**BUGDEFENDER CLOUD BASED ANTIVIRUS**

**Step 1: Creating dynamodb table to save scan data**

1. In the AWS Console, hover over the serices drop-down and select dynamodb
2. Click the create table button
3. Enter a table name, e.g**. s3scanner**
4. Enter primary Key e.g. **userid**
5. Tick add sort key and Enter sort key e.g. **objectkey**
6. Click the create table button

**Step2: Creating SQS queue to s3 upload event notification**

1. In the AWS Console, hover over the serices drop-down and select SQS(Simple Queue Service)
2. Click on CREATE QUEUE button on upper right corner
3. Choose STANDARD QUEUE
4. Enter Queue name e.g. **BugDefenderQueue**
5. In CONFIGURATION SECTION, set Visibility timeout to 3 Minutes
6. In Access policy, add the below policy document

|  |
| --- |
| 1. { 2. "Version": "2008-10-17", 3. "Id": "\_\_default\_policy\_ID", 4. "Statement": [ 5. { 6. "Sid": "\_\_owner\_statement", 7. "Effect": "Allow", 8. "Principal": "\*", 9. "Action": "SQS:\*", 10. "Resource": "arn:aws:sqs:us-east-1:327348694707:BugDefenderQueue", 11. "Condition": { 12. "ArnEquals": { 13. "aws:SourceArn": "arn:aws:s3:::bugdefender" 14. } 15. } 16. } 17. ] 18. } |

1. Now left every thing default, and click CREATE QUEUE button

**Step3: Creating S3 bucket and setup event notification**

1. In the AWS Console, hover over the serices drop-down and select S3
2. Click on CREATE BUCKET on upper right corner
3. Enter BUCKET NAME and choose region same as other servies like ‘us-east-1’
4. Click on CREATE BUCKET button
5. From S3 home page click on your newly created bucket name
6. On your bucket page,switch to properties tab at 2nd position
7. Scroll down to EVENT NOTIFICATION card
8. Click on CREATE EVENT NOTIFICATION
9. Scroll down to EVENT TYPES card and tick **All object create events**
10. Scroll Down to Destination card and choose SQS Queue as destination
11. Specify SQS queue by choosing from your create SQS queue eg. **BugDefenderQueue**
12. Click on save changes button at bottom

**Step4: Creating role for Lambda function**

1. In the AWS Console, hover over the Services drop-down and select IAM.
2. On the left navigation bar, click the "Roles" link.
3. Click the "Create role" button.
4. Click lambda from common use case and click next permissions button
5. From policy page add following permisions
6. [AmazonS3FullAccess](https://console.aws.amazon.com/iam/home?region=us-east-1#/policies/arn%3Aaws%3Aiam%3A%3Aaws%3Apolicy%2FAmazonS3FullAccess)
7. [AmazonEC2ReadOnlyAccess](https://console.aws.amazon.com/iam/home?region=us-east-1#/policies/arn%3Aaws%3Aiam%3A%3Aaws%3Apolicy%2FAmazonEC2ReadOnlyAccess)
8. [AmazonDynamoDBFullAccess](https://console.aws.amazon.com/iam/home?region=us-east-1#/policies/arn%3Aaws%3Aiam%3A%3Aaws%3Apolicy%2FAmazonDynamoDBFullAccess)
9. [AmazonSSMFullAccess](https://console.aws.amazon.com/iam/home?region=us-east-1#/policies/arn%3Aaws%3Aiam%3A%3Aaws%3Apolicy%2FAmazonSSMFullAccess)
10. [AWSLambdaSQSQueueExecutionRole](https://console.aws.amazon.com/iam/home?region=us-east-1#/policies/arn%3Aaws%3Aiam%3A%3Aaws%3Apolicy%2Fservice-role%2FAWSLambdaSQSQueueExecutionRole)
11. [AWSLambdaExecute](https://console.aws.amazon.com/iam/home?region=us-east-1#/policies/arn%3Aaws%3Aiam%3A%3Aaws%3Apolicy%2FAWSLambdaExecute)
12. Click on next tags button, keep default
13. Click next review button
14. Enter role Name e.g. S3ScannerRole
15. Click on create role button

**Step4: Creating SQS triggered Lambda function**

1. In the AWS Console, hover over the Services drop-down and select LAMBDA.
2. Click on CREATE FUNCTION button on upper right corner
3. Choose author from scratch
4. Enter function name e.g. **S3Scanner**
5. Choose runtime Python 3.8
6. Under permissions expand change default execution role
7. Select user exiting role and and choose recently create role “**S3ScannerRole**” from dropdown
8. Click on CREATE FUNCTION button
9. Copy paste code from **s3\_lambda\_scanner.py** from project directory

**Step5: Setting Lambda Function to SQS Lambda triggers**

1. Open SQS and select your created SQS Queue
2. Scroll down and click on lambda triggers tab
3. Click on configure lambda trigger function
4. Choose the lambda function S3Scanner from the dropdown and save

**Step 6: Setting up cognito for user authentication and IAM Policies to upload to s3 and read from dynamodb**

1. In the AWS Console, hover over the Services drop-down and select IAM.
2. On the left navigation bar, click the "Policies" link.
3. Click the "Create policy" button.

a. Click the "Select" button for "Create Your Own Policy"

b. Enter a policy name, e.g. "**policy-bugdefender-cognito-authenticated**"

c. For the policy document, you can use the template below.

**6.a policy for authenticated cognito user**

**Policy name : “policy-bugdefender-cognito-authenticated**”

|  |
| --- |
| {  "Version": "2012-10-17",  "Statement": [  {  "Effect": "Allow",  "Action": [  "mobileanalytics:PutEvents",  "cognito-sync:\*",  "cognito-identity:\*"  ],  "Resource": [  "\*"  ]  },  {  "Effect": "Allow",  "Action": [  "s3:ListBucket"  ],  "Resource": [  "arn:aws:s3:::bugdefender"  ]  },  {  "Effect": "Allow",  "Action": [  "s3:PutObject",  "s3:GetObject",  "s3:DeleteObject"  ],  "Resource": [  "arn:aws:s3:::bugdefender/\*"  ]  },  {  "Effect": "Allow",  "Action": [  "dynamodb:Query"  ],  "Resource": [  "arn:aws:dynamodb:us-east-1:327348694707:table/s3scanner"  ]  }  ]  } |
|  |

**6.b policy for unauthenticated user**

Click the "Create policy" button.

a. Click the "Select" button for "Create Your Own Policy"

b. Enter a policy name, e.g. "**policy-bugdefender-cognito-unauthenticated**"

c. For the policy document, you can use the template below.

|  |
| --- |
| {  "Version": "2012-10-17",  "Statement": [  {  "Effect": "Allow",  "Action": [  "mobileanalytics:PutEvents",  "cognito-sync:\*"  ],  "Resource": [  "\*"  ]  },  {  "Effect": "Allow",  "Action": [  "s3:ListAllMyBuckets"  ],  "Resource": [  "arn:aws:s3:::\*"  ]  },  {  "Effect": "Allow",  "Action": [  "s3:ListBucket"  ],  "Resource": [  "arn:aws:s3:::bugdefender"  ]  },  {  "Effect": "Allow",  "Action": [  "s3:GetObject"  ],  "Resource": [  "arn:aws:s3:::bugdefender/\*"  ]  }  ]  } |

**6.c Create user pool**

In the AWS Console, hover over the Services drop-down and select Cognito.

To start with User Pools, click "Manage your User Pools".

In the top right, click "Create a user pool".

Enter a user pool name, e.g. "bugdefender-user-pool-1"

Click the "Review defaults" button.

Click the "Create pool" button.

Make note of the Pool Id, e.g. "us-east-1\_JuJjoK4XS".

**6.d Create application client**

Scroll down to "App clients" and click the "Add app client..." link. Click the "Add an app client" link.

Enter an app client name, e.g. "bugdefender-app-client-1".

Uncheck the "Generate client secret" checkbox. To simplify implementation later, disable this feature.

Click the "Create app client" button.

Make note of the App client id, e.g. "7a9v27fkto1b7l572i1369l89v".

**6.e Create federated identities pool**

In the top left of your browser, you should see "User Pools | Federated Identities". Click the grayed out "Federated Identities" link.

Click the "Create new identity pool" button.

Enter a federated identities pool name, e.g. "bugdefender\_fed\_ids\_pool\_1". Note that dashes are not supported, so we used underscore instead.

Under Authentication Providers, choose the Cognito tab.

Enter the Pool Id that we took note of above in the User Pool section, e.g. "us-east-1\_JuJjoK4XS".

Enter the App client id that we took note of above in the Application Client section, e.g. "7a9v27fkto1b7l572i1369l89v".

Click the "Create Pool" button.

Cognito will inform you that Cognito identities need access to your resources via IAM roles. Click the "View Details" link to view the role details.

Create new roles for both authenticated and unauthenticated identities. Enter names for each one, e.g. "role-bugdefender-cognito-authenticated" and "role-bugdefender-cognito-unauthenticated"

Note that we will leave the policies as Cognito defaults for now but will replace those policies with the ones we made above.

Click the "Allow" button.

Make note of the Federated Identities pool id, which can be found when you click on the pool, then click on "Edit identity pool", e.g. "us-east-1:0488063e-b4c6-4b1f-86aa-79dd82a88dac".

**6.f Update Cognito roles with previously created policies**

In the AWS Console, hover over the Services drop-down and select IAM.

On the left navigation bar, click the "Roles" link.

Search for the "role-bugdefender-cognito-authenticated". Remove the oneclick\_\* policy and add the policy that we created previously: "policy-bugdefender-cognito-authenticated".

Search for the "role-bugdefender-cognito-unauthenticated". Remove the oneclick\_\* policy and add the policy that we created previously: "policy-bugdefendender-cognito-unauthenticated".

**Step7: Creating Role and policies for EC2 instances**

1. In the AWS Console, hover over the Services drop-down and select IAM.
2. On the left navigation bar, click the "Roles" link.
3. Click the "Create role" button.
4. Click lambda from common use case and click next permissions button
5. From policy page add following permissions

a. [AmazonS3FullAccess](https://console.aws.amazon.com/iam/home?region=us-east-1#/policies/arn%3Aaws%3Aiam%3A%3Aaws%3Apolicy%2FAmazonS3FullAccess)

b. [AmazonSSMManagedInstanceCore](https://console.aws.amazon.com/iam/home?region=us-east-1#/policies/arn%3Aaws%3Aiam%3A%3Aaws%3Apolicy%2FAmazonSSMManagedInstanceCore)

1. Click on next tags button, keep default
2. Click next review button
3. Enter role Name e.g. **Antivirus\_Worker\_Role**
4. Click on create role button

**Step8: Launching EC2 instances**

1. In the AWS Console, hover over the Services drop-down and select EC2
2. Scroll down and click on launch instance button from instance card
3. Choose **Ubuntu Server 18.04 LTS (HVM), SSD Volume Type 64-bit(x86)** and click select button
4. Click next botton configure instance details
5. Choose IAM role created from the dropdown “**Antivirus\_Worker\_Role**”
6. Click on next and keep every thing default and click launch button
7. Choose existing key pair if you have already created and has .pem file available otherwise create new key pair and save
8. Click on launch instance.

**Step9: Configuring EC2 instances**

**9.a connect to your instance via ssh client**

**>> SSH –I “pem file” instance public DNS**

**e.g.** ssh -i "yourpemfile.pem" [ubuntu@ec2-184-72-205-87.compute-1.amazonaws.com](mailto:ubuntu@ec2-184-72-205-87.compute-1.amazonaws.com)

**9.b update the linux instance**

**>>sudo apt-get update**

**>>** **sudo apt-get upgrade –y**

**9.c Swaping on linux instances**

**Check if any swap has already enabled**

>> sudo swapon –show

>> sudo fallocate -l 2G /swapfile

>> sudo chmod 600 /swapfile

>> sudo mkswap /swapfile

>> sudo swapon /swapfile

>> sudo nano /etc/fstab

**Paste this in file and save with ctrl+X**

/swapfile swap swap defaults 0 0

>> sudo swapon --show

**Details reference**

<https://linuxize.com/post/how-to-add-swap-space-on-ubuntu-18-04/>

**9.d install pip3**

**>>sudo apt-get install python3-pip**

**9.e install boto3**

**>>pip3 install boto3**

**9.f copying scanner script to intances**

**>> nano scanner.py**

Copy from python file available in project directory

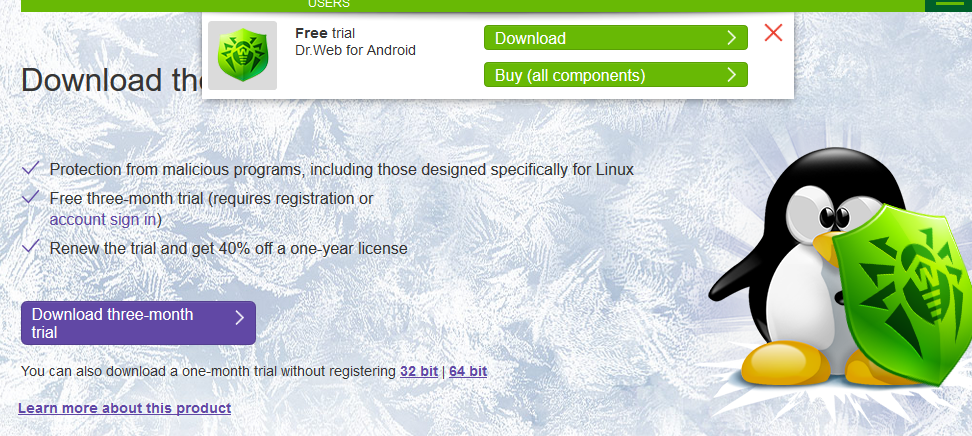
1. Clamav\_scanner.py for clamav installed instance
2. sophos\_scanner.py for sophos installed intance
3. drweb\_scanner.py for drweb installed instance

type CTRL+X to save

**Step10: Installing Antivirus on Instances**

**10.a Instaling drweb antivirus**

Open <https://download.drweb.com/linux/?lng=en>



**10.b right click on one month trail 64 bit and save link location**

**Eg.** [**https://download.geo.drweb.com/pub/drweb/unix/workstation/11.1/drweb-11.1.2-av-linux-amd64.run**](https://download.geo.drweb.com/pub/drweb/unix/workstation/11.1/drweb-11.1.2-av-linux-amd64.run)

**Download installation file on your instance**

**>> wget** [**https://download.geo.drweb.com/pub/drweb/unix/workstation/11.1/drweb-11.1.2-av-linux-amd64.run**](https://download.geo.drweb.com/pub/drweb/unix/workstation/11.1/drweb-11.1.2-av-linux-amd64.run)

**Make file executable on linux**

**>> chmod +x drweb-11.1.2-av-linux-amd64.run**

**>> ./drweb-11.1.2-av-linux-amd64.run**

**Follow the prompt to install**

[**https://download.geo.drweb.com/pub/drweb/unix/workstation/11.1/documentation/html/en/**](https://download.geo.drweb.com/pub/drweb/unix/workstation/11.1/documentation/html/en/)

**10.b Instaling SOPHOS antivirus**

**Download Sophos antivirus file**

[**https://github.com/maliceio/malice-av/raw/master/sophos/sav-linux-free-9.tgz**](https://github.com/maliceio/malice-av/raw/master/sophos/sav-linux-free-9.tgz)

**>> wget** [**https://github.com/maliceio/malice-av/raw/master/sophos/sav-linux-free-9.tgz**](https://github.com/maliceio/malice-av/raw/master/sophos/sav-linux-free-9.tgz)

**>>** tar -xzvf sav-linux-free-9.tgz

**>>** ./sophos-av/install.sh

**For reference**

[**https://docs.sophos.com/esg/SAV-Linux/help/en-us/PDF/sav\_linux\_sg.pdf**](https://docs.sophos.com/esg/SAV-Linux/help/en-us/PDF/sav_linux_sg.pdf)

**10.c Installing CLAMAV antivirus**

**1. Install the “clamav-daemon” Package**

>> sudo apt-get install clamav clamav-daemon

**2. Enable & start the “clamav-daemon” service**

>> sudo systemctl enable clamav-daemon

>> sudo systemctl start clamav-daemon

**Detail reference:**

[**https://aaronbrighton.medium.com/installation-configuration-of-clamav-antivirus-on-ubuntu-18-04-a6416bab3b41**](https://aaronbrighton.medium.com/installation-configuration-of-clamav-antivirus-on-ubuntu-18-04-a6416bab3b41)