

Installazione EJBCA su Ubuntu:

Prerequisiti

Add Ubuntu user with username 'user' during install or another!. The path /home/user means the user's home directory below or another

```
sudo adduser ejbcauser
```

insert your password

Installare le applicazioni come root...

JDK 7 OpenJDK or Oracle JDK, if available OpenJDK is recommended:

```
sudo apt-get install openjdk-7-jdk
```

Settare le variabili di ambiente per java in /etc/environment:

- `vi nano /etc/environment`
- In this file, add the following line (replacing YOUR_PATH by the just copied path):
- `JAVA_HOME="YOUR_PATH"`
- That should be enough to set the environment variable. Now reload this file:
- `source /etc/environment`
- Test it by executing:
- `echo $JAVA_HOME`
- If it returns the just set path, the environment variable has been set successfully. If it doesn't, please make sure you followed all steps correctly.

Setup JCE per JAVA

EJBCA makes use of strong crypto and keystore passwords longer than 7 characters. For this to work you must install the 'Unlimited Strength Jurisdiction Policy Files' for JDK. The policy files can be found at the same place as the JDK download at [Oracle](#). The text "Using exportable cryptography" is shown on the first page in the Admin GUI if you fail to install this package.

Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files for your JDK ([download for Oracle's JDK](#), not required for OpenJDK).

Download the version that matches your installed JVM E.g. UnlimitedJCEPolicyJDK7.zip from the address: <http://www.oracle.com/technetwork/java/javase/downloads/jce-7-download-432124.html> Accept the license and:


[Overview](#)
[Downloads](#)
[Documentation](#)
[Community](#)
[Technologies](#)
[Training](#)

Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files 7 Download

Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files 7

You must accept the [Oracle Binary Code License Agreement for the Java SE Platform Products](#) to download this software.

Thank you for accepting the Oracle Binary Code License Agreement for the Java SE Platform Products; you may now download this software.

Product / File Description	File Size	Download
Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files 7	7.3 K	 UnlimitedJCEPolicyJDK7.zip

- Unzip the downloaded zip
- Copy local_policy.jar and US_export_policy.jar to the \$JAVA_HOME/jre/lib/security (Note: these jars will be already there so you have to overwrite them)

Installare gli altri prerequisiti :

```
sudo apt-get install ant ant-optional unzip ntp
```

Cercare i pacchetti da scaricare: JBOSS ed EJBCA

Trovare il file da scaricare per l'application server :

JBoss Application Server 7.1.x or later or JBoss EAP 6 or later ([download](#), note that the absolute latest may not always be tested).

<http://jbossas.jboss.org/downloads.html>

copy path of download

<http://download.jboss.org/jbossas/7.1/jboss-as-7.1.1.Final/jboss-as-7.1.1.Final.zip>

Trovare le versioni di ejbca:

<http://sourceforge.net/projects/ejbca/files/ejbca6/>

Utilizzata la versione 6.0.4 per l'installazione, la dimensione si aggira sui 37 MB.

http://sourceforge.net/projects/ejbca/files/ejbca6/ejbca_6_0_4/ejbca_ce_6_0_4.zip

(If you are unsure what version of EJBCA you are running, type 'ant ejbcaversion' in the EJBCA_HOME directory)

Partire poi dal punto 4 dell'installazione seconda parte!

INSTALLAZIONE SECONDA PARTE

Ubuntu quick start

This quick start guide assumes EJBCA 6.0, JBoss 7.1.1.Final and Java 7, but other version should also be possible to use by just replacing the versions. We also assume installation is made in a user account with username "user". In your environment simply replace user with the username you are using.

Note that the quick install will create a H2 database as ~/ejbcadb. If you run this multiple times, delete this file in order to remove the old database.

La parte grigia dovreste averla già fatta se avete seguito gli step del paragrafo prima...

1. Install Ubuntu 12.04 server x64, default config, only OpenSSH server selected (or other Ubuntu of your choice, for example Ubuntu Desktop).
 - o Add user with username 'user' during install. The path /home/user means the user's home directory below. Il mio user è ejbcauser
2. Open a new terminal "ejbca".
3. Install needed software from Ubuntu repositories.
 - o `sudo apt-get install openjdk-7-jdk ant ant-optional unzip ntp`
 - o ricordarsi di settare jce per javajdk7

=====

4. Install software not in Ubuntu repositories, JBoss 7.1.1.Final and EJBCA 6.0.

Scaricare ejbca e jboss già nella path giusta, ossia /home/utentedisistemacreato.

Es. root@ejbca:/home/ejbcauser

Posizionarsi in /home/<myuserEjbca>

Di seguito i wget utilizzati con le versioni attuali, presa la versione ejbca_ce_6_0_4 (ne esistono anche di più recenti come la versione ejbca_6_2_0), mentre per jboss utilizzare la 7.1.1 Final:

wget <http://download.jboss.org/jbossas/7.1/jboss-as-7.1.1.Final/jboss-as-7.1.1.Final.zip>

wget http://sourceforge.net/projects/ejbca/files/ejbca6/ejbca_6_0_4/ejbca_ce_6_0_4.zip

- o In caso li abbiate scaricati in un'altra folder copiarli sotto lo user creato es:

```
cp jboss-5.1.0.GA-jdk6.zip \home\user
cp ejbca-5.1.0.GA-jdk6.zip \home\user
```

Unzippare i pacchetti appena scaricati:

Se non installato unzip fare `sudo apt-get install unzip`

- o `unzip jboss-as-7.1.1.Final.zip`

- unzip ejbca_6_0_4.zip

Configurare JBOSS:

Prima di modificare il file originale standalone.conf, è consigliabile farsene una copia. Anche perchè se si commentano alcune parti esse vengono poi rimosse dal file.

1. Abilitare nel file "{JBoss_dir}/bin/standalone.conf" la riga seguente per l'accesso debug da parte di Eclipse
JAVA_OPTS="\$JAVA_OPTS -Xrunjdwp:transport=dt_socket,address=8787,server=y,suspend=n"
2. Verificare che il proprio server JBoss sia accessibile dall'esterno aprendo il file {JBossdir}/standalone/configuration/standalone.xml e impostando opportunamente gli IP nei tag <inet-address> (ad esempio con l'indirizzo universale 0.0.0.0).

Verificare jboss se si vede sulla porta 8080

Settare jboss_home se si vuole.

sotto jboss_home/bin

lanciarlo ./standalone.sh

verificare che sul browser si veda la page di default di jboss; quando è ok proseguire..

Configurare JBOSS_HOME come variabile di ambiente sotto /etc/environment.

Configure EJBCA

- **Da saltare** → echo "appserver.home=/home/user/jboss-as-7.1.1.Final" >> ejbca_6_0_3/conf/ejbca.properties

Compilare appserver.home di jboss poi direttamente in uno dei file di properties di ejbca.

CONFIGURARE DIRETTAMENTE IL FILE DI PROPERTIES RELATIVO ejbca.properties

posso evitare questo comando impostandolo manualmente nel file ejbca_6_0_4/conf/ejbca.properties. Questo file insieme a install.properties, jaxws.properties, web.properties sono stati ottenuti copiandoli dai rispettivi sample presenti nella stessa cartella

Guardare come sono stati configurati i file di properties per trarne spunto...

Configurare bene modificando opportunamente come ad esempio il proprio hostname etc...

Vedi di seguito spiegazioni di come si popolano i files di properties di ejbca.

Configure

*** Configuration files ***

The configuration of EJBCA that can not be configured in the Admin GUI is located in properties files in the *conf* directory. All properties are documented in sample files and to configure an option you copy the sample file, for example copy *conf/ejbca.properties.sample* to *conf/ejbca.properties* and configure *conf/ejbca.properties*. You should at least familiarize your self with the options in *conf/install.properties* and *conf/ejbca.properties*. **Most options, except those in *install.properties* can be changed after installation.**

*** EJBCA configuration ***

1) Copy *conf/install.properties.sample* to *conf/install.properties* and *conf/ejbca.properties.sample* to *conf/ejbca.properties* Customize if needed. **The default values works fine for a test installation.**

You must configure 'appserver.home' in *ejbca.properties* to point to your application server directory. You find examples of how to do this in *ejbca.properties.sample*.

This makes libraries from the application server available to EJBCA during the build.

If you are only testing EJBCA at this stage and is not setting up a production environment, you can skip the rest of this step. There are default configuration options, that should work in a test environment, for everything.

- Customize the CA properties in *conf/ejbca.properties* if you need to do so. For production use you need to do this, don't forget to edit passwords to be secure and secret. Keep *conf/ejbca.properties* as secret as possible. DO NOT forget the passwords, if you need to re-install the software sometime.
- To use a hard ca token from start change *ca.token*, *ca.tokenpassword* and *ca.tokenproperties* in *install.properties*. You also need to add the appropriate values to the *ca.tokenproperties* file for the HSM. Read the HSM documentation for the right values.
- To put the initial superadmin certificate on a smartcard, set *superadmin.batch=false* in *web.properties*. Enroll from public web after the installation is complete, as you would with any other smartcard user. Username is "superadmin" and password is *superadmin.password* from *web.properties*.
- If you are deploying on JBoss EAP you probably want to look at the property 'jboss.config' as well, since 'production' may be the default server to start on JBoss EAP (depends on your configuration).

Do the same with other configuration files that you might want to customize. The default values often works fine and is a safe bet if you are unsure. Most options are well documented in the sample files.

Posizionarsi in: `/home/ejbcauser/ejbca_ce_6_0_4/conf`

E di qui fare una copia dei file che servono ad esempio:

Copiare file/directory: `cp (nome del file o della directory) (directory o nome del file di destinazione)`

Copiare i seguenti file di sample e metterli con i nuovi così come di seguito:

```
sudo cp install.properties.sample install.properties
sudo cp ejbca.properties.sample ejbca.properties
sudo cp jaxws.properties.sample jaxws.properties
sudo cp web.properties.sample web.properties
sudo cp cesecore.properties.sample cesecore.properties
```

Prima di passare al punto 5 popolare correttamente i files, **vedi paragrafo File di properties example**
Dopo che sono stati configurati procedere con il deploy e l'installazione vera e propria dal punto 5.

Well I think the error message is clear. You should not use the same DN for your end entity as the DN for the CA. In any PKI that is not recommended.

User 'tomcat' is not allowed to use same subject DN as the user(s).... Problema di DN uguali.... Nel file di properties...

Vedi dettagli nel **paragrafo § File di properties Example** sotto su come devono essere configurati i vari campi.

=====

5. Open new terminal "jboss" and start JBoss. **TERMINALE1: JBOSS**
 - o jboss-as-7.1.1.Final/bin/ comando ./standalone.sh

aspettare che sia started... poi:
6. Build and deploy EJBCA to JBoss. **TERMINALE2: EJBCA**
 - o cd ejbca_6_0_3
 - o ant deploy (**just press enter if questions show up**)
 - o (now wait a little for JBoss to reload)

Verrà deployato ejbca.ear in jboss che in deployment avrà quindi ejbca.ear e relative file ejbca.deployed

Controllare che non ci siano errori su **TERMINALE1: JBOSS** se ci sono non proseguire con lo step successive ma risolvere prima tutti i bugs....

Errore che si può bypassare che è un errore di jboss:

08:08:26,746 ERROR [org.hibernate.internal.util.xml.ErrorLogger] (MSC service thread 1-3)
HHH000196: Error parsing XML (21) : cvc-complex-type.3.1: Value '1.0' of attribute 'version' of element 'entity-mappings' is not valid with respect to the corresponding attribute use. Attribute 'version' has a fixed value of '2.0'.

Una volta che avete

BUILD SUCCESSFUL su **TERMINALE2: EJBCA**
Verificare anche che sia deployato tutto bene anche su **TERMINALE1: JBOSS**

Proseguite:

7. **TERMINALE2: EJBCA** Run install (in terminal "ejbca") to create initial Management CA and TLS keystores.
 - o ant install **(choose all default values)**

Note: If ant install fails with errors this is typically due to deployment or start of JBoss that did not complete successfully. You will need to look in the server logs to find out why it will not start correctly in your environment, See the [Troubleshooting](#) section for more information.

In caso di errori a questo step....

BUILD FAILED Case:

Stappare jboss **TERMINALE1: JBOSS**

Rimuovere nel deployment di jboss i file ejbca.ear e ejbca.deployed
Andare a rimuovere il db h2 creato, rinominare il file se non si vuole rimuovere:
I file si trovano ...
sotto la root ci sono due file ejbcadb.h2.db e ejbcadb.trace.db ... toglierli o rinominarli

Note that the quick install will create a H2 database as ~/ejbcadb. If you run this multiple times, delete this file in order to remove the old database.

Togliere le key generate se ci sono sotto la folder di ejbca/p12:

- **tomcat.jks** - Holds the actual certificate (and its signing chain) used by jboss to secure the ejbca web portals with TLS.
- **truststore.jks** - Stores a copy of the root CA key that issued the TLS certificate (initially this is the management CA).
- **superadmin.p12** - file that contains the client certificate used to authenticate the default administrator account is also located in /opt/ejbca/p12. It is not copied to the jboss directory with the other keystores.

To clarify what each password is for:

- The "truststore with the CA certificate for https" is the truststore.jks file.
- The "keystore with the TLS key for https" is the tomcat.jks file.
- The "superadmin password" is the password for the superadmin.p12 file.

After creation, these files are copied by the install script to /opt/jboss/standalone/configuration/keystore, and tomcat.jks is renamed keystore.jks in the new directory.

It is critical to understand that while ant creates the keystores in /opt/ejbca/p12, jboss uses the keystores in the /opt/jboss/standalone/configuration/keystore directory for TLS.

e rimuoverle in jboss in caso ci siano in ... jboss_home/standalone/configuration/keystore

Controllare anche il file standalone.xml in caso si abbiano ancora problemi...

(Unfortunately, ejbca keeps clear copies of the keystore passwords in standalone.xml), in caso mettere il file originale...

Una volta ripulito bene, RIFARE TUTTO ossia :

Il deploy di ejbca su jboss started ovviamente e poi ant install fino a che tutto non ha BUILD SUCCESSFULL

Dopo che tutto è ok:

8. Go back to terminal "jboss" **TERMINALE1: JBOSS** and restart JBoss.
 - o ctrl-c
 - o jboss-as-7.1.1.Final/bin/standalone.sh
9. Copy /home/user/ejbca_6_0_3/p12/superadmin.p12 to admin desktop machine and import in web browser (Utilizzare ad esempio Filezilla per trasferirsi il file...)
10. Su chrome importare il certificato superadmin.p12 due volte, una come root trusted etc e una come personale... si vedranno anche due nomi diversi uno specifico e uno SuperAdmin.

(Importare nel browser in impostazioni-mostra impostazioni avanzate- HTTPS/SSL gestione certificati-personale ed in autorità di certificazione radice attendibile(come password usare ejbca)

Confermare i vari popup di root trusted import...

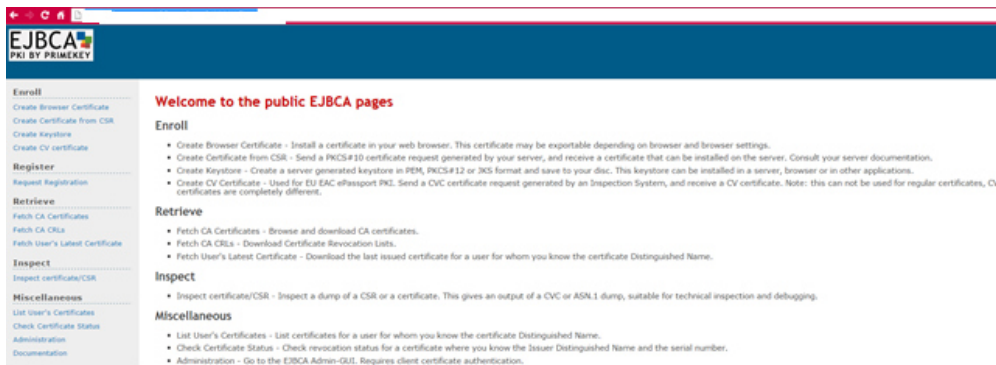
Inserire la chiave messa per superAdmin! Nel file di properties... dovrebbe essere in web.properties file..

Ora verificare che tutto funziona correttamente:

11. Open URL https://server:8443/ejbca/adminweb, where 'server' is the servers name/ip.
12. L'altra page è http://server:8080/ejbca
13. Configure JBoss for logging, see [JBoss 7 and EAP 6 logging](#) for instructions.

Done!

See web interface below like example:



Optional → After installation **do not** forget to **secure** your installation as described in [securing JBoss](#).

JBOSS AS A SERVICE

Una volta che tutto è installato bene, conviene mettere jboss come servizio di Sistema.

EJBCA → File di properties **Example**

Install properties:

----- Administrative CA configuration -----

This installation will create a first administrative CA. This CA will be used to create the first superadministrator and for the SSL server certificate of administrative web server.

When the administrative web server have been setup you can create other CA:s and administrators.

This is only used for administrative purposes,

Enter a short name for the administrative CA.

#ca.name=ManagementCA

ca.name= AdminCA (ad es. Super Admin)

The Distinguished Name of the administrative CA.

This is used in the CA certificate to distinguish the CA.

Note, you can not use DC components for the initial CA, you can create CAs

using DC components later on once the admin GUI is up and running.

#ca.dn=CN=ManagementCA,O=EJBCA Sample,C=SE

ca.dn=CN= AdminCA-hostname.it,O=prisma,C=IT (ad es. CN=Super Admin)

Qui prima dell'opzione O, messo anche OU=MYOrganization (sta per organizzazione..)

ca.tokenype=soft

ca.tokenpassword=null

ca.keyspec=2048

ca.keytype=RSA

ca.signaturealgorithm=SHA1WithRSA

ca.validity=3650

ca.policy=2.5.29.32.0

3650 corrisponde a 10 anni di validità

La policy messa quella suggerita nei commenti nel file...

Altro esempio di un install.properties:

install.properties

```
### Start install.properties ###
```

```
# In every case that "CA" is mentioned in this file, it refers to the  
"management" CA ONLY.
```

```
# This will be the initial name of the management CA instance  
# ejbca will use this for administration purpose, not your production CAs  
# Note that the CN given here is NOT the FQDN of your CA!  
# Why does this matter? This certificate will be temporarily installed  
# on your browser as a trusted root CA, but will not be communicated  
# with.  
ca.name=mgmtca  
ca.dn=CN=mgmtca,O=Your Company,C=US  
ca.tokenType=soft  
ca.tokenPassword=null
```

cesecore.properties :

```
# This password is used internally to protect CA keystores in database (i.e. the CAs private key).  
# foo123 is to keep compatibility with default installations of EJBCA 3.0, please change this if possible  
# Note! If changing this value AFTER installation of EJBCA you must do 'ant clean; ant bootstrap' in order to  
activate changes.  
#ca.keystorepass=foo123  
#ca.keystorepass=!secret!  
ca.keystorepass=Inserire la pwd che si vuole
```

web.properties:

```
# Password for java trust keystore (p12/truststore.jks). Default is changeit  
# This truststore will contain the CA-certificate after running 'ant javatruststore'  
# Run 'ant -Dca.name=FooCA javatruststore' to install the CA-certificate for FooCA instead of the default  
ManagementCA  
java.trustpassword=changeit
```

```
# The CN and DN of the super administrator.
```

```
# Comment out if you want 'ant install' to prompt for this.
```

```
superadmin.cn=SuperAdmin
```

```
# Note that superadmin.dn must start with the same CN as in superadmin.cn.
# example: superadmin.dn=CN=${superadmin.cn},O=EJBCA Sample,C=SE
superadmin.dn=CN=${superadmin.cn}
```

```
# The password used to protect the generated super administrator P12 keystore (to be imported in
browser).
# Choose a good password here.
#superadmin.password=ejbca
superadmin.password=Inserire la pwd che si vuole
# Set this to false if you want to fetch the certificate from the EJBCA public web pages, instead of
# importing the P12-keystore. This can be used to put the initial superadmin-certificate on a smart card.
superadmin.batch=true
```

```
# The password used to protect the web servers SSL keystore. Default is serverpwd
# Choose a good password here.
# If upgrading from EJBCA 3.1, enter here the password found in
# $JBOSS_HOME/server/default/deploy/jbossweb-tomcat55.sar/server.xml
# under the section about 'HTTPS Connector...', the password is in attribute 'keystorePass=...'.
#httpserver.password=serverpwd
httpserver.password=Inserire la pwd che si vuole
```

```
# The CA servers DNS host name, must exist on client using the admin GUI.
#httpserver.hostname=localhost
httpserver.hostname=hostnameAddress → stesso nome sul dns, per convenzione le machine saranno
ca01.miodominio.it ed a seguire ca02.miodominio.it, ca03...
```

```
# The Distinguished Name of the SSL server certificate used by the administrative web gui.
# The CN part should match your hosts DNS name to avoid browser warnings.
#httpserver.dn=CN=${httpserver.hostname},O=EJBCA Sample,C=SE
httpserver.dn=CN=${httpserver.hostname},O=prisma,C=IT
```

Qui prima dell'opzione O, messo anche OU=MyOrganization (sta per organizzazione..)

```
javaxws.properties
ejbcaws.enabled=true
```

```
ejbca.properties
```

```
appserver.home=/home/<myusercreated>/jboss-as-7.1.1.Final
appserver.type=jboss
```

JBOSS ERROR SECTION:

Deployment Error Messages

You'll also see various log messages showing errors on compilation. The following errors can be ignored - they're bugs in jboss:

```
06:05:16,848 ERROR [org.jboss.as.controller.management-operation]
(management-handler-thread - 1) JBAS014612: Operation ("composite")
```

failed - address: ([]): java.lang.IllegalArgumentException

06:05:39,477 ERROR [org.hibernate.internal.util.xml.ErrorLogger] (MSC service thread 1-4) HHH000196: Error parsing XML (21) : cvc-complex-type.3.1: Value '1.0' of attribute 'version' of element 'entity-mappings' is not valid with respect to the corresponding attribute use. Attribute 'version' has a fixed value of '2.0'.

15:29:58,915 SEVERE

[javax.enterprise.resource.webcontainer.jsf.application] (MSC service thread 1-2) JSF1051: Service entry 'org.jboss.as.web.deployment.jsf.JsfInjectionProvider' does not extend DiscoverableInjectionProvider. Entry will be ignored.

This message refers to the fact that the community version of ejbca does not support database integrity protection:

06:06:13,175 INFO [org.cesecore.dbprotection.ProtectedData] (MSC service thread 1-4) No database integrity protection available in this version of EJBCA.

You'll always see these errors no matter what you do. Any other errors and failures should be dealt with before trying to proceed to an installation. But eventually you'll see something like these messages if the deployment is successful:

01:38:38,724 INFO [org.jboss.as] (MSC service thread 1-1) JBAS015874: JBoss AS 7.1.1.Final "Brontes" started in 7761ms - Started 2855 of 2968 services (111 services are passive or on-demand)
01:38:38,769 INFO [org.jboss.as.server] (DeploymentScanner-threads - 2) JBAS018559: Deployed "ejbca.ear"

Finally, note that you will see tons of the following messages returned directly by **ant** during every operation:

```
appserver.error.message:  
    [echo] jndi.properties.file:  
    /opt/ejbca_ce_6_1_1/conf/jndi.properties.jboss7
```

These messages can be safely ignored.

[Di seguito spiegazioni generiche sulle CA :](#)

CA Naming and Certificates

We will always have at least two CAs on a production server because ejbca uses a "Management CA" for administration purposes. Each CA will have its own unique X.500 CN field information.

- I will set the CN of the management CA to be "**mgmtca**", since it is purely an internal CA that will never be resolved via DNS.

This being said, there is a very important thing to understand about the naming of the server and the certificates it will use for administrative purposes.

When you are using the web interface of your ejbca server, it uses a TLS certificate to encrypt HTTPS connections to the web service. The initial version of this certificate will be a "self-signed" one created during ejbca installation. But toward the end of this how-to, we will replace this certificate with one issued by a "Production" CA.

- We do this to ensure that the server itself participates in the PKI that we establish.

Assuming that our "Production" CA will use the FQDN of the server in the CN field of its root certificate, this implies that we will have two certificates issued with the same CN, but for different purposes:

- A "Root CA Certificate" used to identify the Production CA (rootca.yourcompany.net) and to sign new certificates
- A "Server Certificate" used to establish TLS to the web administration pages (at https://rootca.yourcompany.net)

Understanding that there are two certificates, and the purpose of each, is required to have a healthy and sane experience when building an ejbca server!

- The difference between a DNS hostname and a FQDN.
- The basics of PKI, at least to the point of knowing how [root chain validation](#) works.
- A minimum of X.500 notation:
 - CN = Common Name, usually is the FQDN of your CA
 - DN = Distinguished Name, which is the CN followed by information about the organization that owns the CA
 - O = Organization, usually is your company name, and can include spaces
 - C = Country, in ISO 3166-1 alpha-2 format (US, CA, SE, MX, etc)

Example:

```
[echo]
[echo] ----- CA Properties -----
[echo] ca.name                : mgmtca
[echo] ca.dn                   : CN=mgmtca,O=Your Company,C=US
[echo] ca.token_type           : soft
[echo] ca.key_type             : RSA
[echo] ca.key_spec             : 4096
[echo] ca.signature_algorithm  : SHA256WithRSA
[echo] ca.validity              : 3650
[echo] ca.policy               : null
[echo] ca.token_properties     : /opt/ejbca/conf/catoken.properties
[echo] httpserver.hostname     : rootca.yourcompany.net
[echo] httpserver.dn           : CN=rootca.yourcompany.net,O=Your
Company,C=US
```

```
[echo] superadmin.cn           : superadmin
[echo] superadmin.dn           : CN=superadmin,O=Your Company,C=US
[echo] superadmin.batch         : true
[echo] appserver.home           : /opt/jboss
[echo]
```

LINK UTILI:

Dal link : <http://ejbcacentos.blogspot.it/2014/04/how-to-install-ejbca-611-on-centos-65.html>

Dal link: <http://sourceforge.net/p/ejbca/mailman/message/4498419/>

<http://ejbcacentos.blogspot.it/>

Other link:

<http://majic.rs/book/free-software-x509-cookbook/setting-up-ejbca-as-certification-authority>