Quantum Transfer (QT), Receiver AMI Deployment and Configuration Guide

This document provides a comprehensive guide on how to deploy, configure, and run the "Quantum Transfer, Sender - AWS Metered" AMI from the AWS Marketplace.

Part 1: AWS AMI Deployment

Follow these steps to deploy the AMI and set up the necessary AWS permissions.

1. **Find the AMI and Deploy the Instance:**
   - Navigate to VIA on the AWS Marketplace.
   - Select the "Quantum Transfer, Sender - AWS Metered" AMI from results section.
   - Follow the on-screen instructions to launch and deploy the AMI. Choose an appropriate instance size (we recommend a minimum of **t3.medium** or **t3.large**) and configure your network (VPC, security groups) as needed. Ensure your security group allows SSH access (port 22) from your IP address.
2. **Create an S3 Destination Bucket:**
   - Before creating the IAM role, you must create the S3 bucket that will receive the transferred data. For instructions on creating a new S3 bucket, please refer to the AWS documentation on Creating a Bucket and Getting Started with Amazon S3.
3. **Create an IAM Role:**
   - Go to the **IAM** service in your AWS console.
   - Navigate to **Roles** and click **Create role**.
   - Select **AWS service** as the trusted entity type, and choose **EC2** as the use case. Click **Next**.
   - Attach policies that grant the necessary permissions. At a minimum, you will need a policy that allows read access (e.g., s3:PutObject, s3:ListBucket) to your destination S3 bucket. You can create a new custom policy for this. For detailed guidance on permissions, refer to the AWS documentation on IAM Policy Language Overview and S3 Read/Write Policy Examples.
   - When creating a custom policy, you will typically need to select the **JSON tab** to paste the example policy below.

Example Policy (Read-Only Access to a Specific Bucket):

```
JSON
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:PutObject",
                "s3:PutObjectAcl",
                "s3:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3:::YOUR-DESTINATION-BUCKET-NAME",
                "arn:aws:s3:::YOUR-DESTINATION-BUCKET-NAME/*"
            ]
        }
    ]
}
```

- Click **Next**, give the role a descriptive name (e.g., QT-Receiver-S3-Access-Role), and complete its creation.

4. **Attach the Role to the Instance:**
   - Go to the **EC2** service in your AWS console.
   - Select your newly deployed "Quantum Transfer Receiver" instance.
   - Click **Actions** > **Security** > **Modify IAM role**.
   - From the dropdown menu, select the QT-Receiver-S3-Access-Role you just created.
   - Click **Update IAM role**.

Part 1.5: Enabling SSH Access & Network Configuration

1. **Locate Your Instance:** Navigate to the **EC2 Dashboard** and select **Instances**. Find your running "Quantum Transfer Receiver" instance in the list and select it.
2. **Ensure Internet Connectivity (Optional):** If you deployed your instance into a VPC or Subnet that does not have an Internet Gateway attached, you will not be able to connect via SSH using a public IP. To fix this:
   - **Create & Attach Gateway:** Go to the **VPC Dashboard** > **Internet Gateways**, create a new gateway, and attach it to your VPC.
   - **Update Route Table:** In **VPC Dashboard** > **Route Tables**, find the table associated with your instance's subnet. Edit the routes to add `0.0.0.0/0` pointing to the Internet Gateway you just created.

3. **Access Security Settings:** With the instance selected in the EC2 Dashboard, look at the bottom

pane (Instance details). Click on the **Security** tab.

4. **Open Security Group:** Under the **Security groups** section, click the link (ID or Name) of the security group attached to your instance.
5. **Edit Inbound Rules:** Select the Security Group from the list, then click on the **Inbound rules** tab in the bottom pane. Click the **Edit inbound rules** button.
6. **Add SSH Rule:** Click **Add rule** and configure the following:
   - **Type:** Select **SSH**.
   - **Protocol:** TCP (Automatically selected).
   - **Port Range:** 22 (Automatically selected).
   - **Source:** Select **Anywhere-IPv4** (`0.0.0.0/0`) to allow access from any IP, or select **My IP** / **Custom** to restrict access to a specific vetting IP range.
7. **Save Changes:** Click **Save rules** to apply the changes immediately

Part 2: Configure the Deployed Instance

Now, connect to your instance via SSH to set the required environment variables.

1. **Connect to your instance:**
   - Use your EC2 key pair to SSH into the instance.
   - ssh -i /path/to/your-key.pem ec2-user@YOUR-INSTANCE-PUBLIC-IP
2. **Set Environment Variables:**
   - Set the following environment variables. You can add these to your ~/.bash_profile or ~/.bashrc file to make them persistent across reboots.

```
# Set the name of the S3 destination bucket
export DESTINATION_BUCKET="YOUR-DESTINATION-BUCKET-NAME"

# Boolean to send or not to the cloud
export SEND_TO_CLOUD="true"

# If you want or not to store the decrypted file on the file system. Set to true to store the file.
export LOCAL_SAVE="false"

# The location where you want to store the file. The folder needs to exist with read and write permission. Will default to /tmp.
export LOCAL_STORAGE_PATH="/tmp"
```

3. **Load the new variables:**
   - If you added the variables to ~/.bash_profile, run:
     source ~/.bash_profile

Part 3: Run the Script

Once the instance is configured, you can execute the main script.

1. While logged in via SSH, run the following command:

- cd /opt/dcac
- ./run_job.sh

The script will execute, utilizing the environment variables you configured. Upon execution, it automatically generates a unique receiver email address based on your AWS instance ID, using the format `aws-{instance id}@machine.solvewithvia.com`. This critical address, needed for the sending party to initiate transfers, is saved in the file `/opt/dcac/recipient_email.txt`. You can retrieve it by running: `cat /opt/dcac/recipient_email.txt`.