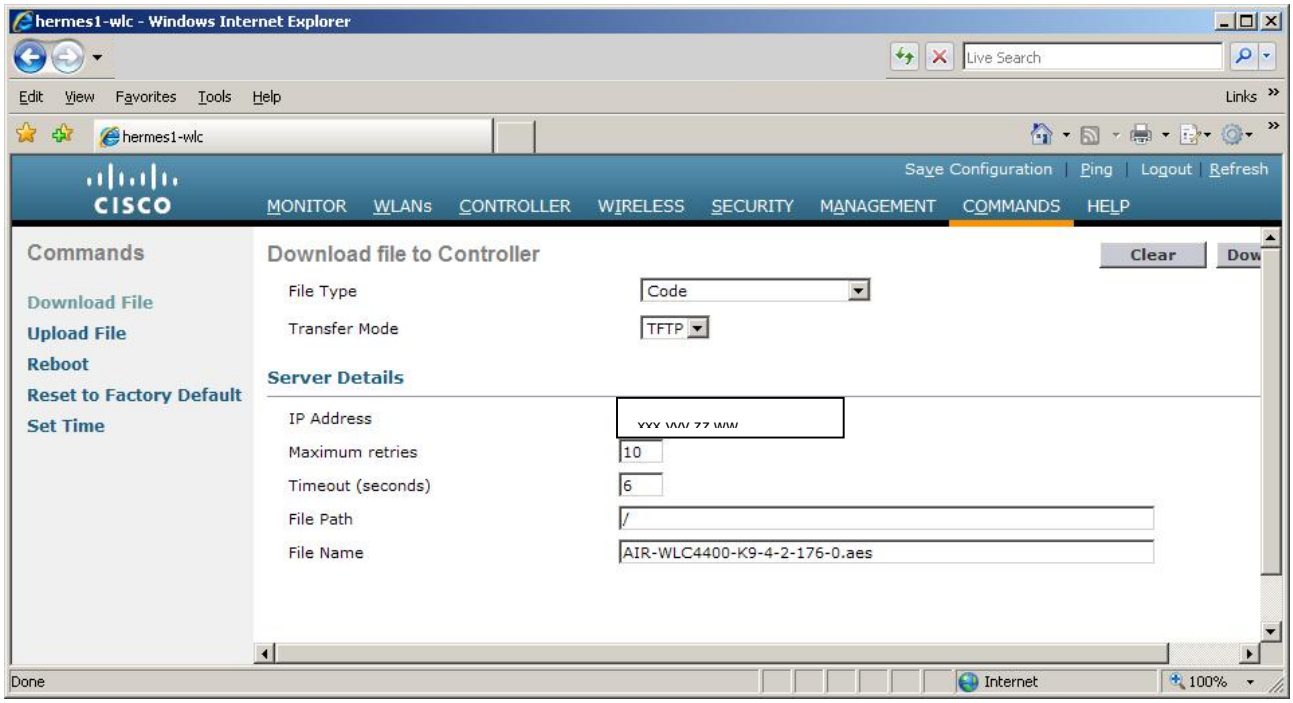

Figure 3. Moving the new software version to the controller.

We also recommend updating the bootloader as depicted in Figure 4.

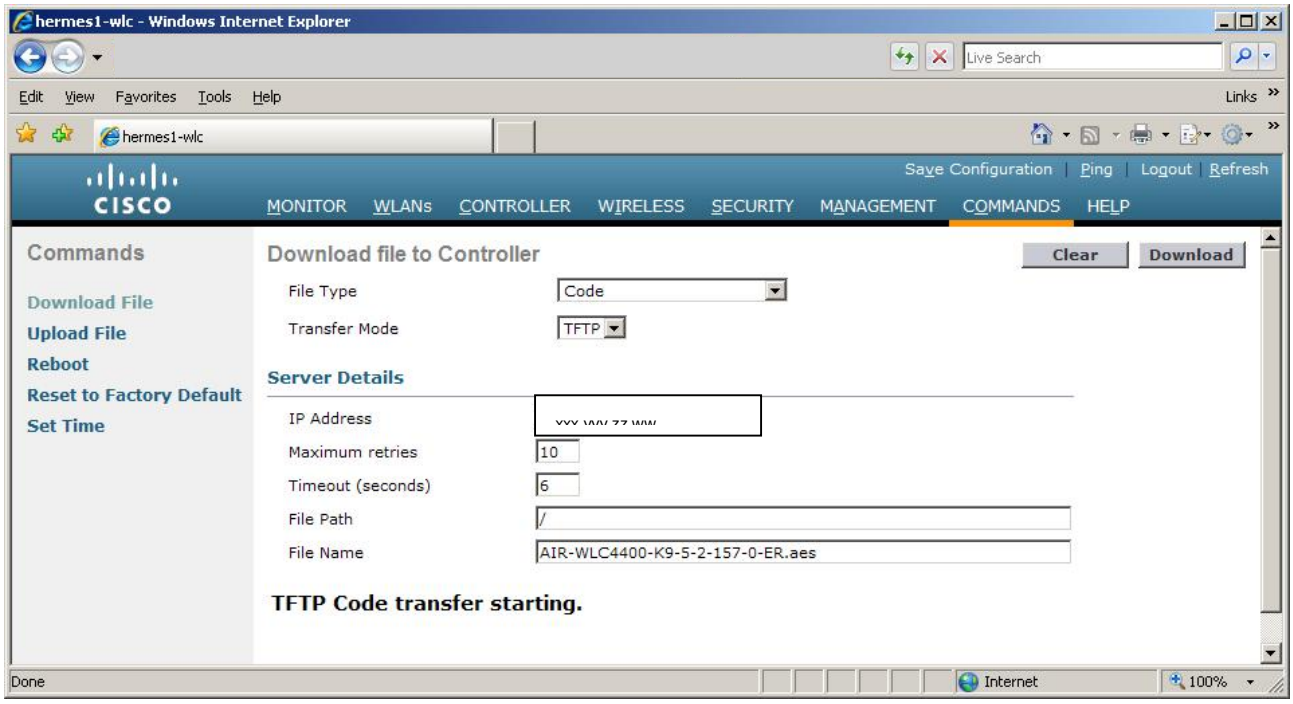

Figure 4. Moving a new bootloader version to the controller.

After the bootloader update, the actual software should be reupdated as depicted in Figure 3.

Taking VLANs into Consideration

If the controller is connected to a local area network using virtual LANs, these are also defined in the controller. You can define the VLANs by adding dynamic interfaces to the controller with the correct identifiers defined. You can add a VLAN identifier by first selecting CONTROLLER from the top bar and then Interfaces from the side bar. Click the New... button and define the VLAN information on the page that opens, for example as depicted in Figure 5.

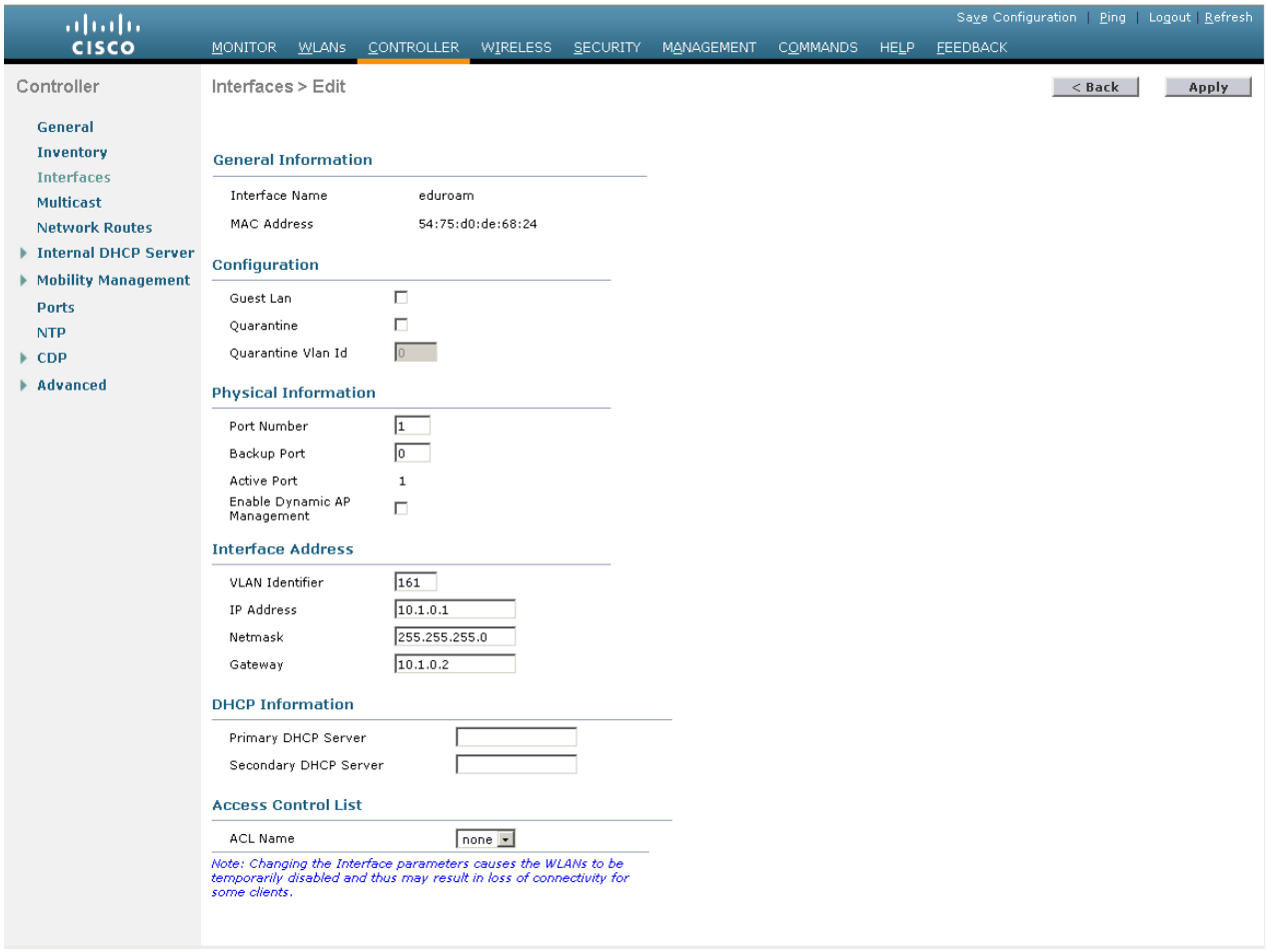

Figure 5. Defining a VLAN identifier.

Click the Apply button. The outcome is as depicted in Figure 6.

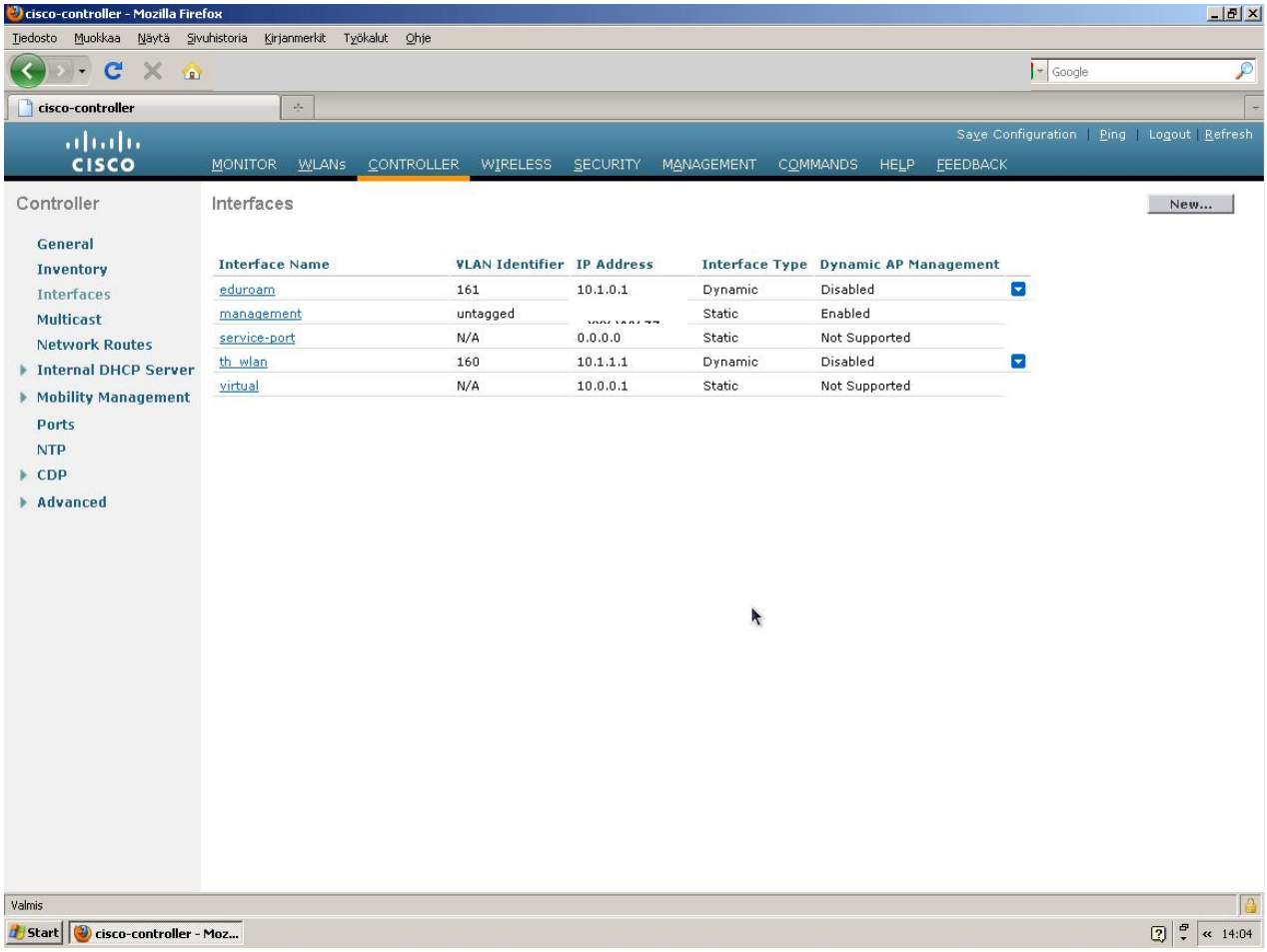

Figure 6. VLANs 161 (eduroam) and 160 (th_wlan) added to the controller as dynamic interfaces.

Once the VLANs in the local area network have been defined in the controller, the user can be directed to the correct VLAN by updating the right VLAN identifier in the Access-Accept packet. However, this is only done when the user is in its home organisation's network, i.e., VLAN identifiers are not sent outside the campus. For more instructions on directing users to the right VLAN, see the FreeRADIUS configuration instructions in **Appendix 3 – FreeRADIUS Configuration**.

Defining an Access Control List

An ACL (Access Control List) is a tool preventing unauthorised access to the controller. You can begin defining an ACL by selecting SECURITY from the top bar and Access Control Lists | Access Control Lists. Create a new list by clicking the New… button, see Figure 7.

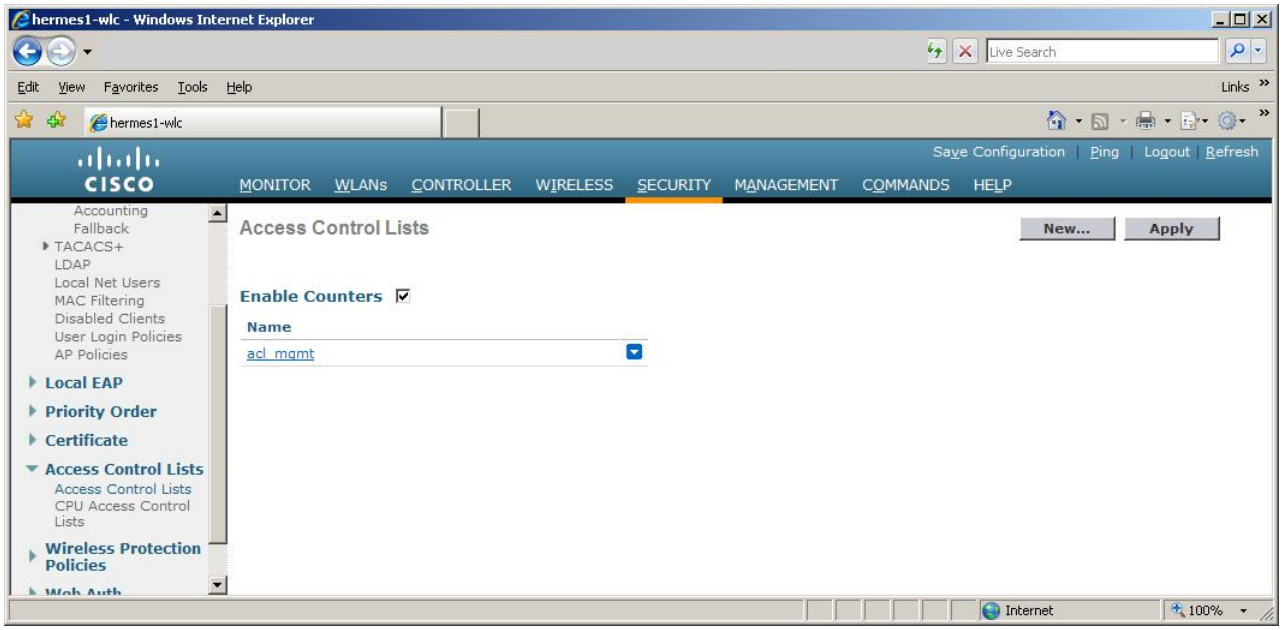

Figure 7. Access Control List creation window.

Next, open the just created Access Control List and add the required rules by clicking the Add New Rule... button. See Figure 8 for an example.

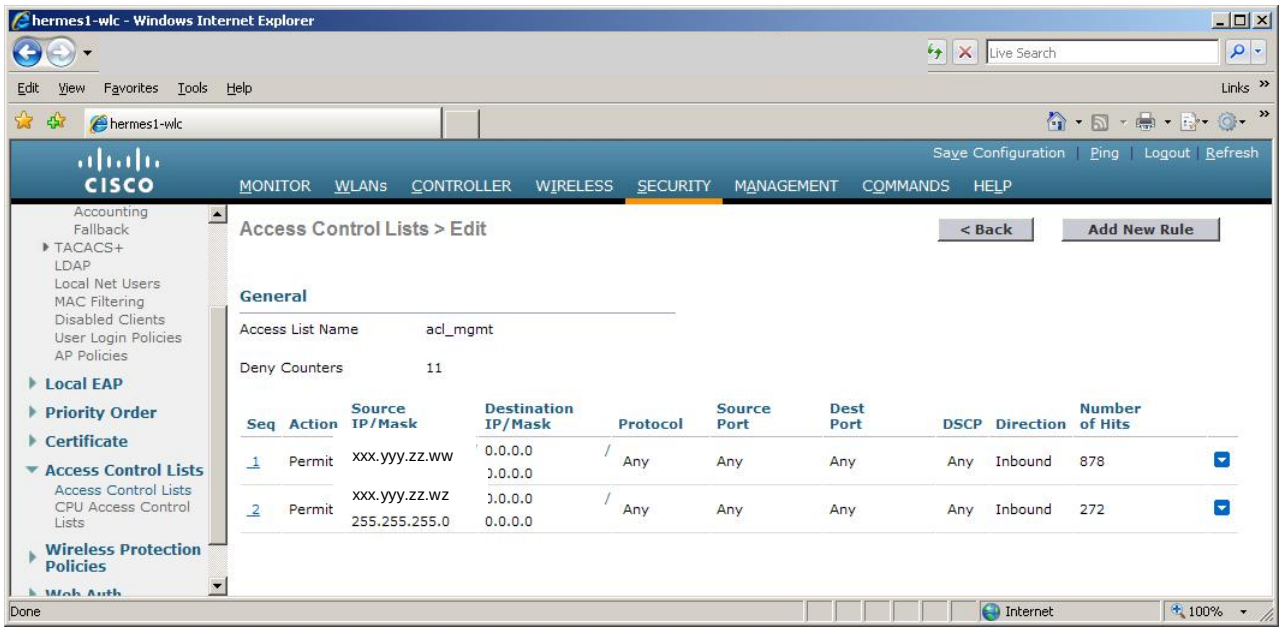

Figure 8. Access Control List rule creation window.

You should add the following rules to the Access Control List:

- Network(s) from which the controller is maintained
- Addresses of any monitoring servers
- Network(s) from which addresses are assigned to WLAN clients and access points
- The address of the RADIUS server through which users are authenticated
- Always respond to a ping, see Figure 9.

When defining the rules, remember that in the Direction section, Inbound means packets coming to the controller and Outbound means packets leaving to the terminal devices. The rules presented above will be defined for the CPU (Central Processing Unit), due to which the direction is always Inbound. No limits can be set for packets sent by the CPU.

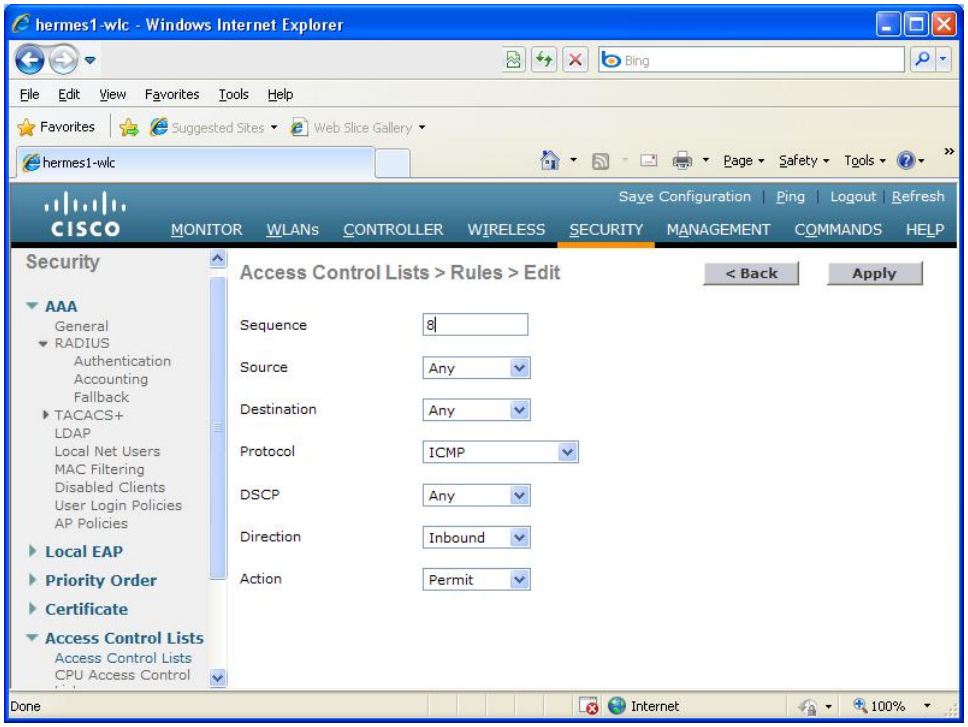

Figure 9. Responding to a ping command.

NOTE: In accordance with the Best Practices document “WLAN Information Security” [1], SMTP connections must be limited in order to prevent the sending of spam messages. SMTP connections from the Internet to WLAN network users must be blocked, and SMTP connections from the users of the WLAN network must be limited so that access is allowed only to the organisation’s own SMTP servers. These limitations can be implemented using Access Control Lists, but as a result, the WLAN network client connection speeds will drop to around 1 Mbps. For this reason, we recommend limiting the SMTP connections elsewhere, for example in the firewall.

The next stage is to implement the ACL for the CPU. You can do this by selecting CPU Access Control Lists from the side bar and filling in the menus as depicted in Figure 10.

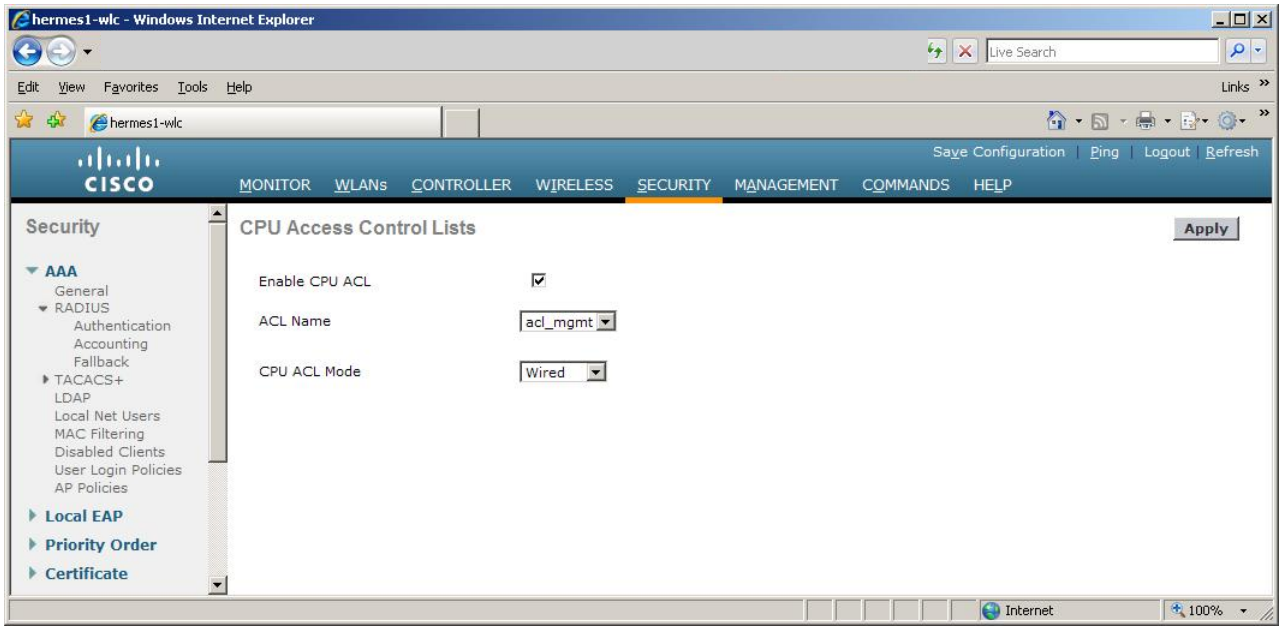

Figure 10. Activating an Access Control List.

Configuration of the Internal DHCP Server (Optional)

The Cisco controller can operate as a DHCP server, and in this case, serve the access points and WLAN terminal devices in the network. The internal DHCP server cannot service other terminal devices. If there is not already a DHCP server in the network, the internal DHCP server can be used. However, its functionality is not very advanced and the distribution of IP addresses may be slow. If possible, we recommend setting up a separate DHCP server for the WLAN network.

The internal DHCP server can be configured by selecting CONTROLLER from the top bar and Internal DHCP Server | DHCP Scope from the side bar. See Figure 11 for a configuration example. We recommend leaving a couple of IP addresses out of the beginning and end of the network IP address space so that they can be distributed to the various interfaces of the controller, and possibly switches and other network hardware.

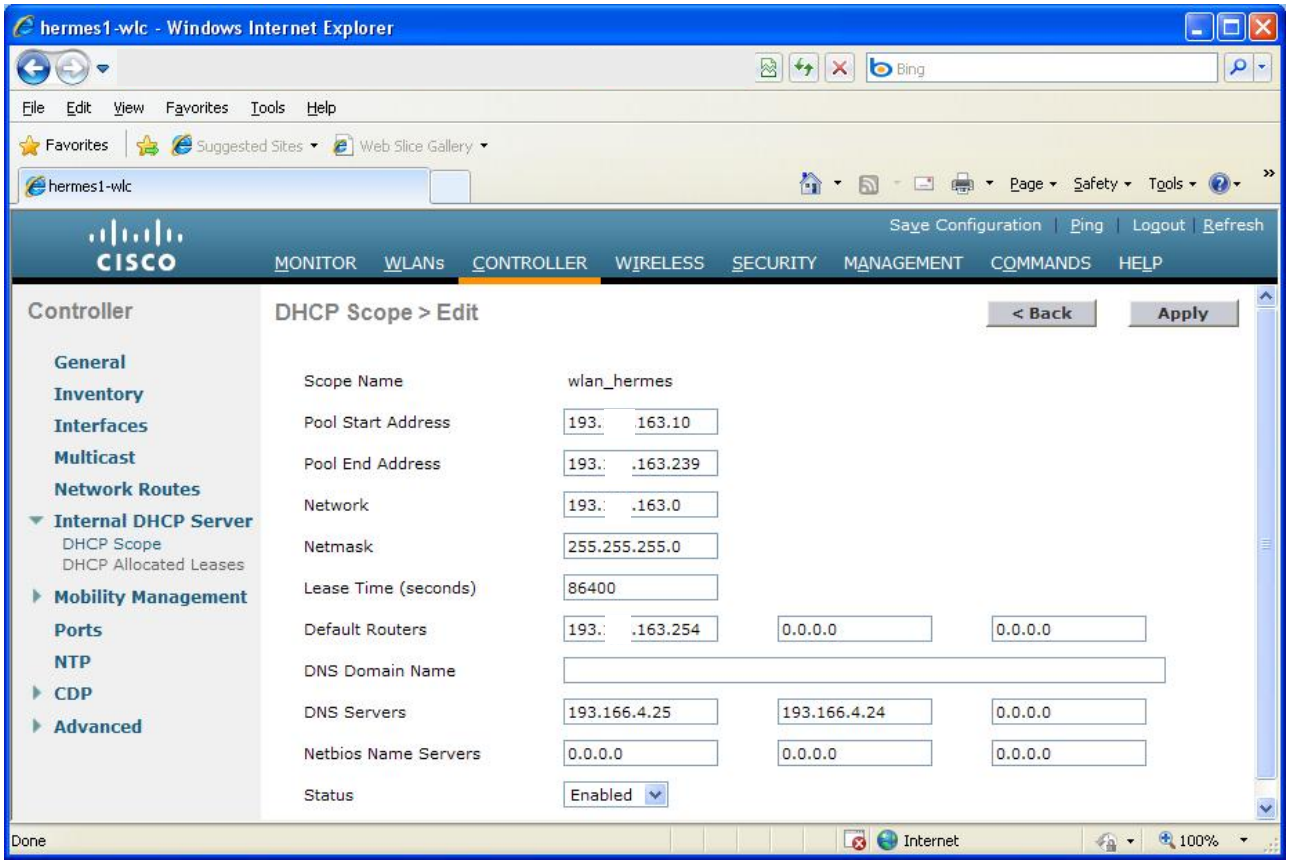

Figure 11. Configuration of the internal DHCP server.

Next, define the DHCP server for the management interface by selecting Interfaces -> Management from the side bar and defining the same address for the Primary DHCP Server as the management interface address, see Figure 12.

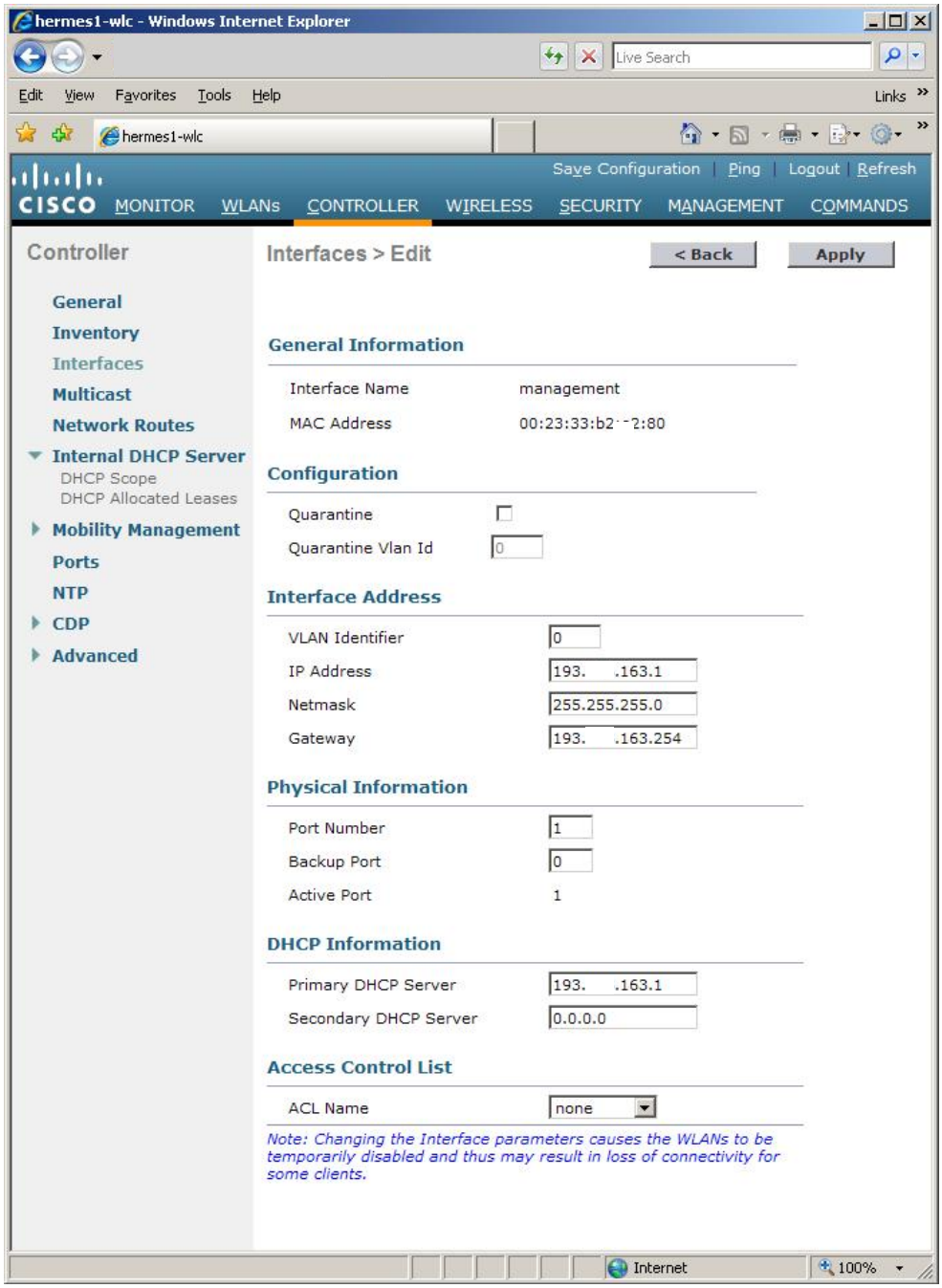

Figure 12. Taking the internal DHCP server into use.

IPv6 addresses can be assigned via a router using the autoconfiguration protocol.

Connecting Access Points to the Network and their Configuration

If the access points are connected to the same network as the controller, they will find the controller automatically and connect to it. In other cases, the controller's IP must be found in the name service as CISCO-LWAPP-CONTROLLER. When a access point has once found the controller, it will store the controller's

address and can connect from any network, as long as access from the network has been opened to the controller's CPU; see Defining an Access Control List.

In the default configuration of at least some models of Cisco access points the 5 GHz radios are on as a default. When the access points have connected to the network, the rest of the configuration related to the air interface can be made. First, set up the 2.4 GHz frequency network by selecting WIRELESS from the top bar and 802.11b/g/n | Network from the side bar. Supporting the 802.11b standard reduces the overall capacity of the network, so we recommend supporting only the 802.11g/n standards in the 2.4 GHz band. For more information on the reduction in overall capacity, see the Best Practices document "WLAN Network Planning and Setup" [10]. You can define network support for the 802.11g standards as depicted in Figure 13.

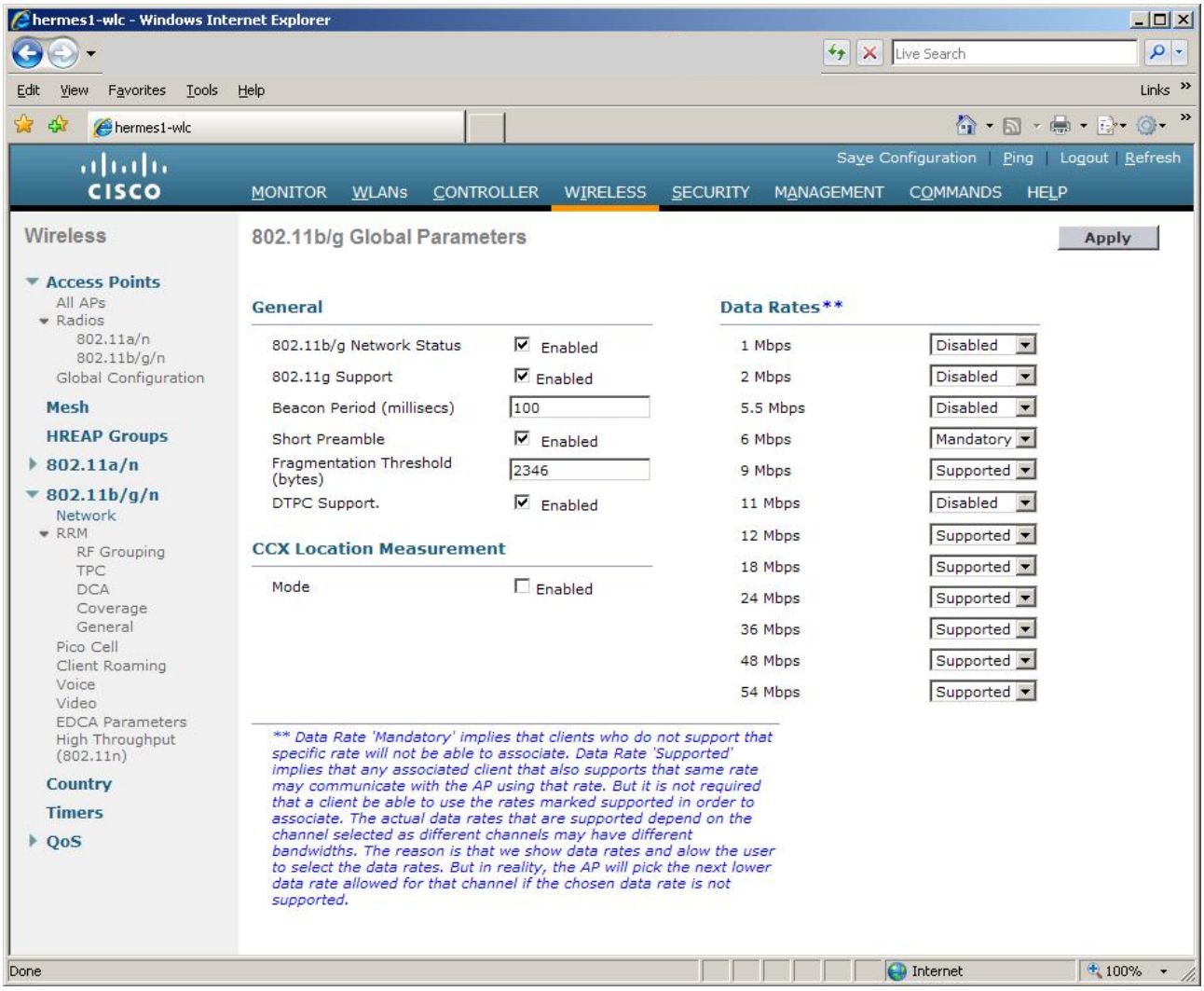

Figure 13. Defining network support for the 802.11g standard.

If you also wish to support the 802.11b standard, edit the supported transfer speeds so that the selection for 1 Mbps is *Mandatory* and for the others, *Supported*.

Next, define the 802.11a standard operating at a 5 GHz frequency as depicted in Figure 14. Open the definition view by selecting 802.11a/n | Network from the side bar.

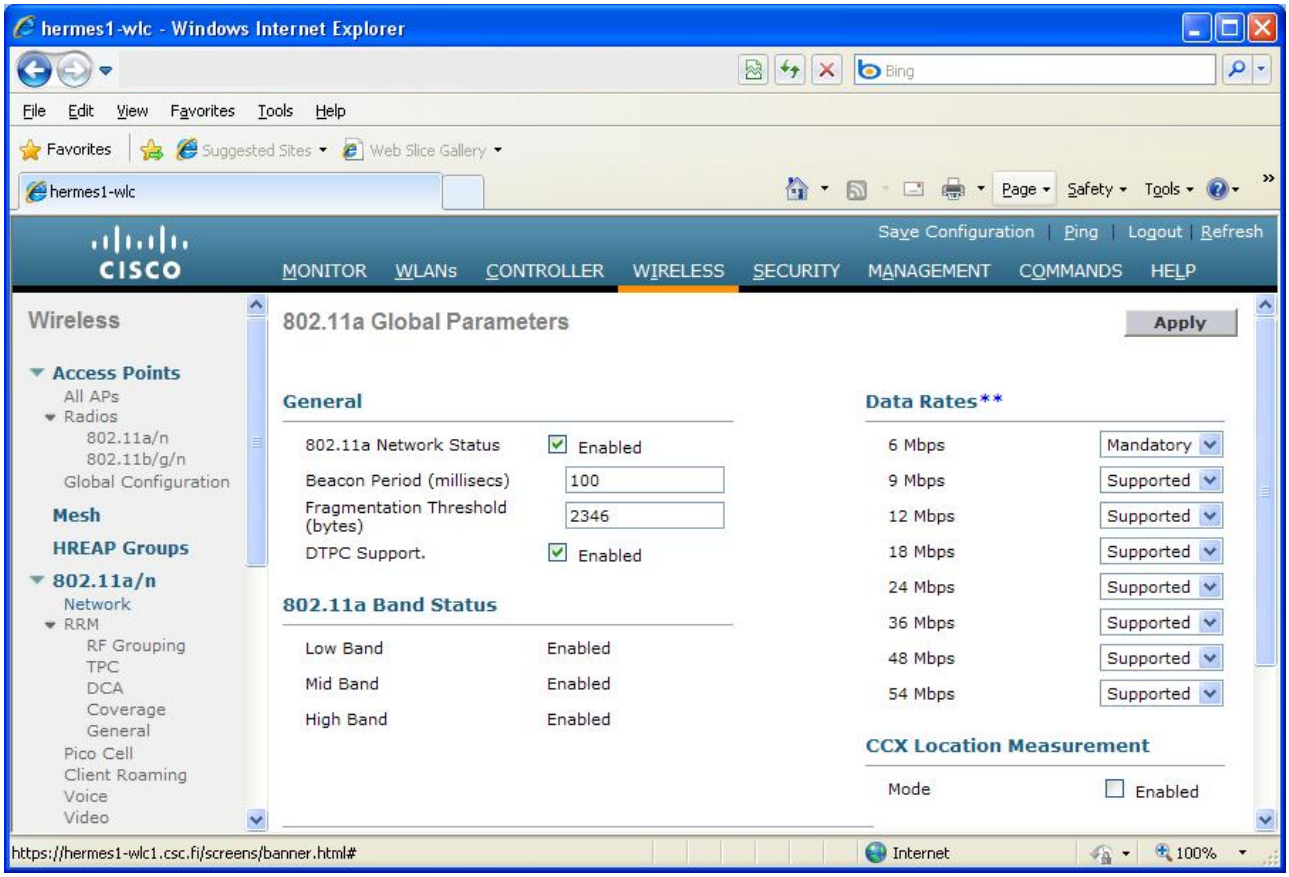

Figure 14. Defining network support for the 802.11a standard.

Next, define network support for the 802.11n standard. This must be done separately for the 2.4 GHz and 5 GHz frequencies. It might make sense to define support for the 802.11n standard for only the 5 GHz band; see the BPD “WLAN Network Planning and Setup”[10]. Open the definition view from the side bar by first selecting 802.11a/n | High throughput (802.11n) and, optionally, then select 802.11b/g/n | High throughput (802.11n) and fill in the information as depicted in Figure 15.

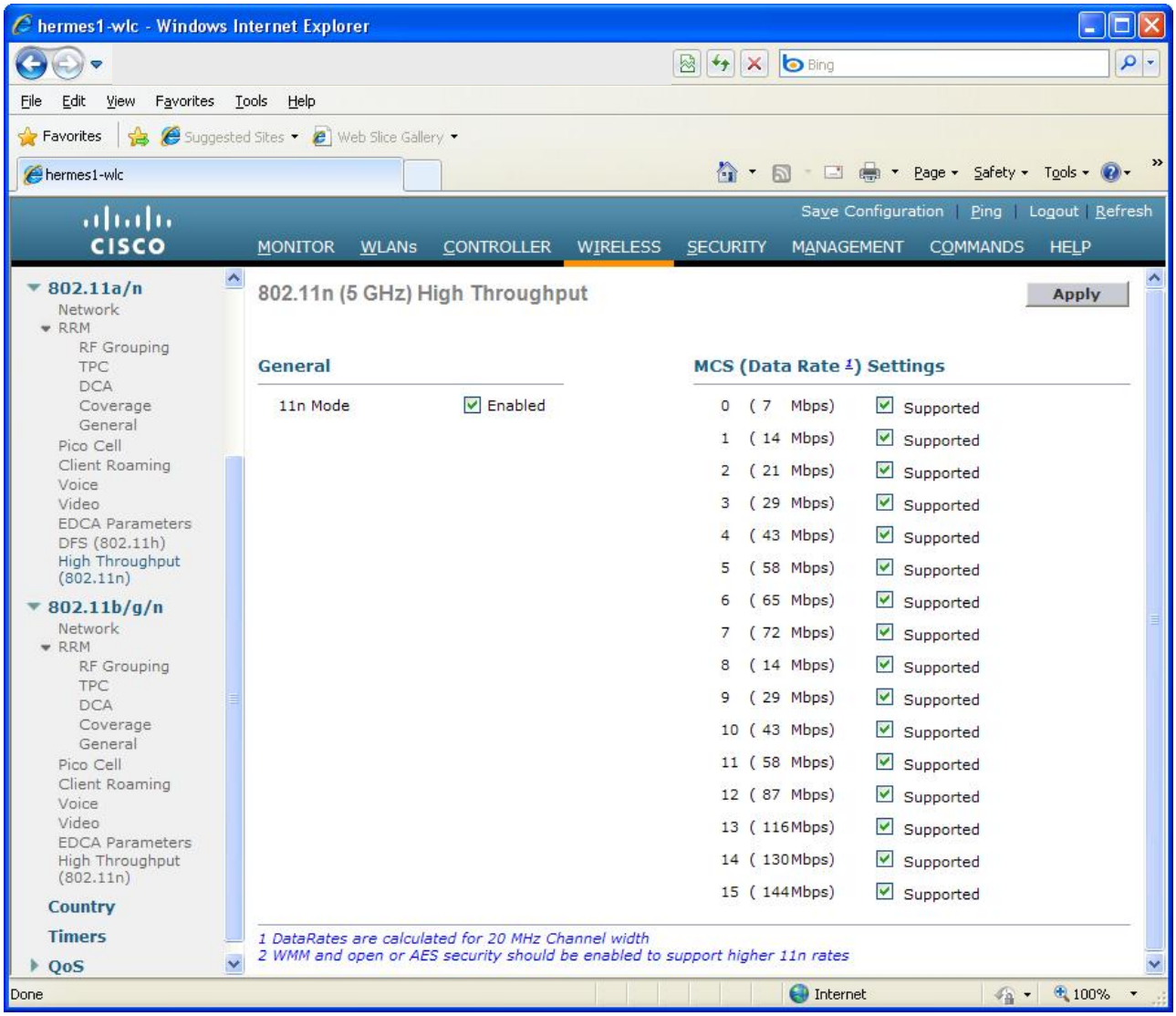

Figure 15. Defining network support for the 802.11n standard.

NOTE: You also need to define the wireless network properties before the network can be connected to. For more information, see Chapter **Defining a Wireless Network**.

Defining a RADIUS Server

You can begin defining an external RADIUS server by selecting SECURITY from the top bar and then selecting AAA | RADIUS | Authentication from the side bar. Define the server as depicted in Figure 16. Unlike in the Figure, Server Index (Priority) is 1 for the first server.

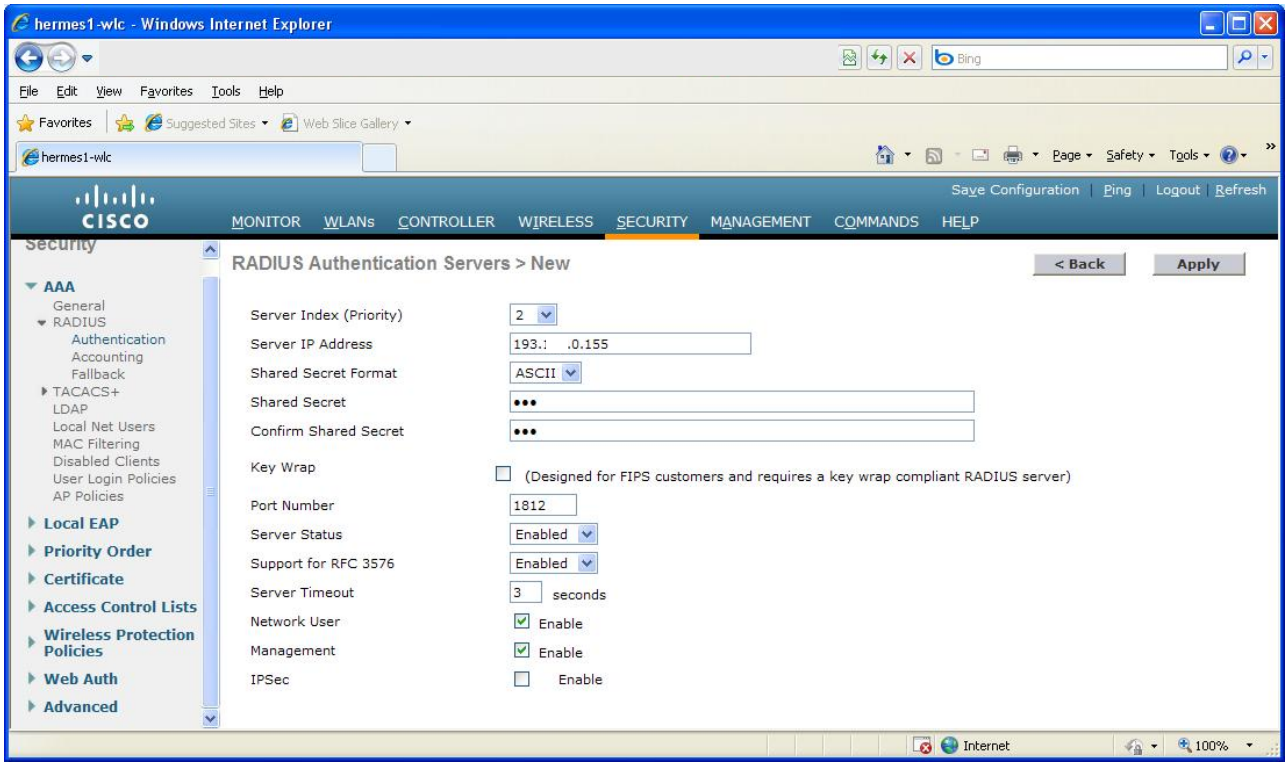

Figure 16. Defining a RADIUS server.

If necessary, also define an accounting server (by choosing Accounting from the side bar) and/or other RADIUS servers.

Defining a Wireless Network

To define a wireless network, open WLANs from the top bar and WLANs | WLANs from the side bar. Select Create New... and define the network. Figure 17 depicts the definition of the eduroam network. Finally, click the Apply button.

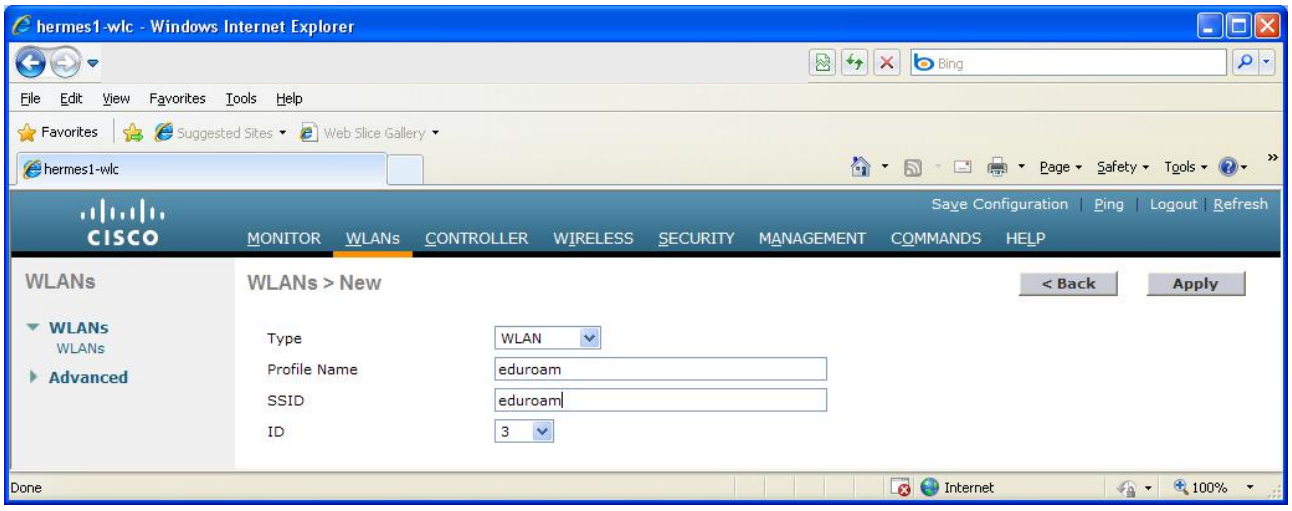

Figure 17. Defining the eduroam network.

Next, define the general network settings. See Figure 18 for eduroam as an example. If you wish to relay traffic of the defined network in a fixed local area network using a certain VLAN, choose the right dynamic interface in the Interface selection. This requires that the dynamic interface has first been defined at CONTROLLER – Interfaces.

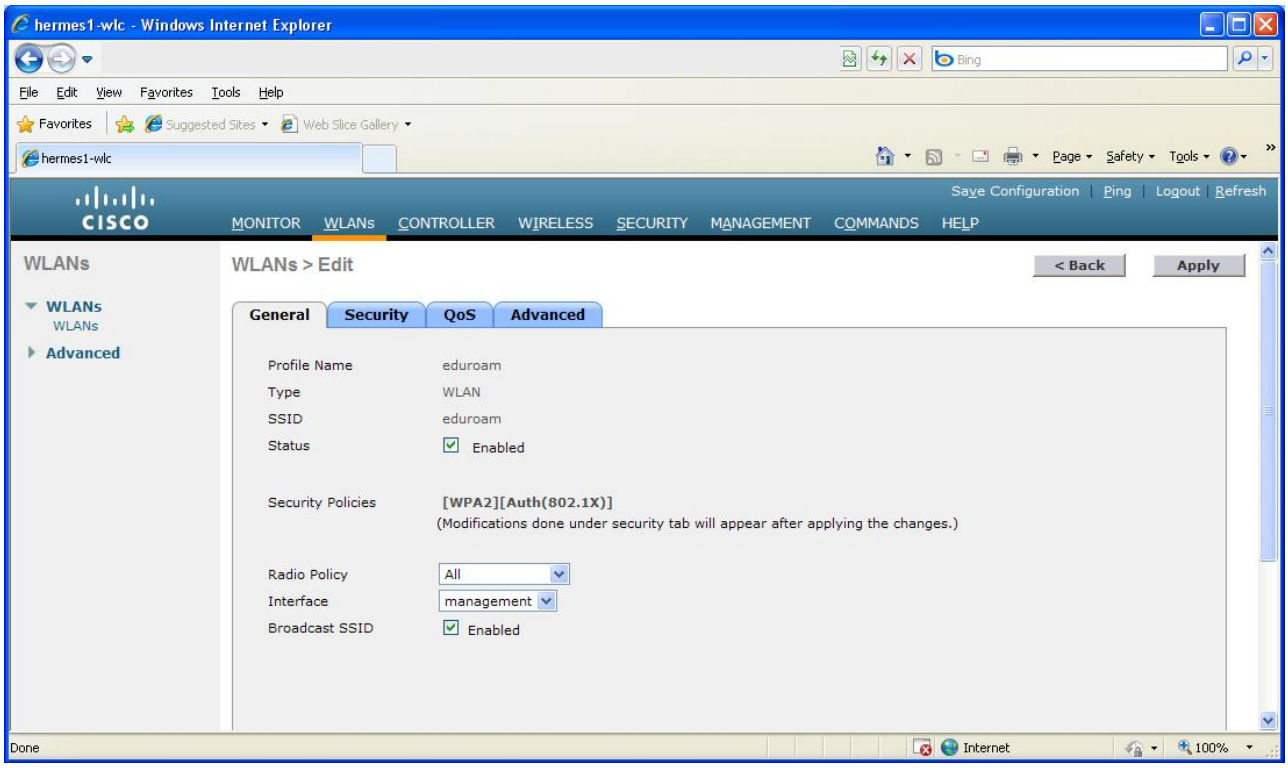

Figure 18. General settings of the eduroam network.

Next, define the security settings by choosing the Security tab. The eduroam network, using only WPA2-AES, is used as an example in Figure 19.

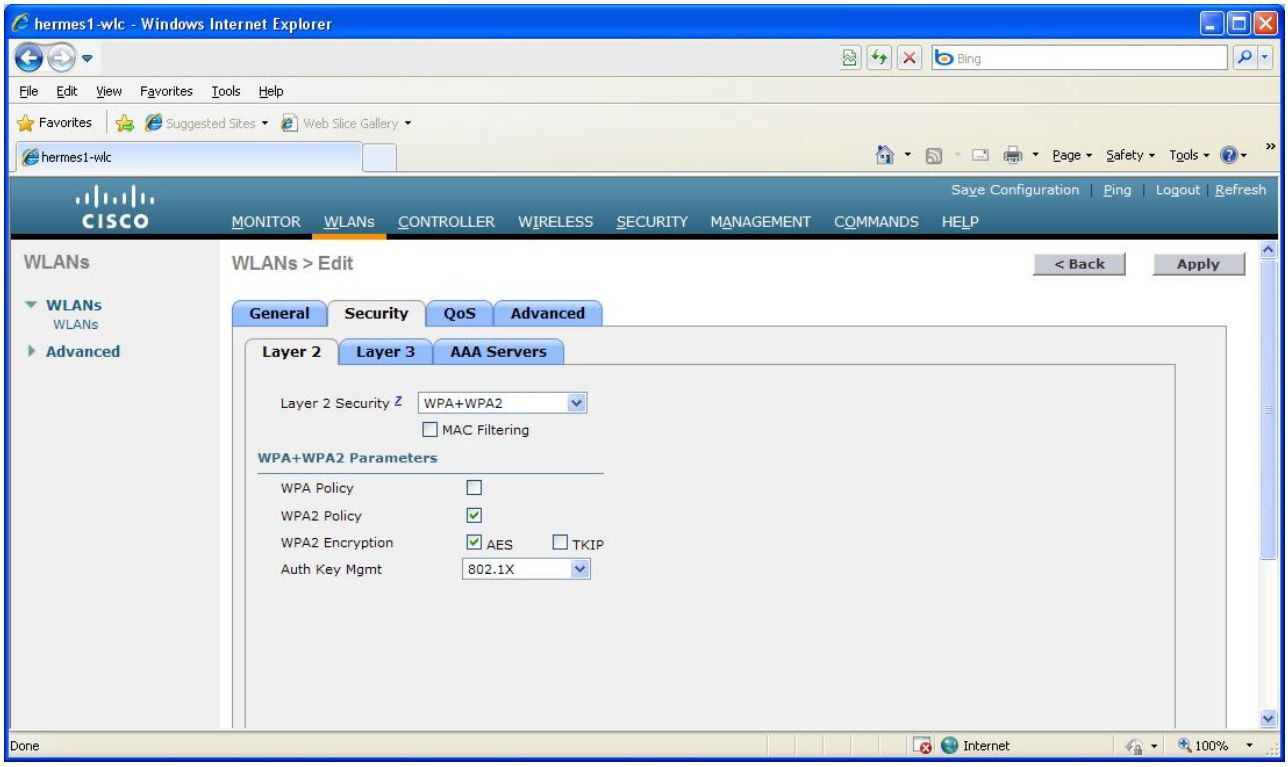

Figure 19. Security settings of the eduroam network.

Next, open the AAA Servers tab and select the defined RADIUS server(s). See Figure 20 for an example.

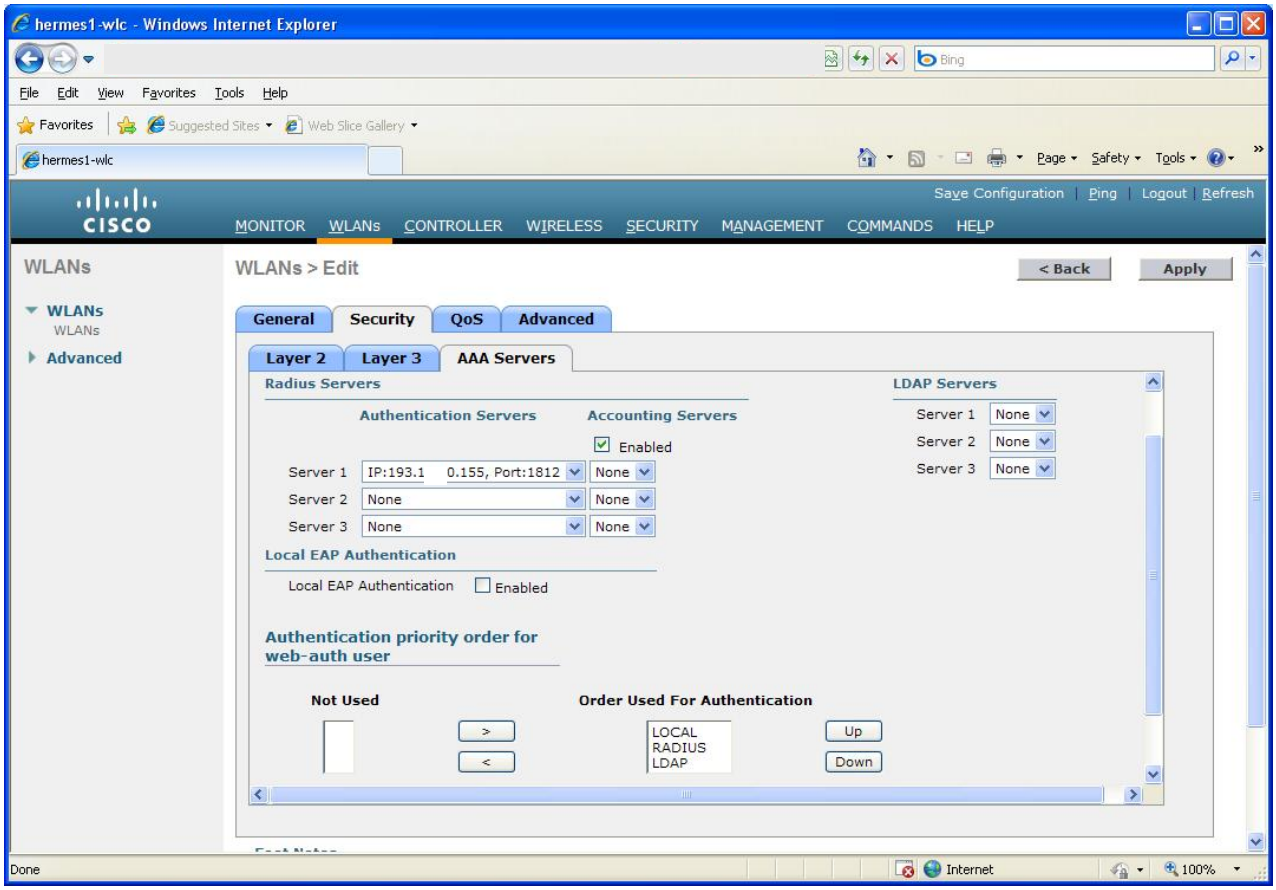

Figure 20. Connecting a RADIUS server to the access control of a certain network.

Next, select the QoS tab and ensure that either Allowed or Required has been selected for WMM Policy. Otherwise, the 802.11n standard speeds will not be supported by the network. See Figure 21 for an example.

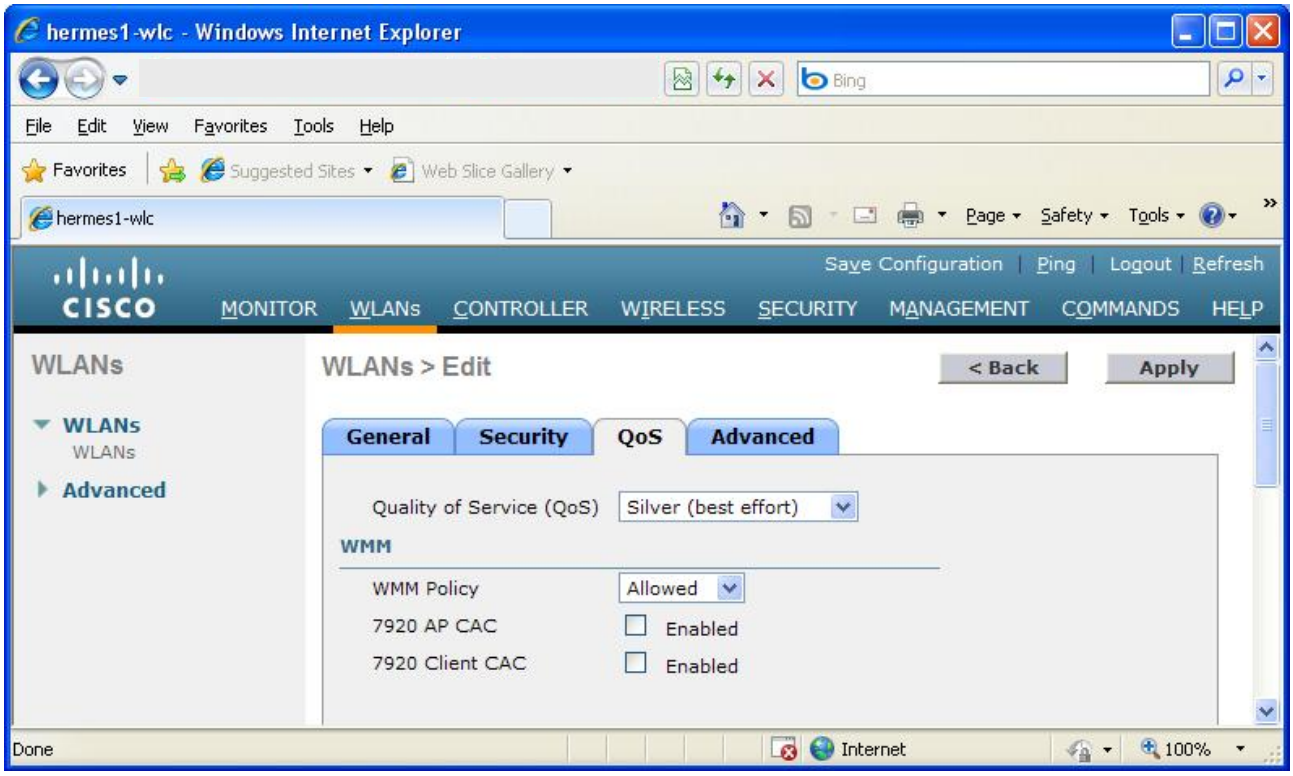

Figure 21. Network QoS settings.

Next, open the Advanced tab and edit the settings as depicted in Figure 22. By changing the value of the P2P Blocking Action parameter to Forward-UpStream, you can prevent direct traffic between WLAN clients in the network in accordance with the BPD "WLAN Information Security" [1]. MFP Client Protection has caused problems and is disabled. Finally, click the Apply button to save the network settings. ***NOTE: If there are VLANs in use in the network, also enable "Allow AAA Override".***

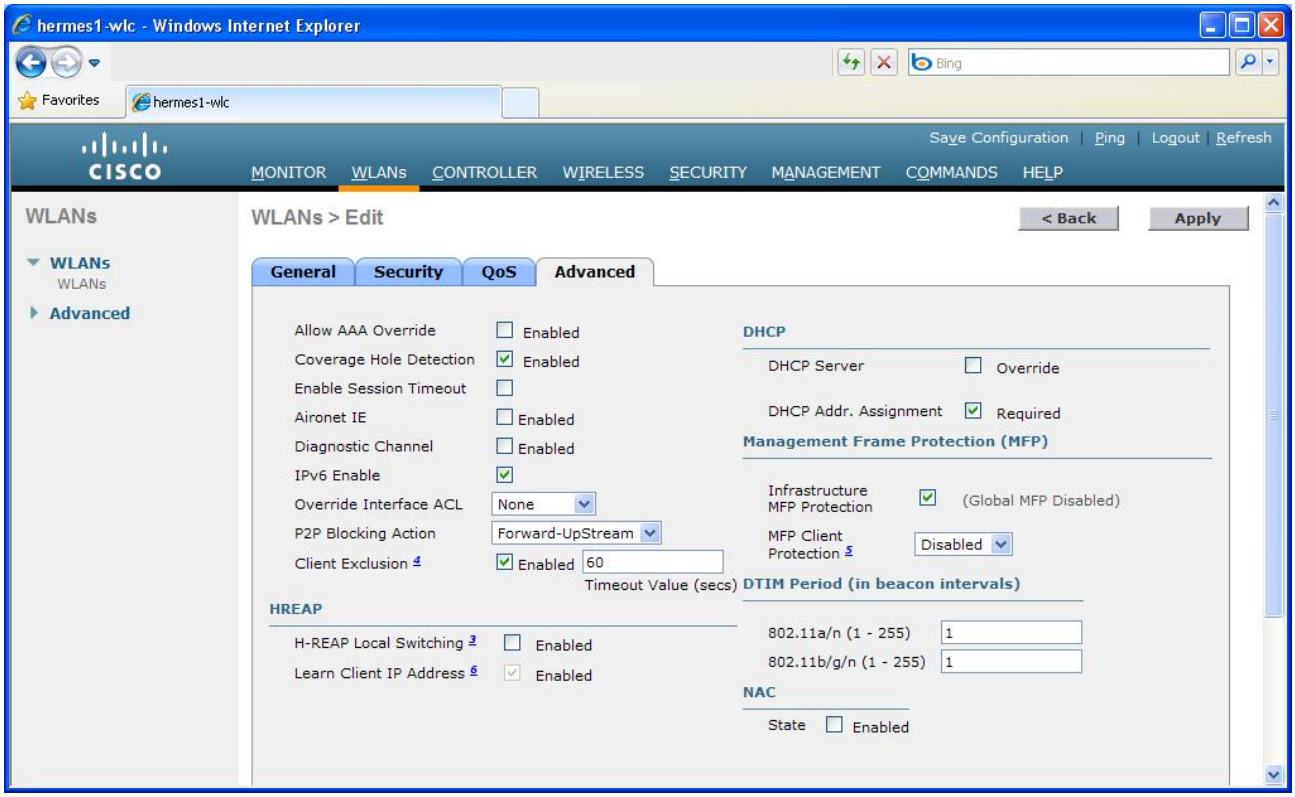

Figure 22. Other network settings.

In Figure 22, the value of Client Exclusion is set at 60s. As a default, Client Exclusion is set a little bit too tightly, so edit the settings at this point by selecting SECURITY from the top bar and Wireless Protection Policies | Client Exclusion Policies from the side bar. Clear all other checkboxes than “IP Theft or IP Reuse” as depicted in Figure 23. Then, click the Apply button.

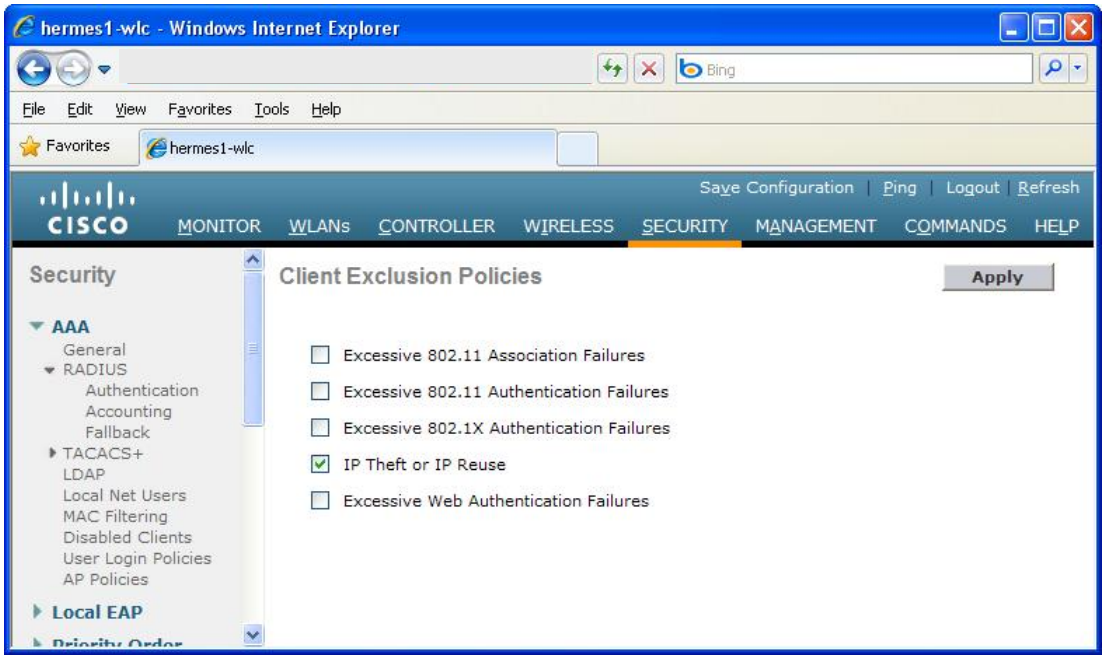

Figure 23. Defining the Client Exclusion Policies settings.

If you define a network other than the eduroam network and wish to use Web authentication for this wireless network instead of 802.1x authentication, the network security settings are defined in a different way. The Layer 2 settings on the Security tab are defined as depicted in Figure 24, and the Layer 3 settings as depicted in Figure 25. Additionally, define the maximum session time on the Advanced tab at Enable Session Timeout as depicted in Figure 26. The maximum session time must not be too short, as the user must re-authenticate when the time runs out, which leads to open sessions being disconnected. ***NOTE: If there are VLANs in use in the network, also enable "Allow AAA Override".***

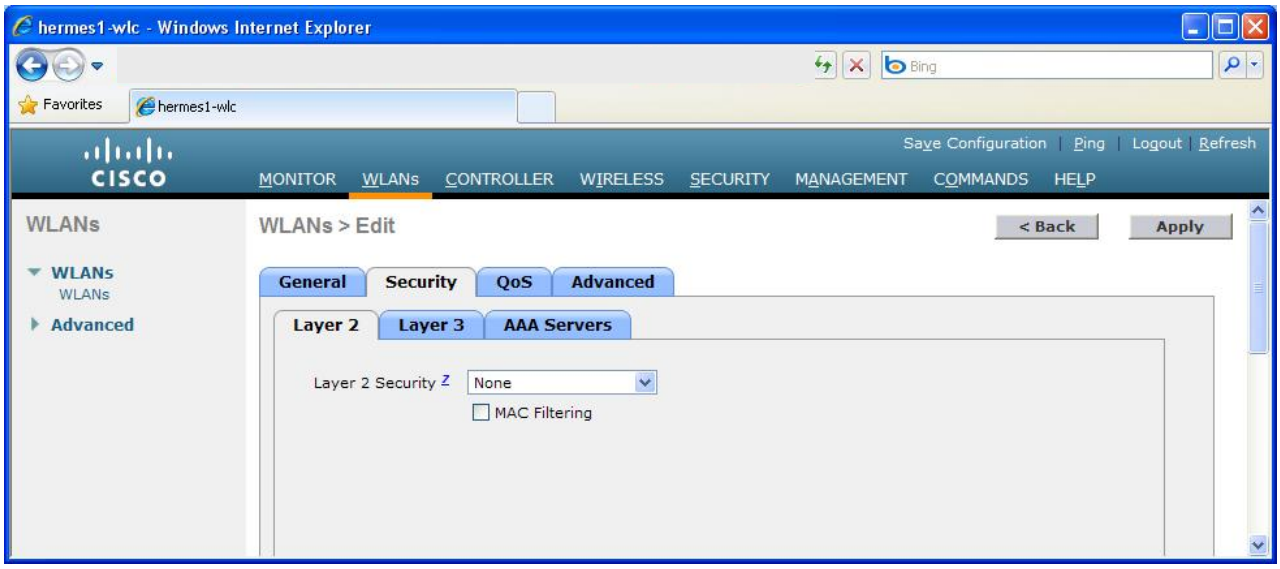

Figure 24. The Security Layer 2 settings of a Web-authenticated network.

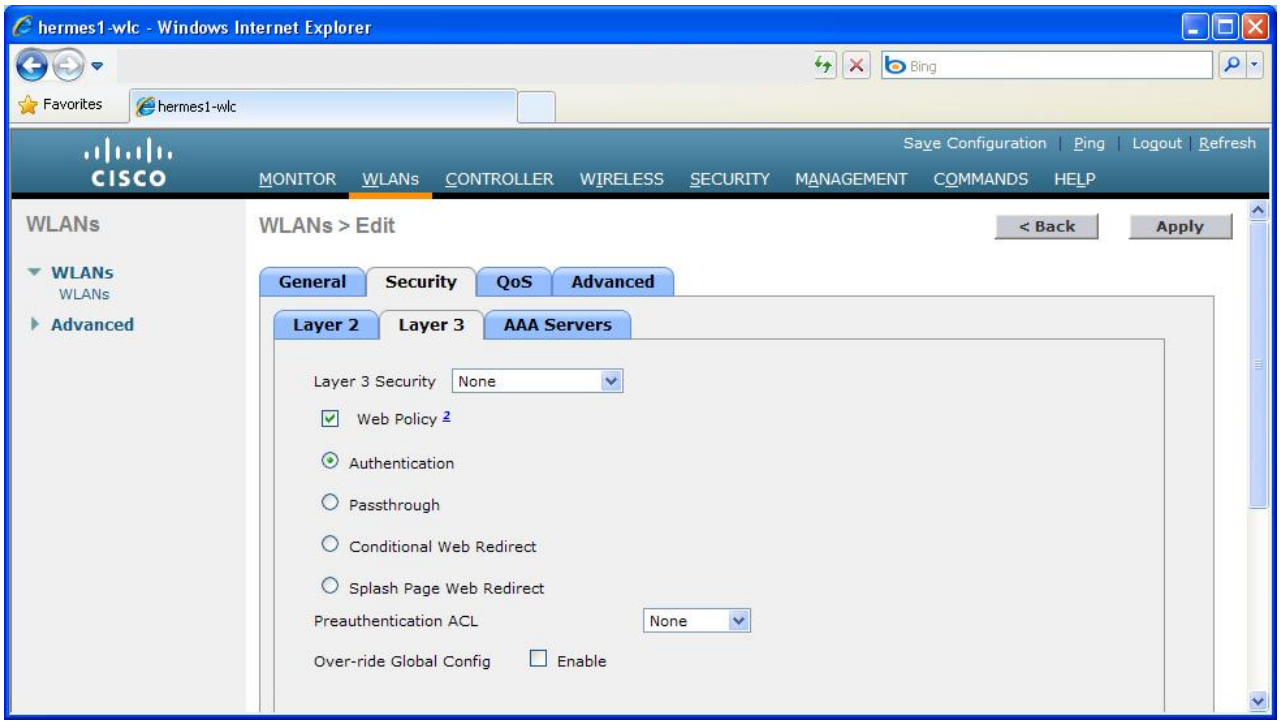

Figure 25. The Security Layer 3 settings of a Web-authenticated network.

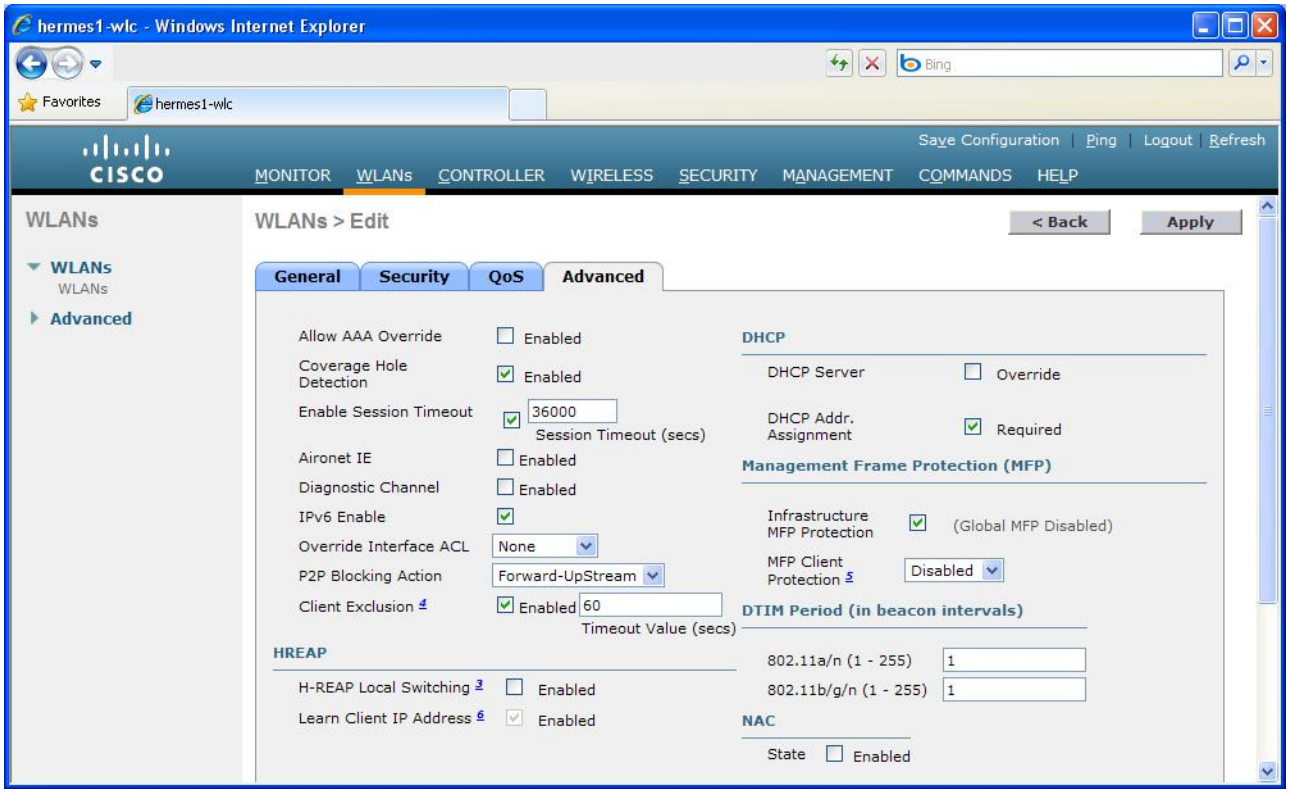

Figure 26. The other settings of a Web-authenticated network, the most important of which is Enable Session Timeout.

Finally, define a login page for the network using Web authentication. See

Figure 27 for an example of how the default page can be edited.

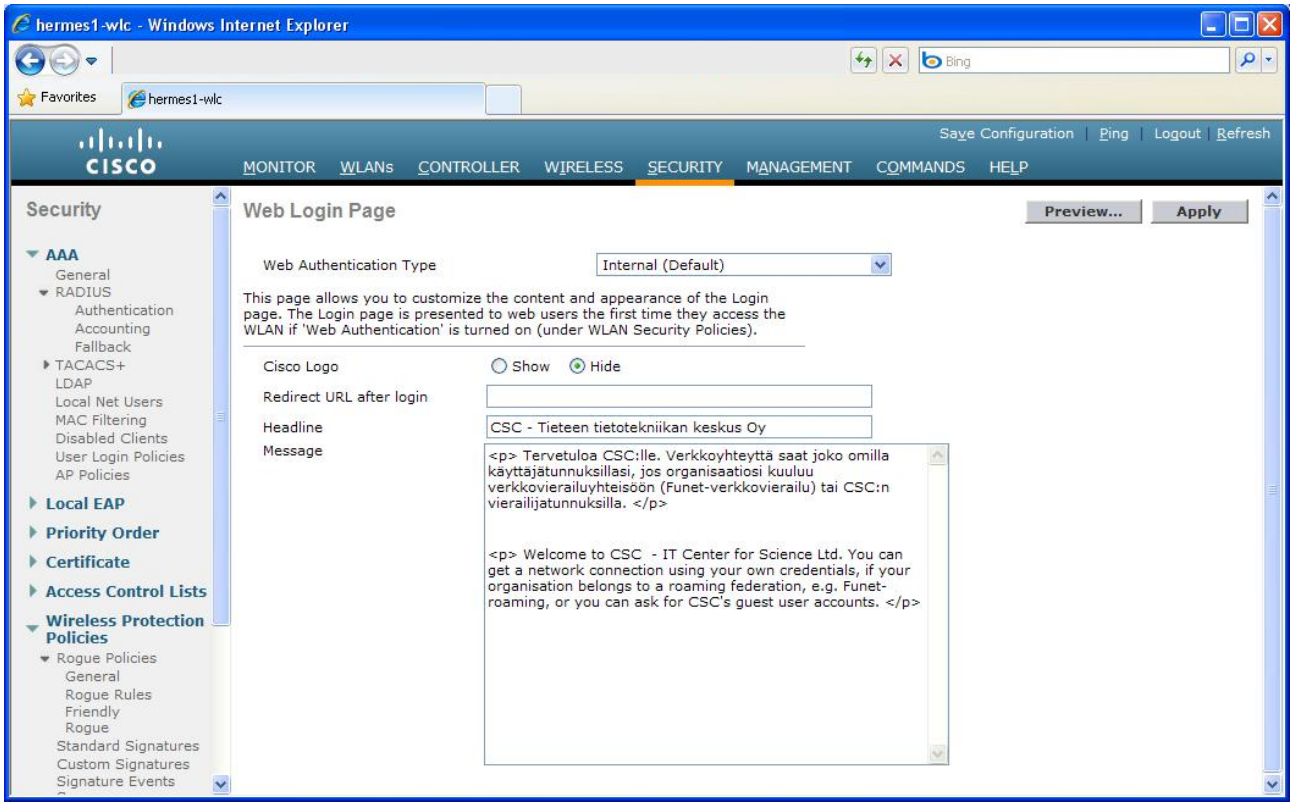

Figure 27. Defining the login page for a Web-authenticated network.

Enabling the Multicast Function

In order to save local area network resources, part of the message traffic can be handled with multicasting between the controller and the access points. The Multicast function must be defined in the controller settings. You can do this by first selecting CONTROLLER from the top bar and then General from the side bar. In the window that opens, define Multicast as the Ethernet Multicast Mode and set the multicast group address; see Figure 28. The multicast group address must be chosen in such a manner that no requests related to this address arrive from the Funet network. If necessary, contact Funet for assistance. Another alternative to protect the multicast transmissions between the controller and the access points would be to define 1 as the Time-To-Live (TTL) value, but this parameter cannot be defined in the controller.

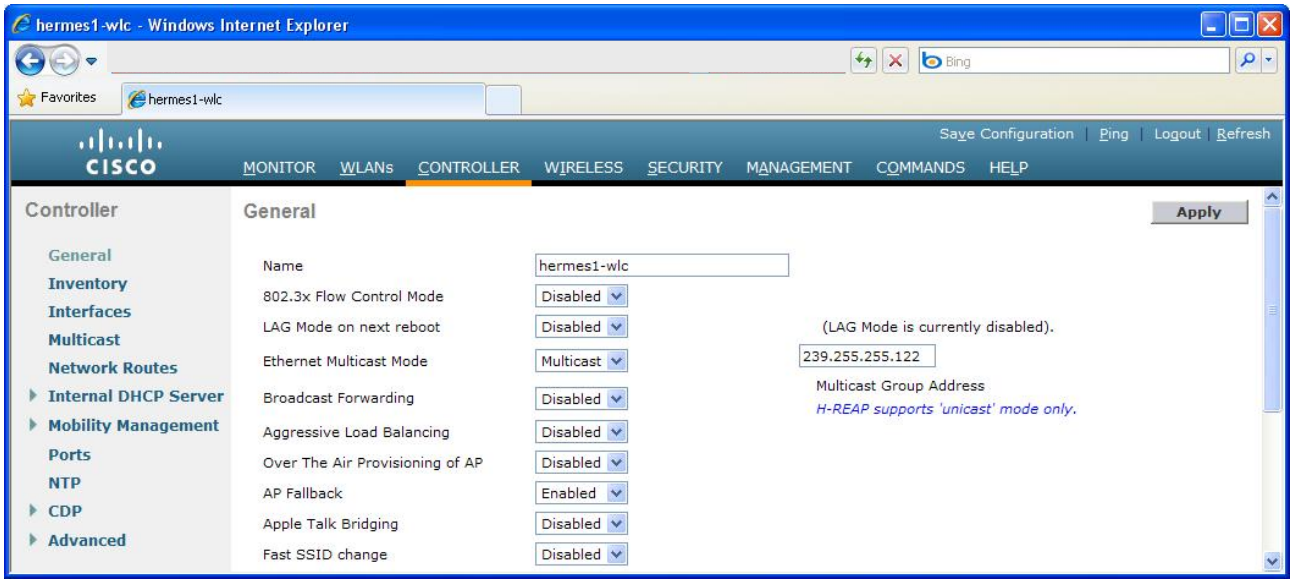

Figure 28. Enabling the Multicast function for traffic between the controller and the access points.

Next, move to the controller's Multicast settings by selecting Multicast from the same side bar. In the window that opens, enable IGMP (Internet Group Management Protocol) Snooping for the multicast, and set the timeout value as something between 30 s and 300 s, for example 60 s. See Figure 29 for an example.

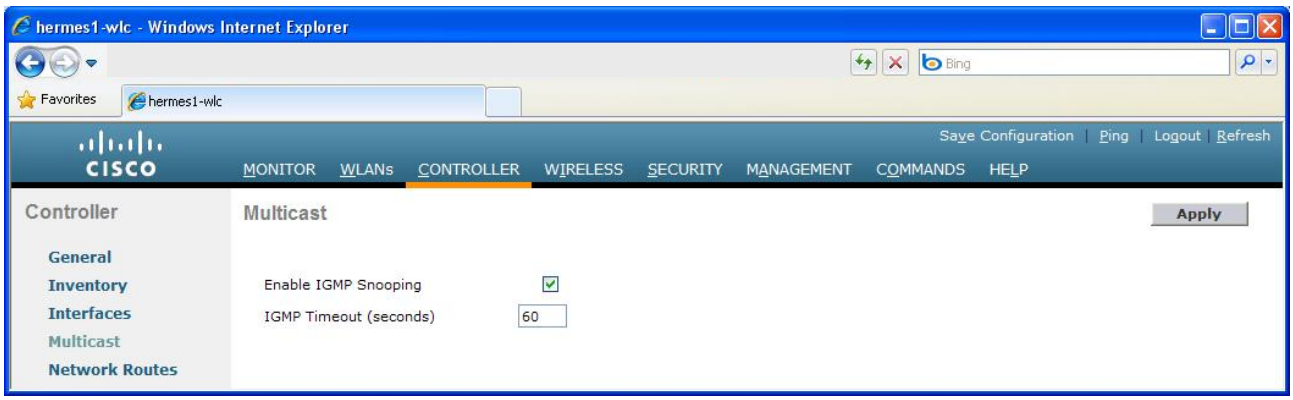

Figure 29. Multicast IGMP settings.

Installing the Certificate

A certificate must be installed on the controller for Web authentication to work logically and securely. Without the certificate, users are displayed a warning in the browser instead of the login page when they attempt to access the network. The certificate must be installed so that the login page will open for the users right away.
NOTE: When requesting the certificate, remember that the name in the certificate's CN (Common Name) field must be the same as the name corresponding to the IP address of the controller's virtual interface on the DNS server.

Once the certificate has been acquired, note that it cannot be moved to the controller without password protection. If the certificate file is not password-protected, you can protect it with the OpenSSL program, for example using the following command:

```
OpenSSL> pkcs12 -export -in myname.pem -inkey mykey.pem -out CA.p12 -clcerts -
passin pass:mypasswd -passout pass:mypasswd
```

Next, set the certificate's password and download the certificate using the CLI (Command Line Interface):

```
(Cisco Controller) >transfer download certpassword check123
Setting password to <check123>
(Cisco Controller) >transfer download start
Mode..................................................... TFTP
Data Type................................................ Site Cert
TFTP Server IP........................................... <TFTP server IP>
TFTP Packet Timeout...................................... 6
TFTP Max Retries......................................... 10
TFTP Path................................................ /
TFTP Filename............................................ myname.pem

This may take some time.
Are you sure you want to start? (y/N) y

TFTP Webauth cert transfer starting.

TFTP receive complete... Installing Certificate.

Certificate installed.
  Reboot the switch to use new certificate.
```

Reboot the controller and then begin configuring the virtual interface by selecting CONTROLLER from the top bar, then selecting Interfaces from the side bar, and finally clicking the *virtual* interface. In the window that opens, ensure that the IP address of the virtual interface is correct, and enter the name in the certificate's CN field in the DNS Host Name field; see Figure 30.

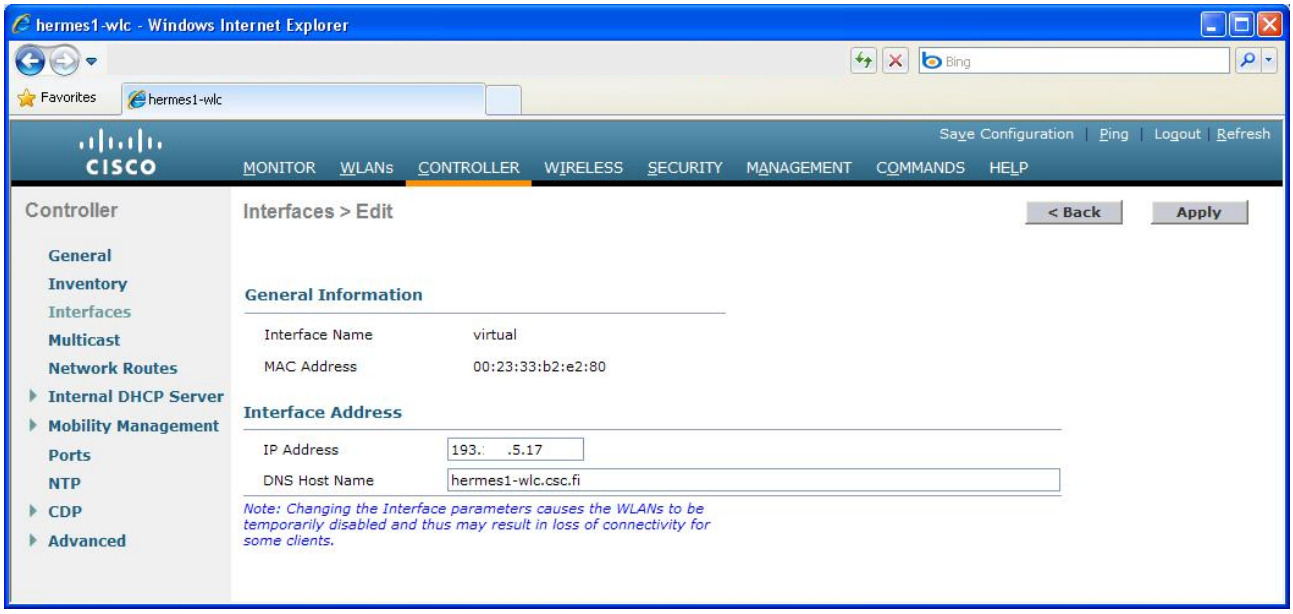

Figure 30. Defining the parameters of the virtual interface.

Connecting Several Controllers into One Mobility Group

If an organisation uses several controllers, we recommend connecting them together. This way, users can move from one access point to another without disruptions, retaining their open sessions; i.e., their IP address remains unchanged. By defining the controllers into the same mobility group, you can implement seamless transition between access points.

Before defining the mobility group, check that you can ping all controllers by selecting ping from the upper right-hand corner and entering the IP address of each controller in turn in the window that opens. At this stage, also compile a list of the IP and MAC addresses of the management interfaces of all controllers. You can find them out by selecting CONTROLLER from the top bar and Mobility Management | Mobility Groups from the side bar.

Also note that the controllers that are connected together as a mobility group must have the same IP address in their virtual interfaces (CONTROLLER – Interfaces – virtual).

If there is a firewall between controllers, check its settings. Ports 16666, 16667, 12222 and 12223, IP protocol 50 and 97, and UPD port 500 must be open. If you are using IPsec, check the settings separately.

Next, define a name for the group by selecting CONTROLLER from the top bar and General from the side bar. Enter the desired name in the Default Mobility Domain Name field if you do not wish to use the name you chose during the first start-up.

The following changes must be made on all controllers belonging to the group. Define the other controllers by selecting CONTROLLER from the top bar and Mobility Management | Mobility Groups from the side bar. At this point, the window that opens only shows the information of the controller in question, see Figure 31.

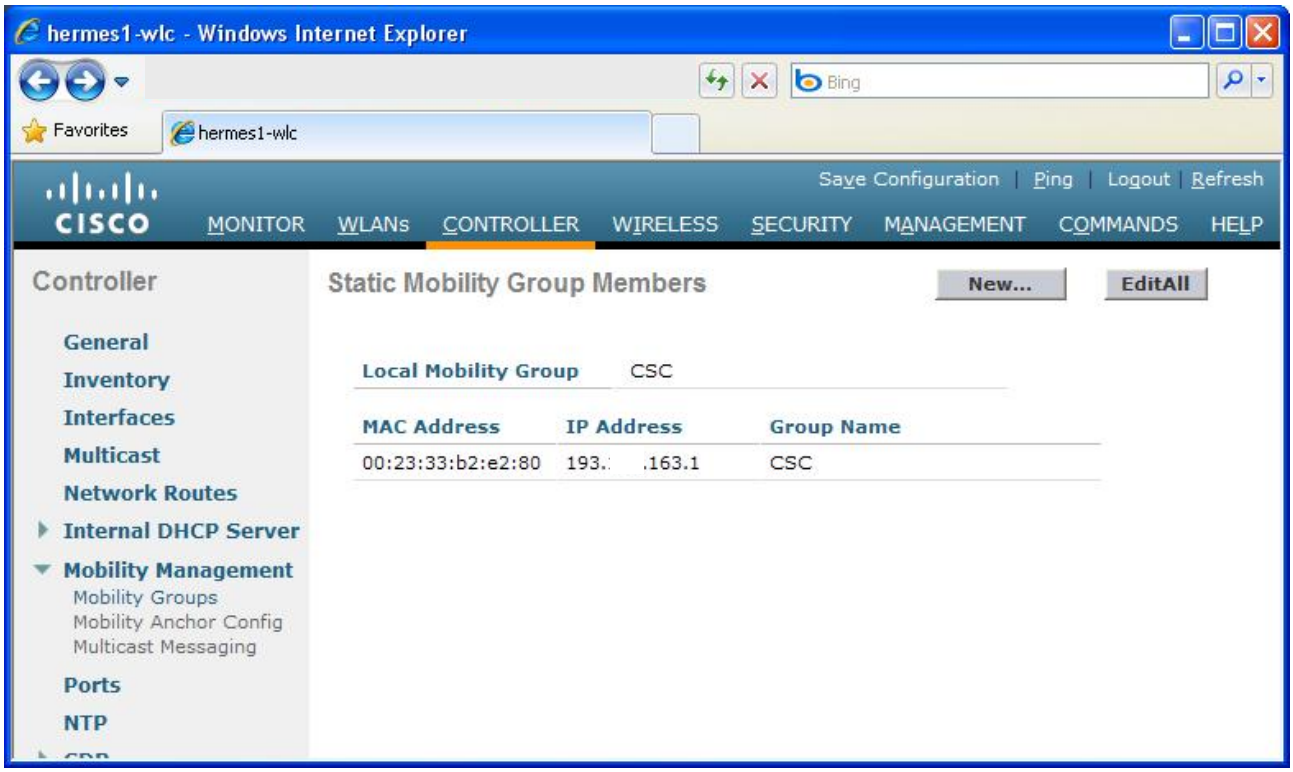

Figure 31. Listing of the controllers belonging to the mobility group.

Next, click the New... button and add the information of the second controller of your organisation. Define the IP address and MAC address of the management interface, see Figure 32. Finally, click the Apply button. Add the information of your organisation's other controllers in the same way. Next, define the controller information in your organisation's other controllers in the same way. You can check the outcome for each controller by listing the information of other controllers defined for the controller by clicking the EditAll button depicted in Figure 31.

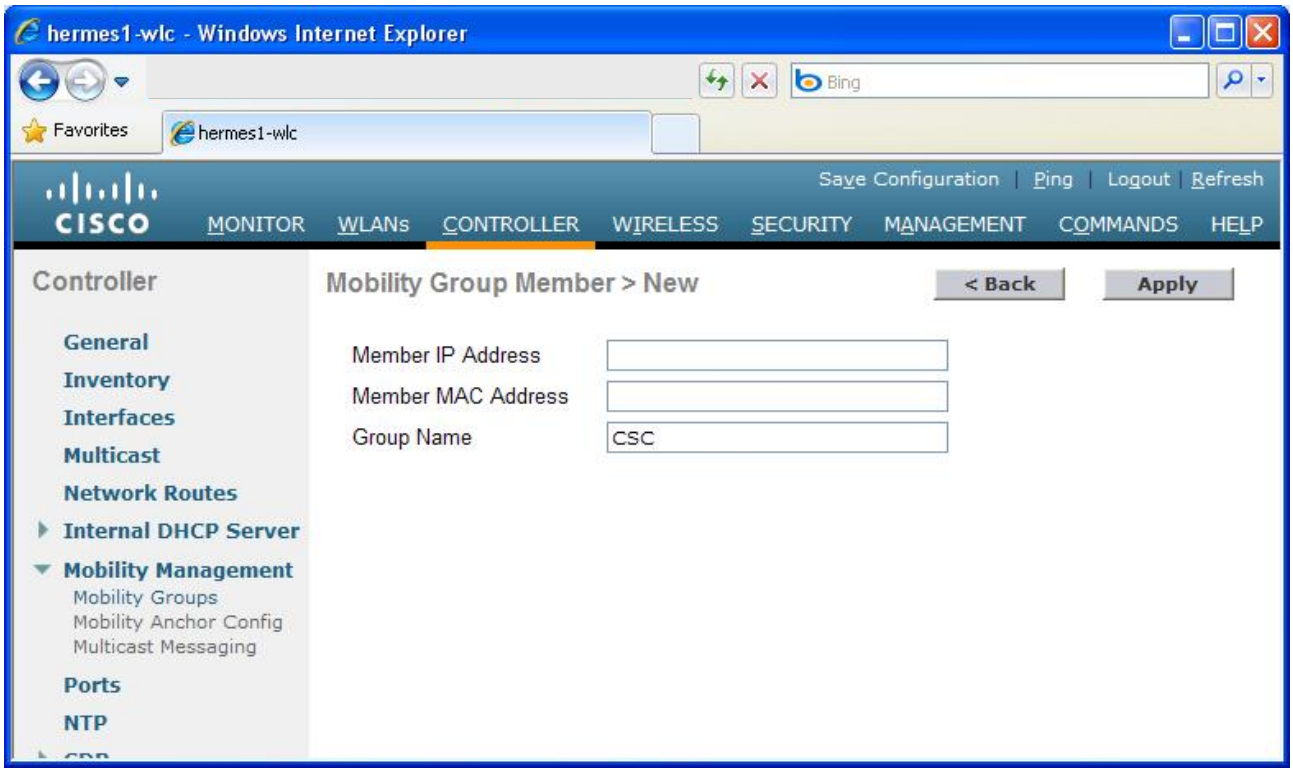

Figure 32. Connecting a controller to a mobility group.

Next, set up the multicast function for communication between the controllers. Select CONTROLLER from the top bar and Mobility Management | Multicast messaging from the side bar. Tick the Enable Multicast Messaging checkbox in the window that opens, and enter the IP address chosen for the group.

Configuration for Loanable Access Points (Optional)

When the network is being built or updated, it may be sensible to acquire a couple of extra access points that can be used as loanable access points. These access points can be used to quickly and handily set up a WLAN network from almost any LAN socket. For example, you can bring the loanable access points along when a course or conference is arranged somewhere else than your organisation's own premises. The access points can be configured so that they automatically contact your organisation's controller and then provide the same networks as all the other access points connected to the controller. The firewall rules of the local network may in principle prevent connection to the controller, but in practice, traffic from the network to the Internet is usually allowed in a flexible manner.

Access Points intended for loan use must always be connected to the network in your home organisation before they can be used elsewhere. At this stage, ensure that the access points find one of your controllers as described in the beginning of Chapter **Connecting Access Points to the Network and their Configuration**. The wireless networks provided by the loanable access points and the settings of their air interfaces will always

be the same as the networks and settings of the other access points connected to the controller. Once a access point has found a controller, it will in principle always have the controller's address in store, but we recommend defining the address(es) of the organisation's controller(s) on the loanable access points just in case. You can do this by selecting WIRELESS from the top bar and Access Points | All APs from the side bar. In the table that opens, click the access point that just connected and enter a descriptive name for it. Next, open the High Availability tab and define your organisation's controllers. See Figure 33 for an example. Finally, click the Apply button.

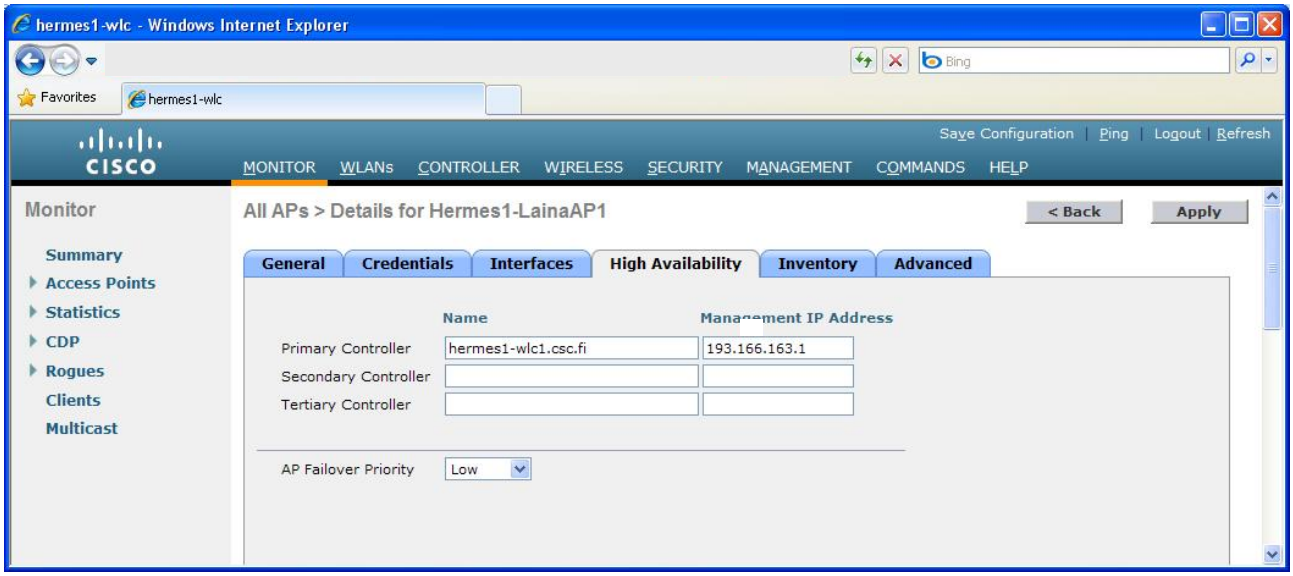

Figure 33. We recommend defining your own controllers and their IP addresses on the loanable access points.

In order for the loanable access points to be able to connect to the controller from any network, you also need to edit the Access Control List. The CAPWAP protocol is currently used for the connection requests of access points, and you need to open UDP ports 5246 and 5247 for this protocol. CAPWAP is a standardised protocol, and it is the successor of the older, Cisco's own LWAPP protocol. Figure 34 depicts how to open port 5246 for the CAPWAP protocol in the Access Control List. Open port 5247 in the same way. For more information on Access Control Lists, see Chapter **Defining an Access Control List**.

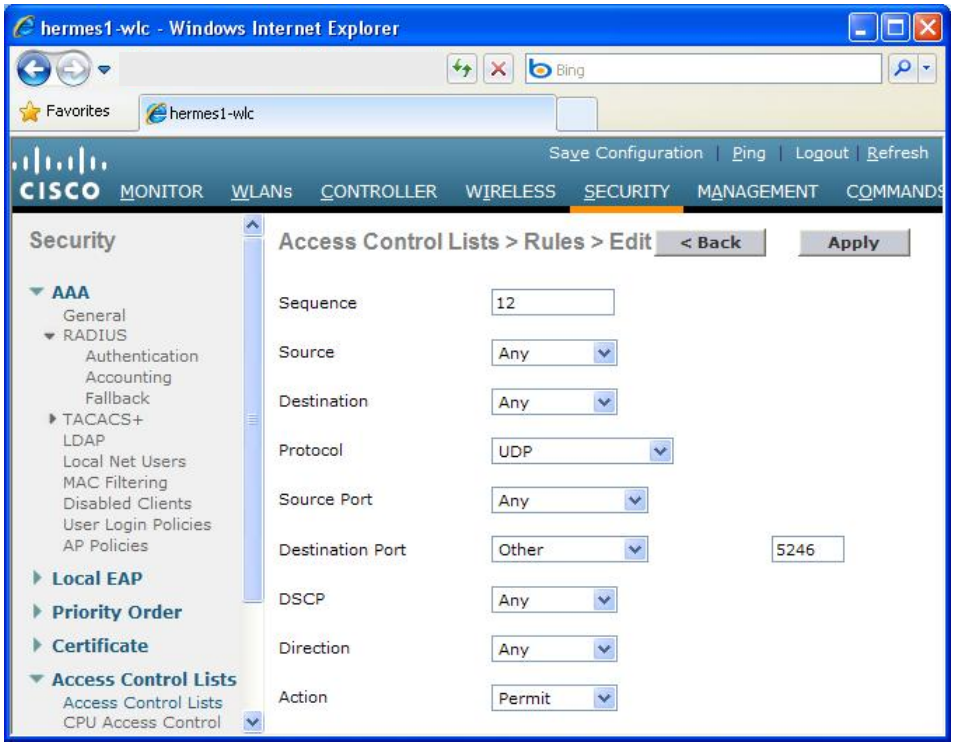

Figure 34. Opening a port for CAPWAP protocol messages.

NOTE: When you open the ports for the CAPWAP protocol in the ACL as depicted above, you cannot prevent the connection of any access point to the controller in advance. If an unknown access point finds out about the organisation's controller's IP address for some reason, the access point can connect to the controller unnoticeably and start providing the organisation's wireless networks in its coverage area. Opening the CAPWAP protocol includes therefore a small information security risk, but many Funet member organisations have accepted this risk. Otherwise, before the access point could connect to the controller, you would have to know what IP address was assigned to the loanable access point by the local network, and this would cause a lot of extra work for IT support.

Appendix 2 – HP Controller Configuration

This document describes the configuration of an HP WLAN controller. The screenshots are from a Procurve MSM760 controller, which means that the command windows of different models may look different.

General

Once an IP address has been specified for the controller and the username has been defined, you can begin configuring the controller with a browser.

In the HP WLAN controller, the WLAN network configuration steps are as follows: First, create a Virtual Service Community (VSC) and define its SSID, authentication method, encryption, etc. Next, arrange the access points connected to the controller into groups. You can create the desired network by defining a certain VSC for a certain group. However, we recommend starting by setting up a RADIUS server for handling the authentication of the WLAN network.

Defining a RADIUS Server

HP Procurve includes an integrated RADIUS server, but as it is designed for small networks, we do not recommend using it in campus networks. Begin setting up an external RADIUS server by selecting Service Controller >> Authentication > RADIUS profiles, and the window depicted in Figure 35 opens.

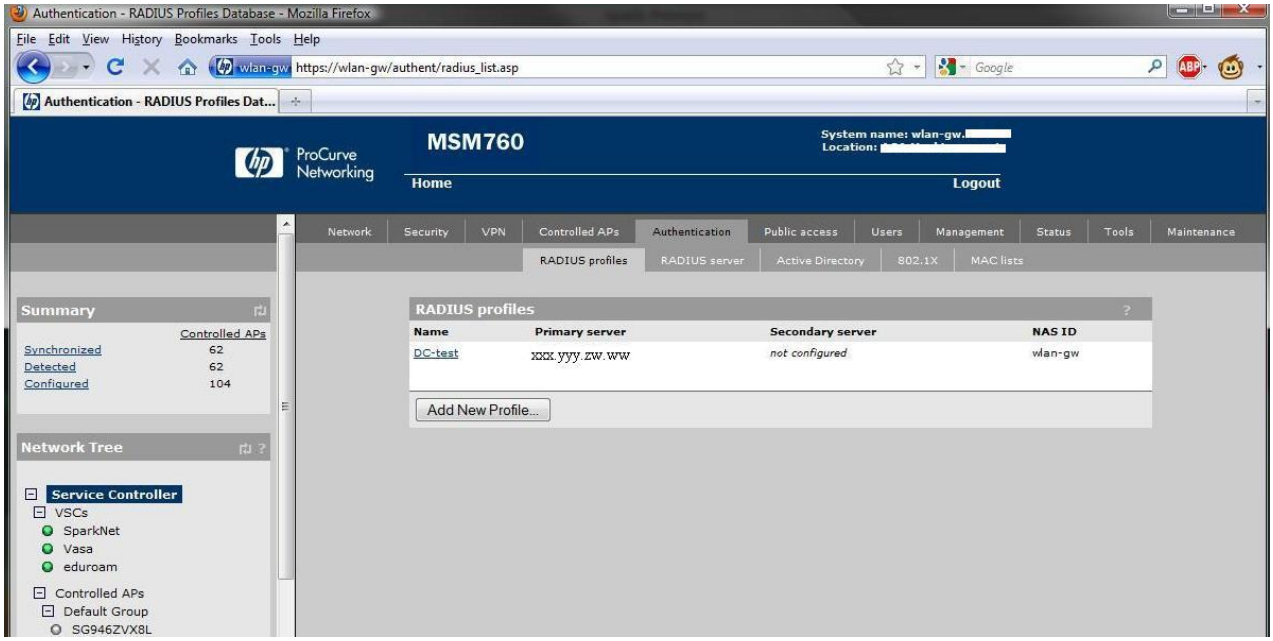

Figure 35. Summary window for RADIUS profiles/servers.

Click the Add New Profile… button and define the RADIUS server as depicted in Figure 36. The options in the Authentication method pull-down menu will not be taken into consideration, if 802.1x authentication is used for the WLAN network, as is the case with eduroam, for example. You can leave the Authentication realms field empty, unless you wish to use different servers for the authentication of different realms.

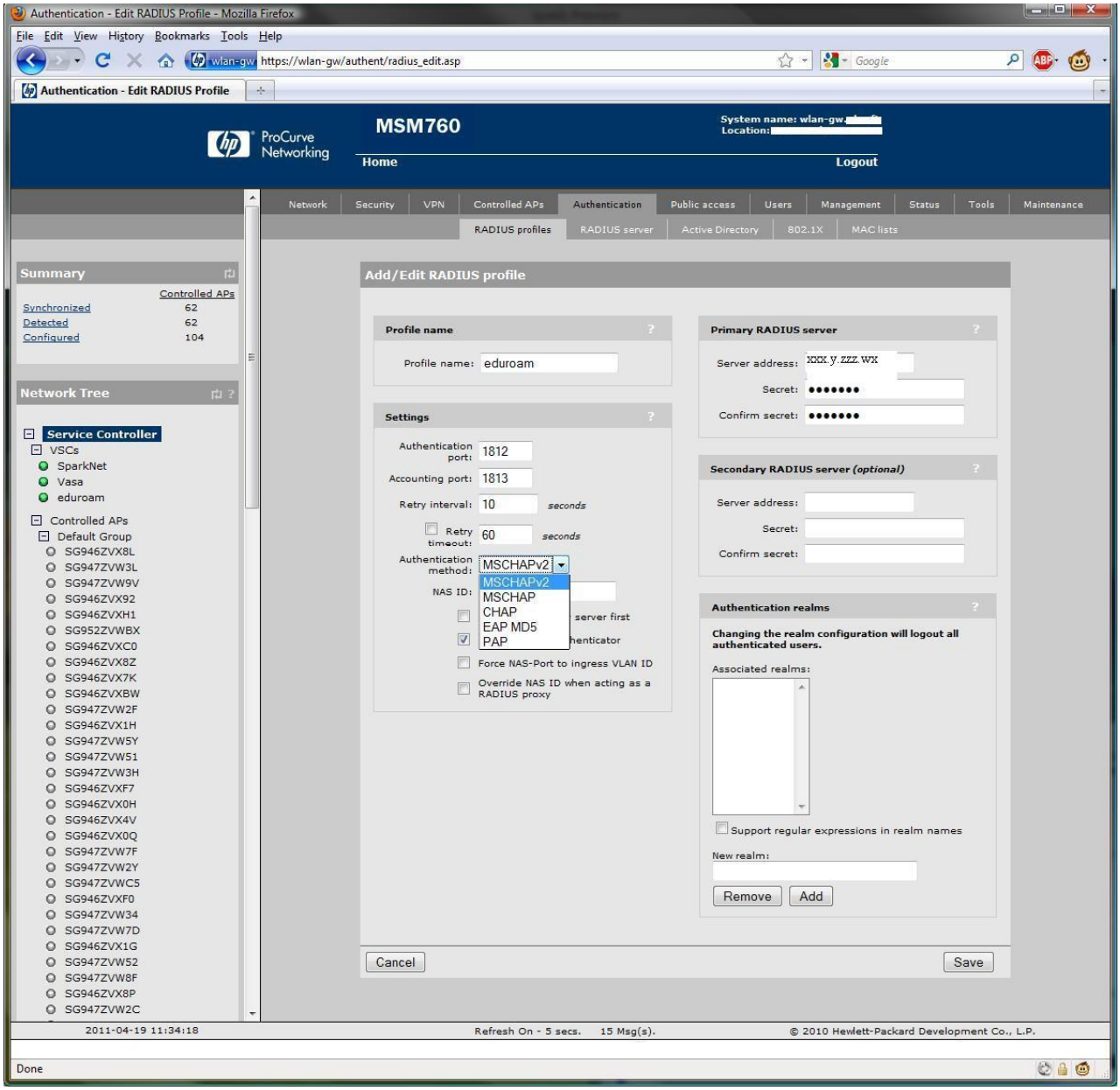

Figure 36. Defining a RADIUS profile, or a RADIUS server.

Defining a Virtual Service Community (VSC)

Begin setting up a VSC server by selecting Service Controller >> Overview > VCS profiles, and the window depicted in Figure 37 opens.

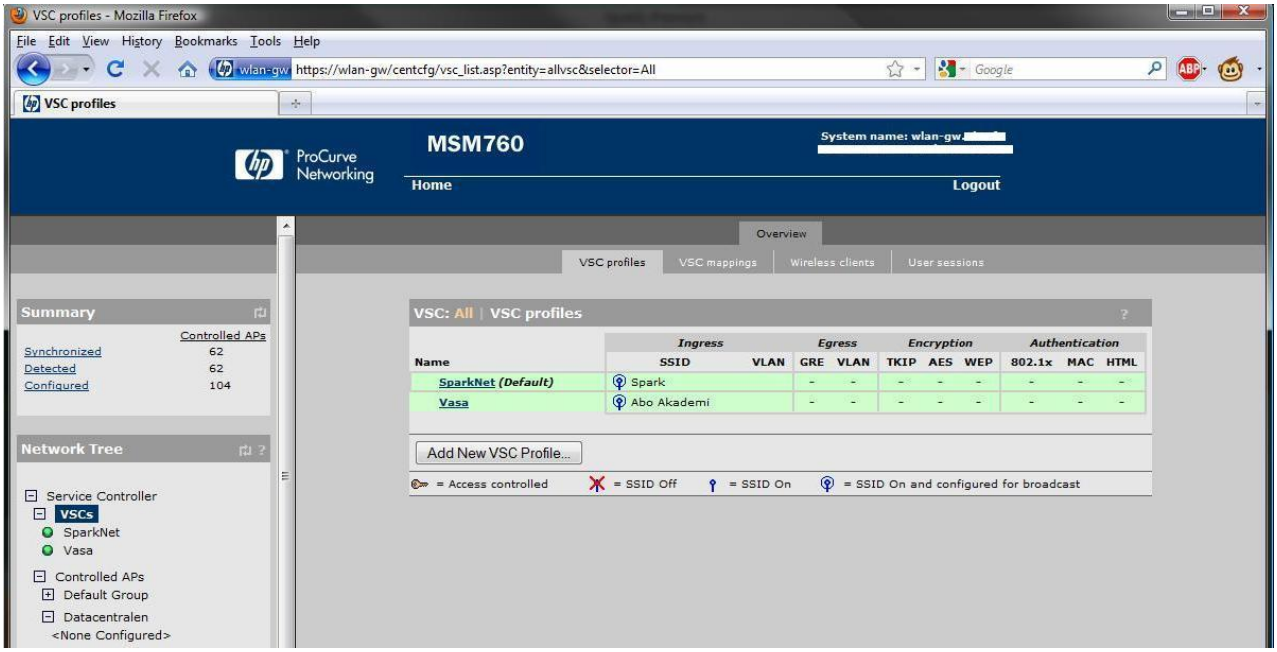

Figure 37. Summary of the VSC profiles.

Click the Add New VSC Profile... button shown in the Figure. The window depicted in Figure 38 opens.

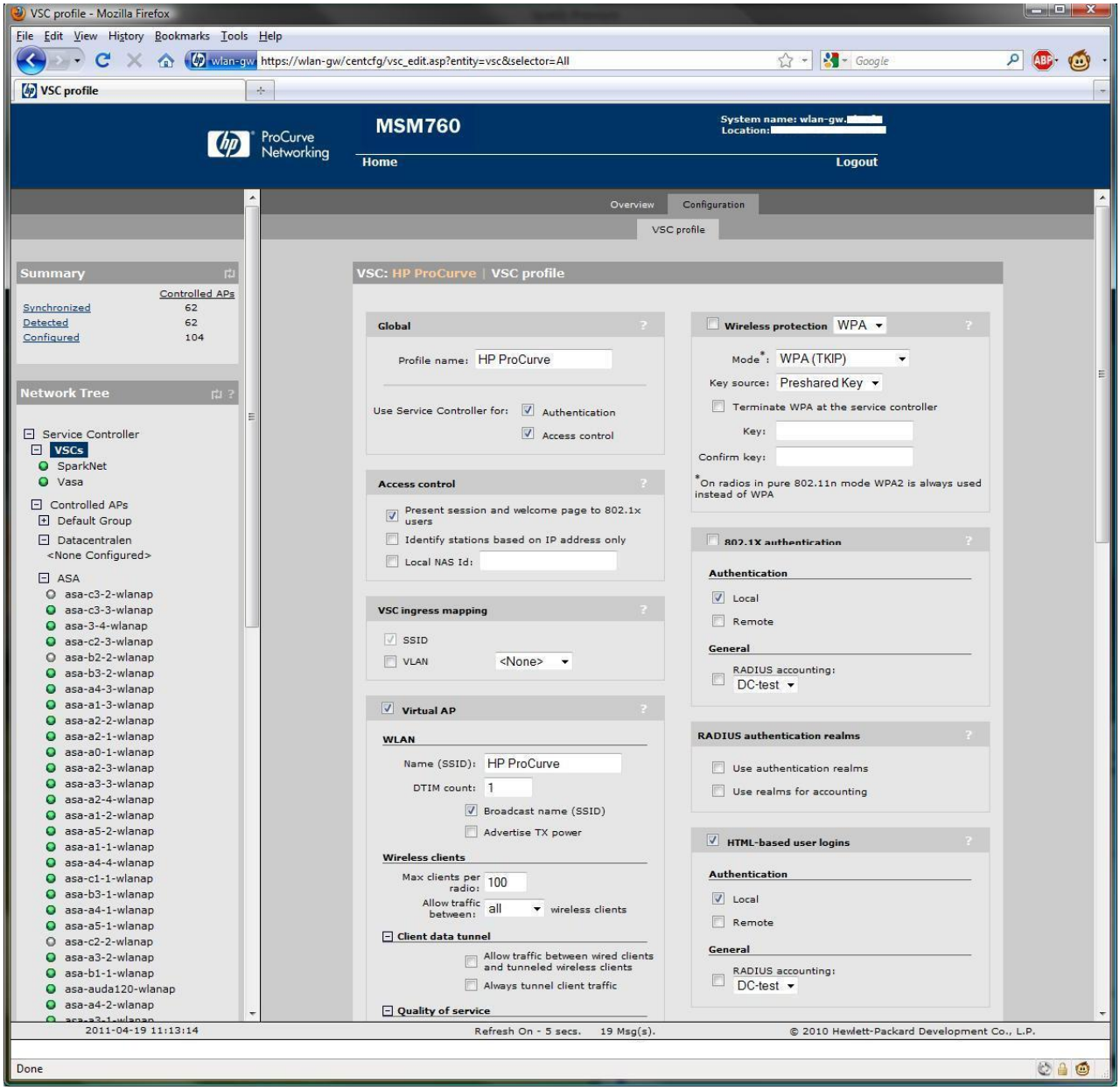

Figure 38. Defining a VSC profile.

If you wish to define a network with Web authentication, see Figure 39 for an example. First, define a suitable name for the profile in the left column, and clear the Authentication and Access control checkboxes. Next, select a suitable SSID for the network. At the bottom of the Figure, you can see part of the transfer rate definition section. Supporting the 802.11b standard reduces the overall capacity of the network, so we recommend supporting only the 802.11a/g/n standards. For more information on the reduction in overall capacity, see the Best Practices document “WLAN network planning and setup” [10]. You can define traffic filters at the bottom of the left column. The filter rules are as a default too strict, and in order to facilitate smooth traffic, we recommend lightening the rules or removing the filter by clearing the Wireless security filters checkbox.

In Figure 39, you can see Wireless protection as the first item in the right-hand column. Because no encryption is used in a Web-authenticated network, leave the Wireless protection checkbox empty. Leave also the next checkbox, 802.1x authentication, empty. Tick the HTML-based user logins checkbox when using Web

authentication, and select the RADIUS profile defined earlier as the RADIUS server by first ticking the Remote checkbox. You can leave the remaining checkboxes in the right-hand column empty.

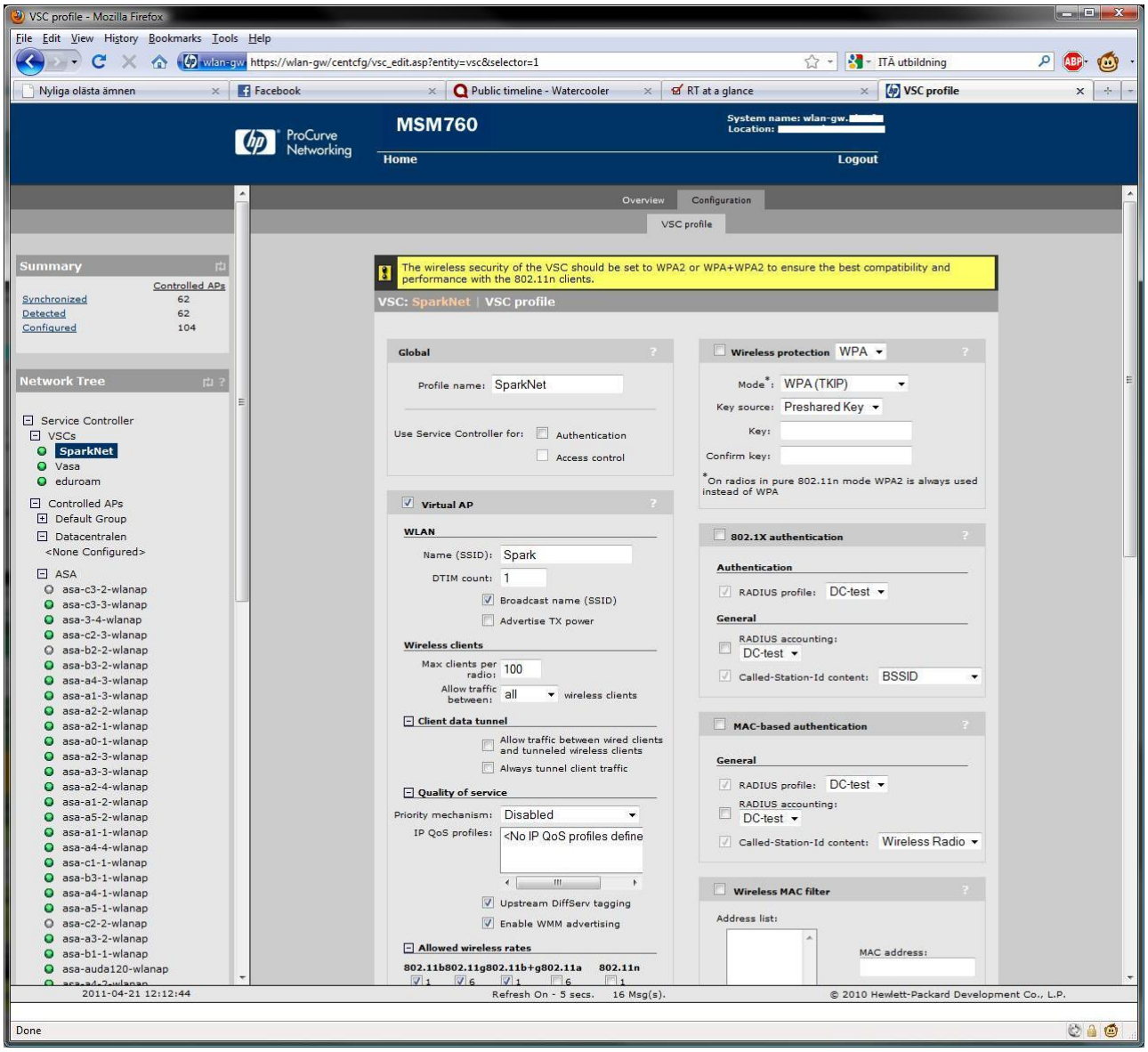

Figure 39. VSC profile for Web authentication. The reason for the warning in yellow is that the full utilisation of the high transfer rates of the 802.11n standard requires support for WPA2 encryption in the network.

If you wish to define a network using 802.1x as the authentication method, see Figure 40 for an example. The profile name has been named ‘eduroam’, as 802.1x authentication is always used in eduroam networks. You can set VSC ingress mapping to SSID, unless, you wish to connect traffic to a certain port of the access points. In an eduroam network, SSID is also always ‘eduroam’. The rest of the settings in the left-hand column can be the same as the settings for a Web-authenticated network.

In the right-hand column, tick the Wireless protection checkbox, and select WPA2 (AES/CCMP) in the Mode pull-down menu. This way, only WPA2 encryption is supported in the network, not WPA, in accordance with the Best Practices document, “WLAN Information Security”, [1]. Select Dynamic in the Key source pull-down menu. Next, tick the 802.1x authentication and Remote checkboxes. Unlike in the Figure, define RADIUS instead of Active directory, and select the RADIUS profile defined earlier from the pull-down menu. You can leave the remaining checkboxes in the right-hand column empty.

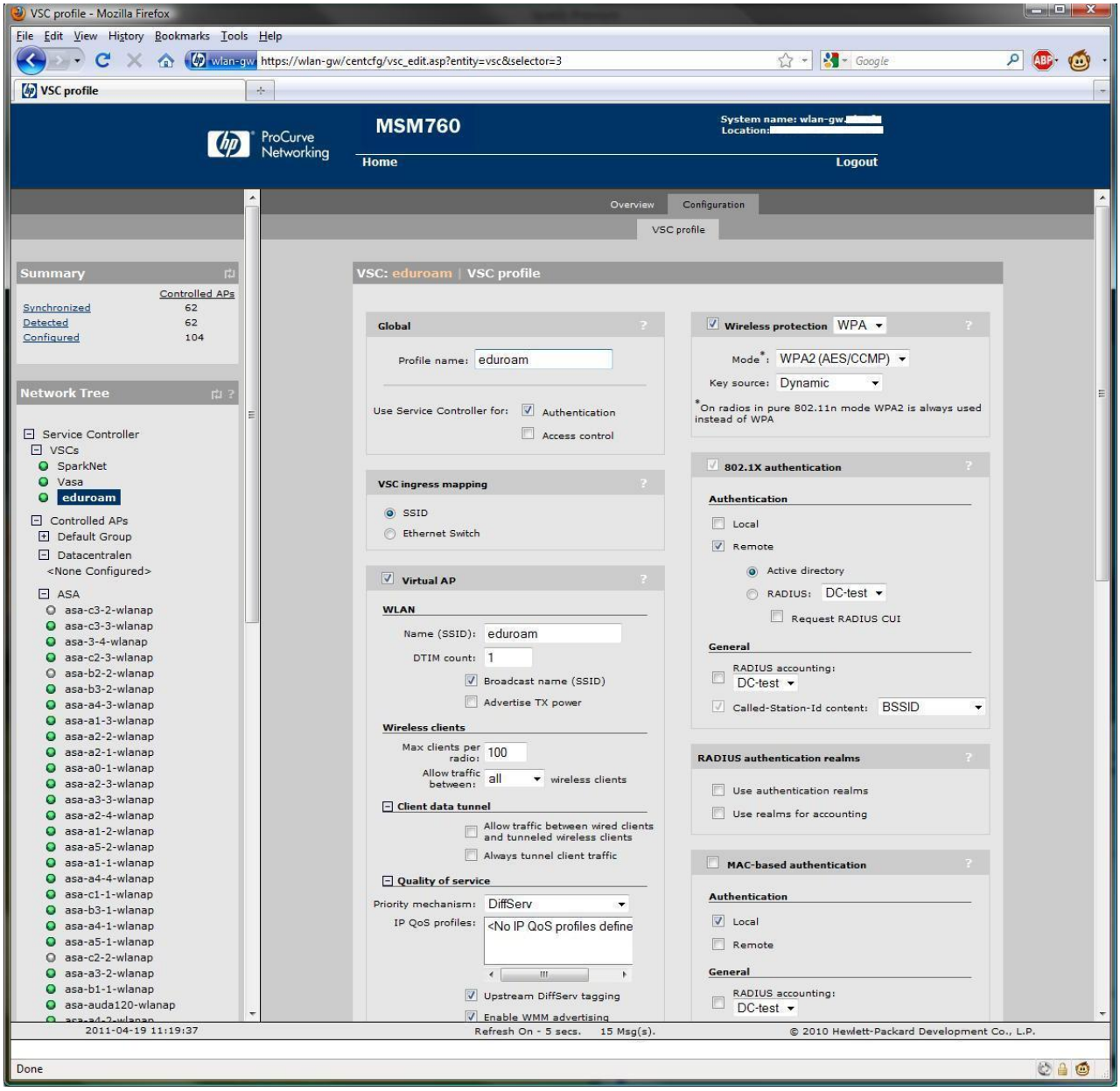

Figure 40. VSC profile for 802.1x authentication.

Connecting Access Points to the Network and their Configuration

The easiest way to get the access points to connect to the controller is to connect them to the same subnet as the controller. In other cases, you need to use DHCP option 43, or add the IP address of the controller to the DNS server under the name cnsrv1.mydomain.com, cnsrv2.mydomain.com or cnsrv3.mydomain.com.

Once an access point has connected to the controller, it will automatically belong to Default group; see Figure 41. Define an own group for network access points in production use by selecting **Controlled APs >> Group management**. Then click the Add New Group... button, enter a name for the group (for

example Datacentralen) and click the Save button. The name will appear in the Controlled APs menu as seen in Figure 41.

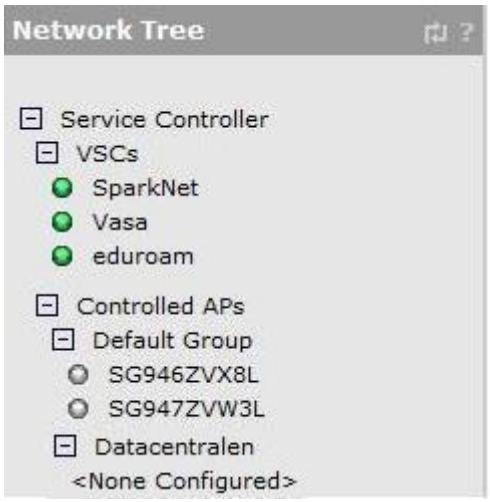

Figure 41. Network structure.

Before the access points are moved from the Default group to the group just defined, you can define the group's network settings. You can do this by connecting a suitable VSC to the group. Select the group from the Controlled APs menu and select **VSC bindings** and **Add New Binding** on the right. Connect the group to a suitable VSC as depicted in Figure 42. At this stage, you can also define the VLAN to be used.

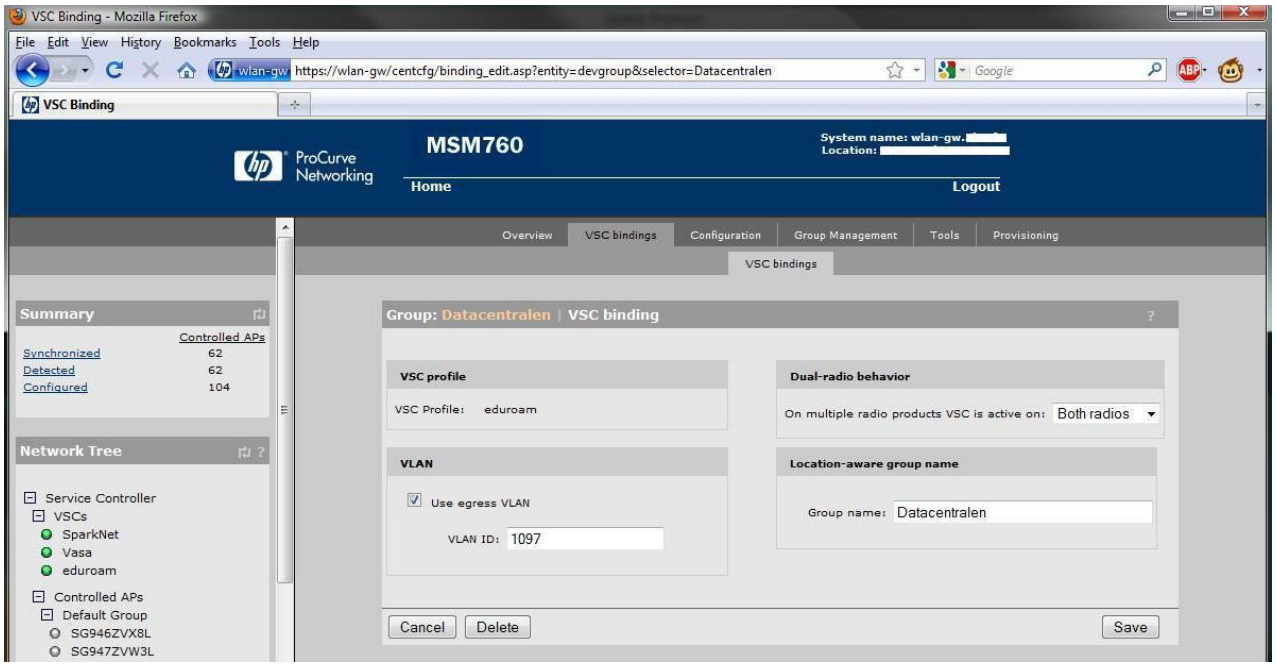

Figure 42. Defining a VSC for a group.

Next, move the access points from the Default group to the defined group. Select an access point and set the desired group for it as depicted in Figure 43.

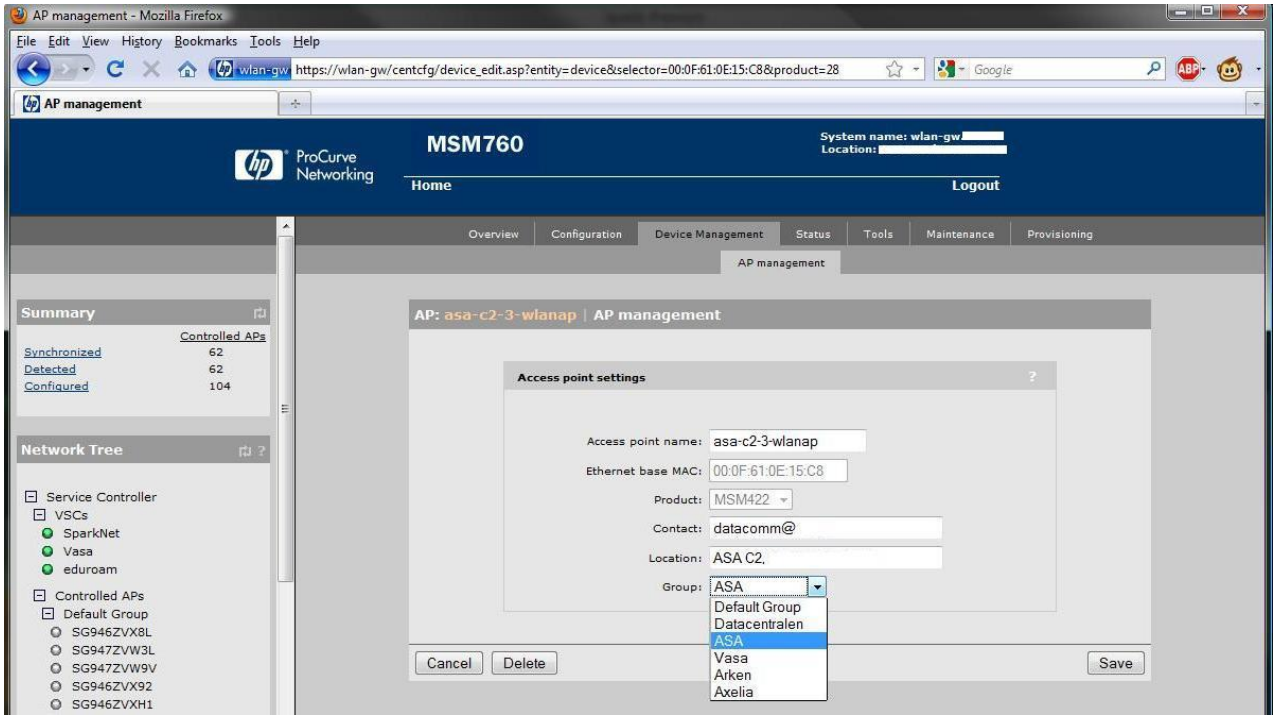

Figure 43. Moving a access point to a certain group.

See Figure 44 for an example of access point details. The access point has been connected to the ASA group.

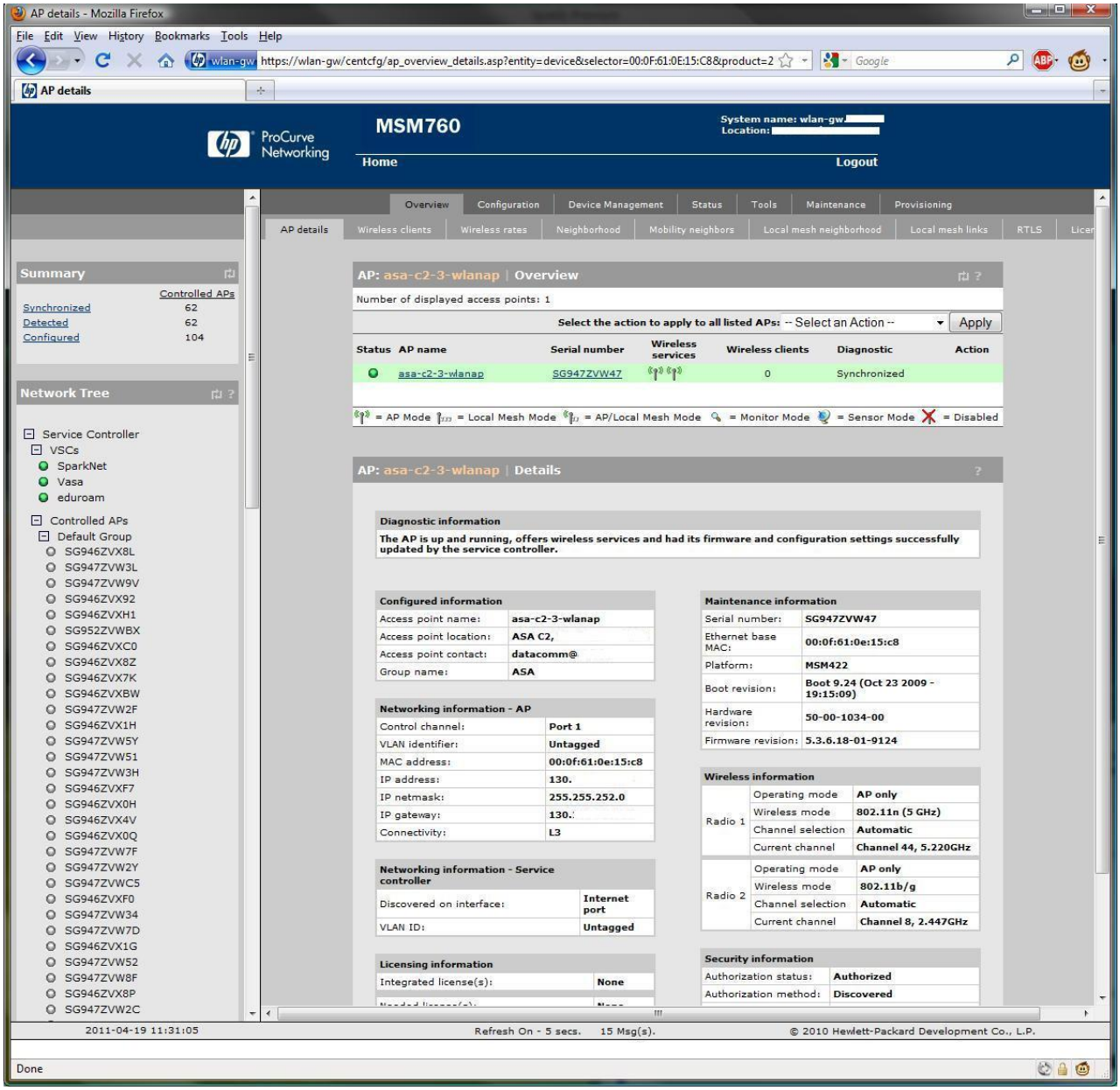

Figure 44. Access Point details.

Appendix 3 – FreeRADIUS Configuration

FreeRADIUS is a RADIUS server based on open source code and running on, for example, several Linux platforms. It allows the forwarding of RADIUS authentication messages in the RADIUS hierarchy, and users in the own organisation can be authenticated locally or by using a connected database. This appendix contains detailed instructions on how to configure a FreeRADIUS server to forward RADIUS messages in the roaming hierarchy as a service provider, and how own users can be authenticated locally (identity provider). Users are always separated by domains, realms.

FreeRADIUS Installation

You can find installation instructions with explanations for FreeRADIUS on the Web: http://wiki.freeradius.org/Build. This section is a summary of the instructions without detailed explanations. In this example, the platform used is *RedHat Enterprise Linux Server 5 64 bit*, and you can find detailed installation instructions for it here: http://wiki.freeradius.org/Red_Hat_FAQ.

FreeRADIUS is available on the Web: http://koji.fedoraproject.org/koji/packageinfo?packageID=298. Download an SRPM file from there – in this case, freeradius-2.1.3-1.fc9.src.rpm. Move the file under /usr/src/redhat/, for example to /usr/src/redhat/FreeRadSrmp. You can extract the SRPM file by running the rpm -ihv command in the directory in which the freeradius-2.1.3-1.fc9.src.rpm is saved: *rpm -ihv freeradius-2.1.3-1.fc9.src.rpm* .

For the next step, you need the yum-builddep tool, found in the yum-utils package. Install it with the following command: *yum install yum-utils*. Next, run *yum-builddep freeradius-2.1.3-1.fc9.src.rpm* in the FreeRadSrmp directory. If the command exits with an error like *Error: No Package found for perl-devel,* this means that you have to manually install some dependencies. You can discover the required dependencies by running *rpmbuild -ba /usr/src/redhat/SPECS/freeradius.spec* in the FreeRadSrmp directory. For example, you can fix *error: Failed build dependencies: gdbm-devel is needed by freeradius-2.1.3-1.x86_64* by running the *yum install gdbm-devel* command. Once you have manually installed the required dependencies, build the rpm file by running the *rpmbuild -ba /usr/src/redhat/SPECS/freeradius.spec* command in the FreeRadSrmp directory. Next, install the required rpms in the /usr/src/redhat/RPMS/x86_64/ directory. In addition to the basic package, you will need at least the libs package. Run the following command: *sudo rpm -Uhv freeradius-2.1.3-1.x86_64.rpm freeradius-libs-2.1.3-1.x86_64.rpm*. You can view a list of the installed freeradius packages with the *rpm -qa freeradius** command.

Configuration as a Service Provider

When FreeRADIUS operates as a service provider, it means that it forwards incoming RADIUS messages in the RADIUS hierarchy, and incoming packets to WLAN access points, controller or switches. It does not authenticate users using a database or a password file.

The files that need to be configured can be found in /etc/raddb/ and are named **clients.conf**, **proxy.conf** and **radiusd.conf**. You also need to create a new file in the sites-enabled directory, and create a symbolic link to it from the sites-available directory. You can find sample files in the FreeRADIUS_SP.zip file from the same web page as this document, [11], but we will also go through the configuration.

clients.conf

In this file, specify the devices that are allowed to send RADIUS messages to the FreeRADIUS server. You can add a device to the configuration as follows:

```
client my_client{
   ipaddr = xxx.yyy.zzz.www
   netmask = aa
   secret = v8493nfnkwenGYEj     # The client must also know this password
   require_message_authenticator = no
   shortname = my_client
   nastype = other     #or cisco if you are using cisco hardware.
   virtual_server =  eduroam     # Or some other suitable name
}
```

Additionally, we recommend leaving a localhost block for testing purposes:

```
client localhost {
   ipaddr = 127.0.0.1
   netmask = 32
   secret      = testing123
   require_message_authenticator = no
   shortname   = loopback
   nastype     = other
   virtual_server = eduroam
}
```

proxy.conf

In this file, you can specify how the RADIUS messages received from a client are forwarded in the RADIUS hierarchy. The following lines are enough:

```
proxy server {
        default_fallback        = yes
}

home_server next_radius_server_in_hierarchy{
        type                    = auth+acct
        ipaddr                  = xxx.yyy.zzz.www
        port                    = 1812
        secret                  = MfhurewrbDm886PR   #This password must be shared
```

```
        # with the next_radius_server_in_hierarchy. With regard to the Finnish root
server, please contact noc@funet.fi.
        response_window         = 20
        zombie_period           = 40
        revive_interval         = 60
        check_interval          = 30
        num_answers_to_alive    = 3
}

home_server_pool EDUROAM-FTLR {
        type                    = fail-over
        home_server       = next_radius_server_in_hierarchy
}

realm LOCAL {
     nostrip
}

realm NULL {
     nostrip
}

realm DEFAULT {
        pool                    = EDUROAM-FTLR
        nostrip
}
```

/sites-available/eduroam

In the sites-available directory, set up a virtual server that was defined to be used in the clients.conf file (virtual_server = eduroam). The name of the file is the same as the name of the virtual server, i.e., *eduroam* in this case. The server contents are presented in FreeRADIUS_SP.zip [11].

/sites-enabled/eduroam

You need to create a symbolic link to the virtual server (eduroam) in the sites-enabled directory. You can do this by running the *ln -s ../sites-available/eduroam eduroam* command in the sites-enabled directory.

radiusd.conf

First, check that

```
user = radiusd
group = radiusd
```

are not commented out, i.e., FreeRADIUS does not have to be run as root.

Next, modify the listen blocks as follows:

```
listen {
   type = auth
   ipaddr = *  # Define the IP addresses if necessary.
```

```
   port  = 1812
}

listen {
    ipaddr = * # Define the IP addresses if necessary.
           port = 1813
           type = acct
}
```

It is also possible to add support for IPv6.
Add the following blocks at the end of the file:

```
detail auth_log {
        detailfile = ${radacctdir}/%Y%m%d/eduroam/auth-detail
        detailperm = 0600
}

realm suffix {
        format = suffix
        delimiter = "@"
}

attr_filter attr_filter.pre-proxy {
        attrsfile = ${confdir}/attrs.pre-proxy
}
```

Testing and troubleshooting

Start up the server in the /etc/raddb/ directory by running the *radius –X* (debug mode) command. If necessary, you can find the command in the /usr/sbin/ directory.

In the first stage, you can test the server configuration using the localhost client. In the /usr/src/redhat/BUILD/freeradius-server-2.1.3/src/main/ directory, run the following command: */usr/src/redhat/BUILD/freeradius-server-2.1.3/src/main/radtest username@myorganisation.fi PaSsWoRd localhost 1 testing123*

If you run into troubles, check that the firewalls (iptables) are configured so that the FreeRADIUS server can receive RADIUS messages from the specified clients, and that next_server_in_hierarchy can receive the RADIUS messages sent by the server.

Configuration as an Identity Provider

The next step is to change the configuration so that the server works as both a service provider and an identity provider. The domain information of arriving RADIUS messages is checked, and own users are authenticated and the messages from visitors are forwarded in the hierarchy.

You need to modify the following files: **proxy.conf**, **/sites-available/eduroam**, **eap.conf** and **users**. You also need to create a /sites-available/eduroam-inner-tunnel file, and create a symbolic link to this file in the sites-enabled directory. You can find sample files in the FreeRADIUS_IdP. zip file from the same web page as this document, [11], but we will also go through the configuration.

proxy.conf

After the home_server_pool EDUROAM-FTLR block, add a block defining that own users (mydomain.fi) form a special case:

```
realm mydomain.fi {
        nostrip
}
```

/sites-available/eduroam

At the end of the authorize and authenticate branches of this file (virtual server), define that eap must be used:

```
authorize {
          auth_log
          suffix
          eap
}

authenticate {
          eap
}
```

eap.conf

The EAP methods used are defined in the eap.conf file; we recommend using PEAP and TTLS. In order for these to work, you also need to define the TLS settings! See FreeRADIUS_IdP.zip [11] for the configuration details.

users

This file contains usernames and passwords for the users that will be authenticated locally, i.e., the mydomain.fi users. In the example below, on user is defined with a password in cleartext and one user with a password in NT-hash format:

```
mina@mydomain.fi Cleartext-Password := "hello"

#sina@mydomain.fi NT-Password := "goodbye"
sina@mydomain.fi NT-Password := "CAC331BC07EC8830CA1563716472A22C"
```

The cleartext password will only work with TTLS-PAP and the NT-hash password will work with TTLS-MSCHAPv2 and PEAP-MSCHAPv2.

/sites-available/eduroam-inner-tunnel and /sites-enabled/eduroam-inner-tunnel

The virtual server eduroam-inner-tunnel was defined to be used in the PEAP and TTLS branches of the eap.conf file in the case where the user is authenticated locally and the EAP method is one of these. The virtual server is created in the same directory as the eduroam virtual server, i.e. the sites-available directory. You can find the configuration details in FreeRADIUS_IdP.zip [11]., but the most important blocks are the authorize and authenticate blocks, where the authentication methods to be used are defined:

```
authorize {
   auth_log
   files
   mschap
   pap
        eap {
                ok = return
        }
}
```

```
authenticate {
   Auth-Type PAP{
      pap
   }
   Auth-Type MS-CHAP{
      mschap
   }
        #  Allow EAP authentication.
        eap
}
```

In these blocks, you can define LDAP in the same way, if the server is connected to an LDAP database. In the case of LDAP, you also need to edit the radius.conf file.

You also need to create a symbolic link to the eduroam-inner-tunnel virtual server by running the *ln -s ../sites-available/eduroam-inner-tunnel eduroam-inner-tunnel* command in the sites-enabled directory.

Testing

You can further test the forwarding of RADIUS messages locally by using the localhost client. The best way of testing local authentication is by connecting an access point to the server and testing over a wireless network, for example by using Linux wpa_supplicant. This way, you can ensure that the server works in production conditions.

Summary

Configured this way, the server will successfully forward authentication requests and authenticates mydomain.fi users locally using TTLS-PAP, TTLS-MSCHAPv2 or PEAP-MSCHAPv2.

Implementing a Certificate

FreeRADIUS includes testing certificates to make configuration and initial testing easier, but these certificates are not used in production. For production use, you can either create a self-signed certificate or request a certificate for the server froma certificate service. You can find instructions for creating a self-signed certificate in the README file located in the /etc/raddb/certs directory.

You can implement a certificate by editing the TLS block of the eap.conf file as follows (alternatively, see the eap_with_cert.conf –file in FreeRADIUS_IdP.zip [11]):

```
tls {
   private_key_password = YoUrPaSsWoRd
   private_key_file = /pathToCert/my_server.pem
   certificate_file = /pathToCert/my_server.crt
   CA_file = /pathToCert/cert.pem

   #  make_cert_command  =  "${certdir}/bootstrap"  This  line  is  used  to  create  testing
certificates, and must be commented out at this stage.
}
```

If there are intermediate certificates between the root certificate and the server certificate, such as is the case with the certificates currently issued by Funet's certificate service, the cert.pem file must contain the whole certificate chain. The correct cert.pem file, with the intermediate certificates, can be created out of the intermediate certificates using the following command:

```
cat CA_Root.pem Intermediate1.pem Intermediate2.pem > cert.pem
```

Once the server has been configured in this way, authentication will succeed if the one of the intermediate certificates has been selected as a trusted certificate in the user's supplicant and the correct server name has been defined. Authentication will fail, if the server name is incorrect or none of the intermediate certificates have been defined as trusted certificates. Because quite a lot of authentication security details can be defined in a supplicant, authentication will succeed even if the certificate and server name are left undefined. However, IT support should ensure to the extent possible that the certificate used and the server name are defined in the users' supplicants.

Taking Virtual LANs into Consideration

If VLANs are in use, users can be placed in different VLAN networks, for example when you wish to grant more rights to the users in your own organisation than to visitors. The VLANs must also naturally be defined in the WLAN controller and in the other elements in the LAN, for example switches. It may be a good idea to define a VLAN intended for the visitors as the default, and move users from the own realm (user@myorganisation.fi) to the VLAN intended for them once authentication has been successful. Within this context, you must also take into consideration that users from your own realm are placed in a VLAN intended for them only if they are in their own organisation's network. When they are visiting (roaming), the visiting organisation decides the VLAN in which they are placed, and VLAN definitions are not sent outside the own organisation.

You can place your own users in the VLAN intended for them by adding the following lines in the post-auth block of the /sites-available/eduroam-inner-tunnel file:

```
post-auth {
.
.
.
   if ("%{User-name}" =~ /@myorganisation.fi$/ && "%{NAS-IP-Address}" =~ /^xyz.zyx.xzy./){
   # In this case, the organisation's access points use addresses in the xyz.zyx.xzy.0/24
address space.
      update reply {
         Tunnel-Type := 13
         Tunnel-Medium-Type := 6
         Tunnel-Private-Group-ID := desired_VLAN_id_nr
         }
     }
}
```

References

[1] W. Backman et. al. “WLAN Information Security”, Best Practice Document producet by Funet led working group on wireless systems and mobility (MobileFunet), June 2010, Available at http://www.terena.org/activities/campus-bp/bpd.html

[2] http://www.ciscointernethome.org/en/US/tech/tk722/tk809/technologies_configuration_example09186a00807cc3b8.shtml

[3] http://www.eduroam.org/downloads/docs/GN2-08-230-DJ5.1.5.3-eduroamCookbook.pdf

[4] http://wiki.eduroam.cz/dead-realm/docs/dead-realm.html

[5] http://deployingradius.com/documents/protocols/compatibility.html

[6] http://www.csc.fi/funet/status/tools/roaming.pl

[7] http://cbt.geant2.net/repository/eduroam_supplicants/setting_up_eduroam_supplicants.html

[8] https://info.funet.fi/wiki/BCP/EduroamAsennusohjelma

[9] http://deployingradius.com/scripts/eapol_test/

[10] W. Backman et. al. “WLAN Network Planning and Setup”, Best Practice Document producet by Funet led working group on wireless systems and mobility (MobileFunet), December 2010, Available at http://www.terena.org/activities/campus-bp/bpd.html

[11] http://www.terena.org/activities/campus-bp/bpd.html

Glossary

ACL	Access Control List
AD	Active Directory
AES	Advanced Encryption Standard
CA	Certification Authority
CCMP	Counter Mode with Cipher Block Chaining Message Authentication Code Protocol
CLI	Command Line Interface
CN	Common Name
CPU	Central Processing Unit
CSC	CSC - IT Center for Science Ltd.
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name System
EAP	Extensible Authentication Protocol
EAPOL	EAP over LAN
Funet	Finnish University and Research Network
H-REAP	Hybrid Remote Edge Access Point
IGMP	Internet Group Management Protocol
IP	Internet Protocol
LDAP	Lightweight Directory Access Protocol
MAC	Medium Access Control
MobileFunet	a working group on wireless systems and mobility led by Funet
MSCHAPV2	Microsoft Challenge-Handshake Authentication Protocol version 2
PAP	Password Authentication Protocol
PEAP	Protected EAP
PoE	Power over Ethernet
RADIUS	Remote Authenticated Dial-In User Service
SMTP	Simple Mail Transfer Protocol
SSH	Secure SHell
SSID	Service Set Identification
TFTP	Trivial File Transfer Protocol
TLS	Transport Layer Security
TTL	Time To Live
TTLS	Tunneled TLS
VLAN	Virtual Local Area Network
XML	eXtensible Markup Language
VSC	Virtual Service Community
WLAN	Wireless Local Area Network

WPA and WPA2 Wi-Fi Protected Access