User manual- Digital Signature

JnānaBhūmi

Enabling Excellence

1. Introduction:

A **digital signature** (not to be confused with a **digital** certificate) is a mathematical technique used to validate the authenticity and integrity of a message, software or **digital** document.

The digital signature of an electronic document aims to fulfill the following requirements:

- That the recipient can verify the identity of the sender (**authenticity**);
- That the sender cannot deny that he signed a document (**non-repudiation**);
- That the recipient is unable to invent or modify a document signed by someone else (**integrity**).

A typical digital signature scheme consists of three algorithms:

- An **algorithm** for generating the key that produces a key pair (PK, SK): PK (public key) is the public key signature verification while SK (Secret Key) is the private key held by the applicant, used to sign the document.
- ✓ A **signature algorithm** which, taken as input a message m and a private key SK produces a signature σ.
- ✓ A **verification algorithm** which, taken as input the message m, public key PK and a signature σ, accepts or rejects the signature.

To generate a digital signature it is necessary to use the digital asymmetric key pair:

- The **private key** is known only by the owner, it is used to generate the digital signature for a specific document;
- The **public key** is used to verify the authenticity of the signature.

Digital Signature Process:

A Digital signature is a one-way hash, of the original data, that has been encrypted with the signer's private key. A digital signature process is composed by the following steps:

- The signer calculates the hash for the data he needs to sign. The message digest is a file that contains some sort of control code that refers to the document.
- The signer, using his private key, encrypt the hash calculate.

Signer sends the original data and the digital signature to the receiver. The pair (document and signature) is a signed document or a document to which was attached a signature.

Verification:

- ✓ The receiver first uses the signer's public key to decrypt the hash, and then it uses the same hashing algorithm that generated the original hash to generate a new one-way hash of the same data.
- The receiver compares the new hash against the original hash. If the two hashes match, the data has not changed since it was signed.

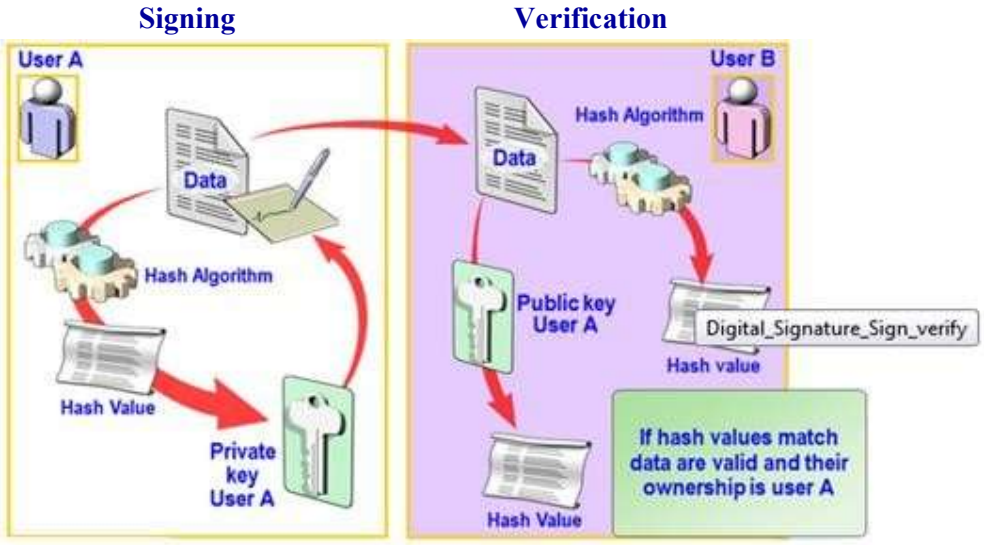


Fig: Digital Signature Process

What is Digital Certificate?

A Digital Signature Certificate (DSC) is a secure digital key that certifies the identity of the holder, issued by a Certifying Authority (CA). It typically contains your identity (name, email, country, APNIC account name and your public key etc.). Digital Certificates use Public Key Infrastructure meaning that data has been digitally signed or encrypted by a private key can only be decrypted by its corresponding public key.

What is an eToken?

USB e-Token can be password protected so that Digital Signature is never lost when computer is formatted or internet explorer changed. A virus cannot affect USB Token, and the digital certificate stored would always be secure

Difference between Encryption and Signing

Message encryption provides confidentiality. Allows users to encrypt document with the public key which can be decrypted only with the corresponding private key. To put it in simple terms when encrypting, you use their public key to write message and recipient uses their private key to read it. One of the most secure way protecting confidential documents.

Message signing binds the identity of the message source to this message. It ensures data integrity, message authentication, and non-repudiation altogether. When signing, you use your private key to write message's signature, and they use your public key to check if it's really yours.

Caution: Please completely read the User Manual before commencing the installation, it may well save you a lot of time.

2. Technical Requirements:

Requirements for Digital Signature:

S.No	Software	Recommended	More Information
1.	JRE	1.8	https://www.java.com/en/download/win10.jsp
2.	USB eToken Driver	-	https://jnanabhumi.ap.gov.in/downloads/ePass2003-Setup.zip

System Configuration Details:

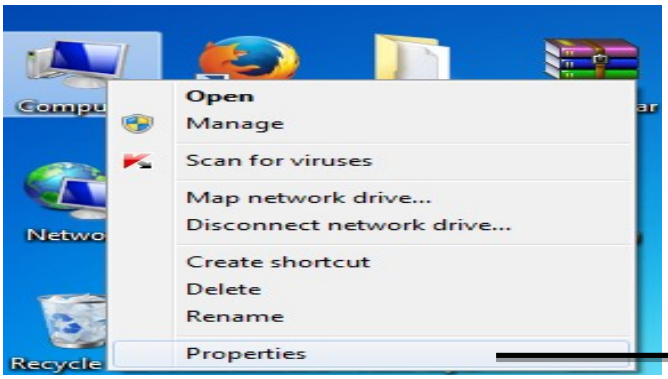
Follow the following steps to find out the System basic information:

Step 1:

Check the basic information of the computer like "**System type"** i.e. (Operating System, bit version) to know which version of JRE and eToken driver to install.

Step 2:

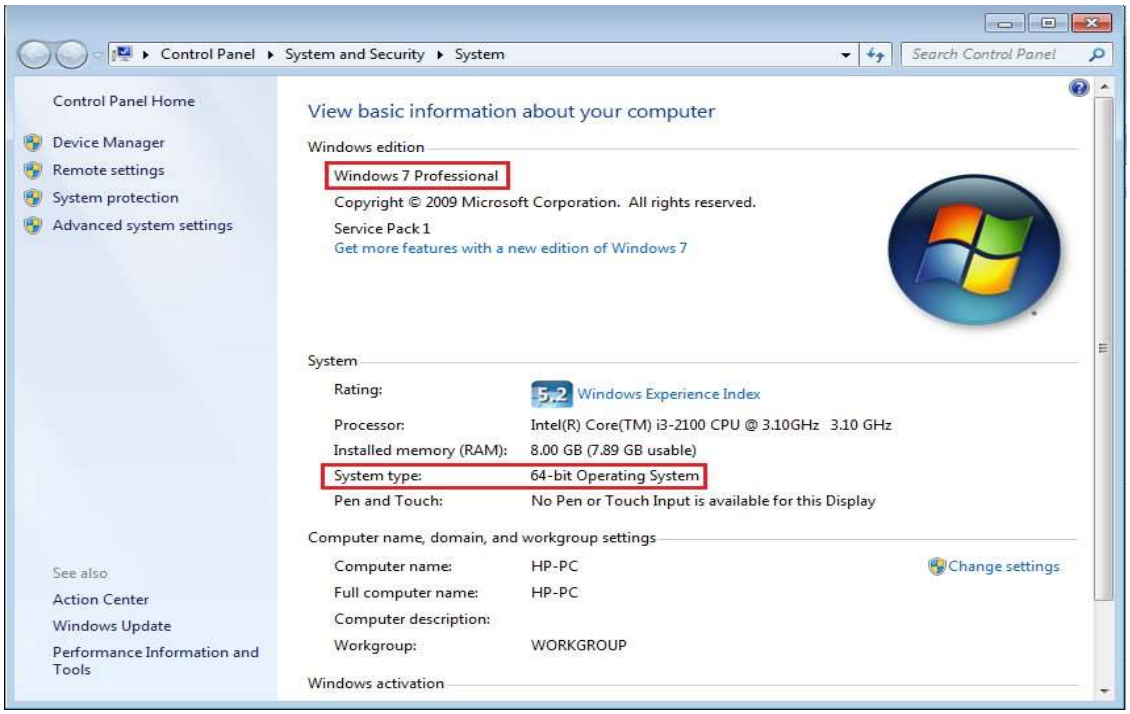
To know the system type, right clicks on the "Computer" desktop icon then Select "Properties" as shown below



Step 3:

After clicking on the 'Properties' below screen is displayed. From the screen as shown below - basic information of the computer is displayed.

Basic details such as - Windows edition/System Information 32 – bits operating system or 64 – bits operating system)



Step 4:

Check the version of operating System bit (E.g.: **SystemType32-Bit or 64 bit**).

Step 5:

After gathering the basic information like System type (32 or 64 bit) download and install the required software's.

Note: For any further information / support, call the help desk.

3. Installation:

JRE Installation:

1. Download JRE 1.8 (32 bit) from the following **url: https://www.java.com/en/download/win10.jsp**

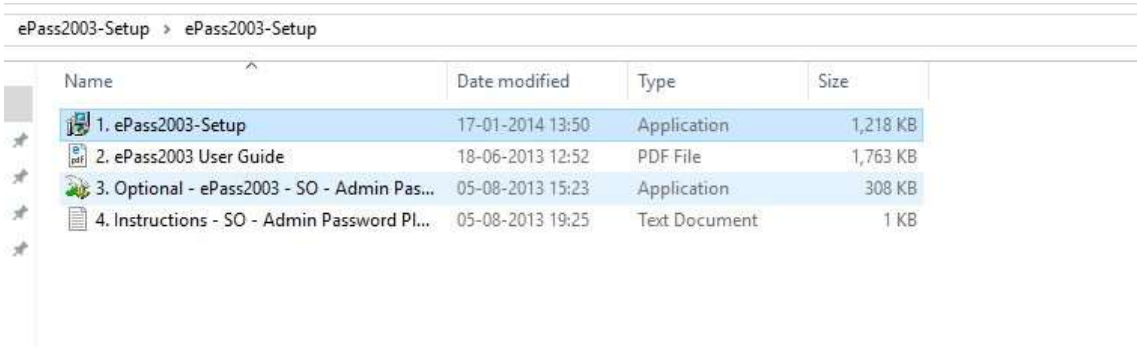
USB eToken Driver Installation:

1. Plugin the eToken (dongle) to the system.
2. If your eToken have the facility to install the driver automatically then let that install it.
3. If that does not install then go to the respective eToken website and install the driver in system

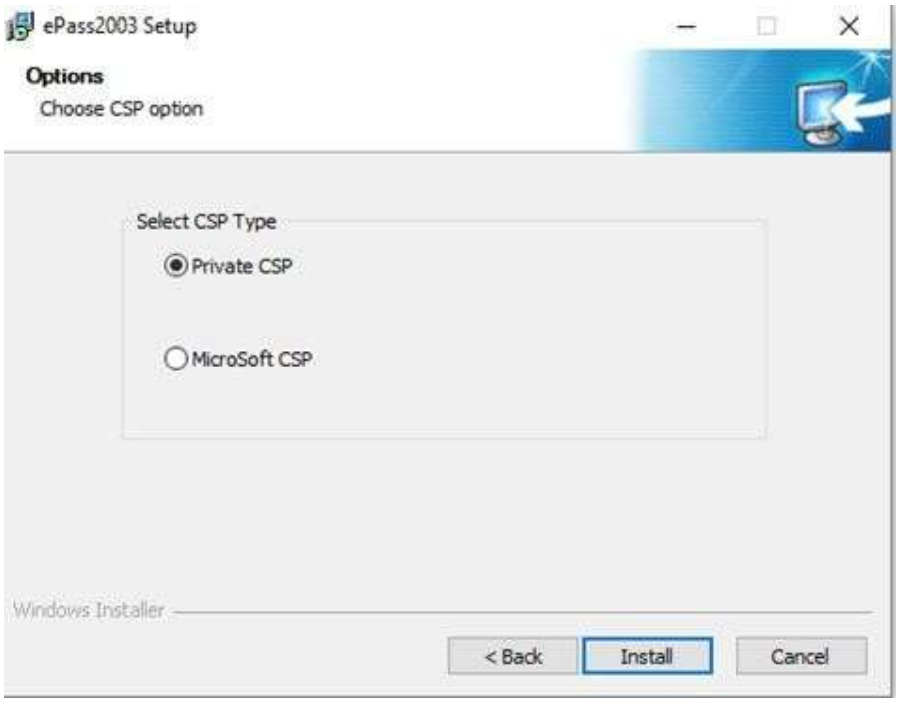
NOTE: Generally the eToken provider will provide USB eToken driver software along with USB eToken. If you have the driver software, you can install it or you can download it from the following URL.

http://jnanabhumi.ap.gov.in/downloads/ePass2003-Setup.zip

Caution: If JRE and USB eToken driver installed already then leave **Installation** step

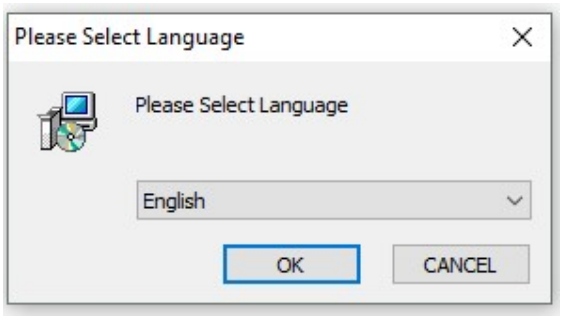


1) Download ePass2003-Setup.zip → unzip it and Open the folder double click on ePass-Setup as shown below.

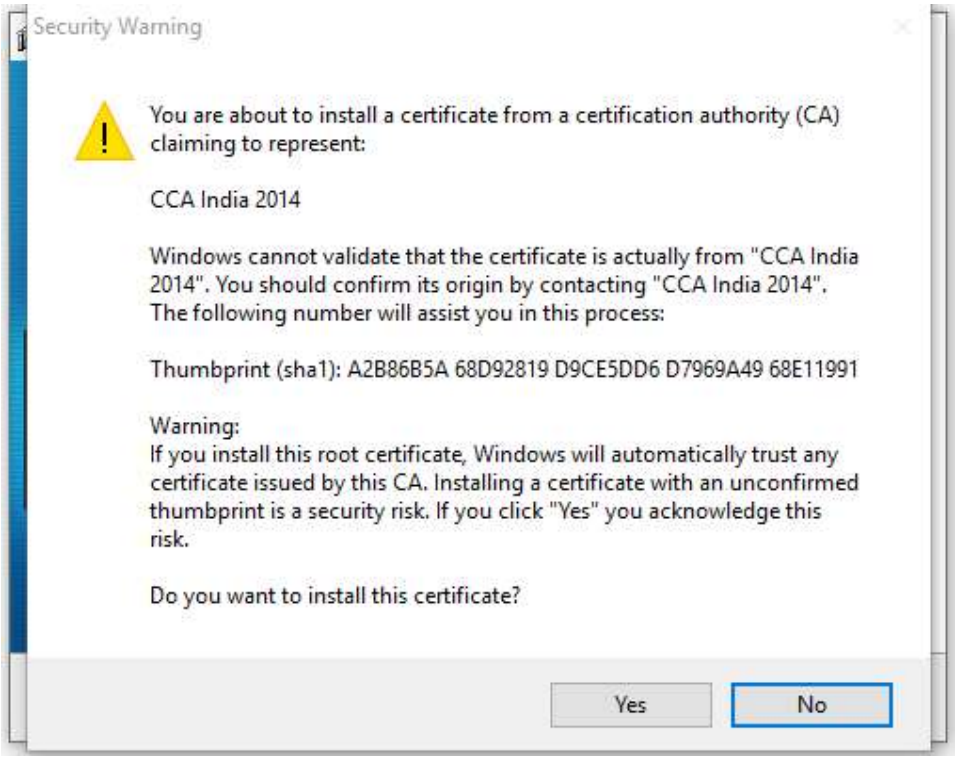


2) Select Privarte CSP and click on install.

3) Above screen appears select next .



4) Language dialogue box appears .Select English as preferred language as shown in above image.



5) Select Yes in the security warning dialog box.

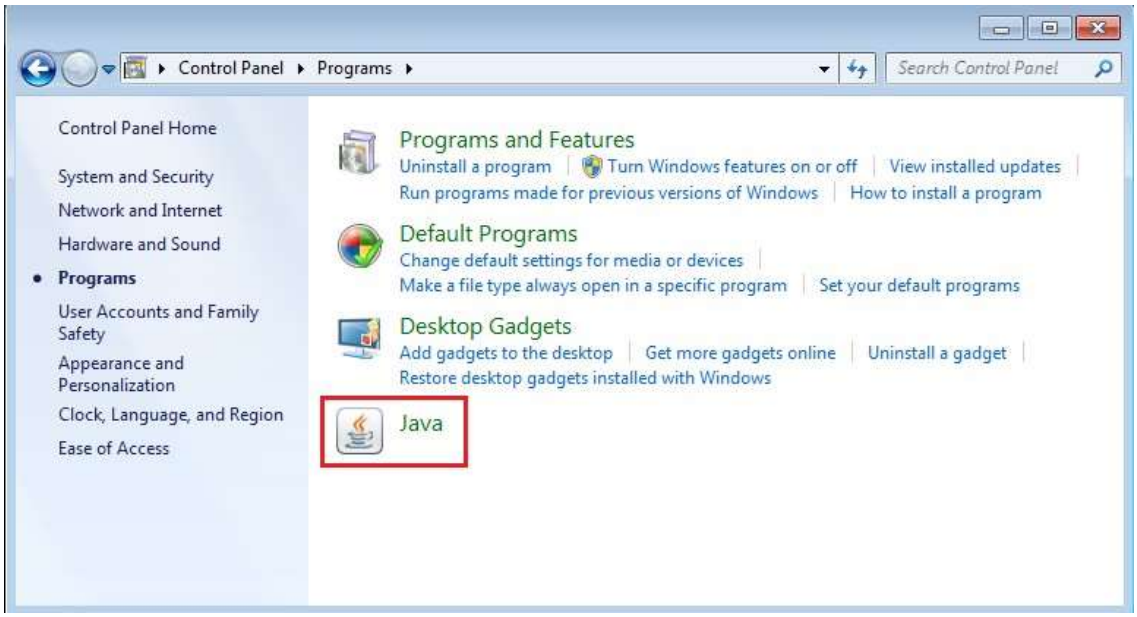
6) Your token driver installations are completed successfully.

4. Configure Java Security Settings:

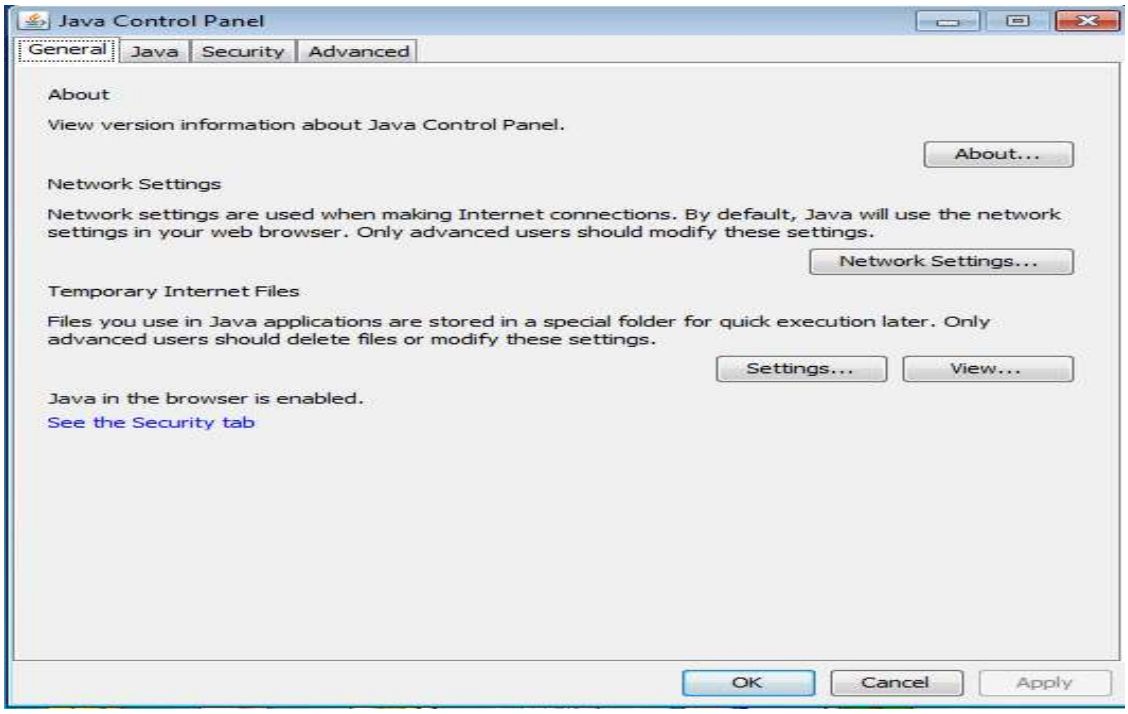
- To make the user system less vulnerable to external exploits, java does not allow users to run applications that are not signed (unsigned) and self-signed (not signed by trusted authority).
- By adding URL of the blocked application to the **Exception Site list** allows the blocked application to run with some warnings.
- To run the JNANABHUMI application without blocked by security checks add the https://jnanabhumi.ap.gov.in/ URL to the Exception site list as shown below

    - Go to the "Control Panel".
    - Go to "Programs" Folder.

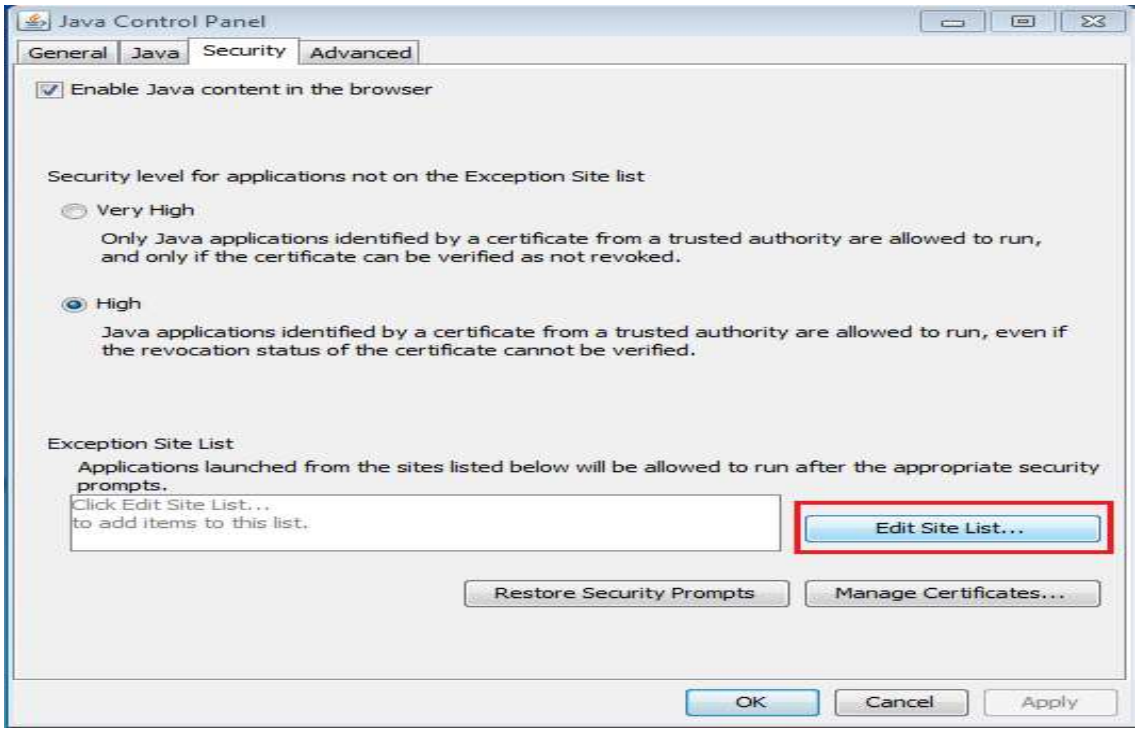
After clicking on the "Programs" below screen will be shown. Click on "JAVA"



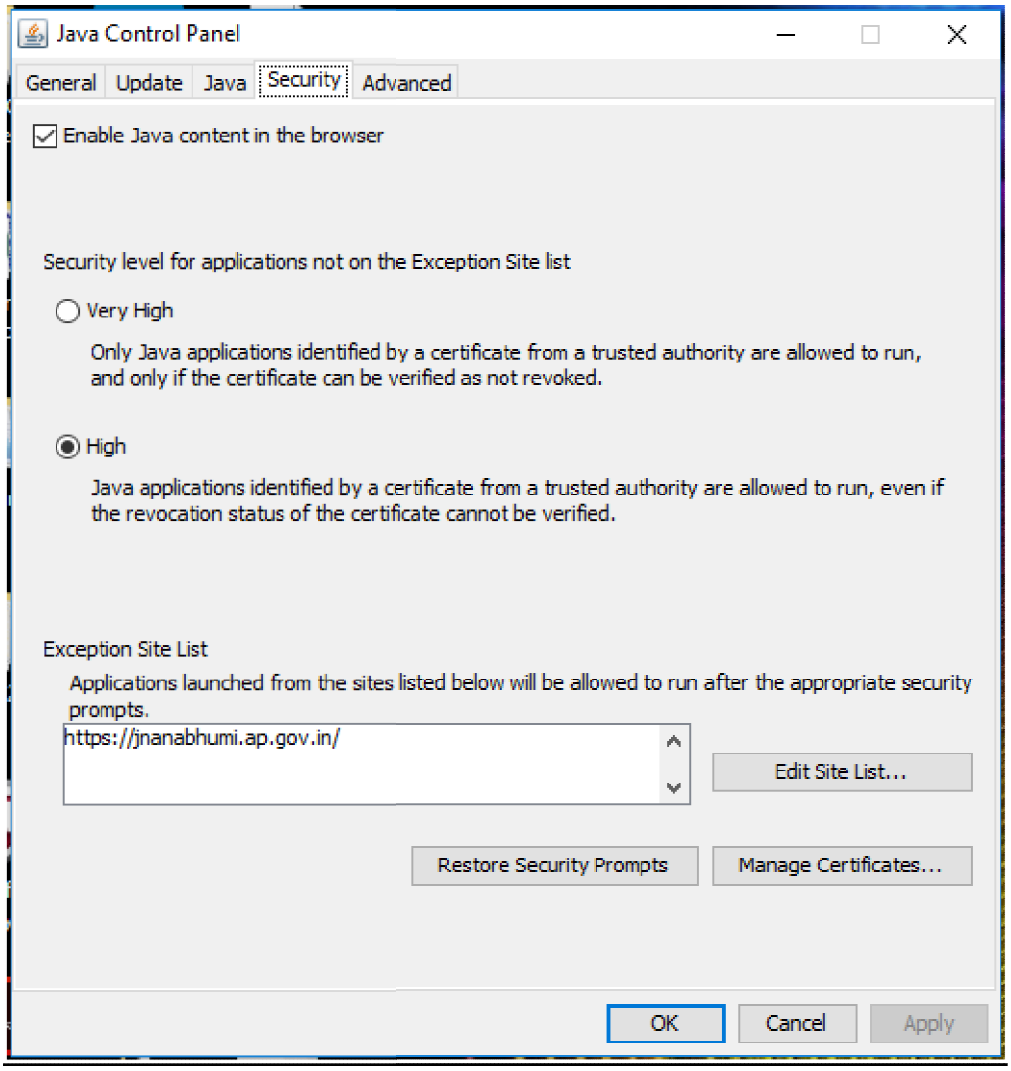
Java when opened below screen as shown below is displayed



Click on **"Security" tab.** Then click on **"Edit Site List"** button to make change in text box as shown below.



Write https://jnanabhumi.ap.gov.in/ URL in the text box and click on "add" Button and then "OK" button. As shown in the below screen



After making the changes Restart the browser and check whether the browsers java plugin updated to installed JRE version.