Connect to your Linux instance if you lose your private key

PDF (ec2-ug.pdf#replacing-lost-key-pair)
Kindle (https://www.amazon.com/dp/B076452RSZ)
RSS (amazon-ec2-release-notes.rss)

If you lose the private key for an EBS-backed instance, you can regain access to your instance. You must stop the instance, detach its root volume and attach it to another instance as a data volume, modify the `authorized_keys` file with a new public key, move the volume back to the original instance, and restart the instance. For more information about launching, connecting to, and stopping instances, see Instance lifecycle (./ec2-instance-lifecycle.html) .

This procedure is only supported for instances with EBS root volumes. If the root device is an instance store volume, you cannot use this procedure to regain access to your instance; you must have the private key to connect to the instance. To determine the root device type of your instance, open the Amazon EC2 console, choose **Instances**, select the instance, and check the value of **Root device type** in the details pane. The value is either `ebs` or `instance store`.

In addition to the following steps, there are other ways to connect to your Linux instance if you lose your private key. For more information, see How can I connect to my Amazon EC2 instance if I lost my SSH key pair after its initial launch? (http://aws.amazon.com/premiumsupport/knowledge-center/user-data-replace-key-pair-ec2/)

Steps for connecting to an EBS-backed instance with a different key pair

Step 1: Create a new key pair

Create a new key pair using either the Amazon EC2 console or a third-party tool. If you want to name your new key pair exactly the same as the lost private key, you must first delete the existing key pair. For information about creating a new key pair, see Create a key pair using Amazon EC2 (./ec2-key-pairs.html#having-ec2-create-your-key-pair) or Create a key pair using a third-party tool and import the public key to Amazon EC2 (./ec2-key-pairs.html#how-to-generate-your-own-key-and-import-it-to-aws) .

Step 2: Get information about the original instance and its root volume

Make note of the following information because you'll need it to complete this procedure.

To get information about your original instance

1. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/ (https://console.aws.amazon.com/ec2/) .
2. Choose **Instances** in the navigation pane, and then select the instance that you'd like to connect to. (We'll refer to this as the *original* instance.)
3. On the **Details** tab, make note of the instance ID and AMI ID.
4. On the **Networking** tab, make note of the Availability Zone.
5. On the **Storage** tab, under **Root device name**, make note of the device name for the root volume (for example, `/dev/xvda`). Then, under **Block devices**, find this device name and make note of the volume ID (for example, vol-0a1234b5678c910de).

Step 3: Stop the original instance

Choose **Instance state**, **Stop instance**. If this option is disabled, either the instance is already stopped or its root device is an instance store volume.

Warning

When you stop an instance, the data on any instance store volumes is erased. To keep data from instance store volumes, be sure to back it up to persistent storage.

Step 4: Launch a temporary instance

Choose **Launch instances**, and then use the launch wizard to launch a *temporary* instance with the following options:

- On the **Choose an AMI** page, select the same AMI that you used to launch the original instance. If this AMI is unavailable, you can create an AMI that you can use from the stopped instance. For more information, see Create an Amazon EBS-backed Linux AMI (./creating-an-ami-ebs.html) .
- On the **Choose an Instance Type** page, leave the default instance type that the wizard selects for you.
- On the **Configure Instance Details** page, specify the same Availability Zone as the original instance. If you're launching an instance in a VPC, select a subnet in this Availability Zone.
- On the **Add Tags** page, add the tag `Name=Temporary` to the instance to indicate that this is a temporary instance.
- On the **Review** page, choose **Launch**. Choose the key pair that you created in Step 1, then choose **Launch Instances**.

Step 5: Detach the root volume from the original instance and attach it to the temporary instance

1. In the navigation pane, choose **Volumes** and select the root device volume for the original instance (you made note of its volume ID in a previous step). Choose **Actions**, **Detach Volume**, and then select **Yes, Detach**. Wait for the state of the volume to become `available`. (You might need to choose the **Refresh** icon.)
2. With the volume still selected, choose **Actions**, and then select **Attach Volume**. Select the instance ID of the temporary instance, make note of the device name specified under **Device** (for example, `/dev/sdf`), and then choose **Attach**.

   Note

If you launched your original instance from an AWS Marketplace AMI and your volume contains AWS Marketplace codes, you must first stop the temporary instance before you can attach the volume.

Step 6: Add the new public key to `authorized_keys` on the original volume mounted to the temporary instance

1. Connect to the temporary instance.

2. From the temporary instance, mount the volume that you attached to the instance so that you can access its file system. For example, if the device name is `/dev/sdf`, use the following commands to mount the volume as `/mnt/tempvol`.

   Note

   The device name might appear differently on your instance. For example, devices mounted as `/dev/sdf` might show up as `/dev/xvdf` on the instance. Some versions of Red Hat (or its variants, such as CentOS) might even increment the trailing letter by 4 characters, where `/dev/sdf` becomes `/dev/xvdk`.

   1. Use the **lsblk** command to determine if the volume is partitioned.

      ```
      [ec2-user ~]$ lsblk
      NAME    MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
      xvda    202:0    0    8G  0 disk
      └─xvda1 202:1    0    8G  0 part /
      xvdf    202:80   0  101G  0 disk
      └─xvdf1 202:81   0  101G  0 part
      xvdg    202:96   0   30G  0 disk
      ```

      In the preceding example, `/dev/xvda` and `/dev/xvdf` are partitioned volumes, and `/dev/xvdg` is not. If your volume is partitioned, you mount the partition (`/dev/xvdf1`) instead of the raw device (`/dev/xvdf`) in the next steps.

   2. Create a temporary directory to mount the volume.

      ```
      [ec2-user ~]$ sudo mkdir /mnt/tempvol
      ```

   3. Mount the volume (or partition) at the temporary mount point, using the volume name or device name that you identified earlier. The required command depends on your operating system's file system. Note that the device name might appear differently on your instance. See note (#device-name) in this section for more information.

      - Amazon Linux, Ubuntu, and Debian

        ```
        [ec2-user ~]$ sudo mount /dev/xvdf1 /mnt/tempvol
        ```

      - Amazon Linux 2, CentOS, SUSE Linux 12, and RHEL 7.x

        ```
        [ec2-user ~]$ sudo mount -o nouuid /dev/xvdf1 /mnt/tempvol
        ```

   Note

If you get an error stating that the file system is corrupt, run the following command to use the **fsck** utility to check the file system and repair any issues:

```
[ec2-user ~]$ sudo fsck /dev/xvdf1
```

3. From the temporary instance, use the following command to update `authorized_keys` on the mounted volume with the new public key from the `authorized_keys` for the temporary instance.

   Important

   The following examples use the Amazon Linux user name `ec2-user`. You might need to substitute a different user name, such as `ubuntu` for Ubuntu instances.

   ```
   [ec2-user ~]$ cp .ssh/authorized_keys /mnt/tempvol/home/ec2-user/.ssh/authorized_keys
   ```

   If this copy succeeded, you can go to the next step.

   (Optional) Otherwise, if you don't have permission to edit files in `/mnt/tempvol`, you must update the file using **sudo** and then check the permissions on the file to verify that you are able to log into the original instance. Use the following command to check the permissions on the file.

   ```
   [ec2-user ~]$ sudo ls -l /mnt/tempvol/home/ec2-user/.ssh
   total 4
   -rw------- 1 222 500 398 Sep 13 22:54 authorized_keys
   ```

   In this example output, `222` is the user ID and `500` is the group ID. Next, use **sudo** to re-run the copy command that failed.

   ```
   [ec2-user ~]$ sudo cp .ssh/authorized_keys /mnt/tempvol/home/ec2-user/.ssh/authorized_keys
   ```

   Run the following command again to determine whether the permissions changed.

   ```
   [ec2-user ~]$ sudo ls -l /mnt/tempvol/home/ec2-user/.ssh
   ```

   If the user ID and group ID have changed, use the following command to restore them.

   ```
   [ec2-user ~]$ sudo chown 222:500 /mnt/tempvol/home/ec2-user/.ssh/authorized_keys
   ```

Step 7: Unmount and detach the original volume from the temporary instance, and reattach it to the original instance

1. From the temporary instance, unmount the volume that you attached so that you can reattach it to the original instance. For example, use the following command to unmount the volume at `/mnt/tempvol`.

   ```
   [ec2-user ~]$ sudo umount /mnt/tempvol
   ```

2. Detach the volume from the temporary instance (you unmounted it in the previous step): From the Amazon EC2 console, select the root device volume for the original instance (you made note of the volume ID in a previous step), choose **Actions**, **Detach Volume**, and then choose **Yes, Detach**. Wait for the state of the volume to become `available`. (You might need to choose the **Refresh** icon.)

3. Reattach the volume to the original instance: With the volume still selected, choose **Actions**, **Attach Volume**. Select the instance ID of the original instance, specify the device name that you noted earlier in Step 2 (#step-2-get-info-about-original-instance) for the original root device attachment (`/dev/sda1` or `/dev/xvda`), and then choose **Attach**.

   Important

   If you don't specify the same device name as the original attachment, you cannot start the original instance. Amazon EC2 expects the root device volume at `sda1` or `/dev/xvda`.

Step 8: Connect to the original instance using the new key pair

Select the original instance, choose **Instance state**, **Start instance**. After the instance enters the `running` state, you can connect to it using the private key file for your new key pair.

Note

If the name of your new key pair and corresponding private key file is different from the name of the original key pair, ensure that you specify the name of the new private key file when you connect to your instance.

Step 9: Clean up

(Optional) You can terminate the temporary instance if you have no further use for it. Select the temporary instance, and choose **Instance state**, **Terminate instance**.