

Certification Authority

{scrollbar}

This release of Apache Geronimo allows you to define your own Certification Authority (CA) and issue certificates in reply to Certificate Signing Requests (CSR). The Certification Authority portlet is available by clicking **Certificate Authority** on the left menu in the Geronimo Administration Console.

- [#Configuring a Certification Authority](#)
- [#Signing certificate requests](#)

Configuring a Certification Authority

The first time you call this portlet the CA is not yet configured so you will see a screen similar to this one.

Certification Authority	[view]
<p>This portlet allows you to setup a Certification Authority (CA) and issue certificates in reply to Certificate Signing Requests (CSRs). <i>Setup Certification Authority</i> function allows to initialize the CA by providing CA Identity details, algorithm parameters for CA's key pair and self-signed certificate and a password to protect the CA's private key. This password is to be used to unlock the CA to access CA functions. Once the CA is initialized, CSRs can be processed using <i>Issue New Certificate</i> function. Previously issued certificates can be viewed using <i>View Issued Certificate</i> function.</p> <p>CA is not running or the CA may not have been initialized. Please initialize the CA using the link provided below.</p> <p>Setup Certification Authority</p>	

Click on **Setup Certification Authority** to configure Geronimo as a CA.

This process is somewhat similar to defining keystores and certificates as covered in the [Administering certificates](#), this is in the sense that you should be prepared to provide similar type of information.

The first step is defining the Certification Authority details as illustrated in the following image. The information entered in this form will be used to create the Certification Authority and respective self-signed key pairs.

Certification Authority

[\[view\]](#)

Setup Certification Authority - Step 1: Enter CA details

On this screen you can enter the Certification Authority (CA) details, algorithm parameters for CA's keypair, algorithm for CA's self signed certificate and a password to protect the CA's private key. The next screen will let you review this information before generating the CA's keypair and self-signed certificate.

Certification Authority's Identity

Common Name (CN):
Division/Business Unit (OU):
Company/Organization (O):
City/Locality (L):
State/Province (ST):
Country Code (2 char) (C):

Key Details

Alias:
Key Algorithm:
Key Size:
Password:
Confirm Password:

Certificate Details

Certificate Serial Number:
Valid From Date(mm/dd/yyyy):
Valid To Date(mm/dd/yyyy):
Signature Algorithm:

[Cancel](#)

This is an "information gathering" step, at this point you are not creating any certificates yet. Click on **Review CA Details** and then on **Setup Certification Authority**.

Once created you will see a confirmation message **CA Setup is successful!** along with the details for the certificate you just created.

Certification Authority Details

This screen shows the details of CA's certificate and keypair. *Highest Serial Number* shows the highest of serial number of any certificate issued by this CA. *Certificate Text* shows the CA's certificate in base64 encoded form. This text can be used by the certificate requestors to designate this CA as a trusted CA in their software.

CA Setup is successful!

Certificate Details

Version: 3

Subject: C=LK, ST=State, L=City, O=Apache, OU=Geronimo, CN=Geronimo's CA

Issuer: C=LK, ST=State, L=City, O=Apache, OU=Geronimo, CN=Geronimo's CA

Serial Number: 1

Valid From: Thu May 03 00:00:00 LKT 2007

Valid To: Sat May 03 00:00:00 LKT 2008

Signature Alg: MD5withRSA

Public Key Alg: RSA

Key Size: 1024

Finger prints: SHA1 = 2F:AC:A3:10:5E:B2:A5:4E:B5:59:55:10:D3:37:AA:1B:1E:DB:57:E6
MD5 = 2D:C3:42:C0:60:59:8B:AD:A6:7E:BA:2D:64:AE:1C:F6

Highest Serial Number: 1

Base64 encoded Certificate Text

```
-----BEGIN CERTIFICATE-----
MIIECQDCCAaugAwIwBAGlTBATALBgkqhkiG9w0BAQQwaDENMBQGAlUEAxMNR2Vybz25pbW8ncyBDQTER
MA8GA1UECxMIR2Vyb25pbW8xZzAnBgNVBAoTBkFwYWN0ZTENMASGA1UEBXMtMEQwLmF0eTEOMAwGA1UE
CBMFU3RhdGUxZzA3BgNVBAYTAkxLMXB4XDTA3MDUwMjE4MDAwMFowDENMBQGAlUEAxMNR2Vybz25pbW8ncyBDQTERMA8GA1UECxMIR2Vyb25pbW8xZzAnBgNVBAoTBkFwYWN0ZTENMASGA1UEBXMtMEQwLmF0eTEOMAwGA1UECBMFU3RhdGUxZzA3BgNVBAYTAkxLMiGFMA0GC SqGSib3
dQEBAQUAA4GNADCBiQKBgQC8DFuP0D6bEdh/YsnfFTlUZRYH6mo07hoxrYlLpJtaFmpwoCldLTG
rgWk8GvOVnkpB+hzqbtKUAIK3uYiZYlCU4VRBr/Oaqxz0SzGwqXZJvw2Lx2giPY7nOGMamHvt7xn
Wo97gn97bSlod+fwdge8LTCT04uX0+L0xJ5b5kdnjQIDAQBAMASGC SqGSib3dQEBBAOBGBQAurZg/
7oFuX A7MYnPDS G8ER/yZcxQ/xDWfb JxXzyBzJhDh3JFZh9KQca/vK/Cs7puXSkPt5q6joTNoj2Y
Fh6Ryc94Gb yDVX7W6SNFd+5CX7YQ6k+Cv+m77MiEkZBMooUo8+jyxI4vl3U00Y2wCkkaAk2JCyli
UHBSls7GEcgklw==
-----END CERTIFICATE-----
```

[Back to CA home](#)

Next time you access the **Certification Authority** portlet you should see the the CA you just created. From this portlet now you can manage CSRs, review and issue certificates.

This portlet allows you to setup a Certification Authority (CA) and issue certificates in reply to Certificate Signing Requests (CSRs). *Setup Certification Authority* function allows to initialize the CA by providing CA identity details, algorithm parameters for CA's key pair and self-signed certificate and a password to protect the CA's private key. This password is to be used to unlock the CA to access CA functions. Once the CA is initialized, CSRs can be processed using *Issue New Certificate* function. Previously issued certificates can be viewed using *View Issued Certificate* function.

CA has been initialized. CA functions can be accessed using the links provided below.

Lock CA

[View CA Details](#)

Publish CA Certificate

Requests to be verified

Requests to be fulfilled

[Issue New Certificate](#)

[View Issued Certificate](#)

Signing certificate requests

The [Certificate Properties File Realm](#) section cover in great detail how to create a new keystore and certificate and how to create a CSR and then import the CA's reply. In this section we will focus on how the CA manages and signs the client CSR.

We will start from the point where you generate the CSR, here is the example we used for the [Certificate Properties File Realm](#) section.