Table of Contents

[Installation of Python 2](#_Toc533609961)

[RHEL 2](#_Toc533609962)

[Windows 2](#_Toc533609963)

[Installation of the Libraries 2](#_Toc533609964)

[Installation of AWS CLI 2](#_Toc533609965)

[Editing configuration files 3](#_Toc533609966)

[Proxy.properties 3](#_Toc533609967)

[Making cacerts.pem 3](#_Toc533609968)

[Option 1 3](#_Toc533609969)

[Option 2 3](#_Toc533609970)

[Runing the program 4](#_Toc533609971)

# Installation of Python

**Note:** Linux might already have older python version. Then python3.5.2 will be installed as alternative. Make sure that you are invoking python3.5.2. In the following instructions for linux, this python is invoked as python3.5/pip3.5 command

## RHEL

Run the following commands to install Python 3.5.2 on linux

1. Yum install gcc
2. Cd /opt
3. wget <https://www.python.org/ftp/python/3.5.2/Python-3.5.2.tgz>
4. tar xzf Python-3.5.2.tgz
5. cd Python-3.5.2
6. ./configure
7. make altinstall
8. python3.5 --version
9. pip3.5 –version (version shows 8.1.1)
10. pip3.5 install --upgrade pip (version shows 8.1.2)

## Windows

1. Download appropriate windows installer from <https://www.python.org/downloads/release/python-352/> and run the installer.
2. Run the command pip install --upgrade pip

# Installation of the Libraries

Run the following commands on command prompt.

1. pip3.5 install requests
2. pip3.5 install boto3

# Installation of AWS CLI

Run the following commands on the command prompt.

1. pip3.5 install awscli --ignore-installed six
2. aws help //to make sure aws is installed
3. aws configure //to create credentials file. We are not going to use these credentials.
   1. AWS Access Key ID [None]: *AKIAJIGAW5UADUMZP37Q*
   2. AWS Secret Access Key [None]: *wIw3vAJaIBrfaXQC6JWZFyuBJDTwRfuPg7i+Fh6f*
   3. Default region name [None]: *us-west-2*
   4. Default output format [None]: *text*
4. cat ~/.aws/credentials //for linux
5. Check c:/User/<username>/.aws/credentials file //for windows.

# Editing configuration files

1. Unzip aws-idaptive-v1.zip
2. Change directory to aws-idaptive-v1 - cd python-aws-v7

## Proxy.properties

In this properties file, if your organization uses a proxy server then mention the proxy server details. Else, mention proxy=no.

* 1. [Properties] //Don’t modify
  2. proxy=no //If you are using proxy, modify it as yes
  3. http\_proxy= //If you are using proxy, modify with the proxy value
  4. https\_proxy= //If you are using proxy, modify with the proxy value
  5. proxy\_user= //User for proxy server authentication (and not the Idaptive instance authentication)
  6. proxy\_password= //If above value is no, and if you are using proxy server then update this value with base64 encoded value of proxy server password

# Making cacerts.pem

## Option 1

1. Open your Idaptive instance in the browser (firefox) e.g. instance.idaptive.com
2. Click on the lock icon on the left hand side of the URL
3. Click on the arrow ">" and "More Information" to view Security information
4. Click on View Certificate button.
5. In the Certificate Viewer, there are two buttons "General" and "Detail"
6. Click on the details and the certificate chain can be seen. (a)leaf certificate - \*.instance.idaptive.com (b) intermediate certificate - Go Dadday Secure certificate authority - G2 (c) root certificate - Go Daddy Root Certificate authority - G2
7. Click on each certificate and save the files in .crt format.
8. Open the leaf certificate i.e. \*.instance.idaptive.com
9. Open the intermediate certificate and copy the contents, paste it after the leaf certificate
10. Open the root certificate and copy the contents, paste it after the contents.
11. Save the file as cacerts\_<tenant\_name>.pem in the root directory of the script i.e. where AWSCLI.py file exists. [Refer point no 2 in below section]

## Option 2

* 1. Run below command on unix
  2. openssl s\_client -connect *your\_tenant*.idaptive.com:443 -showcerts 2>&1 | sed -ne '/-BEGIN CERTIFICATE-/,/-END CERTIFICATE-/p' > cacerts\_ *your\_tenant*.pem
  3. Open cacerts\_ your\_tenant.pem. Copy the following text from ---BEGIN CERTIFICATE---- to ----END CERTIFICATE----- and append it in the file.

-----BEGIN CERTIFICATE-----

MIIDxTCCAq2gAwIBAgIBADANBgkqhkiG9w0BAQsFADCBgzELMAkGA1UEBhMCVVMx

EDAOBgNVBAgTB0FyaXpvbmExEzARBgNVBAcTClNjb3R0c2RhbGUxGjAYBgNVBAoT

EUdvRGFkZHkuY29tLCBJbmMuMTEwLwYDVQQDEyhHbyBEYWRkeSBSb290IENlcnRp

ZmljYXRlIEF1dGhvcml0eSAtIEcyMB4XDTA5MDkwMTAwMDAwMFoXDTM3MTIzMTIz

NTk1OVowgYMxCzAJBgNVBAYTAlVTMRAwDgYDVQQIEwdBcml6b25hMRMwEQYDVQQH

EwpTY290dHNkYWxlMRowGAYDVQQKExFHb0RhZGR5LmNvbSwgSW5jLjExMC8GA1UE

AxMoR28gRGFkZHkgUm9vdCBDZXJ0aWZpY2F0ZSBBdXRob3JpdHkgLSBHMjCCASIw

DQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBAL9xYgjx+lk09xvJGKP3gElY6SKD

E6bFIEMBO4Tx5oVJnyfq9oQbTqC023CYxzIBsQU+B07u9PpPL1kwIuerGVZr4oAH

/PMWdYA5UXvl+TW2dE6pjYIT5LY/qQOD+qK+ihVqf94Lw7YZFAXK6sOoBJQ7Rnwy

DfMAZiLIjWltNowRGLfTshxgtDj6AozO091GB94KPutdfMh8+7ArU6SSYmlRJQVh

GkSBjCypQ5Yj36w6gZoOKcUcqeldHraenjAKOc7xiID7S13MMuyFYkMlNAJWJwGR

tDtwKj9useiciAF9n9T521NtYJ2/LOdYq7hfRvzOxBsDPAnrSTFcaUaz4EcCAwEA

AaNCMEAwDwYDVR0TAQH/BAUwAwEB/zAOBgNVHQ8BAf8EBAMCAQYwHQYDVR0OBBYE

FDqahQcQZyi27/a9BUFuIMGU2g/eMA0GCSqGSIb3DQEBCwUAA4IBAQCZ21151fmX

WWcDYfF+OwYxdS2hII5PZYe096acvNjpL9DbWu7PdIxztDhC2gV7+AJ1uP2lsdeu

9tfeE8tTEH6KRtGX+rcuKxGrkLAngPnon1rpN5+r5N9ss4UXnT3ZJE95kTXWXwTr

gIOrmgIttRD02JDHBHNA7XIloKmf7J6raBKZV8aPEjoJpL1E/QYVN8Gb5DKj7Tjo

2GTzLH4U/ALqn83/B2gX2yKQOC16jdFU8WnjXzPKej17CuPKf1855eJ1usV2GDPO

LPAvTK33sefOT6jEm0pUBsV/fdUID+Ic/n4XuKxe9tQWskMJDE32p2u0mYRlynqI

4uJEvlz36hz1

-----END CERTIFICATE-----

* 1. Save the cacerts\_ *your\_tenant*.pem in the root directory of the script i.e. where AWSCLI.py file exists. [Refer point no 2 in below section]

# Runing the program

1. Run the command - Python3.5 AWSCLI.py -h [-t|-tenant] <tenant> [-r|-region] <region> [-v|-version]

Command line arguments

* 1. -h / -help

Provides help for the program.

|  |
| --- |
| D:\python-aws-v7>**python AWSCLI.py -h**  usage: AWSCLI.py [-h] [-tenant TENANT] [-region REGION] [-debug]  Enter Idaptive Credentials and choose AWS Role to create AWS Profile. Use this  AWS Profile to run AWS commands.  optional arguments:  -h, --help show this help message and exit  -tenant TENANT, -t TENANT  Enter tenant url or name e.g. cloud.idaptive.com or  cloud  -region REGION, -r REGION  Enter AWS region. Default is us-west-2  -debug, -d This will make debug on |

* 1. -t / -tenant

The tenant parameter can be used in the following way. If the value is not provided, then the default value is cloud.idaptive.com

|  |
| --- |
| D:\python-aws-v7>**python AWSCLI.py -t cloud**  **OR**  D:\python-aws-v7>**python AWSCLI.py -t cloud.idaptive.com**  **OR**  D:\python-aws-v7>**python AWSCLI.py -tenant station** |

* 1. -r / -region

You can specify AWS region in the following way. If the value is not provided, then the default value is us-west-2

|  |
| --- |
| D:\python-aws-v7>**python AWSCLI.py -t cloud -r ap-southeast-1**  **OR**  D:\python-aws-v7>**python AWSCLI.py -t cloud -region ap-southeast-1** |

* 1. -d / -debug

Makes debug ON. This makes program logs input request with the values. It may log sensitive data.

|  |
| --- |
| D:\python-aws-v7>**python AWSCLI.py -t cloud -d**  **OR**  D:\python-aws-v7>**python AWSCLI.py -t cloud -r ap-southeast-1 -d** |

* 1. -v / -version

It prints the version and exits.

|  |
| --- |
| D:\python-aws-v7>**python AWSCLI.py -v**  **OR**  D:\python-aws-v7>**python AWSCLI.py -version** |

1. The tenant name can be either only “cloud” or “cloud.idaptive.com”. In these cases, the cacerts file should be named as “cacerts\_cloud.pem”. If the tenant name is “**mytenant**.somedomain.idaptive.com” then the cacerts filename should be cacerts\_**mytenant**.pem
2. Give username for Idaptive instance.
3. If proxy server is used and proxy\_ask\_password option is ‘yes’, then it will be asked. Enter that value.
4. Enter Idaptive instance password.
5. Choose role by entering the number from the options e.g. 1
6. The credential file will be updated with saml profile
7. You can use the profile name to run the AWS command e.g. - aws s3 ls --profile <profile\_name>