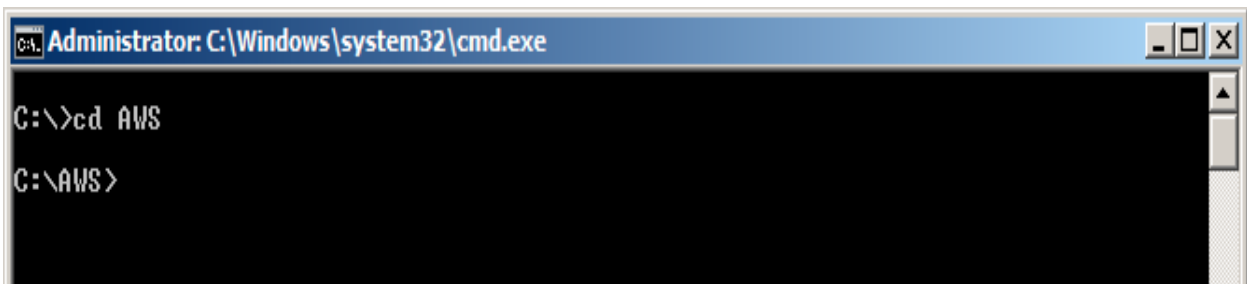


# How to Install AWS CLI to Windows

## A) Downloading SDK APIs

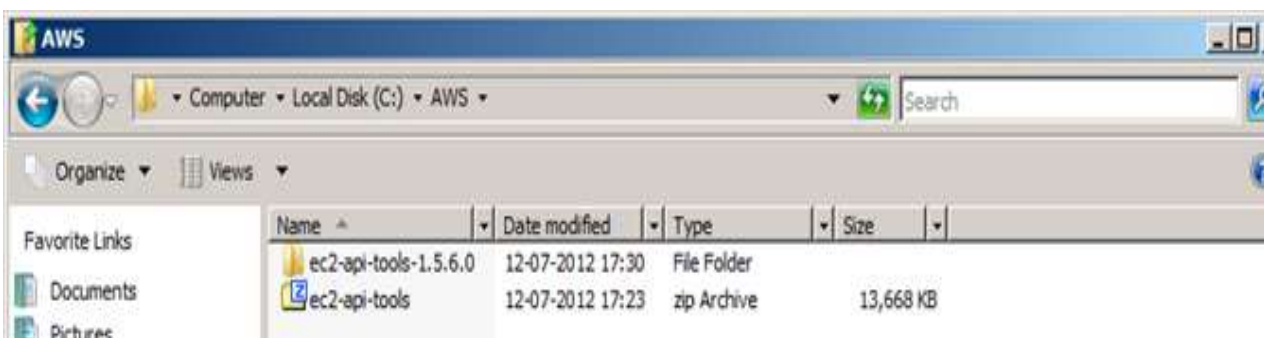
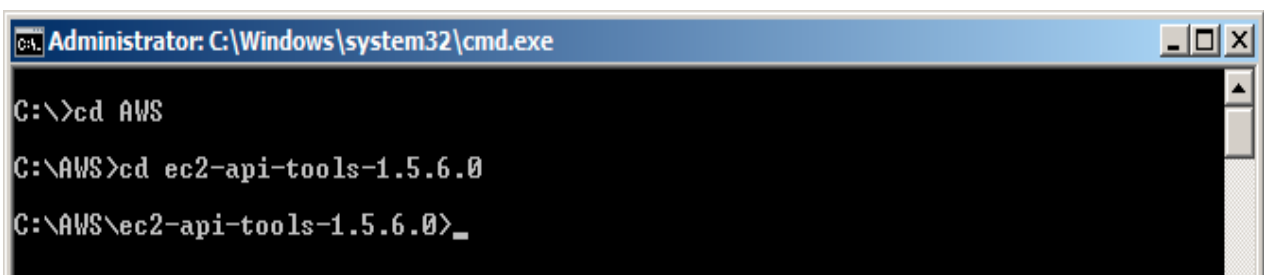
1. Create a folder to store your APIs in your local drive. E.g. C:\AWS



2. Download the Amazon AWS SDK API tools for Windows (.zip) file from the following link.

<http://s3.amazonaws.com/ec2-downloads/ec2-api-tools.zip> and save in the folder created in step#1.

3. Unzip the file and Extract it to local drive



## B) Install and setup Java

1. If JDK / JRE is not installed and environment variables are not set please follow below steps else jump to section 'C'

2. Install and download JDK 5 or above. The JDK download is free and JDK 7 is available for download

at <http://www.oracle.com/technetwork/java/javase/downloads/index.html>

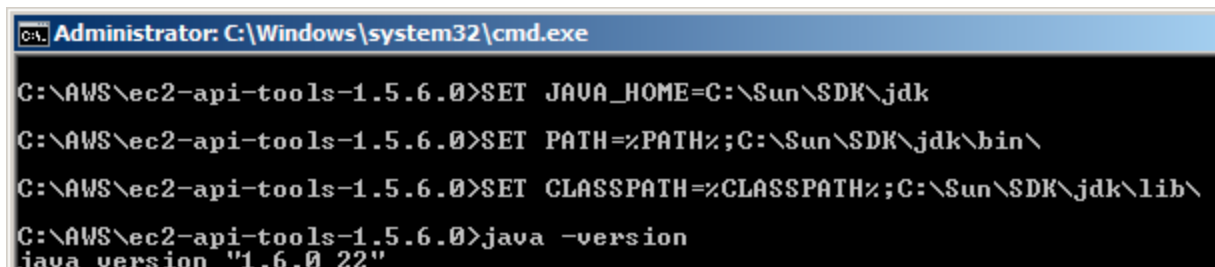
3. Set environment variable as following

i. JAVA\_HOME=<JRE / JDK PATH>

ii. PATH=%PATH%;<JAVA\_HOME>bin

iii. CLASSPATH=%PATH%;<JAVA\_HOME>lib

4. Run command `java -version` and check if it displays the correct version of your JDK / JRE.



```
C:\>Administrator: C:\Windows\system32\cmd.exe

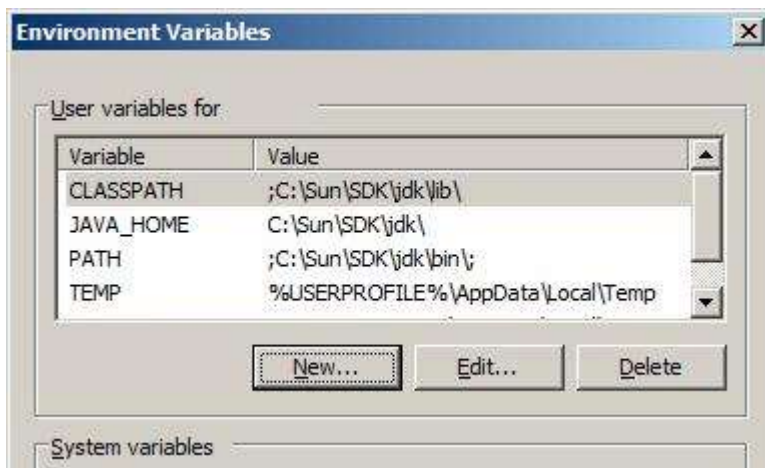
C:\AWS\ec2-api-tools-1.5.6.0>SET JAVA_HOME=C:\Sun\SDK\jdk
C:\AWS\ec2-api-tools-1.5.6.0>SET PATH=%PATH%;C:\Sun\SDK\jdk\bin\
C:\AWS\ec2-api-tools-1.5.6.0>SET CLASSPATH=%CLASSPATH%;C:\Sun\SDK\jdk\lib\
C:\AWS\ec2-api-tools-1.5.6.0>java -version
java version "1.6.0_22"
```

5. If you setup above commands through command window it will be valid for the session of this command window only.

6. Please set all above parameters through Environment Variables. You can access Environment variables through for windows 7 / Vista: MyComputer -> Right Click and Select Properties. -> select "Advanced System Settings" from left menu and go to Environment variables.

For Windows XP Right Click on Computer -> Select Properties -> Select Advanced Tab and click -> Environment variables.

7. Set the variables as shown below



C) Download and set AWS Certificate File and Private Keys. (Some of the data is masked or removed in the screen for confidentiality purpose).

1. Go to AWS Account section. <http://aws.amazon.com/account>

2. in the left menu click on “Security Credentials” as selected below:

The screenshot shows the AWS Management Console interface. At the top, there is a navigation bar with the Amazon Web Services logo, a 'Sign Up' button, and links for 'My Account / Console' and 'English'. Below this is a secondary navigation bar with 'AWS Products & Solutions', a search bar, 'AWS Product Information', and links for 'Developers' and 'Support'. On the left side, there is a sidebar menu under the 'Account' heading, listing various account management options. The 'Security Credentials' option is highlighted with a red box. The main content area features a yellow informational box stating that the page is for managing root account credentials and that IAM users should be managed via the AWS Management Console. Below this, a paragraph explains that access to AWS services is secure and requires special credentials. It then lists three types of credentials: Access Credentials (Access Keys, X.509 Certificates, and Key Pairs), Sign-In Credentials (E-mail Address, Password, and AWS Multi-Factor Authentication Device), and Account Identifiers (AWS Account ID and Canonical User ID). At the bottom, there is a link to 'Find out which AWS Security Credentials you need'.

**Account**

- Account Activity
- AWS Identity and Access Management
- AWS Management Console
- Consolidated Billing
- DevPay
- Manage Your Account
- Payment Method
- Personal Information
- Security Credentials**
- Usage Reports
- Billing Alerts
- Billing Preferences

**Welcome D1 | Sign Out**  
Account Number 0 | 182 18:0 6

This page allows you to manage the root account credentials for your AWS Account. To manage IAM Users, their permissions, and security credentials, use the AWS Management Console.

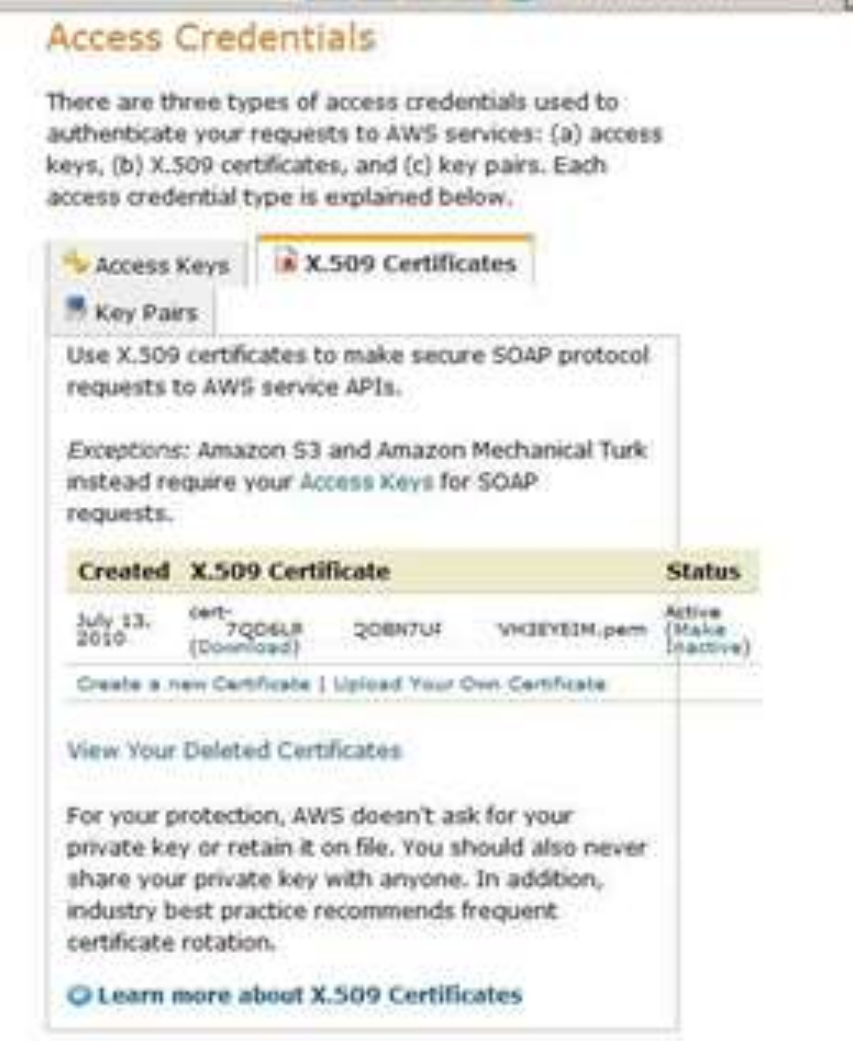
Access to applications and services within AWS cloud is secure and protected in multiple ways. Accessing those applications and services requires the use of special credentials that are associated with your account. There are three types of credentials currently offered by AWS. If you know which security credentials you need, simply select one of the links below:

- Access Credentials:** Your Access Keys, X.509 Certificates, and Key Pairs
- Sign-In Credentials:** Your E-mail Address, Password, and AWS Multi-Factor Authentication Device
- Account Identifiers:** Your AWS Account ID and Canonical User ID

If you are not sure which security credentials you should use, the link below will help you identify the credentials you need for the task you want to accomplish:

[Find out which AWS Security Credentials you need](#)

3. Go to Access Credentials – > X.509



**Access Credentials**

There are three types of access credentials used to authenticate your requests to AWS services: (a) access keys, (b) X.509 certificates, and (c) key pairs. Each access credential type is explained below.

**Access Keys** | **X.509 Certificates** | **Key Pairs**

Use X.509 certificates to make secure SOAP protocol requests to AWS service APIs.

Exceptions: Amazon S3 and Amazon Mechanical Turk instead require your [Access Keys](#) for SOAP requests.

Created	X.509 Certificate	Status
July 13, 2010	cert-7QD6L8 (Download)    Q0BN7U4    VHC3EYB1M.pem	Active (Make Inactive)

[Create a new Certificate](#) | [Upload Your Own Certificate](#)

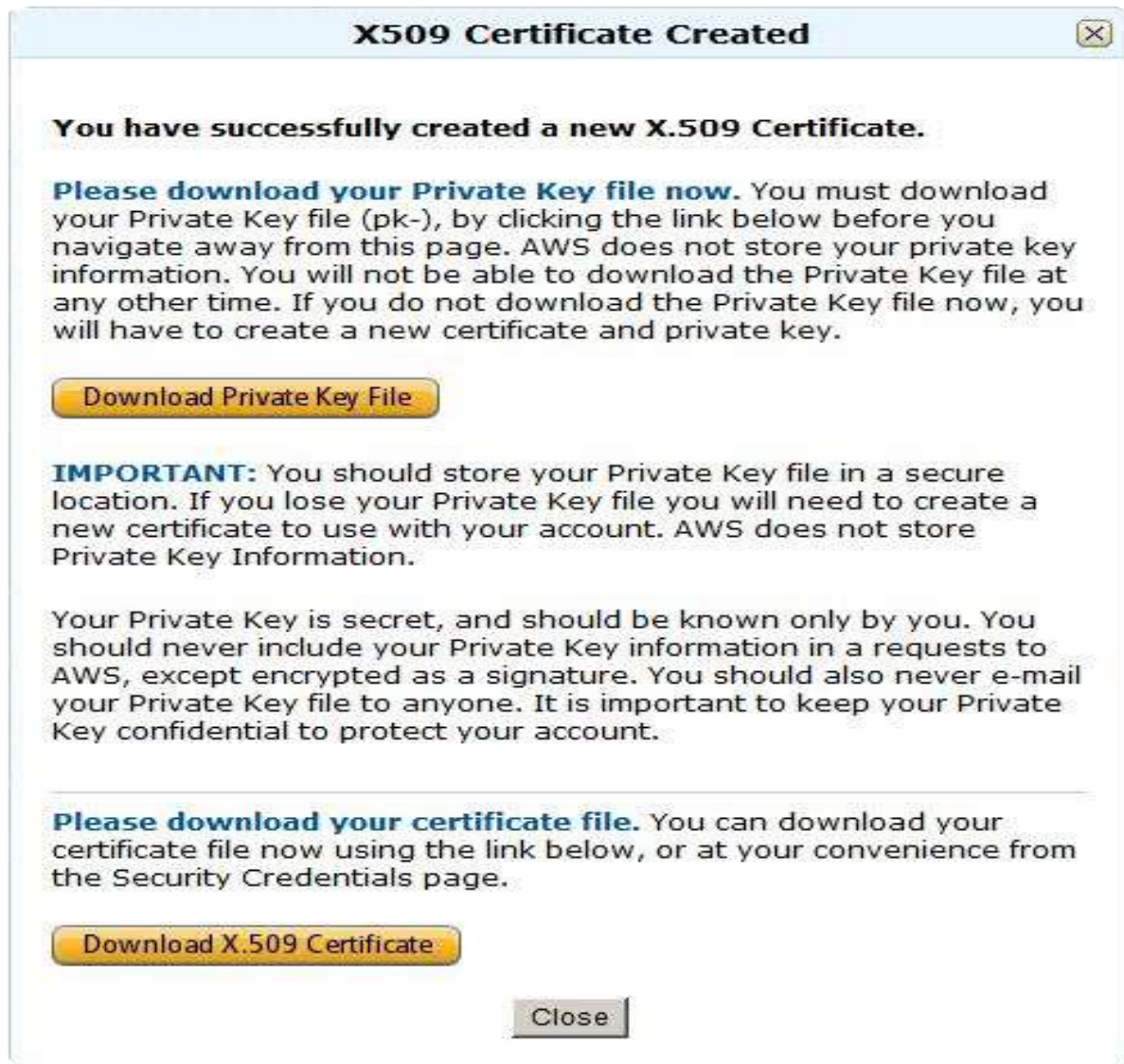
[View Your Deleted Certificates](#)

For your protection, AWS doesn't ask for your private key or retain it on file. You should also never share your private key with anyone. In addition, industry best practice recommends frequent certificate rotation.

[Learn more about X.509 Certificates](#)

4. It will show all existing active / Inactive certificates.

5. Create a new Certificate by clicking “Create a new Certificate”. It will show screen as below:



6. Download your private key file and X.509 to local folder. (E.g. C: AWSkeys).

7. If you fail to save Private Key file, AWS does not store it for you and you will lose it permanently.

8. If the case #7 happens, delete the new created certificate and follow steps #1 – #6 to save the file again.

9. Store the downloaded pk & cert file into local directory (e.g c: AWSkeys)

10. Set the AWS Keys in environment as below: (For going to Environment variable follow step#6 of section 'B')

i. EC2\_HOME= < <path where you have downloaded ec2 tools extracted as section 'A'>, e.g. C:AWSec2-api-tools-1.5.6.0

ii. EC2\_CERT=<fully qualified path where cert-xxxxx.pem file placed>

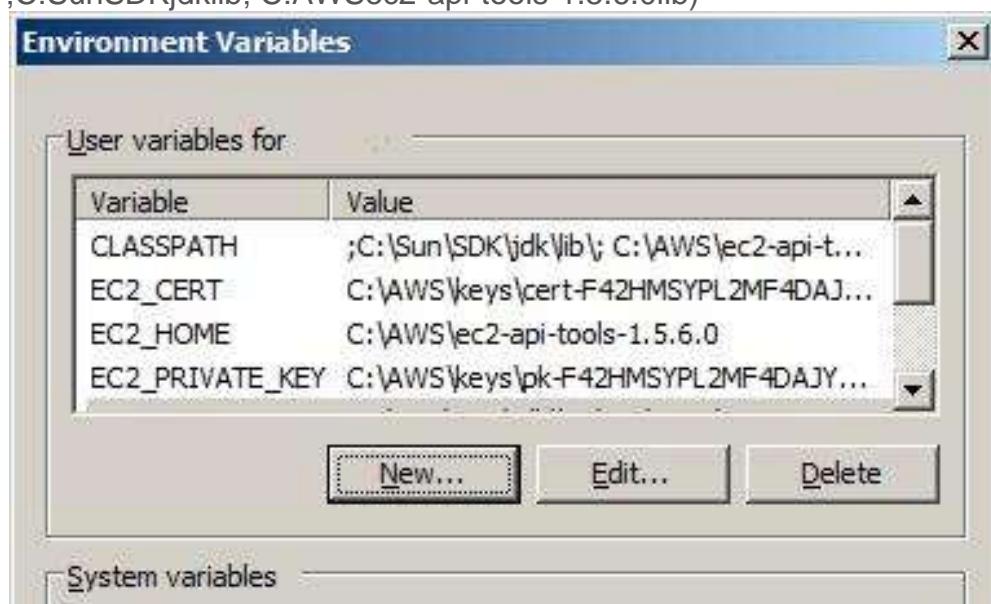
e.g. EC2\_CERT= c:AWSkeys cert-F42xxxxxxxxxAR2xxxxxxxxUBA438xxxxD.pem

iii. EC2\_PRIVATE\_KEY=<fully qualified path where pk-xxxx.pem file placed>

e.g. EC2\_PRIVATE\_KEY= c:Cloudkeys pk-F42xxxxxxxxxAR2xxxxxxxxUBA438xxxxD.pem

iv. PATH=; <JAVA\_HOME>bin;< EC2\_HOME >bin (e.g. PATH=;C:\SunSDK\jdk\bin;C:\AWSec2-api-tools-1.5.6.0\bin)

v. CLASSPATH= ; <JAVA\_HOME>lib; < EC2\_HOME >lib (e.g. CLASSPATH= ;C:\SunSDK\jdk\lib; C:\AWSec2-api-tools-1.5.6.0\lib)





11. Test your setup by executing following command in command line. ec2-run-instances or

ec2-describe-images -o amazon (Lists all public AMIs of Amazon)

```
C:\AWS>ec2-run-instances
Required parameter 'AMI' missing (-h for usage)

C:\AWS>ec2-describe-images -o amazon ! more
IMAGE   aki-d4ca2dbd    aki-linux-2.6.18.92-92.el5xen-xfs/vmlinuz-2.6.18.92-92.e
l5xen.i386.aki.manifest.xml    amazon    available    public    i386
kernel
instance-store    paravirtual    xen
IMAGE   aki-46e7002f    aki-linux.2.6.21.7-2.fc8xen-xfs/vmlinuz.manifest.xml
amazon    available    public    i386    kernel
instance-store    paravirtual    xen
IMAGE   ami-32dc075b    amazon/.NET Beanstalk HostManager v1.0.0.3    amazon
available    public    x86_64    machine    windows ebs
hvm    xen
BLOCKDEVICEMAPPING    /dev/sda1    snap-96ebaaeb    35
IMAGE   ami-6c9c3105    amazon/.NET Beanstalk HostManager v1.0.0.4    amazon
available    public    x86_64    machine    windows ebs
hvm    xen
BLOCKDEVICEMAPPING    /dev/sda1    snap-608be71e    30
IMAGE   ami-cbd47ba2    amazon/Amazon Elastic MapReduce 2012-07-09-23-50-37 pvm/
ebs    amazon    available    public    x86_64    machine    aki-4e7d9527
ebs    paravirtual    xen
BLOCKDEVICEMAPPING    /dev/sda    snap-9c817ded    10
IMAGE   ami-e9d27d80    amazon/Amazon Elastic MapReduce 2012-07-10-00-42-47 pvm/
ebs    amazon    available    public    x86_64    machine    aki-4e7d9527
ebs    paravirtual    xen
BLOCKDEVICEMAPPING    /dev/sda    snap-180bf769    10
IMAGE   ami-8bb21de2    amazon/Amazon Elastic MapReduce 2012-07-10-16-56-36 pvm/
ebs    amazon    available    public    x86_64    machine    aki-4e7d9527
ebs    paravirtual    xen
BLOCKDEVICEMAPPING    /dev/sda    snap-ce28eebf    10
```

If it shows above output your setup of AWS API is complete.