Configuring an AWS Account

Create AWS Account

This is a **one-time** step.

To create an AWS account, follow these steps:

1. **Open the AWS Sign-Up Page:**
   - Go to AWS's account creation page and click on "Create an AWS Account".
2. **Enter Your Account Information:**
   - Provide your email address and AWS account name. After entering the details, click on "Verify email address". You'll receive an email with a verification code that you need to enter on the signup page.
   - Set up your root user password and confirm it.
3. **Add Contact Information:**
   - Choose between a Personal or Professional account. The features and functions are the same, but the information required might differ slightly.
   - Fill in your contact information and read the AWS Customer Agreement. Once done, click "Continue".
4. **Enter Payment Information:**
   - Provide a valid payment method (credit/debit card). This step is mandatory even if you plan to use the free tier services. You may also need to verify your payment method by entering a code sent by AWS.
5. **Verify Your Phone Number:**
   - AWS will ask you to provide a phone number to verify your identity. You'll receive a call or SMS with a verification code that you need to enter to proceed.
6. **Select a Support Plan:**
   - Choose one of the available support plans (Basic, Developer, Business, or Enterprise). The Basic support plan is free.
7. **Complete the Sign-Up Process:**
   - After selecting your support plan, AWS will begin activating your account. This process usually takes a few minutes but can take up to 24 hours. Once activation is complete, you'll receive a confirmation email.

After your account is activated, you can sign in to the AWS Management Console and start using AWS services.

Login to AWS Account

To log in to your AWS account, follow these steps:

1. **Navigate to the AWS Management Console:**

    - Open your web browser and go to the AWS Management Console login page.
2. **Enter Your Account Credentials:**
    - On the login page, you'll see two options:
        - Root user: Use this if you are logging in as the account owner or the main administrator.
        - IAM user: Use this if you have been assigned a user account within an AWS organization.
    - If you are the root user, click on "Root user" and enter your email address associated with the AWS account. Click "Next".
    - If you are an IAM user, click on "IAM user" and enter your IAM username along with the account ID or alias. Click "Next".
3. **Enter Your Password:**
    - Enter the password for your account, and click "Sign In".
4. **Multi-Factor Authentication (MFA) (if enabled):**
    - If Multi-Factor Authentication (MFA) is enabled for your account, you will be prompted to enter the code from your authentication device (like a mobile app). Enter the code and proceed.
5. **Access the AWS Management Console:**
    - After successful authentication, you will be directed to the AWS Management Console, where you can start managing your AWS services.

Create an S3 Bucket

This is a **one-time** step.

You will now create an AWS S3 bucket that will contain all of your Nanopore POD5 sample files. To create an S3 bucket in your AWS account, follow these steps:

1. **Access the S3 Service:**
    - Once you're in the console, type "S3" into the search bar at the top and select "S3" from the drop-down list. This will take you to the Amazon S3 dashboard.
2. **Start the Bucket Creation Process:**
    - On the S3 dashboard, click the "Create bucket" button.
3. **Configure the Bucket Settings:**
    - Bucket Name: Enter a name for the bucket in the Bucket name field.
        - Bucket names must be globally unique across all existing bucket names in Amazon S3. If you choose a name that is already taken, you'll have to try other names until you find a unique name.
        - The name must not contain spaces and must adhere to the bucket naming rules (e.g., lowercase letters, numbers, and hyphens).
    - Region: Select the region "US East (N. Virginia) us-east-1". This is the region where the S3 bucket will be created.

4. **Set Bucket Options:**
   - Block Public Access Settings: By default, S3 buckets are set to block all public access. You can leave this option enabled unless you specifically need the bucket to be public.
   - Bucket Versioning: If you want to enable versioning, which keeps multiple versions of an object in the same bucket, you can do so here. This is optional.
   - Tags, Object Lock, and Encryption: You can add tags, enable object lock, and set default encryption if needed, but these are optional and can be configured later.
5. **Review and Create the Bucket:**
   - After configuring the settings, review your choices and click "Create bucket".
6. **Verify Bucket Creation:**
   - After the bucket is created, you will be taken back to the S3 dashboard, where you should see Bucket name listed among your S3 buckets.

Upload POD5 Files to S3 Bucket

There are several methods available for uploading POD5 files to S3 buckets. These methods are beyond the scope of this document. For human whole genome sequencing, the aggregate size of these files is quite large, ca. 1TB or more, and may need special methods to upload in reasonable time. Please contact us at support@thesequencingcenter.com to review and discuss options for uploading large datasets.

Store POD5 Files in S3 Bucket

If you already have an AWS account and an S3 bucket with POD5 files in it, you have two choices. You can use the existing Bucket name with Sniffles. Or you can copy or move the POD5 files from the existing bucket to another bucket within the same AWS account. To copy or move files, follow these instructions:

Using the AWS Management Console

1. **Navigate to the Source Bucket:**
   - In the S3 dashboard, find and click on the bucket that contains the POD5 files you want to copy or move.
2. **Select the Files:**
   - Browse through the folders in your source bucket and select the files you want to copy or move. You can select multiple files by holding the Ctrl key (or Cmd on Mac) while clicking.
3. **Choose the Copy or Move Option:**
   - After selecting the files, click on the "Actions" button at the top of the screen, and choose either "Copy" or "Move".

4. **Specify the Destination Bucket:**
   - A dialog box will appear asking where you want to copy or move the files.
   - In the "Destination" field, click "Browse S3" and navigate to the destination bucket.
   - You can specify the exact location within the bucket by selecting a folder or leave it blank to place the files directly in the bucket's root.
5. **Confirm the Operation:**
   - Once the destination is set, click on "Copy" or "Move" to begin the operation. Depending on the size and number of files, this may take a few moments.
6. **Verify the Transfer:**
   - Navigate to the destination bucket and verify that the files have been successfully copied or moved.

Important Note: - In the S3 bucket that contains the POD5 files, make sure that all of the files are in the directory just below the S3 bucket name. There should be no other subdirectories under the main directory.

For example: If your bucket name is "s3://your-bucket-name/", all of the POD5 files should be immediately below this directory and there should be no subdirectories under "s3://your-bucket-name/".

Create a Key Pair

This is a one-time step.

To create a key pair in AWS, follow these steps:

Step 1: Access the EC2 Dashboard

1. **Go to EC2 Service:**
   - In the AWS Management Console, type "EC2" in the search bar and select "EC2" from the drop-down menu.

Step 2: Create a Key Pair

1. **Open the Key Pairs Section:**
   - On the EC2 Dashboard, look for the "Key Pairs" option under the "Network & Security" section on the left-hand side menu.
   - Click on "Key Pairs" to open the key pair management page.
2. **Create a New Key Pair:**
   - Click the "Create Key Pair" button.
3. **Configure Key Pair Settings:**
   - Name: Enter a name for your key pair, such as "sniffles-keypair". The name must be unique within your AWS region.
   - Key Pair Type: Choose the key pair type, typically "ED25519".

  - Private Key File Format: Choose the format for the private key file. Options include:
    - .pem: For SSH clients (like OpenSSH) on Linux or macOS.
    - .ppk: For PuTTY, a popular SSH client on Windows.
  - Tags: (Optional) You can add tags to your key pair for easier management.
4. **Create the Key Pair:**
   - Click on "Create key pair". AWS will generate the key pair and automatically download the private key file to your computer.

Step 3: Secure the Private Key

- **Save the Private Key:** The private key file will automatically download to your computer. This file is crucial for connecting to your EC2 instances securely, so store it in a safe location. If you lose this file, you will not be able to connect to your instances using this key pair.

- **Set Permissions (Linux/macOS):** If you are using the key pair on a Linux or macOS system, run the following command to set the correct permissions:

  ```
  chmod 400 /path/to/your-keypair.pem
  ```

- **Store Securely:** Make sure the private key is stored securely. Do not share it with anyone or commit it to version control systems like Git.

Important Notes: - One-Time Download: The private key file can only be downloaded once at the time of creation. If you lose the private key, you will need to create a new key pair and update your instances to use the new key. - **Accessing EC2 Instances:** When launching a new EC2 instance, you'll be able to select this key pair for SSH access. Ensure the correct permissions are set on your private key file before attempting to connect.

Create a Security Group

This is a one-time step.

To create a security group in AWS for EC2 that allows SSH logins on port 22, follow these steps:

Step 1: Access the EC2 Dashboard

1. **Go to EC2 Service:**
   - In the AWS Management Console, type "EC2" in the search bar and select "EC2" from the drop-down menu.

Step 2: Create a Security Group

1. **Open the Security Groups Section:**
    - On the EC2 Dashboard, locate the “Security Groups” option under the “Network & Security” section in the left-hand menu.
    - Click on “Security Groups” to open the security group management page.
2. **Create a New Security Group:**
    - Click on the “Create security group” button.
3. **Configure Security Group Settings:**
    - Name: Enter a name for your security group, such as “sniffles-sg”.
    - Description: Provide a brief description of the security group, like “Security group for SSH access to EC2 instances”.
    - VPC: Select the appropriate VPC (Virtual Private Cloud) where you want this security group to be used. If you only have one VPC, it will be selected by default.

Step 3: Add an Inbound Rule for SSH Access

1. **Configure Inbound Rules:**
    - In the Inbound rules section, click on “Add Rule”.
    - Type: From the drop-down list, select “SSH”. This will automatically set the Port Range to 22.
    - Source:
        - You can choose “My IP” to allow SSH access only from your current IP address.
        - Alternatively, select “Anywhere” (0.0.0.0/0) to allow SSH access from any IP address, but note that this is less secure and should be used with caution.
        - You can also specify a custom IP range if you only want to allow SSH access from specific IP addresses or ranges.

Step 4: Review and Create the Security Group

1. **Review the Rules:**
    - Review the rules you’ve added to ensure they match your requirements.
2. **Create the Security Group:**
    - Once satisfied with the configuration, click on “Create security group”.

Step 5: Attach the Security Group to an EC2 Instance

1. **During EC2 Instance Launch:**
    - When launching a new EC2 instance, you can select this security group under the “Configure Security Group” section.
2. **For Existing Instances:**
    - If you want to assign this security group to an existing instance, go to the “Instances” section, select your instance, click “Actions”

> “Networking” > “Change Security Groups”, and then select the “sniffles-sg” security group.

Important Notes: - Security Considerations: Allowing SSH access from any IP address (0.0.0.0/0) is convenient but can expose your instance to potential attacks. It’s recommended to restrict access to specific IP addresses whenever possible. **- Firewall Settings:** Ensure that any local firewalls on your machine or network allow outbound connections on port 22.