Step 1: Form the main account create the IAM user that will allow you use the user created, do not use root user for any purpose,

Download and save csv file of user which container username secret key access key etc.

Step 2 : How to Install AWS CLI on windows.

* Use **ubuntu bash** extension for latest Windows10 for older one use **CYGWin**
* Enable developer mode in Settings it will allow you to download ubuntu bash(restart)
* Programs and features 🡪 Turn Windows features on and off 🡪 Check Windows Subsystem for Lunix
* Open Bash CLI in windows
* Check sudo apt-get install awscli
* aws
* aws configure

Installed AWS CLI MSI instead of Ubuntu bash

Step 3 setup aws configuration

C:\Users\Roopesh>aws configure

AWS Access Key ID [None]: AKIAYEBVGVZGFEH6CNOQ

AWS Secret Access Key [None]: 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

Default region name [None]: ap-south-1

Default output format [None]: json

Step 4 : Create a Security Group

C:\Users\Roopesh>aws ec2 create-security-group --group-name lamp --description "Security group for Lixux Apache maven PHP"

{

"**GroupId": "sg-0e21a6583dcce81c1**"

}

Step 5: Assign permissions for ports and ip address to allow all the IP range use 0.0.0.0/0 as ip instead of specific

aws ec2 authorize-security-group-ingress --group-name lamp --protocol tcp --port 80 --cidr 0.0.0.0/0

aws ec2 authorize-security-group-ingress --group-name lamp --protocol tcp --port 22 --cidr 0.0.0.0/0

Step 6: To check the details of the security group

aws ec2 describe-security-groups --group-name lamp

{

"SecurityGroups": [

{

"Description": "Security group for Lixux Apache maven PHP",

"GroupName": "lamp",

"IpPermissions": [

{

"FromPort": 80,

"IpProtocol": "tcp",

"IpRanges": [

{

"CidrIp": "0.0.0.0/0"

}

],

"Ipv6Ranges": [],

"PrefixListIds": [],

"ToPort": 80,

"UserIdGroupPairs": []

},

{

"FromPort": 22,

"IpProtocol": "tcp",

"IpRanges": [

{

"CidrIp": "0.0.0.0/0"

}

],

"Ipv6Ranges": [],

"PrefixListIds": [],

"ToPort": 22,

"UserIdGroupPairs": []

}

],

"OwnerId": "558457335372",

"GroupId": "sg-0e21a6583dcce81c1",

"IpPermissionsEgress": [

{

"IpProtocol": "-1",

"IpRanges": [

{

"CidrIp": "0.0.0.0/0"

}

],

"Ipv6Ranges": [],

"PrefixListIds": [],

"UserIdGroupPairs": []

}

],

"VpcId": "vpc-ee445f86"

}

]

}

Step 8: before creating an instance create a ssh key pair to access the ec2 instance

Amazon aws restricts its access by requiring public private key by ssh key pair

Public key in machine side

Private key is with client side – can only be use with its public key

Inorder to create that use

**ec2 create-key-pair --key-name LampKey --output text**

-----BEGIN RSA PRIVATE KEY-----

MIIEogIBAAKCAQEAqdy2fXhxDO+qTd3a3LmEG3tzX7bznMNksrEqiLmblEWobshNO+Ifmv1yD+BO

Q9x7o63Ki+htNJQkJXSnmqNcam8Bvsb4jLJoSkFzYWEc7aLG+ut7O0sO6Icr8vnEqxG2sfMzdkvm

HCCynfFmWEwNwyf18geoBZCJjLQzh1IBFvWZoaDHHjoYlfs8e4Pz3PAxVMihlroBHLIvc4XrMldi

l7jktzjLxorgF5T5f09gnuXmrozXzR5D6fIkawXTdW7P0Kuxih+e9nvCtH0qasy3TxZ8T4LP+S+I

8n2D7R7f+OxdnCWaJETIvzDZyBZNCH6ymz9mPevqdqllfdIrJ94jFQIDAQABAoIBAG+sKVXH7xsA

HcjfiFoApTekK6/LidGQLA3GqB4HWJ9cRGXAzfYGrZyp72k8oHfA5pAvW7AreFBO7Iyrat0hdUqL

/Gu+R/3H9QSMjxyAShJ5J7QjSHss13pME7rENmX5WVEYgMbPE1N+s69e21hFd+7LwMgD/JTb93JH

rnDSmuDKNeki1pMnQo4ZLgNn2TbLRQgVcEYaFU72YSZzz1lxeLX9NL8L4e+Bnnni3diVXfwDdTSJ

dt1mQLKFkdz9IHU2pn7CYrh2BZ2GHY3iqg/q+JOqrB7KILC5zd14gRPjwhRjYxHzy7wEdrWDlVjF

CfuwnjSCJ9Ad7SOl/ED/Lt+eVWkCgYEA8EmaoI8gc4qDOma1o433l0gItV+JX2Gw0Akpwuj6NNTQ

qLJ82vXEQNjJnlanK3Gv82ZICws606W6AizVwIATr3hqdp44KqMm+z3M00AMiNVBAz2/ucgCWli6

rUAPAtYfiRAA4VHyQztShLQE43P5JwivhXFopY/bvz62HoRve38CgYEAtPgxgJYTnBm9+c6c1A3G

5TP8Olaqnc2FcSn+rF92PVHWwwKgiDwqxhs4TY5JdaKDYs5nBuupxMyNwwsPcjA+GO7hksr1nmUQ

AS4tGkYip58XmrCvx5FVHjTDdAf5trRrjrnCCKRnieNlWCFZ5UGb4uI2icnUJpQBnYGrt9jQ+2sC

gYAtseGc7Pr/3fRss8GRE9tkwocdbsiiAeLs/QEbFtEWIO55VlAWF6y1pERbPyFVSq5qPCYxK3aJ

bze7WCliDTkxIlYkrpwBx7sPBkagWcrIbr2JT0HZo6aG5d7COa9hG7UtjISOVNYcYPilUdKoKpLU

EHvIhbOf4IaqjNN3ZDoLIwKBgEA2/aHKHyU6AS5eQUoBpXz5SFwPGrWqEtrH8wWJTjSOEK3cBcix

yN/JEt56SdjDFOn/oZFUGEar6vW1vAJh2fcQ9NwYxrUw2VAB6aE0F+Hw0vhiD8255UE1qdFHbzWE

qdHYWmCkrdTA6tot4281K7KDYKTeCDi0NiqEs8El32oHAoGAMWMjC/K5SVEPO9ifhn69nV3ymBU7

KUNSHIVZq4dJLFPSDkE7t6swAUIaBvdQMe1Idb+3ZwzbWPQtJUvQvfEZN/1ecy7RDZ5Fv2Oto8X2

GD8IgQr6vZB0Z+XtUzo90dTX/PQzHT85zgC7eSZLwgaixA+eBXp5mGhvXHngAwyqw70=

-----END RSA PRIVATE KEY----- LampKey key-09cc2e6a89ae08f28

Step 9 : Create an ec2 instance in cli

**aws ec2 run-instances --image-id ami-08706cb5f68222d09 --instance-type t2.micro --key-name LampKey --security-group-ids sg-0e21a6583dcce81c1**

{

"Groups": [],

"Instances": [

{

"AmiLaunchIndex": 0,

"ImageId": "ami-08706cb5f68222d09",

"InstanceId": "i-0468c45b665e7fcf4",

"InstanceType": "t2.micro",

"KeyName": "LampKey",

"LaunchTime": "2020-07-15T06:15:43+00:00",

"Monitoring": {

"State": "disabled"

},

"Placement": {

"AvailabilityZone": "ap-south-1a",

"GroupName": "",

"Tenancy": "default"

},

"PrivateDnsName": "ip-172-31-37-135.ap-south-1.compute.internal",

"PrivateIpAddress": "172.31.37.135",

"ProductCodes": [],

"PublicDnsName": "",

"State": {

"Code": 0,

"Name": "pending"

},

"StateTransitionReason": "",

"SubnetId": "subnet-b4babbdc",

"VpcId": "vpc-ee445f86",

"Architecture": "x86\_64",

"BlockDeviceMappings": [],

"ClientToken": "40b3f08c-3a3e-4267-935d-3191e83bc29a",

"EbsOptimized": false,

"Hypervisor": "xen",

"NetworkInterfaces": [

{

"Attachment": {

"AttachTime": "2020-07-15T06:15:43+00:00",

"AttachmentId": "eni-attach-0d52d819222234d89",

"DeleteOnTermination": true,

"DeviceIndex": 0,

"Status": "attaching"

},

"Description": "",

"Groups": [

{

"GroupName": "lamp",

"GroupId": "sg-0e21a6583dcce81c1"

}

],

"Ipv6Addresses": [],

"MacAddress": "02:a8:81:2b:37:00",

"NetworkInterfaceId": "eni-06c9555bd3f7c2080",

"OwnerId": "558457335372",

"PrivateDnsName": "ip-172-31-37-135.ap-south-1.compute.internal",

"PrivateIpAddress": "172.31.37.135",

"PrivateIpAddresses": [

{

"Primary": true,

"PrivateDnsName": "ip-172-31-37-135.ap-south-1.compute.internal",

"PrivateIpAddress": "172.31.37.135"

}

],

"SourceDestCheck": true,

"Status": "in-use",

"SubnetId": "subnet-b4babbdc",

"VpcId": "vpc-ee445f86",

"InterfaceType": "interface"

}

],

"RootDeviceName": "/dev/xvda",

"RootDeviceType": "ebs",

"SecurityGroups": [

{

"GroupName": "lamp",

"GroupId": "sg-0e21a6583dcce81c1"

}

],

"SourceDestCheck": true,

"StateReason": {

"Code": "pending",

"Message": "pending"

},

"VirtualizationType": "hvm",

"Description": "",

"Groups": [

{

"GroupName": "lamp",

"GroupId": "sg-0e21a6583dcce81c1"

}

],

"Ipv6Addresses": [],

"MacAddress": "02:a8:81:2b:37:00",

"NetworkInterfaceId": "eni-06c9555bd3f7c2080",

"OwnerId": "558457335372",

"PrivateDnsName": "ip-172-31-37-135.ap-south-1.compute.internal",

"PrivateIpAddress": "172.31.37.135",

"PrivateIpAddresses": [

{

"Primary": true,

"PrivateDnsName": "ip-172-31-37-135.ap-south-1.compute.internal",

"PrivateIpAddress": "172.31.37.135"

}

],

"SourceDestCheck": true,

"Status": "in-use",

"SubnetId": "subnet-b4babbdc",

"VpcId": "vpc-ee445f86",

"InterfaceType": "interface"

}

],

"RootDeviceName": "/dev/xvda",

"RootDeviceType": "ebs",

"SecurityGroups": [

{

"GroupName": "lamp",

"GroupId": "sg-0e21a6583dcce81c1"

}

],

"SourceDestCheck": true,

"StateReason": {

"Code": "pending",

"Message": "pending"

},

"VirtualizationType": "hvm",

Step 10: To check the instance run the command

**aws ec2 describe-instance-status --instance-ids i-0468c45b665e7fcf4**

{

"InstanceStatuses": [

{

"AvailabilityZone": "ap-south-1a",

"InstanceId": "i-0468c45b665e7fcf4",

"InstanceState": {

"Code": 16,

"Name": "running"

},

"InstanceStatus": {

"Details": [

{

"Name": "reachability",

"Status": "passed"

}

],

"Status": "ok"

},

"SystemStatus": {

"Details": [

{

"Name": "reachability",

"Status": "passed"

}

],

"Status": "ok"

}

}

]

}