Here are the instructions for leveraging the AWS CLI via SSO for either general usage or just to generate temp keys for other tools (eg S3 Browser).

1.       Install Python 2.7 (Not 3.x, must be 2.x)

a.       Navigate to <https://www.python.org/downloads/>

b.       Click "Download Python 2.7.XX"

c.       Run the Installation Package and accept default options

2.       Install AWS CLI

a.       Navigate to <https://aws.amazon.com/cli/>

b.       Click "64-bit" in upper right

c.       Run the installation package and accept default options

3.       Run aws configure:

a.       Open Command Prompt

b.       Type aws configure (if this isn’t done, the .aws directory will not exist)

c.       Hit enter through the fist two prompts (AWS Access, AWS Secret)

d.       Enter a default region (eg us-east-1) on the third prompt and hit enter on the 4th

e.       Command Prompt 
licrosoft Windows [Version 18.8.18586] 
(c) 2815 microsoft Corporation. All rights reserved . 
: \Users\jburch>aws configure 
WS Access Key ID [None]: 
WS Secret Access Key [None] : 
Default region name [None]: us-east-I 
Default output format [None] : 
: burch> 

4.       Deploy Necessary files

a.       Copy file credentials to C:\Users\%username%\.aws

b.       Copy file pythonAwsSSO2.py to c:\Python27

5.       Use pip to install requisite packages (requests.ntlm, boto and bs4)

a.       Open command prompt

b.       Navigate to c:\Python27\scripts

c.       Run pip install requests.ntlm

d.       Run pip install boto

e.       Run pip install bs4

f.        Packages should download and install, output similar to the following:

g.       Command Prompt 
Collecting ipaddress; python_version < "3" (from cryptography>=1.3->requests.nt1m) 
Downloading ipaddress-l .8.19 . tar .gz 
ollecting pycparser (from cffi>=1.7; platform_python_implementation .3->requests . ntlm) 
Downloading pycparser-2.18.tar.gz (245kB) 
188% | 
1 256kB 1.Sb1B/s 
Installing collected packages: chardet, certifi, ur11ib3, idna, requests, six, 
ntlm- auth, pycparser, cffi, enum34, 
rypto, ipaddress, cryptography, requests . ntlm 
Running setup. py install for pycparser 
done 
Running setup. py install for ipaddress 
done 
asnlc 
Successfully installed asn1crypto-ø.24.ø certifi-2818.1.18 cffi-1.11.4 chardet-3.ø.4 cryptography-2.1.4 enum34-1.1.6 idn 
a-2.6 ipaddress-1.ø.19 nt1m-auth-1.ø.6 pycparser-2.18 requests-2.18.4 requests.ntlm six-I. 11.8 ur11ib3-1.22 
: install boto 
ollecting boto 
Downloading boto-2.48.ø-py2.py3-none-any. whl (1.414B) 
188% | 
1 1.41B 591kB/s 
Installing collected packages: boto 
Successfully installed boto-2.48.ø 
: install bs4 
ollecting bs4 
Downloading bs4-ø.ø.1.tar.gz 
ollecting beautifulsoup4 (from bs4) 
Downloading beautifulsoup4-4.6.ø-py2-none-any. whl (86kB) 
188% | 
1 92kB 1.Sb1B/s 
Installing collected packages: beautifulsoup4, bs4 
Running setup. py install for bs4 
done 
Successfully installed beautifulsoup4-4.6.ø bs4-ø.ø.1 
: \ p hon27\Scri 

6.       Connect to AWS via SAML

a.       Open command prompt

b.       Navigate to c:\Python27\

c.       Run 'python pythonAwsSSO2.py'

d.       Enter your PCLC0 usermame and password

e.       Roles you are allowed to assume are enumerated, select the one you would like to leverage

f.        You should receive a confirmation statement indicating that your access key pair was stored and an expiration time for the pair

1.       Test command aws cli commands including '--profile saml'

a.       Execute  'aws --profile saml ec2 describe-instances'

b.       Machine generated alternative text:
Command Prompt 
: - -profile saml ec2 describe-instances 
"Reservations" 
"Instances" 
"monitoring" 
"State" : 
"enabled" 
"PublicDnsName" : 
"State": 
"Code": 1 
"Name" 
"running 