User Authentication using FreeRadius and Novell eDirectory

Georgi Todorov

October 31, 2006

Contents

Chapter 1

Introduction

1.1 What is this paper about?

The purpose of this paper is to serve as a guide to people who need to add automated user authentication and billing capabilities of unknown wireless and/or wired users to their existing eDirectory based network. The basic idea is the following. For AAA (Authentication, authorization and accounting) FreeRADIUS will be used. It will make the process a lot easier due to its modular structure. It is capable of working with Novell's eDirectory which is important for the purpose of this paper. In general FreeRADIUS makes future changes of the back-end LDAP/SQL schema or authentication server easy task. Recent Cisco's IOS support AAA trough RADIUS, for the ones that do not, TACACS+ can be used to bridge to RADIUS. Nortel also recommends RADIUS for AAA[Nortel, 2003]. The wireless technology that will be used is EAP (802.1x EAP) or more precisely EAP-TLS[Aboba and Simon, 1999]. Once we have FreeRADIUS integrated with eDirectory, there are two cases to consider, wireless users and wired users. For both, the process will be similar. Once an unknown user connects to the network, he/she will be assigned to a VLAN that will send all traffic to a specific router and that router will forward all traffic to port 80 on a local server. There a billing web page will welcome the new user and provide him with step-by-step tutorial on how to obtain an account, and how to setup his client computer to use the network. Once the the user completes the process that web page will add/update the users and the authentication will be complete. At that point, the user is mapped to the second VLAN, which is the "trusted" VLAN and traffic is allowed according to the internal use policy.

1.2 freeRADIUS

"FreeRADIUS is the premiere open source RADIUS server. While detailed statistics are not available, we believe that FreeRADIUS is well within the top 5 RADIUS servers world-wide, in terms of the number of people who use it daily for authentication. It scales from embedded systems with small amounts of memory, to systems with millions of users. It is fast, flexible, configurable, and supports more authentication protocols than many commercial servers."[FreeRADIUS, 2006]

1.3 802.1X

IEEE 802.1X is an IEEE standard for port-based Network Access Control; it is part of the IEEE 802 (802.1) group of protocols. It provides authentication to devices attached to a LAN port, establishing a point-to-point connection or preventing access from that port if authentication fails. It is used for certain closed wireless access points, and is based on the EAP, Extensible Authentication Protocol (RFC 3748). [Wikipedia, 2006b]

1.4 eDirectory

"Novell eDirectory (formerly called Novell Directory Services) is an X.500 compatible directory service software product released in 1993 by Novell, Inc. for centrally managing access to resources on multiple servers and computers within a given network. It is used in over 80% of Fortune 1000 companies and competes with Microsoft's Active Directory, Sun's Java System Directory Server and Red Hat's Fedora Directory Server.

eDirectory is a hierarchical, object oriented database that represents all the assets in an organisation in a logical tree. Assets can include people, posimnotions, servers, workstations, applications, printers, services, groups, etc. The use of dynamic rights inheritance and equivalence allows both global and fine grained access controls to be implemented efficiently. Access rights between objects in the tree are determined at the time of the request and is determined by the rights assigned to the objects by virtue of their location in the tree, any security equivalences and individual assignments. eDirectory supports partitioning at any point in the tree and replication of that partition to any number of servers. Replication between each server occurs periodically using deltas of the objects to reduce LAN/WAN

traffic. Each server can act as a master of the information it holds (providing the replica is not read only). Additionally, replicas may be filtered to only include defined attributes to increase speed (eg a replica may be configured to only include a users name and phone number for use in a corporate address book).

Referential integrity, multi-master replication and the modular authentication architecture are other major advantages.

eDirectory can be accessed via LDAP, XML, DSML, SOAP, ODBC, JDBC, JNDI, EJB, Perl, Active X and ADSI and has been proven to scale to over 1 billion objects."[Wikipedia, 2006c]

Chapter 2

Installation and Configuration

NOTE: This paper will assume that eDirectory is already installed and configured, and will not cover that process.

2.1 FreeRADIUS

2.1.1 From Source

First you will need some prerequisites before you compile freeRADIUS. If you have the following packages, you have a good chance to compile most modules:

- cyrus-sasl-devel
- db-devel
- heimdal-devel
- libiodbc
- libiodbc-devel
- mysql-devel
- mysql-shared
- openldap2-cliet
- openldap2-devel
- openssl
- openssl-devel

- postgresql-devel
- postgresql-libs
- python
- python-devel

Once you have some good number of the above, you need to obtain the latest copy of freeRADIUS from `http://www.freeradius.org/`. When you extract the latest archive, go to the extracted directory and type:

```
# ./configure –with-edir
```

The option *-with-edir* tells freeRADIUS to enable eDirectory support, which is disabled by default. After the configuration script is done without errors type:

```
# make
# make install
```

The last two commands will compile and install the software if there were no errors.

2.1.2 From a package

Depending on the distribution you are using, follow the standard approach to install the latest available freeradius package.

2.2 Certificate Authority(CA)

Certification authority (CA) is an entity which issues digital certificates for use by other parties.[Wikipedia, 2006a]. /etc/ssl is the default location for the configuration files of openssl, but it may vary from distribution to distribution. If you do not wish to enter the information asked by openssl (state, country name etc.) you need to edit openssl.cnf in /etc/ssl (or wherever this file is located on your linux) and edit the following keys:

```
# dir = ./radiusCA # this is the directory that will contain the CA files
```

and further down in the file just enter the values for countryName_default, stateOrProvinceName_default and 0.organizationName_default:

```
#countryName_default = US
#stateOrProvinceName_default = New-York
#0.organizationName_default = HighWiFi Organization
```

After these entries are changed, the CA.sh script needs to be updated as well to point to the correct directory *./radiusCA*. The line that has to be edited is CATOP=./radiusCA. After the script is ready, go to /etc/ssl and execute the script with *-newca* option:

```
# /usr/share/ssl/misc/CA.sh -newca
```

When asked for a password, enter a good password and make sure you will not forget it, because you will not be able to use the certificate anymore if you do.

Once the script is done, there should be a radiusCA directory in your ssl configuration directory (/etc/ssl). The default name of the certificate is*cacert.pem*. This file is very important and needs to be provided to the radius server and to be copied to each wireless client.

Since Windows XP will be a possible client, an extensions file is required. While in the ssl configuration directory, create a file called xpextensions that contaions:

```
[ xpclient_ext ]
extendedKeyUsage = 1.3.6.1.5.5.7.3.2
[ xpserver_ext ]
extendedKeyUsage = 1.3.6.1.5.5.7.3.1
```

For OS X we need a DER version of the CA certificate which is generated using the following:

```
#openssl x509 -in cacert.pem -out cacert.der -outform DER
```

In addition to the CA certificate EAP-TLS requires at least two more certificates. One for the freeRADIUS server and one for each wireless client. The following command will generate a server certificate signing request. Make sure to enter the full name of your server for Common Name (eg: fradius.mydomain.com):

```
# openssl req -new -nodes -keyout server_key.pem -out server_req.pem -days 730 -config ./openssl.cnf
```

The file *server_req.pem* is being generated and it is used to generate a signed by the CA certificate. Use the next command to sign the the request and generate the actual certificate:

```
# openssl ca -config ./openssl.cnf -policy policy_anything -out server_cert.pem -extensions xpserver_ext
-extfile ./xpextensions/-infiles ./server_req.pem
```

You will be asked for the CA password and a *server_cert.pem* file will be generated. The next step is to open the file with any text editor and cleanup everything above ––*BEGIN CERTIFICATE*––.

After you have both *server_cert.pem* and *server_key.pem* a new key-certificate file should be created that contains both files:

```
# cat server_key.pem server_cert.pem > server_keycert.pem
```

Now the key and certificate for the server are ready. Note that the private key is not password-protected, so extra copies of this file should not be spread around. The next step is to generate the client certificate using the above procedure. Generate signing request with passwords:

```
# openssl req -new -keyout client_key.pem -out client_req.pem -days 730 -config ./openssl.cnf
```

Signing the signing request:

```
# openssl ca -config ./openssl.cnf -policy policy_anything -out client_cert.pem  -extensions xpclient_ext
-extfile ./xpextensions -infiles ./client_req.pem
```

Linux machines require the certificate to be clean, so you can remove again everything above ––*BEGIN CERTIFICATE*–– and concatenating them into a

single file is optional here.

For Windows XP and OS X, the certificate needs to be in PKCS12-format, and since WinXP requires the file to be without a password, do not enter one (not a good practice in general):

```
# openssl pkcs12 -export -in client_cert.pem -inkey client_key.pem -out client_cert.p12 -clcerts
```

The generated file *client_cert.p12* contains both signed certificate and a private key.

The above process must be repeated for every new client we have on the network. A simple script can be used with the *-batch* and *-passin=password -passout=password* options of openssl.

One last thing is to extract the self-signed certificate of the certificate authority for eDirectory and freeRADIUS purposes. Since iManager[iManager, 2006] is used follow the steps provided by Novell: `http://www.novell.com/documentation/crt27/index.html?page=/documentation/crt27/crtadmin/data/a2ebopb.html#a2ebopd` Just make sure when you save the extracted certificate to remember the location for freeRADIUS later[Bauer, 2005]

2.3 eDirectory for Radius

Using the iManager plugin, the following changes must be performed[Novell, 2005]:

1. Enable Universal Password for eDirectory users Use the following Novell documentation to setup the Universal Password:
   `http://www.novell.com/documentation/nmas23/index.html?page=/documentation/nmas23/admin/data/allq21t.html`

2. Creating the RADIUS Administrator Object
   Use the following Novell documentation the create a new Object:
   `http://www.novell.com/documentation/edir873/index.html?page=/documentation/edir873/edir873/data/a4jgpgc.html#a3olp4k`

3. Granting Administration Rights for the RADIUS Administrator
   Grant the RADIUS administrator the write right over the ACL attribute of the user object whose universal password has to be read. By granting this right, the RADIUS administrator will gain the administrative rights over that

user object.
The eDirectory administrator can also be the RADIUS administrator. For more information on eDirectory rights, refer to the Novell eDirectory Administration Guide
http://www.novell.com/documentation/edir873/index.html?page=/documentation/edir873/edir873/data/fbachifb.html#fbachifb

4. Granting Rights to RADIUS Administrator to Retrieve Password
By default, the administrator does not have the right to read universal password. eDirectory administrator will modify the password policy to enable the RADIUS Administrator to read universal password.
if you have iManager's Password Management 2.0.2 installed do the following:

1 In iManager, click the Roles and Tasks button .
2 Click Passwords - Password Policies
    2a Select the password policy being used.
    2b Click Edit.
3 Click Universal Password - Configuration Options.
    3a Select Allow admin to retrieve passwords from
    Universal Password Retrieval.
    3b Click OK.

Else follow these steps:

1 In iManager, click the Roles and Tasks button .
2 Click eDirectory Administration - Modify Object.
    2a Select Security Container from the Object Selector.
    2b Select Universal Password On from Password Policies.
    2c Click OK
3 Select General tab.
    3a Edit the nspmConfigurationOptions attribute and add 32
    to the value already shown..
    3b Click OK.

2.4 freeRADIUS configuration

2.4.1 Certificate Access Control

We need to take care of some permissions of several files in your raiddb/certs directory.[Bauer, 2005] We need to make *cacert.pem* owned by root and only readable with the following commands:

```
# chmod 444 cacert.pem

# chown root:users cacert.pem
```

Than we need set *server_keycert.pem* to be owned by nobody:

```
# chmod 400 cacert.pem

# chown nobody:users cacert.pem
```

Also the */var/log/radius/radius.log* and */var/run/radiusd/* must be writable by the user nobody (which is the user that will run radius)

```
# chown -R nobody:users /var/run/radiusd/

# chown -R nobody:users /var/log/radius/
```

The next step is to create the so called Difie-Hellman parameters file for negotiation of TLS session keys in raidb/certs/:

```
# openssl dhparam -check -text -5 512 -out dh

# chown nobody:users dh

# chmod 644 dh
```

Also we need a random bitstream that will be used in TLS operations. Again from raiddb/certs:

dd if=/dev/urandom of=random count=2

chown nobody:users random

chmod 644 random

2.4.2 radiusd.conf

In etc/raiddb/ we need to edit just a few files for the actual authentication. One of them is radiusd.conf. The first two things we need to specify are the user and the group"

user = nobody

group = nobody

If you wish to use different users here make sure you update the file permissions accordingly in the previews subsection.

Also make sure the user has close to no privileges on the machine and set the login shell to something line */bin/false* or */bin/true-this* - whatever is available on your system.

For the eDirectory part we need to change the following in the ldap section Some of the entries we will configure later in this chapter:

server = hostname of eDir LDAP server
identity = FDN of the RADIUS server in eDirectory
password = password of the RADIUS server object in eDirectory
basedn = the DN of the container that stores the RADIUS users and profile objects
filter = (cn=%{Stripped-User-Name:-%{User- Name}})
start_tls = yes
tls_mode = conditional
tls_cacertfile = /full/path/to/cacert.pem
tls_require_cert = demand dictionary_mapping = ${raddbdir}/ldap.attrmap
password_attribute = nspmPassword
edir_account_policy_check = yes

```
access_attr = dialupAccess
```

In the post-auth section do the following:

```
post-auth{
ldap
Post-Auth-Type REJECT { ldap
}
}
```

Feel free to look at the rest of the options in radiusd.conf and tweak them, but you shouldn't need to configure them. Also for some help on ldap and freeRADIUS check the *doc/ldap_howto.txt* file.

2.4.3 eap.conf

In this file we will define our authentication methods and there are quite a few changes to be made. Fortunately the file is full with example entries. The basic entry we need is the following:

```
eap {
tls {
# The following parameters tell radiusd where to
# find its certs and keys, plus dh & random files:
private_key_password = the_pass_for_the_key
private_key_file = ${raddbdir}/certs/server_keycert.pem
certificate_file = ${raddbdir}/certs/server_keycert.pem
CA_file = ${raddbdir}/certs/cacert.pem
dh_file = ${raddbdir}/certs/dh
random_file = ${raddbdir}/certs/random
}
}
```

Since we used -nodes when we generated the password for the private key, the actual entry should hava an empty password, but in general it is not wise to use

-nodes. You might as well generate your key with a password to be on the safe side.

Another security issue we need to deal with is that *eap.conf* contains passwords and should be readable only by root and nobody else:

```
# chown root:root eap.conf

# chmod 600 eap.conf
```

2.4.4 clients.conf

When we setup the network we need to create an entry for every access point that will use the server. Here is an example of an AP entry in clients.conf:

```
clients 10.1.2.3/32 {
secret=someAPpass
shortname=CiscoAP1
}
```

The *secret* option specifies the string that the AP will use as an encryption key for all queries to the FreeRADIUS server. *shortname* will be the name that will appear in the logs when referring to that AP.

Now FreeRADIUS should be able to run with the provided *rc.radiusd* script using the following command:

```
rc.radiusd start
```

If something goes wrong... google it.

2.5 Nortel Routing Switch Configuration

Due to the source of the information needed for this section I will not include a step-by-step configuration. Please refer to your Switch Documentation provided to you by Nortel. For the 5500 series refer to document NN47200-502_2.03 page 73. The basic idea is to add our radius server and enable per port EAPOL(EAP Over Lan

) authentication. This will separate the two VLANs - if the user is authenticated he can access the internal network, if not, the second VLAN will redirect all his requests to the accounting web page.

2.6 Access Point Configuration

The AP's are very easy to configure. Regardless of the brand (Cisco, 3com, Nortel etc) all that is needed is te IP of the server and the corresponding *secret* value for that AP. The basic entries in every AP are :

Security mode (Encryption) : WPA with RADIUS
Algorithm: TKIP
Port: 1312 (or the port you configured)
Shared Key: the secret value

You will need to refer to your documentation in order to configure the two VLANs needed (one for authenticated users and the other for guests that will be redirected to the accounting web page).

2.7 Windows XP Client

Unfortunately this is not a one-click task, but for our needs a nice step by step web page can be used to show every windows user what needs to be done with screenshots and etc. The basic steps are:

1. Run the command *mmc* from Start?Run....
2. In Microsoft Management Console, select File?Add/Remove Snap-in, add the Certificates snap-in and set it to manage certificates for My user account and, on the next screen, only for the Local computer.
3. Copy your CA (cacert.pem) certificate to your Windows system's hard drive, for example, to C:
   cacert.pem.
4. From within MMC, expand Console Root and Certificates - Current User and right-click on Trusted Root Certification Authorities. In the pop-up menu,

select All Tasks?Import. Tell the subsequent wizard to import the file C:
cacert.pem and to store it in Trusted Root Certification Authorities.

5. Copy your client certificate/key file to your Windows system, for example, to C:
client_cert.p12.

6. From within MMC?Console Root?Certificates, expand Personal and right-click on Certificates. In the pop-up menu, select All Tasks?Import. Tell the subsequent wizard to import the file C:
client_cert.p12.

7. The certificate-import wizard then prompts you for the certificate's passphrase. In the same dialog, it offers the option to enable strong private key protection. Unfortunately, enabling this breaks WPA, so be sure to leave this option unchecked. Also, leave the option to mark this key as exportable unchecked-you're better off backing up the password-protected file you just imported rather than allowing the imported nonprotected version to be exportable.

8. In the subsequent screen, let the wizard Automatically select the certificate store.

The rest is up to the particular driver to setup an WPA wireless. in Windows XP SP1 with a Centrino system and a native WPA supplicant, the Network Authentication is set to WPA, Data encryption to TKIP, EAP type to Smart Card or other Certificate and than the rest is done by the OS.

But this is to be presented to the user. And we need to be able to "expire" user access to our network. In the next chapter we will take care of the automation business.

2.8 Mac OS X Client

In OS X it is easier than Win XP and again a step-by-step web page can be used to show the users of OS X how to import the *cacert.der* and the *cliet_cert.p12*. Go to Applications - Utilities - Keychain Access. From File - Import, select your *cacert.der* and from the drop-down menu "Keychain" select *X509Anchors*. Click OK Repeat the steps for *client_cert.p12* but this time import it into *login*

You can now try from the wireless icon Select "Other". In the next Window called "Closed Network", enter the Network Name (ESSID), Select "WPA2 Enterprise" and change the 802.1X Configuration to Automatic.

If you want to have the setup stored on your system or you don't have a wireless icon in your system bar: Go to System Preferences - Network Click on AirPort Click on the + sign. Fill the information as in the above example.

2.9 Linux Client

To configure linux a copy of wpa_supplicant is needed. The following url: `http://hostap.epitest.fi/wpa_supplicant/` is the place to seek help. The basic info that is needed in *wpa_supplicant.conf* is:

```
network={
ssid="example"
scan_ssid=1
key_mgmt=WPA-EAP WPA-PSK IEEE 8021X NONE
pairwise=CCMP TKIP
group=CCMP TKIP WEP104 WEP40
psk="very secret passphrase"
eap=TTLS PEAP TLS
identity="user@example.com"
password="foobar"
ca_cert="/etc/cert/cacert.pem"
client_cert="/etc/cert/user.pem"
private_key="/etc/cert/user.prv"
private_key_passwd="password"
phase1="peaplabel=0"
}
```

2.10 eDirectory Users for freeRADIUS Authentication

2.10.1 Configuring iManager Plug-in for RADIUS

In order for RADIUS to work with iManager plug-in, the plu-ing must be configured with SSL/TLS connection to eDirectory. If both the plugin and the iManager are on the same machine SSL/TLS is setup by default. In the other case it has to be done manually. To do that refer to : `http://www.novell.com/documentation/imanager20/index.html?page=/documentation/lg/imanager20/imanager20/data/am4ajce.html`

2.10.2 Extending the eDirectory Schema for RADIUS

There are three scenarios:

If mapping already exists between RADIUS:Profile to rADIUSProfile

1 In iManager, click the Roles and Tasks button .
2 Click LDAP - LDAP Overview.
    2a Select View LDAP Groups.
    2b Select Class Map from the drop down list.
    2c Select the RADIUS:Profile to rADIUSProfile mapping.
    2d Click Edit.
    2e Change the primary LDAP class name to anything other than rADIUSProfile, for example, novellradiusprofile.
    2f Click Apply.
3 Refresh LDAP server.
4 Click RADIUS - Extend schema for RADIUS.
    4a Click OK

If mapping does not exists between RADIUS:Profile to rADIUSProfile

1 In In iManager, click the Roles and Tasks button
2 Click LDAP - LDAP Overview.
    2a Select View LDAP Groups.
    2b Select Class Map from the drop down list.
    2c Click Add mapping button
    2d In the eDirectory class drop down list, select RADIUS:Profile.

2e Change the primary LDAP class name to anything other than rADIUSProfile, for example, novellradiusprofile.
2f Click OK.
3 Refresh LDAP server.
4 Click RADIUS - Extend schema for RADIUS.
4a Click OK

If mapping already exists between RADIUS:Profile to any name other than rADIUS-Profile

1 In In iManager, click the Roles and Tasks button
2 Click RADIUS - Extend schema for RADIUS
2a Click OK.

2.10.3 Managing RADIUS Objects/Users

Creating RADIUS Users

1. In iManager, click the Roles and Tasks button.
2. Click RADIUS - Create RADIUS User.
3. Specify the User object to create either by typing in the object name or using the object selector.
4. (Optional) Specify the Profile object you want to associate with the user by typing in its name or using the object selector.
5. Click OK.

Modifying RADIUS Users

1. In iManager, click the Roles and Tasks button
2. Click RADIUS - Modify RADIUS User
3. Specify the User object to modify either by typing in the object name or using the object selector.

4. (Optional) Specify or modify the RADIUS attributes for the User object.
5. Click OK.

Deleting RADIUS Users

1. In iManager, click the Roles and Tasks button
2. Click RADIUS - Delete RADIUS User
3. Specify the User object to delete either by typing in the object name or using the object selector.
4. Click OK.

2.10.4 Managing RADIUS Profiles

Creating RADIUS Profiles

1. In iManager, click the Roles and Tasks button.
2. Click RADIUS - Create RADIUS Profile.
3. Specify the context for the Profile object to create either by typing in the object name or using the object selector.
4. Click OK.

Modifying RADIUS Profiles

1. In iManager, click the Roles and Tasks button
2. Click RADIUS - Modify RADIUS Profiles
3. Specify the RADIUS Profile object to modify either by typing in the object name or using the object selector.
4. (Optional) Specify or modify the RADIUS attributes for the Profile object.
5. Click OK.

Deleting RADIUS Profiles

1. In iManager, click the Roles and Tasks button
2. Click RADIUS - Delete RADIUS Profile
3. Specify the RADIUS Profile object to delete either by typing in the object name or using the object selector.
4. Click OK.

Chapter 3

Conclusion

Well, if you have managed to configure everything so far, it means your infrastructure is ready for very good user authentication. The last thing to do is to create the web pages that will generate user accounts, interact with your eDirectory to add the accounts and will explain the new users how to configure their computers to be able to use your network. All new (unauthenticated) user requests are redirected to the accounting web page and once they can authenticate they are sent trough the internal VLAN and that's it!

Hopefully the above info is somewhat correct by the time you read it, and at least serves as a guide, if not as a complete solution.

I would like to thank Todor Bukov for the initial lead-in on the topic.

Glossary

802.1x

An IEEE standard for port-based Network Access Control 3

AAA

Authentication, Authorization and Accounting Protocol 3

ACL

Access Control List 10

Active X

Distributed object system and protocol developed by Microsoft 5

ADSI

Active Directory Service Interfaces 5

AP

Wireless Access Point 15, 16

CA

Certificate Authority or Certification Authority 7–9, 16

DER

Distinguished Encoding Rules 8

DSML

Directory Service Markup Language 5

EAP

Extensible Authentication Protocol 3, 4, 8, 15, 17

eDirectory

An X.500 compatible directory service software product released in 1993 by Novell, Inc. for centrally managing access to resources on multiple servers and computers within a given network. 3–7, 10, 11, 13, 18, 19, 23

EJB

Enterprise Java Bean 5

FreeRADIUS

A free open source RADIUS server 3, 15

IEEE

Institute of Electrical and Electronics Engineers 4

IOS

Internetwork Operating System 3

JDBC

An API for the Java programming language that defines how a client may access a database 5

JNDI

Java Naming and Directory Interface 5

LDAP

Lightweight Directory Access Protocol 3, 5, 13, 19

ODBC

Open Database Connectivity 5

Perl

Practical Extraction and Report Language 5

PKCS

Public Key Cryptography Standards 10

RADIUS

Remote Authentication Dial In User Service 3

RFC

Request for Comments 4

SOAP

A protocol for exchanging XML-based messages over a computer network, normally using HTTP 5

SQL

Structured Query Language 3

SSL

Secure Sockets Layer 18

TACACS

Terminal Access Controller Access-Control System 3

TLS

Transport Layer Security 3, 8, 12, 18

VLAN

Virtual Local Area Network 3, 15, 16, 23

WPA

Wi-Fi Protected Access 17

XML

Extensible Markup Language 5

Bibliography

[Aboba and Simon, 1999] Aboba, B. and Simon, D. (1999). Ppp eap tls authentication protocol. http://tools.ietf.org/html/rfc2716.

[Bauer, 2005] Bauer, M. (2005). Securing WLANs with WPA and FreeRADIUS.
Part I http://www.linuxjournal.com/node/8017/print
Part II http://www.linuxjournal.com/node/8095/print
Part III http://www.linuxjournal.com/node/8151/print.

[FreeRADIUS, 2006] FreeRADIUS, T. P. (2006). FreeRADIUS. http://www.freeradius.org/.

[iManager, 2006] iManager, N. (2006).
urlhttp://www.novell.com/documentation/imanager26/index.html.

[Nortel, 2003] Nortel (2003). Securing the network infrastructure. http://www.nortel.com/solutions/security/snf.html.

[Novell, 2005] Novell (2005). Integrating novell edirectory with freeradius. http://www.novell.com/documentation/edir_radius/pdfdoc/radadmin/radadmin.pdf.

[Thomas, 2005] Thomas, R. (2005). Novell Contributes eDirectory Developer Interfaces to Open Source. http://www.linuxelectrons.com/article.php/20050214091355776.

[Wikipedia, 2006a] Wikipedia (2006a). Certificate authority. http://en.wikipedia.org/wiki/Certificate_Authority.

[Wikipedia, 2006b] Wikipedia, t. f. e. (2006b). IEEE 802.1X. http://en.wikipedia.org/wiki/802.1x.

[Wikipedia, 2006c] Wikipedia, t. f. e. (2006c). Novell eDirectory. http://en.wikipedia.org/wiki/EDirectory.