

Zentyal

for Network Administrators

VERSION 2.2



Preparation for the certification exam
Zentyal Certified Associate (ZeCA)

VERSION 2.2



Credits

❑ Produced by:



eBox Technologies S.L.
CEEI Aragon, Nave 19
C/ Maria de Luna 11
50018 - Zaragoza (Spain)
www.zentyal.com

❑ Copyright notice

Copyright © 2011 eBox Technologies S.L. All rights reserved. No part of this manual shall be reproduced, stored in a retrieval system, transmitted by any means, electronic, mechanical, photocopying, recording, or otherwise, or translated to any language without the written permission of eBox Technologies S.L. No patent liability is assumed with respect to the use of the information contained herein.

Although every precaution has been taken in the preparation of this training guide, eBox Technologies S.L. assumes no responsibility for errors or omissions. Nor is any liability assumed for damages resulting from the use of the information contained herein. The information provided is on an "as is" basis and no warranty or fitness is implied.

The copyright of this manual is owned by eBox Technologies S.L. Zentyal ® and the Zentyal logo are registered trademarks of eBox Technologies S.L. Other trademarks and registered trademarks referred to in this manual are the property of their respective owners, and are used for identification purposes only.

Index

1. Introduction to Zentyal	9
1.1. Presentation	9
1.1.1. SMBs and ITC	9
1.1.2. Zentyal: Linux server for SMBs	10
1.1.3. About this manual	12
1.2. Installation	12
1.2.1. Zentyal installer	13
1.2.2. Initial configuration	19
1.2.3. Hardware requirements	23
1.3. First steps with Zentyal	24
1.3.1. Administrative web interface of Zentyal	24
1.3.2. Location in a Zentyal network	30
1.3.3. Network configuration with Zentyal	31
1.3.4. Practical examples	37
1.4. Software updates	37
1.4.1. Management of Zentyal components	38
1.4.2. System updates	39
1.4.3. Automatic updates	40
1.4.4. Proposed exercises	40
1.5. Zentyal Cloud client	41
1.5.1. Subscribing Zentyal server to Zentyal Cloud (Basic Subscription)	41
1.5.2. Configuration backup in Zentyal Cloud	43
1.5.3. Other available services in the Basic Subscriptions	44
1.6. Self-assessment questions	46
2. Zentyal Infrastructure	47
2.1. Domain Name System (DNS)	47
2.1.1. DNS cache server configuration with Zentyal	51
2.1.2. Transparent DNS proxy	53
2.1.3. DNS forwarders	53
2.1.4. Configuration of an authoritative DNS server with Zentyal	54
2.1.5. Practical examples	57
2.1.6. Proposed exercises	59
2.2. Time synchronization service (NTP)	60
2.2.1. NTP Client Configuration	60
2.2.2. Configuring an NTP server with Zentyal	62
2.2.3. Practical examples	63

2.3. Network configuration service (DHCP)	64
2.3.1. DHCP server configuration with Zentyal	65
2.3.2. Practical examples	69
2.3.3. Proposed exercises	70
2.4. Certification authority (CA)	70
2.4.1. Public Key Infrastructure (PKI)	70
2.4.2. Importing certificates in clients	71
2.4.3. Certification Authority configuration with Zentyal	78
2.4.4. Practical examples	82
2.5. Web data publication service (HTTP)	83
2.5.1. HTTP server configuration with Zentyal	85
2.5.2. Practical examples	86
2.5.3. Proposed exercises	86
2.6. File Transfer Protocol (FTP)	87
2.6.1. Configuration of a FTP client	87
2.6.2. FTP server configuration with Zentyal	91
2.6.3. Practical examples	91
2.7. Virtualization manager	92
2.7.1. Creating virtual machines with Zentyal	93
2.7.2. Virtual machine maintenance	95
2.8. Self-assessment questions	97
3. Zentyal Gateway	99
3.1. High-level Zentyal abstractions	99
3.1.1. Network objects	100
3.1.2. Network services	102
3.1.3. Practical examples	103
3.1.4. Proposed exercises	104
3.2. Firewall	105
3.2.1. Firewall configuration with Zentyal	105
3.2.2. Port redirection with Zentyal	109
3.2.3. Practical examples	110
3.2.4. Proposed exercises	111
3.3. Routing	112
3.3.1. Configuring routing with Zentyal	112
3.3.2. Configuring traffic balancing with Zentyal	114
3.3.3. Configuring wan-failover in Zentyal	115
3.3.4. Practical examples	117
3.3.5. Proposed exercises	120
3.4. Quality of Service (QoS)	120
3.4.1. Quality of service configuration in Zentyal	121
3.4.2. Practical examples	122

3.4.3. Proposed exercises	123
3.5. Network authentication service (RADIUS)	123
3.5.1. Configuring an access point with RADIUS	123
3.5.2. Configuration of the RADIUS client	125
3.5.3. Configuring a RADIUS server with Zentyal	132
3.5.4. Practical examples	133
3.6. Captive portal	133
3.6.1. Configuring a captive portal with Zentyal	134
3.6.2. List of users	134
3.6.3. Bandwidth Monitor	135
3.6.4. Using the captive portal	136
3.6.5. Proposed exercises	136
3.7. HTTP Proxy Service	137
3.7.1. Configuring the web browser to use the HTTP Proxy	137
3.7.2. HTTP Proxy configuration in Zentyal	141
3.7.3. Blocking ads from the web	143
3.7.4. Limiting downloads with Zentyal	143
3.7.5. Content filtering with Zentyal	145
3.7.6. Practical examples	147
3.7.7. Proposed exercises	148
3.8. Self-assessment questions	149
4. Zentyal Unified Threat Manager	151
4.1. HTTP Proxy advanced configuration	152
4.1.1. Configuration of filter profiles	152
4.1.2. Filter profile per object	152
4.1.3. User group based filtering	153
4.1.4. User group based filtering for objects	154
4.1.5. Practical examples	155
4.1.6. Proposed exercises	156
4.2. Virtual private network (VPN) service with OpenVPN	156
4.2.1. Configuration of a VPN client	157
4.2.2. Configuration of a VPN server with Zentyal	160
4.2.3. Configuration of a VPN server for interconnecting networks	163
4.2.4. Practical examples	164
4.2.5. Proposed exercises	166
4.3. Virtual private network (VPN) service with IPsec	167
4.3.1. Configuring an IPsec tunnel in Zentyal	167
4.4. Virtual private network (VPN) service with PPTP	168
4.4.1. PPTP client configuration	169
4.4.2. Configuring a PPTP server in Zentyal	174
4.5. Intrusion Detection System (IDS)	175

4.5.1. Configuring an IDS with Zentyal	176
4.5.2. IDS Alerts	177
4.5.3. Practical examples	177
4.5.4. Proposed exercises	178
4.6. Self-assessment questions	178
5. Zentyal Office	181
5.1. Directory Service (LDAP)	181
5.1.1. Configuring Zentyal servers in master/slave mode	182
5.1.2. Configuring Zentyal as a slave of Windows Active Directory	184
5.1.3. Configuration of an LDAP server with Zentyal	187
5.1.4. User's corner	191
5.1.5. Practical examples	191
5.1.6. Proposed exercises	192
5.2. File sharing and authentication service	192
5.2.1. Configuring a file server with Zentyal	193
5.2.2. Samba client configuration	196
5.2.3. Configuring a Zentyal authentication server	197
5.2.4. PDC client configuration	198
5.2.5. Proposed exercises	199
5.3. Printers sharing service	200
5.3.1. Printer server configuration with Zentyal	200
5.3.2. Proposed exercises	204
5.4. Backup	204
5.4.1. Design of a backup system	204
5.4.2. Zentyal configuration backup	204
5.4.3. Data backup configuration in a Zentyal server	206
5.4.4. How to recover from a disaster	211
5.4.5. Proposed exercises	213
5.5. Self-assessment questions	214
6. Zentyal Unified Communications	215
6.1. Electronic Mail Service (SMTP/POP3-IMAP4)	216
6.1.1. SMTP/POP3-IMAP4 server configuration with Zentyal	218
6.1.2. Email client configuration	224
6.1.3. Practical examples	230
6.1.4. Proposed exercises	232
6.2. Mail filter	232
6.2.1. Mail filter schema in Zentyal	233
6.2.2. External connection control lists	239
6.2.3. Transparent proxy for POP3 mailboxes	239
6.2.4. Practical examples	240
6.2.5. Proposed exercises	241

6.3. Webmail service	241
6.3.1. Configuring a webmail in Zentyal	242
6.4. Groupware service (Zarafa)	243
6.4.1. Configuration of a groupware server (Zarafa) with Zentyal	243
6.4.2. Zarafa basic use cases	245
6.5. Instant Messaging Service (Jabber/XMPP)	248
6.5.1. Configuring a Jabber/XMPP server with Zentyal	248
6.5.2. Setting up a Jabber client	250
6.5.3. Setting up Jabber MUC (Multi User Chat) rooms	254
6.5.4. Practical examples	258
6.5.5. Proposed exercises	258
6.6. Voice over IP service	259
6.6.1. VoIP server configuration with Zentyal	263
6.6.2. Configuring a softphone to work with Zentyal	267
6.6.3. Using Zentyal VoIP features	269
6.6.4. Practical examples	269
6.6.5. Proposed exercises	270
6.7. Self-assessment questions	270
7. Zentyal maintenance	273
7.1. Logs	274
7.1.1. Zentyal log queries	274
7.1.2. Configuration of Zentyal logs	276
7.1.3. Log Audit for Zentyal administrators	277
7.1.4. Practical examples	278
7.1.5. Proposed exercises	279
7.2. Events and alerts	279
7.2.1. Events and alerts configuration in Zentyal	279
7.2.2. Practical examples	281
7.2.3. Proposed exercises	281
7.3. Monitoring	282
7.3.1. Metrics	282
7.3.2. Bandwidth Monitoring	285
7.3.3. Alerts	286
7.3.4. Proposed exercises	287
7.4. Support tools	288
7.4.1. Configuration report	288
7.4.2. Remote access support	288
8. Advanced use of Zentyal	291
8.1. Importing configuration data	291

Index

Zentyal 2.2 for Network Administrators

8.2. Advanced service customization	292
8.3. Development environment of new modules	294
8.4. Release policy	294
8.4.1. Zentyal Release Cycle	295
8.4.2. Support policy	296
8.5. Bug management policy	296
8.5.1. Patches and security updates	296
8.6. Technical support	297
8.6.1. Community support	297
8.6.2. Commercial support	297
8.7. Proposed exercises	297
Appendix A. Test environment with VirtualBox	299
A.1. About virtualization	299
A.2. VirtualBox	300
A.2.1. Creating a virtual machine on VirtualBox	300
A.2.2. Configuring a virtual machine on VirtualBox	305
A.2.3. VirtualBox snapshots	310
A.2.4. Adding an additional virtual hard disk	311
Appendix B. Advanced network scenarios	313
B.1. Scenario 1: Virtualized Zentyal server with Internet connection and access from the host machine and other client	313
B.2. Scenario 2: Virtualized Zentyal server with access from the host and another client with Internet connection through two gateways	316
B.3. Scenario 3: Virtualized Zentyal server with Internet connection and access from the host and two clients	317
B.4. Scenario 4: Virtualized Zentyal server connected to another virtualized Zentyal linking two separate networks	318
B.5. Scenario 5: Virtualized Zentyal server with Internet access, access from the host machine, clients in Internal and External networks	320
Appendix C. LVM	321
C.1. LVM	321
C.2. Practical examples	322
C.2.1. Practical example A	322
C.2.2. Practical example B	323
Appendix D. Answers to Self-Assessment	325

5. **ACTION.** Finally verify from *Dashboard* that the address appearing in the *DHCP leases widget* is displayed¹⁷.

2.3.3

PROPOSED EXERCISES

EXERCISE A

Configure DHCP service to always assign the same IP address to a host. Check that the host is working properly.

EXERCISE B

Change the maximum lease time. Change the parameters of **dhclient** in your configuration file to send a request with a lease greater than allowed. What happens? Check the lease contents in `/var/lib/dhcp3/dhclient.leases`.

EXERCISE C

Integrate DHCP and DNS modules to serve names of DHCP clients. Provide examples using static and dynamic domains by using static ranges and assignments.

2.4

CERTIFICATION AUTHORITY (CA)

2.4.1

PUBLIC KEY INFRASTRUCTURE (PKI)

Encryption technologies ensure authenticity, privacy and integrity in data transmission. However, the main problem in all the encryption schemes is how to distribute the keys to their users without interception by third parties. The solution consists of the use of a *Public Key Infrastructure*¹⁸ (PKI). This technology allows the use of the key in insecure medium while avoiding forgery, interception or modification of keys by anyone who snoops the communication.

PKI means each participant generates two keys: a *public key* and a *private key*. The public key can be distributed publicly and the private one must remain secret. Any participant who wants to encrypt a message can do it with his own private key and the recipient's public key. Therefore, the message can only be decrypted with the recipient's private key. Moreover, as the message has been encrypted with the sender's private key, it is possible to guarantee its integrity by checking the sender's public key.

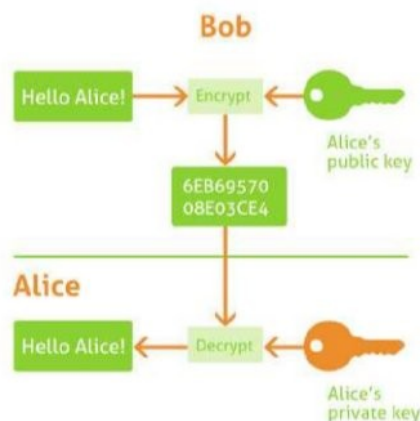


Image 2.30. Public key encryption.

¹⁷ Bear in mind that the static allocations do not appear in the DHCP widget.

¹⁸ http://en.wikipedia.org/wiki/Public_key_infrastructure.



Image 2.31. Public key signature.

However, this solution creates a new problem: if anyone could present a public key, how can you guarantee that a participant is really who he claims to be and is not impersonating another identity? To solve this problem, **certificates** were created¹⁹.

A certificate is a file that contains a public key, signed for someone that is trusted. This trusted participant is used to verify identities and is called **Certification Authority (CA)**²⁰.



Image 2.32. Diagram to issue a certificate.

Zentyal uses **OpenSSL**²¹ for the management of the Certification Authority and the life cycle of the issued certificates.

2.4.2

IMPORTING CERTIFICATES IN CLIENTS

To validate any certificate issued by a *Certification Authority* operated by Zentyal, you should import the certificate into the system.

¹⁹ http://en.wikipedia.org/wiki/Public_key_certificate

²⁰ http://en.wikipedia.org/wiki/Certification_authority

²¹ www.openssl.org/

In Windows XP you should click on *Start* → *Configuration* → *Control Panel*, and once in this window select *Network and Internet connections*.



Image 2.33. Control panel.

Next, select *Internet options*.



Image 2.34. Network and Internet connections.

You will access a new *Internet properties* window where you should select the *Content*: tab and go to *Certificates...*

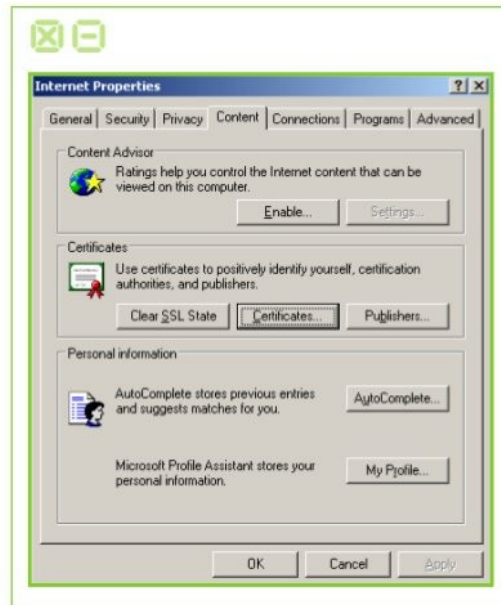


Image 2.35. Internet properties.

In the *Certificates* window you can see different tabs classifying the different types of stored certificates. To import your certificate, just click on *Import...*

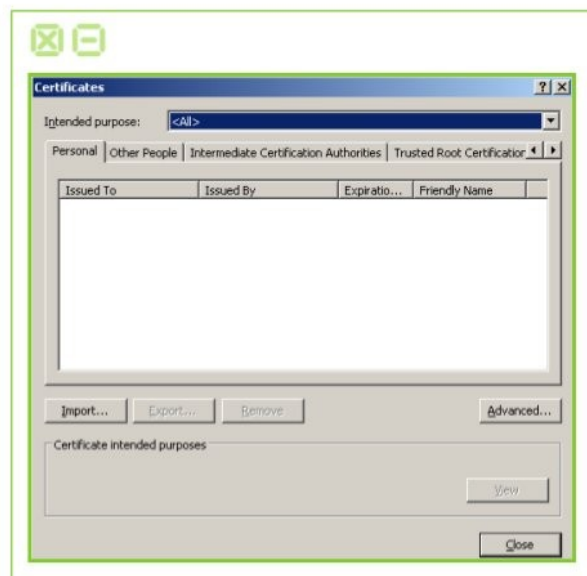


Image 2.36. Certificates.

A wizard will start to import new certificates. Just continue with the *Next* step.



Image 2.37. Certificate import wizard.

In *File name* select the path to the file with your certificate by using *Examine....* Once you have selected the certificate to import just continue to the next step.

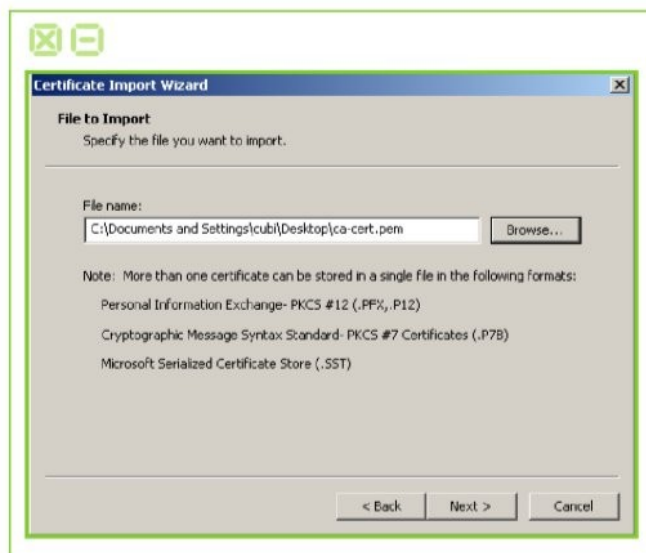


Image 2.38. Certificate import wizard.

On the *Certificate store* window select the *Automatically select the certificate store based on the type of certificate* option and continue again to the next step.

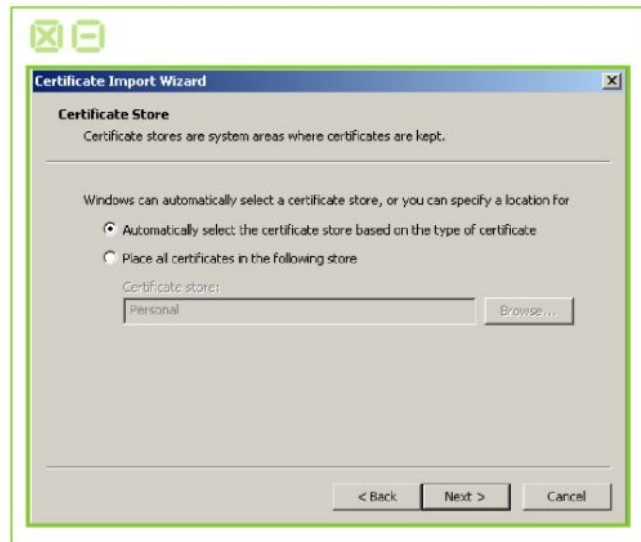


Image 2.39. Certificate import wizard.

You will see a summary of the actions to be executed and you just have to click on *Finish*.

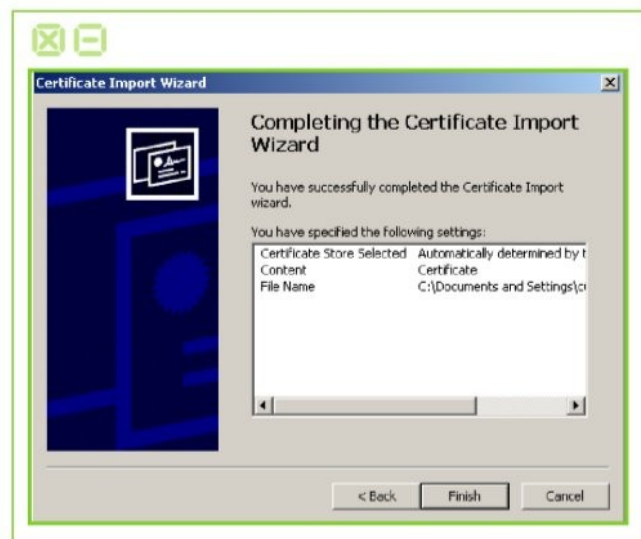


Image 2.40. Certificate import wizard.

If the import was successful, you will see a confirmation message.



Image 2.41. Certificate import wizard.

Verify that the certificate from your CA appears in the list of certificates.

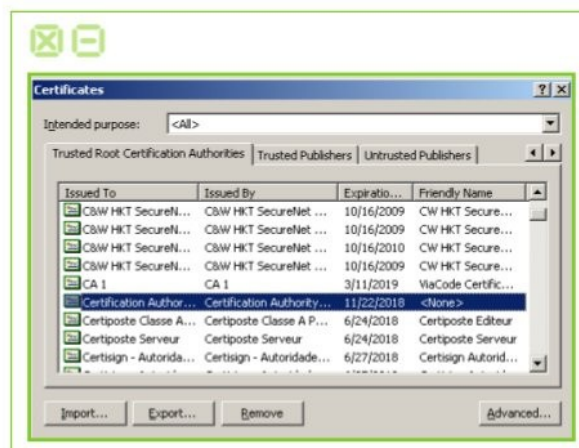


Image 2.42. Certificates.

You can also add a certificate from the CA to a web browser such as Mozilla Firefox. Let's see how to do this on Ubuntu.

The first step is to launch the browser and go to *Edit* → *Preferences*. In this window select the *Advanced* tab, then the *Encryption* tab and click on *View certificates*.



Image 2.43. Advanced preferences.

Much like in the previous case, a list of different types of certificates classified in different tabs will be displayed. Select the *Authorities* tab.



Image 2.44. Certificates from Authorities.

Proceed to *Import* the CA certificate by selecting the file where it is stored. The following window will be displayed to select in which situations you want to trust this new **Certification Authority**.

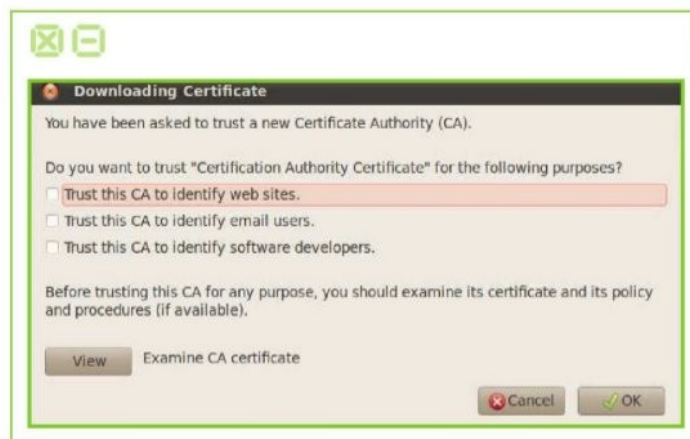


Image 2.45. Import new certificate.

By clicking **View** you will see the details of the certificate.



Image 2.46. Details of the certificate.

Once you have verified that the certificate is correct and you have selected the situations in which you are going to trust this CA, you just have to click on **Accept** and verify that the certificate appears on the list.



Image 2.47. Certificates.

2.4.3

CERTIFICATION AUTHORITY CONFIGURATION WITH ZENTYAL

In Zentyal, the *Certification Authority* module is self-managed, which means that it does not need to be enabled in *Module status*. However, you have to initialize the CA to make the functionality of the module available.

Go to *Certification Authority* → *General* and you will find the form to create the CA. You are required to fill in the *Organization Name* and *Days to expire* fields. Optionally, it is possible

to specify the *Country code* (a two-letter acronym following the ISO-3166-1 standard²²), *City* and *State*.

Image 2.48. Create the CA certificate.

When setting the expiration date you have to take into account that at the moment of expiration all certificates issued by this CA will be revoked, stopping all services depending on those certificates.

Once the CA has been initialised, you will be able to issue certificates. The required data are the *Common Name* of the certificate and the *Days to expire*. This last field is limited by the fact that no certificate can be valid for a longer time than the CA. In case you are using the certificate for a service such as a web server or mail server, the *Common Name* of the certificate should match the domain name of that server. For example, if you are using the domain name *zentyal.home.lan* to access the web administrative interface in Zentyal, you will need a certificate with the same *Common Name*. In case you are setting a user certificate, the *Common Name* will usually be the user's email address.

Optionally, you could set *Subject Alternative Names*²³ for the certificate. These are useful when setting common names to a certificate: a domain name or an IP address for a HTTP virtual host or an email address when signing email messages.

Once the certificate is issued, it will appear in the list of certificates and it will be available for the administrator and for the rest of modules. Through the certificate list you can perform several actions on the certificates:

- Download the public key, private key and the certificate.
- Renew the certificate.
- Revoke the certificate.
- Reissue a previously revoked or expired certificate.

²² http://en.wikipedia.org/wiki/ISO_3166-1

²³ For more information about subject alternative names, visit http://www.openssl.org/docs/apps/x509v3_config.html#Subject_Alternative_Name



Image 2.49. Certificate list page.

The package with the keys contains also a PKCS12 file with the private key and the certificate and it can be installed directly into other programs such as web browsers, mail clients, etc.

If you renew a certificate, the current certificate will be revoked and a new one with the new expiration date will be issued. And if you renew the CA, all certificates will be renewed with the new CA trying to keep the old expiration date. If this is not possible because it is after the date of expiry of the CA, then the date of expiration is set as the one of the CA.



Image 2.50. Renew a certificate.

If you revoke a certificate you will not be able to use it anymore as this action is permanent and it cannot be undone. Optionally, you can select the reason of the certificate revocation:

- **unspecified:** reason non specified,
- **keyCompromise:** the private key has been compromised,
- **CACompromise:** the private key for the certification authority has been compromised,
- **affiliationChanged:** the issued certificate has changed its affiliation to another certification authority from other organization,
- **superseded:** the certificate has been renewed and it is now replaced by a new one,
- **cessationOfOperation:** the certification authority has ceased its operations,
- **certificateHold:** certified suspended,
- **removeFromCRL:** currently unimplemented, it provides delta CRLs support, that is, lists of certificates whose revoked status has changed.



Image 2.51. Revoke a certificate.

When a certificate expires all the modules are notified. The expiration date of each certificate is automatically checked once a day and every time you access the certificate list page.

Services Certificates

On *Certification Authority* → *Services Certificates* you can find the list of Zentyal modules using certificates for their operation. Each module generates its own self-signed certificates, but you can replace them with others issued by your CA.

You can generate a certificate for each service by defining its *Common Name*. If a previous certificate with the name does not exist, the CA will create it automatically.



Image 2.52. Services Certificates.

Once enabled, you need to restart the service to force the module to use the new certificate. This also applies if you renew a certificate for a module.

TIP. If you want to use certificates signed by a commercial Certification Authority, you have to follow the usual procedure to generate a private key and then a CSR (Certificate Signing Request) so they will forward the signed certificate that you will use in the service. Let's see an example on how to do this for your mail server:

1.- Generate the private key

```
openssl genrsa -out dominio.tld.key 2048
```

2.- Using the private key, generate the CSR:

```
openssl req -new -key dominio.tld.key -out dominio.tld.csr
```

3.- Send the CSR file to the Certification Authority, which will return a certificate that you will save in a file named domain.tld.crt.

4.- Check where to store the certificate in the configuration files of the desired service; in the case of Postfix, this directory is `/etc/postfix/sasl/postfix.pem`, so proceed to overwrite that file setting exclusive read permissions for the root user:

```
cat dominio.tld.crt dominio.tld.key > /etc/postfix/sasl/postfix.pem
chmod 400 /etc/postfix/sasl/postfix.pem
```

5.- You can proceed in the same way for Zarafa:

```
cat dominio.tld.crt dominio.tld.key > /etc/zarafa/ssl/ssl.pem
chmod 400 /etc/zarafa/ssl/ssl.pem
```

6.- It is a good practice to save a backup of the private key and the certificate, and also double-check that all files containing the private key can only be read by the root user and/or the username used by the service.

2.4.4

PRACTICAL EXAMPLES

□ PRACTICAL EXAMPLE A

The company ContaFoo SL is implementing security protocols in internal communications to comply with current legislation. The various intranets will work under HTTPS and email will use SSL/TLS, but for that they need to import the certificate of the **Certification Authority** operated in their Zentyal server. Create the CA and then import its certificate in the Windows XP clients.

1. **ACTION.** Go to *Certification Authority* → *General*. In the form called *Create Certification Authority Certificate*, fill in the fields *Organization Name* and *Days to expire* with reasonable values. Press *Create* to generate the Certification Authority.

EFFECT. The key pair of the Certification Authority is generated and its certificate will be issued. Your new CA will be displayed in the certificate list. The form for creating the CA will be replaced by another one intended to issue normal certificates.

2. **ACTION.** From the certificate list, you will download the one of the CA, a file with a name such as *CA-key-and-cert.tar.gz* containing the public key *ca-public-key.pem* and the certificate *ca-cert.pem*. Following the previously described procedure, import the certificate file *ca-cert.pem* on the Windows XP clients.

EFFECT. The new certificates will appear in the certificate list, and any certificate issued by this CA will be accepted by the Windows XP clients.