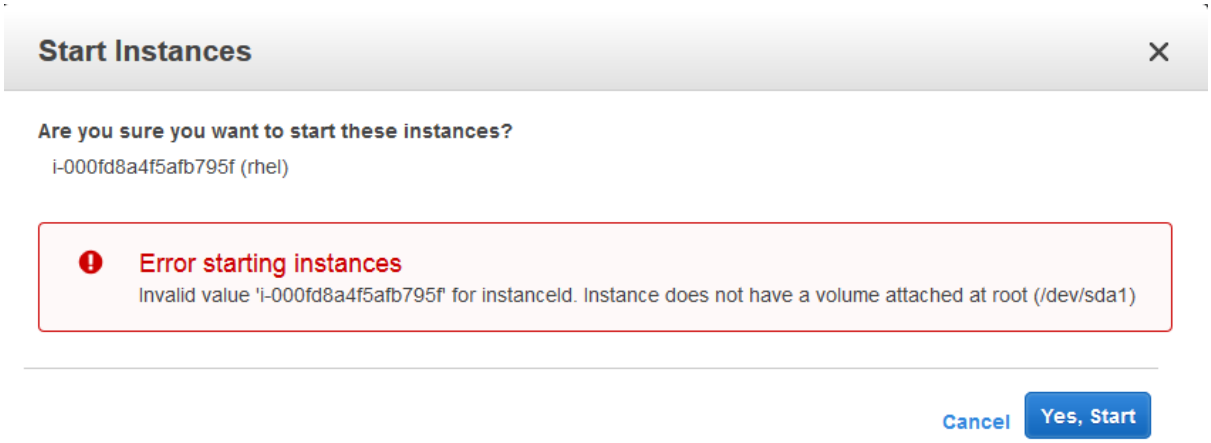
Troubleshoot instance launch issues

Error-1



Description

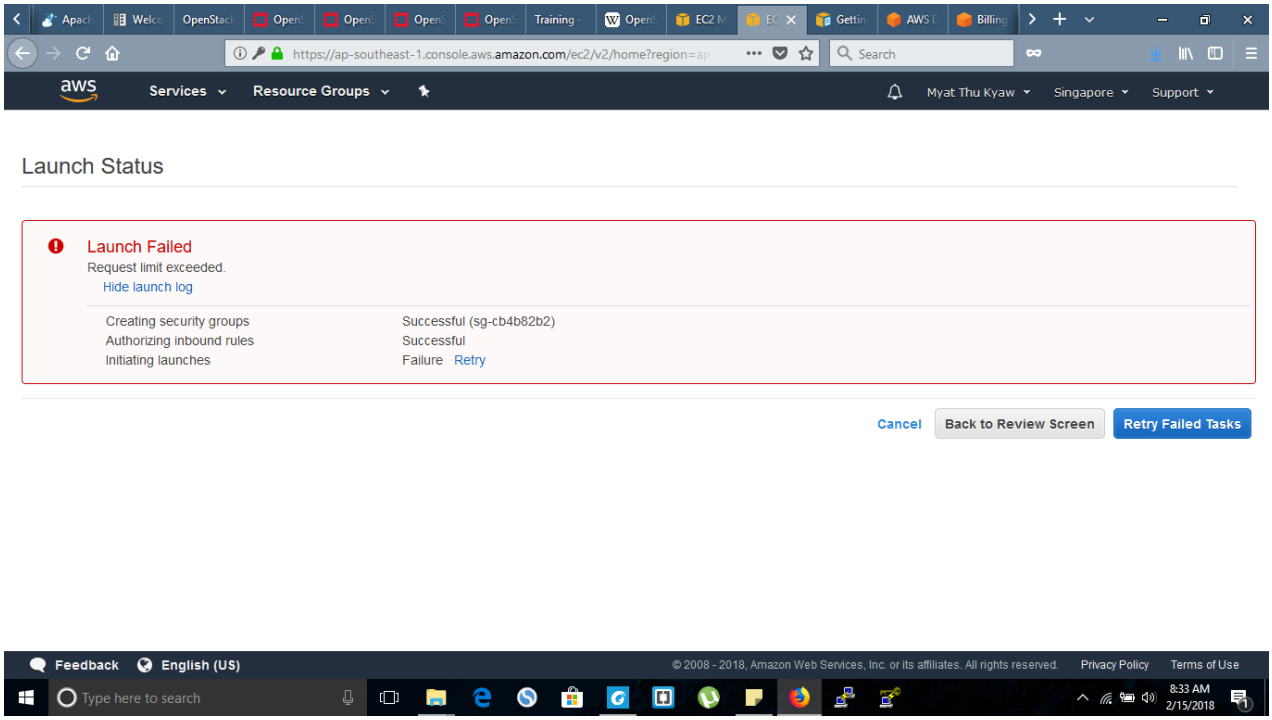
You get the `Invalid device name` *`device_name`* error when you try to launch a new instance.

Cause

If you get this error when you try to launch an instance, the device name specified for one or more volumes in the request has an invalid device name. Possible causes include:

- The device name might be in use by the selected AMI.
- The device name might be reserved for root volumes.
- The device name might be used for another volume in the request.
- The device name might not be valid for the operating system.

Error-2



Instance limit exceeded

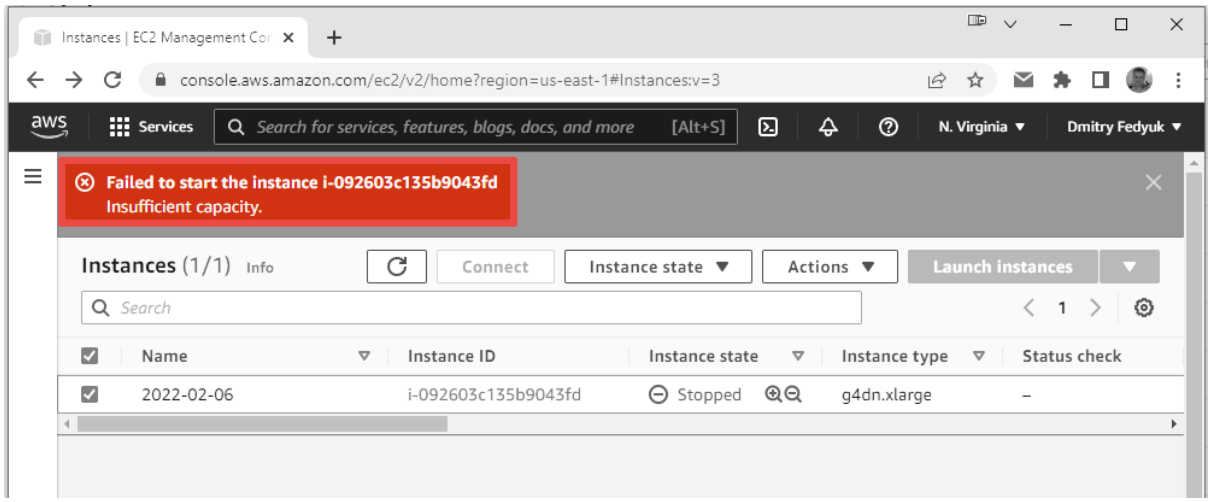
Description

You get the `InstanceLimitExceeded` error when you try to launch a new instance or restart a stopped instance.

Cause

If you get an `InstanceLimitExceeded` error when you try to launch a new instance or restart a stopped instance, you have reached the limit on the number of instances that you can launch in a Region. When you create your AWS account, we set default limits on the number of instances you can run on a per-Region basis.

Error-3



Insufficient instance capacity

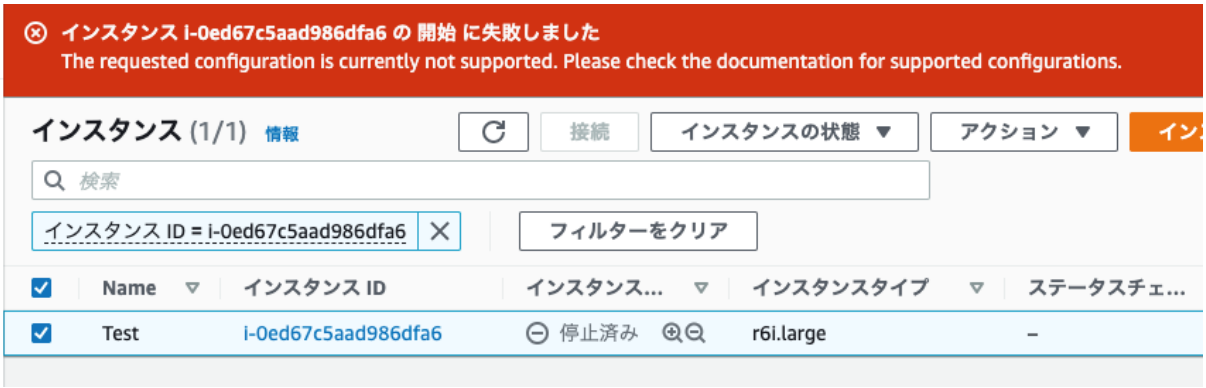
Description

You get the `InsufficientInstanceCapacity` error when you try to launch a new instance or restart a stopped instance.

Cause

If you get this error when you try to launch an instance or restart a stopped instance, AWS does not currently have enough available On-Demand capacity to fulfill your request.

Error-4



The requested configuration is currently not supported. Please check the documentation for supported configurations.

Description

You get the `Unsupported` error when you try to launch a new instance because the instance configuration is not supported.

Cause

The error message provides additional details. For example, an instance type or instance purchasing option might not be supported in the specified Region or Availability Zone.

Error-5

Instance terminates immediately

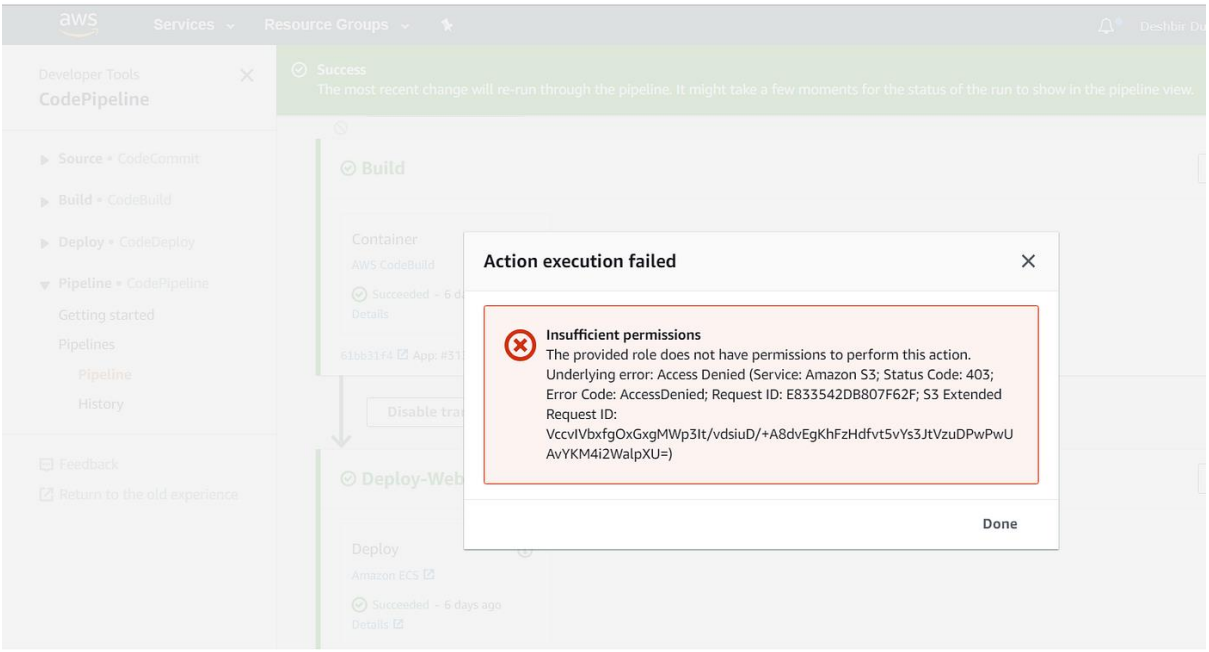
Description

Your instance goes from the `pending` state to the `terminated` state.

Cause

The following are a few reasons why an instance might immediately terminate:

- You've exceeded your EBS volume limits. For more information, see Instance volume limits.
- An EBS snapshot is corrupted.
- The root EBS volume is encrypted and you do not have permissions to access the KMS key for decryption.
- A snapshot specified in the block device mapping for the AMI is encrypted and you do not have permissions to access the KMS key for decryption or you do not have access to the KMS key to encrypt the restored volumes.
- The instance store-backed AMI that you used to launch the instance is missing a required part (an image.part.*xx* file).



Insufficient permissions

Description

You get the `"errorMessage": "You are not authorized to perform this operation."` error when you try to launch a new instance, and the launch fails.

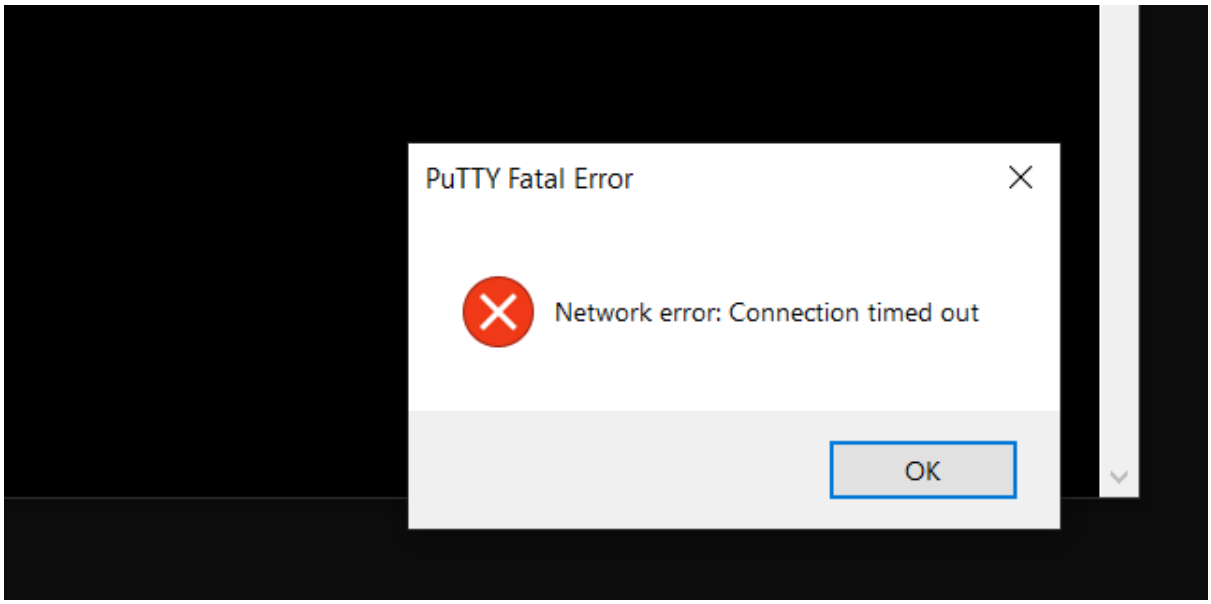
Cause

If you get this error when you try to launch an instance, you don't have the required IAM permissions to launch the instance.

Possible missing permissions include:

- `ec2:RunInstances`
- `iam:PassRole`

Error-7

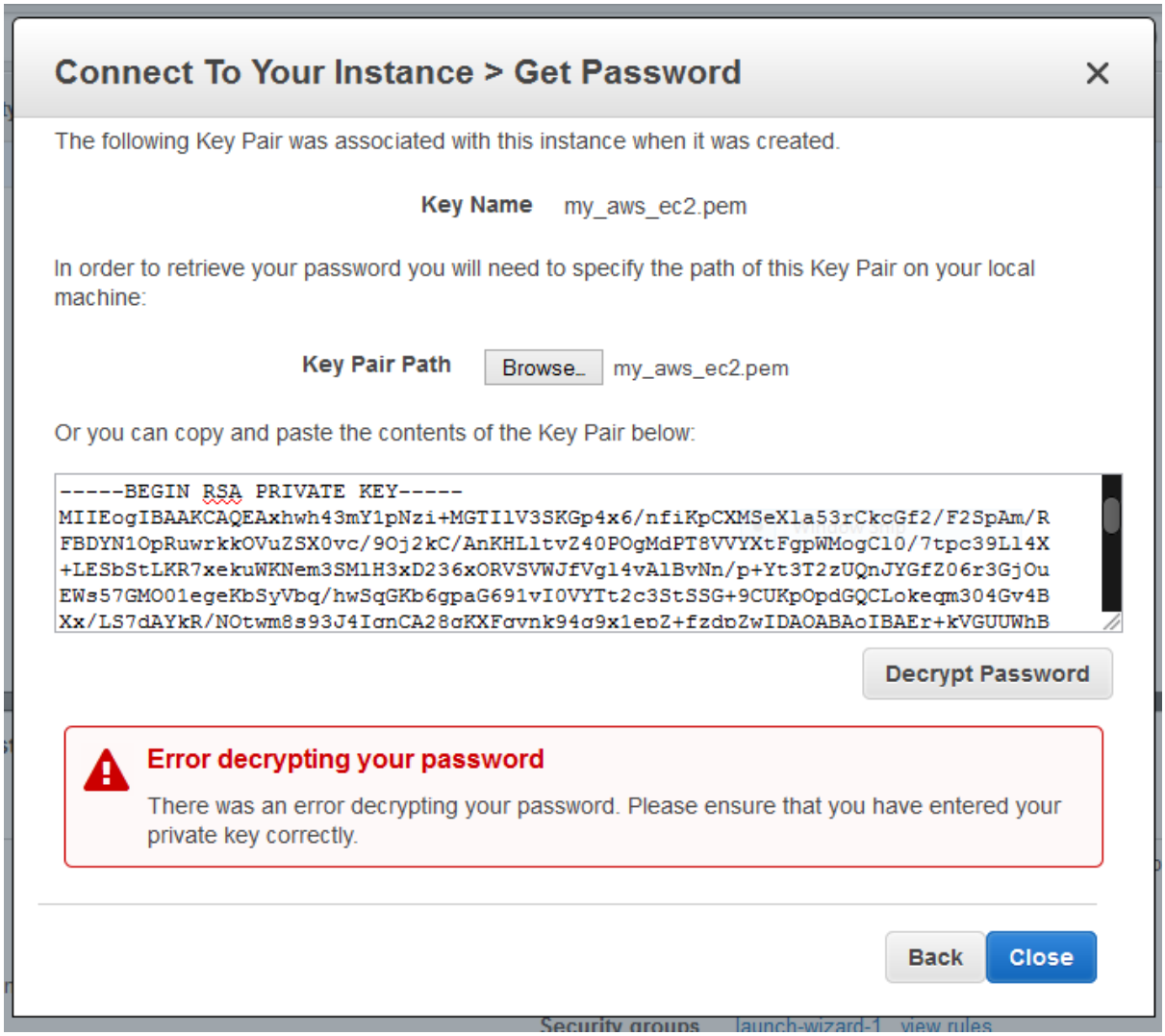


Error connecting to your instance: Connection timed out

Check

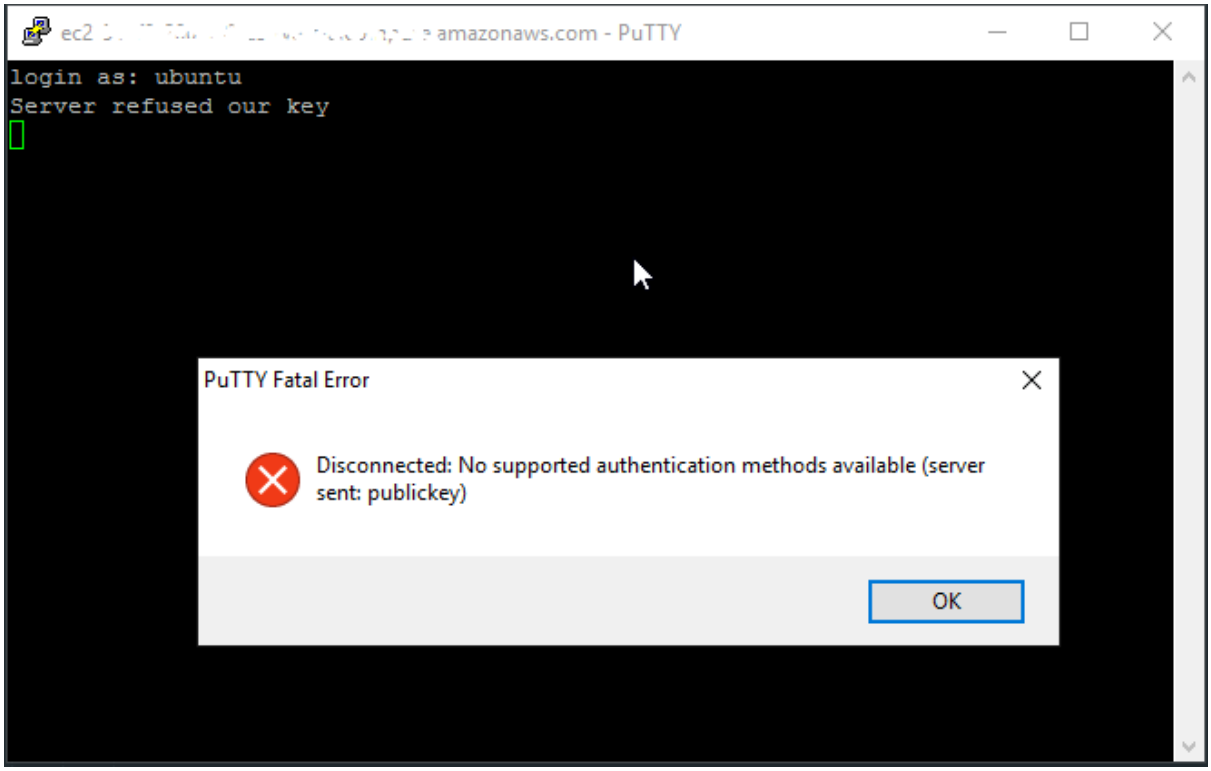
1) SecurityGroups
2) Check the route table for the subnet.
3) Check the network access control list (ACL) for the subnet.
4) If your computer is on a corporate network
5) Check that your instance has a public IPv4 address.
6) Check the CPU load on your instance; the server may be overloaded.

Error-8



If you try to connect to your instance and get the error message, `unable to load key ... Expecting: ANY PRIVATE KEY`, the file in which the private key is stored is incorrectly configured. If the private key file ends in `.pem`, it might still be incorrectly configured. A possible cause for an incorrectly configured private key file is a missing certificate.

Error-9



- Use `ssh -vvv` to get triple verbose debugging information while connecting:

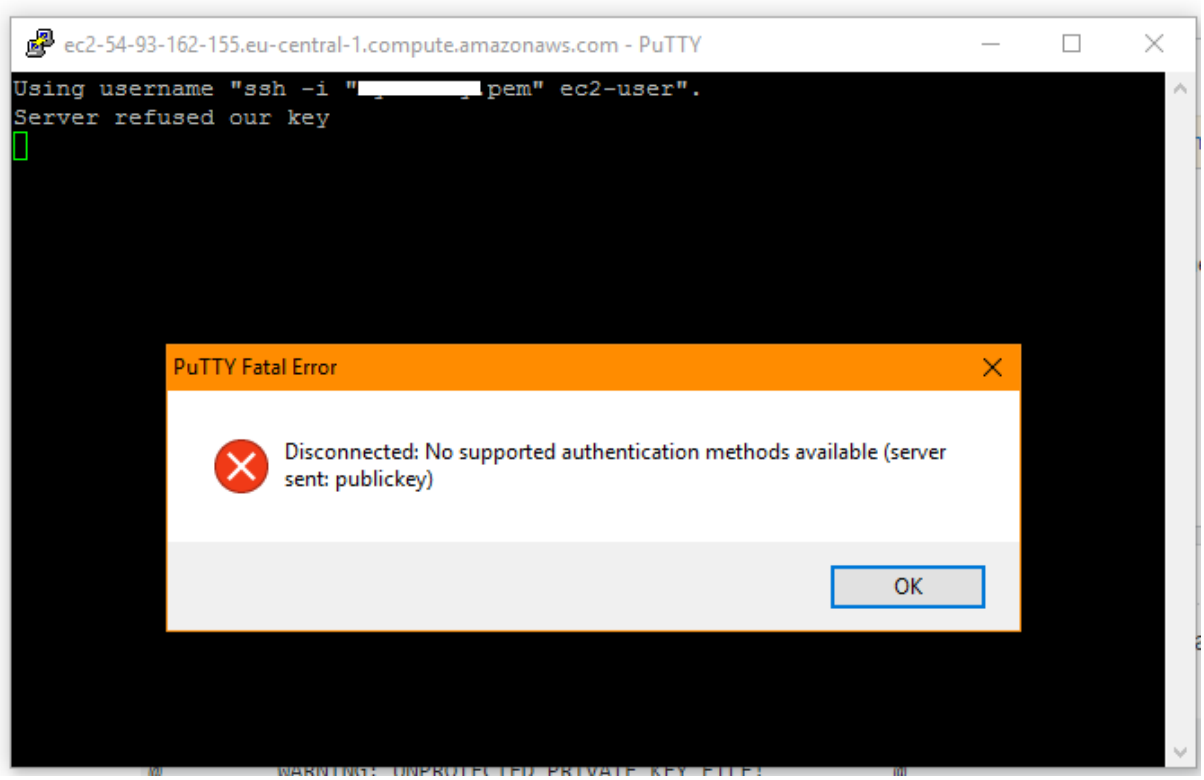
  ```
  ssh -vvv -i path/key-pair-name.pem instance-user-name@ec2-203-0-113-25.compute-1.amazonaws.com
  ```

  The following sample output demonstrates what you might see if you were trying to connect to your instance with a key that was not recognized by the server:

  ```
  open/ANT/myusername/.ssh/known_hosts).

  debug2: bits set: 504/1024

  debug1: ssh_rsa_verify: signature correct

  debug2: kex_derive_keys

  debug2: set_newkeys: mode 1

  debug1: SSH2_MSG_NEWKEYS sent

  debug1: expecting SSH2_MSG_NEWKEYS

  debug2: set_newkeys: mode 0
  ```

```
debug1: SSH2_MSG_NEWKEYS received

debug1: Roaming not allowed by server

debug1: SSH2_MSG_SERVICE_REQUEST sent

debug2: service_accept: ssh-userauth

debug1: SSH2_MSG_SERVICE_ACCEPT received

debug2: key: boguspem.pem ((nil))

debug1: Authentications that can continue: publickey

debug3: start over, passed a different list publickey

debug3: preferred gssapi-keyex,gssapi-with-
mic,publickey,keyboard-interactive,password

debug3: authmethod_lookup publickey

debug3: remaining preferred: keyboard-interactive,password

debug3: authmethod_is_enabled publickey

debug1: Next authentication method: publickey

debug1: Trying private key: boguspem.pem

debug1: read PEM private key done: type RSA

debug3: sign_and_send_pubkey: RSA
9c:4c:bc:0c:d0:5c:c7:92:6c:8e:9b:16:e4:43:d8:b2

debug2: we sent a publickey packet, wait for reply

debug1: Authentications that can continue: publickey

debug2: we did not send a packet, disable method

debug1: No more authentication methods to try.

Permission denied (publickey).
```

Error-10

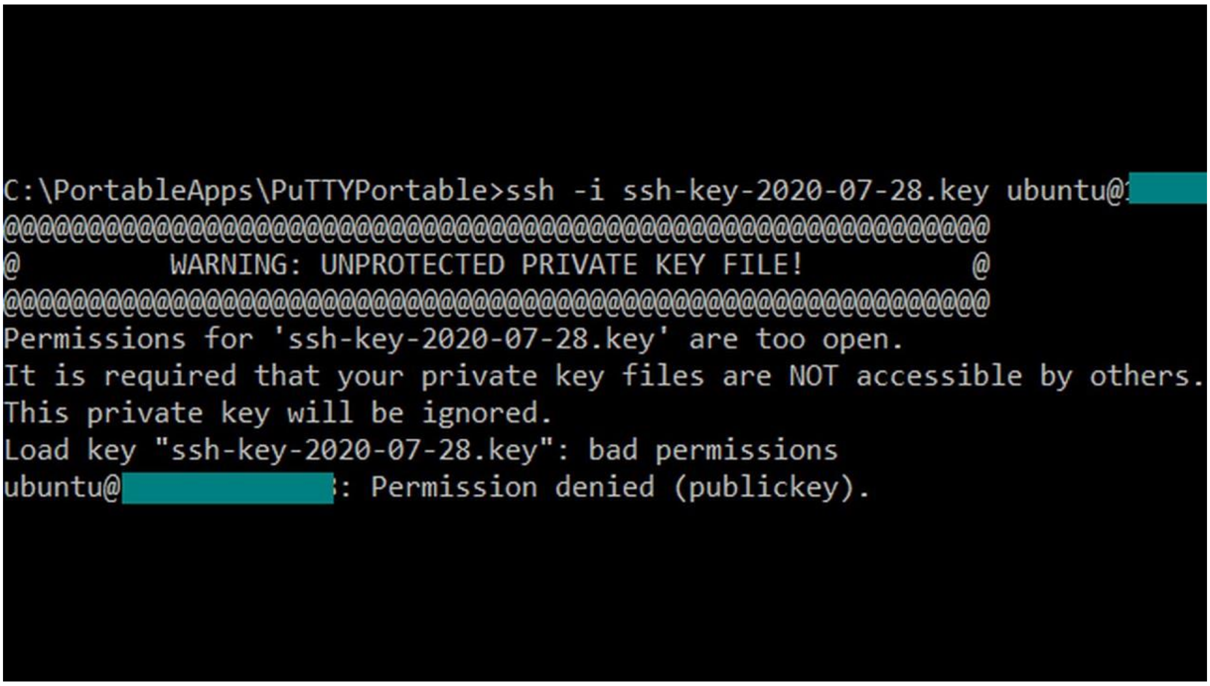


If you connect to your instance using SSH and get any of the following errors, `Host key not found in [directory]`, `Permission denied (publickey)`, `Authentication failed, permission denied`, or `Connection closed by [instance] port 22`, verify that you are connecting with the appropriate user name for your AMI *and* that you have specified the proper private key (`.pem`) file for your instance.

AMI used to launch instance	Default username
AL2023 Amazon Linux 2 Amazon Linux	`ec2-user`
CentOS	`centos` or `ec2-user`
Debian	`admin`
Fedora	`fedora` or `ec2-user`
RHEL	`ec2-user` or `root`
SUSE	`ec2-user` or `root`
Ubuntu	`ubuntu`

AMI used to launch instance	Default username
Oracle	`ec2-user`
Bitnami	`bitnami`
Rocky Linux	`rocky`
Other	Check with the AMI provider

Error-11



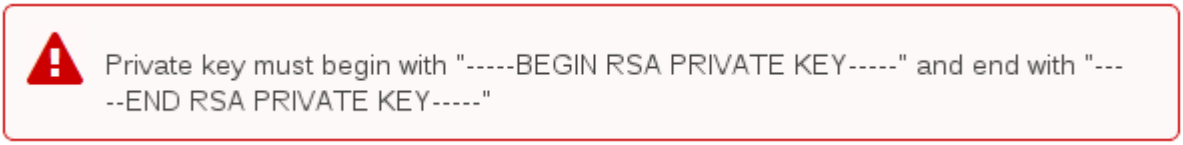
Your private key file must be protected from read and write operations from any other users. If your private key can be read or written to by anyone but you, then SSH ignores your key and you see the following warning message below.

```
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@

@         WARNING: UNPROTECTED PRIVATE KEY FILE!          @

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@

Permissions 0777 for '.ssh/my_private_key.pem' are too open.
```

```
It is required that your private key files are NOT accessible by
others.

This private key will be ignored.

bad permissions: ignore key: .ssh/my_private_key.pem

Permission denied (publickey).
```

If you see a similar message when you try to log in to your instance, examine the first line of the error message to verify that you are using the correct public key for your instance. The above example uses the private key `.ssh/my_private_key.pem` with file permissions of `0777`, which allow anyone to read or write to this file. This permission level is very insecure, and so SSH ignores this key.

Error-12

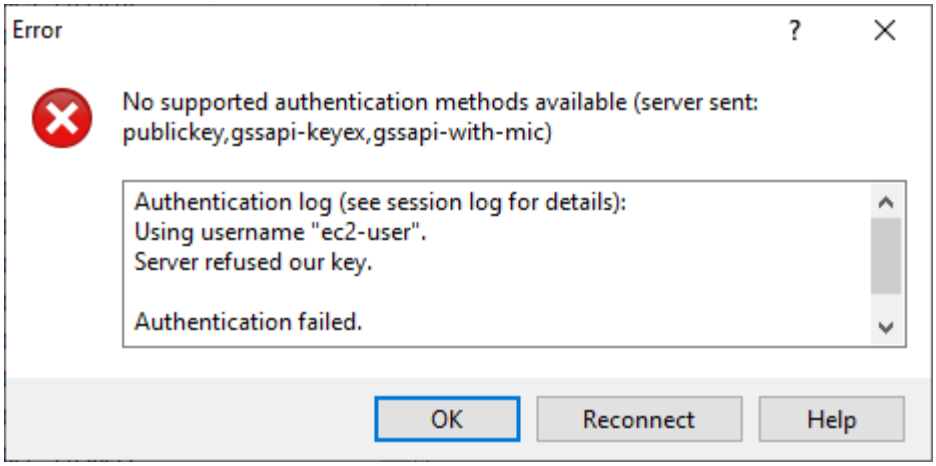


If you use a third-party tool, such as **ssh-keygen**, to create an RSA key pair, it generates the private key in the OpenSSH key format. When you connect to your instance, if you use the private key in the OpenSSH format to decrypt the password, you'll get the error `Private key must begin with "-----BEGIN RSA PRIVATE KEY-----" and end with "-----END RSA PRIVATE KEY-----"`.

To resolve the error, the private key must be in the PEM format. Use the following command to create the private key in the PEM format:

```
ssh-keygen -m PEM
```

Error-13



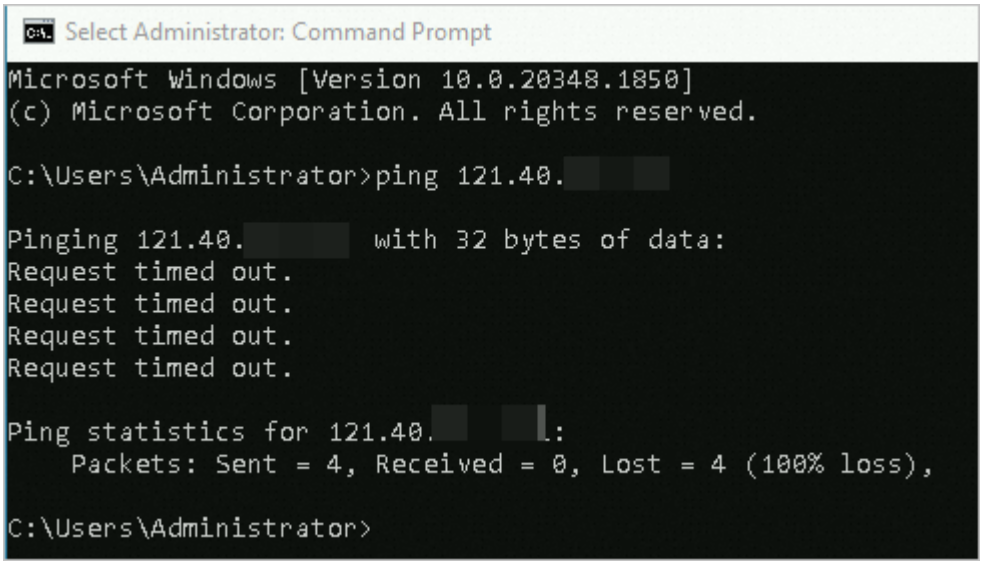
If you use PuTTY to connect to your instance and get either of the following errors, `Error: Server refused our key` or `Error: No supported authentication methods available`, verify that you are connecting with the appropriate user name for your AMI. Type the user name in **User name** in the **PuTTY Configuration** window.

The appropriate user names are as follows:

AMI used to launch instance	Default username
AL2023 Amazon Linux 2 Amazon Linux	`ec2-user`
CentOS	`centos` or `ec2-user`
Debian	`admin`
Fedora	`fedora` or `ec2-user`
RHEL	`ec2-user` or `root`
SUSE	`ec2-user` or `root`
Ubuntu	`ubuntu`
Oracle	`ec2-user`
Bitnami	`bitnami`
Rocky Linux	`rocky`

AMI used to launch instance	Default username
Other	Check with the AMI provider

Error-14

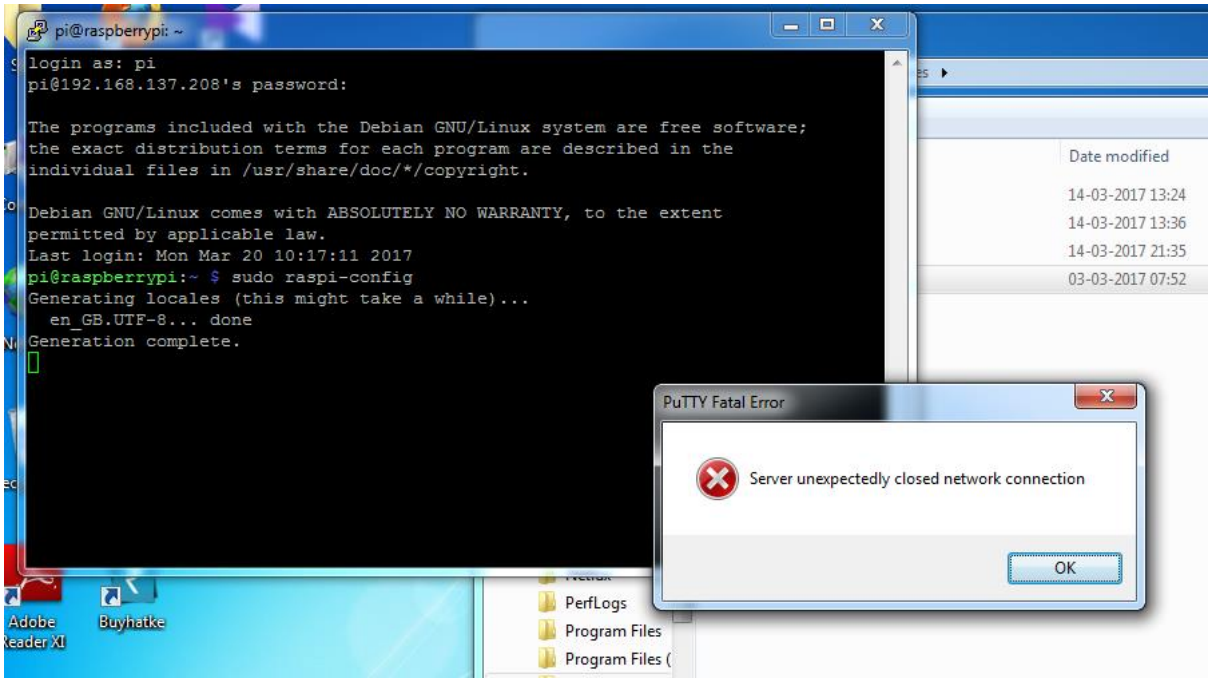


The `ping` command is a type of ICMP traffic — if you are unable to ping your instance, ensure that your inbound security group rules allow ICMP traffic for the `Echo Request` message from all sources, or from the computer or instance from which you are issuing the command.

If you are unable to issue a `ping` command from your instance, ensure that your outbound security group rules allow ICMP traffic for the `Echo Request` message to all destinations, or to the host that you are attempting to ping.

`Ping` commands can also be blocked by a firewall or time out due to network latency or hardware issues. You should consult your local network or system administrator for help with further troubleshooting.

Error-15



Error: Server unexpectedly closed network connection

If you are connecting to your instance with PuTTY and you receive the error "Server unexpectedly closed network connection," verify that you have enabled keepalives on the Connection page of the PuTTY Configuration to avoid being disconnected. Some servers disconnect clients when they do not receive any data within a specified period of time. Set the Seconds between keepalives to 59 seconds.

If you still experience issues after enabling keepalives, try to disable Nagle's algorithm on the Connection page of the PuTTY Configuration.

Error-16

```
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@    WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!     @
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
IT IS POSSIBLE THAT SOMEONE IS DOING SOMETHING NASTY!
Someone could be eavesdropping on you right now (man-in-the-middle attack)!
It is also possible that a host key has just been changed.
The fingerprint for the ECDSA key sent by the remote host is
SHA256:uQa5wz5y7xR19yGt7xqzXQ.
Please contact your system administrator.
Add correct host key in /home/user/.ssh/known_hosts to get rid of this message.
Offending ECDSA key in /home/user/.ssh/known_hosts:3
ECDSA host key for 192.168.1.100 has changed and you have requested strict checking.
Host key verification failed.
```

If you rotate your instance host keys, the new host keys are not automatically uploaded to the AWS trusted host keys database. This causes host key validation to fail when you try to connect to your instance using the EC2 Instance Connect browser-based client, and you're unable to connect to your instance.

To resolve the error, you must run the `eic_harvest_hostkeys` script on your instance, which uploads your new host key to EC2 Instance Connect. The script is located at `/opt/aws/bin/` on Amazon Linux 2 instances, and at `/usr/share/ec2-instance-connect/` on Ubuntu instances.

Error-17

EC2 Instance Connect | Session Manager | **SSH client** | EC2 Serial Console

No associated key pair
This instance is not associated with a key pair. Without a key pair, you can't connect to the instance through SSH.

You can connect using EC2 Instance Connect with just a valid username. You can connect using Session Manager if you have been granted the necessary permissions.

Can't connect to Ubuntu instance using EC2 Instance Connect

If you use EC2 Instance Connect to connect to your Ubuntu instance and you get an error when attempting to connect, you can use the following information to try to fix the issue.

Possible cause

The `ec2-instance-connect` package on the instance is not the latest version.

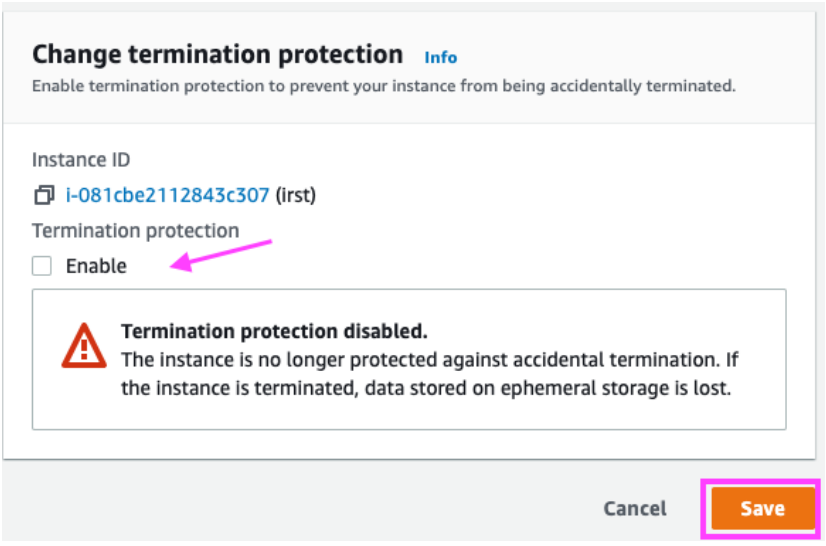
Solution

Update the `ec2-instance-connect` package on the instance to the latest version, as follows:

1. Connect to your instance using a method other than EC2 Instance Connect.
2. Run the following command on your instance to update the `ec2-instance-connect` package to the latest version.

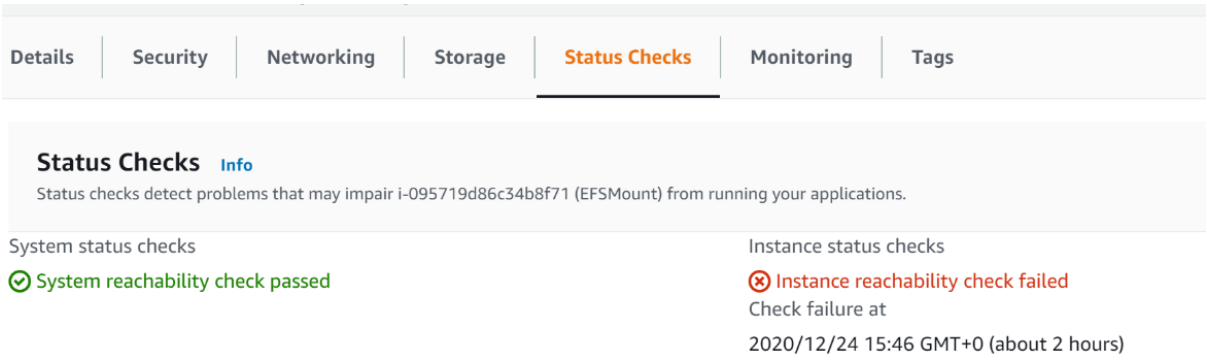
```
apt update && apt upgrade
```

Error-18



If you try to terminate an instance and get the `The instance instance_id may not be terminated. Modify its 'disableApiTermination' instance attribute` error message, it indicates that the instance has been enabled for termination protection. Termination protection prevents the instance from being accidentally terminated.

Error-19



If a system status check has failed, you can try one of the following options:

- Create an instance recovery alarm. For more information, see Create alarms that stop, terminate, reboot, or recover an instance.
- If you changed the instance type to an instance built on the AWS Nitro System, status checks fail if you migrated from an instance that does not have the required ENA and NVMe drivers. For more information, see Compatibility for changing the instance type.
- For an instance using an Amazon EBS-backed AMI, stop and restart the instance.
- For an instance using an instance-store backed AMI, terminate the instance and launch a replacement.
- Wait for Amazon EC2 to resolve the issue.

Troubleshoot system log errors for Linux-based instances

For Linux-based instances that have failed an instance status check, such as the instance reachability check, verify that you followed the steps above to retrieve the system log. The following list contains some common system log errors and suggested actions you can take to resolve the issue for each error.

Memory Errors

- Out of memory: kill process
- ERROR: mmu_update failed (Memory management update failed)

Out of memory: kill process

An out-of-memory error is indicated by a system log entry similar to the one shown below.

```
[115879.769795] Out of memory: kill process 20273 (httpd) score
1285879

or a child

[115879.769795] Killed process 1917 (php-cgi) vsz:467184kB, anon-

rss:101196kB, file-rss:204kB
```

Potential cause

Exhausted memory

Suggested actions

For this instance type	Do this
Amazon EBS-backed	Do one of the following: • Stop the instance, and modify the instance to use a different instance type, and start the instance again. For example, a larger or a memory-optimized instance type. • Reboot the instance to return it to a non-impaired status. The problem will probably occur again unless you change the instance type.
Instance store-backed	Do one of the following: • Terminate the instance and launch a new instance, specifying a different instance type. For example, a larger or a memory-optimized instance type.

For this instance type	Do this
	• Reboot the instance to return it to an unimpaired status. The problem will probably occur again unless you change the instance type.

ERROR: mmu_update failed (Memory management update failed)

Memory management update failures are indicated by a system log entry similar to the following:

```
...

Press `ESC' to enter the menu... 0   [H[J  Booting 'Amazon Linux
2011.09 (2.6.35.14-95.38.amzn1.i686)'




root (hd0)


 Filesystem type is ext2fs, using whole disk


kernel /boot/vmlinuz-2.6.35.14-95.38.amzn1.i686 root=LABEL=/
console=hvc0 LANG=


en_US.UTF-8 KEYTABLE=us


initrd /boot/initramfs-2.6.35.14-95.38.amzn1.i686.img
```

```
ERROR: mmu_update failed with rc=-22
```

Potential cause

Issue with Amazon Linux

Device Errors

- I/O error (block device failure)
- I/O ERROR: neither local nor remote disk (Broken distributed block device)

I/O error (block device failure)

An input/output error is indicated by a system log entry similar to the following example:

```
[9943662.053217] end_request: I/O error, dev sde, sector 52428288

[9943664.191262] end_request: I/O error, dev sde, sector 52428168

[9943664.191285] Buffer I/O error on device md0, logical block
209713024

[9943664.191297] Buffer I/O error on device md0, logical block
209713025

[9943664.191304] Buffer I/O error on device md0, logical block
209713026

[9943664.191310] Buffer I/O error on device md0, logical block
209713027

[9943664.191317] Buffer I/O error on device md0, logical block
209713028

[9943664.191324] Buffer I/O error on device md0, logical block
209713029
```

```
[9943664.191332] Buffer I/O error on device md0, logical block
209713030

[9943664.191339] Buffer I/O error on device md0, logical block
209713031

[9943664.191581] end_request: I/O error, dev sde, sector 52428280

[9943664.191590] Buffer I/O error on device md0, logical block
209713136

[9943664.191597] Buffer I/O error on device md0, logical block
209713137

[9943664.191767] end_request: I/O error, dev sde, sector 52428288

[9943664.191970] end_request: I/O error, dev sde, sector 52428288

[9943664.192143] end_request: I/O error, dev sde, sector 52428288

[9943664.192949] end_request: I/O error, dev sde, sector 52428288

[9943664.193112] end_request: I/O error, dev sde, sector 52428288

[9943664.193266] end_request: I/O error, dev sde, sector 52428288

...
```

Potential causes

Instance type	Potential cause
Amazon EBS-backed	A failed Amazon EBS volume
Instance store-backed	A failed physical drive

Suggested actions

For this instance type	Do this
Amazon EBS-backed	Use the following procedure: 1. Stop the instance. 2. Detach the volume. 3. Attempt to recover the volume. Note It's good practice to snapshot your Amazon EBS volumes often. This dramatically decreases the risk of data loss as a result of failure. 4. Re-attach the volume to the instance. 5. Start the instance.
Instance store-backed	Terminate the instance and launch a new instance. Note Data cannot be recovered. Recover from backups. Note It's a good practice to use either Amazon S3 or Amazon EBS for backups. Instance store volumes are directly tied to single host and single disk failures.

I/O ERROR: neither local nor remote disk (Broken distributed block device)

An input/output error on the device is indicated by a system log entry similar to the following example:

```
...

block drbd1: Local IO failed in request_timer_fn. Detaching...



Aborting journal on device drbd1-8.
```

```
block drbd1: IO ERROR: neither local nor remote disk

Buffer I/O error on device drbd1, logical block 557056

lost page write due to I/O error on drbd1

JBD2: I/O error detected when updating journal superblock for drbd1-8.
```

Potential causes

Instance type	Potential cause
Amazon EBS-backed	A failed Amazon EBS volume
Instance store-backed	A failed physical drive

Suggested action

Terminate the instance and launch a new instance.

For an Amazon EBS-backed instance you can recover data from a recent snapshot by creating an image from it. Any data added after the snapshot cannot be recovered.

Kernel Errors

- request_module: runaway loop modprobe (Looping legacy kernel modprobe on older Linux versions)
- "FATAL: kernel too old" and "fsck: No such file or directory while trying to open /dev" (Kernel and AMI mismatch)
- "FATAL: Could not load /lib/modules" or "BusyBox" (Missing kernel modules)
- ERROR Invalid kernel (EC2 incompatible kernel)

request_module: runaway loop modprobe (Looping legacy kernel modprobe on older Linux versions)

This condition is indicated by a system log similar to the one shown below. Using an unstable or old Linux kernel (for example, 2.6.16-xenU) can cause an interminable loop condition at startup.

```
Linux version 2.6.16-xenU (builder@xenbat.amazonsa) (gcc version
4.0.1

20050727 (Red Hat 4.0.1-5)) #1 SMP Mon May 28 03:41:49 SAST 2007


BIOS-provided physical RAM map:


 Xen: 0000000000000000 - 0000000026700000 (usable)


0MB HIGHMEM available.

...


request_module: runaway loop modprobe binfmt-464c


request_module: runaway loop modprobe binfmt-464c


request_module: runaway loop modprobe binfmt-464c


request_module: runaway loop modprobe binfmt-464c
```

```
request_module: runaway loop modprobe binfmt-464c
```

Suggested actions

For this instance type	Do this
Amazon EBS-backed	Use a newer kernel, either GRUB-based or static, using one of the following options: Option 1: Terminate the instance and launch a new instance, specifying the `-kernel` and `-ramdisk` parameters. Option 2: 1. Stop the instance. 2. Modify the kernel and ramdisk attributes to use a newer kernel. 3. Start the instance.
Instance store-backed	Terminate the instance and launch a new instance, specifying the `-kernel` and `-ramdisk` parameters.

"FATAL: kernel too old" and "fsck: No such file or directory while trying to open /dev" (Kernel and AMI mismatch)

This condition is indicated by a system log similar to the one shown below.

```
Linux version 2.6.16.33-xenU (root@dom0-0-50-45-1-a4-ee.z-
2.aes0.internal)

(gcc version 4.1.1 20070105 (Red Hat 4.1.1-52)) #2 SMP Wed Aug 15
17:27:36 SAST 2007

...

FATAL: kernel too old

Kernel panic - not syncing: Attempted to kill init!
```

Potential causes

Incompatible kernel and userland

Suggested actions

For this instance type	Do this
Amazon EBS-backed	Use the following procedure: 1. Stop the instance. 2. Modify the configuration to use a newer kernel. 3. Start the instance.
Instance store-backed	Use the following procedure: 1. Create an AMI that uses a newer kernel. 2. Terminate the instance. 3. Start a new instance from the AMI you created.

"FATAL: Could not load /lib/modules" or "BusyBox" (Missing kernel modules)

This condition is indicated by a system log similar to the one shown below.

```
[    0.370415] Freeing unused kernel memory: 1716k freed

Loading, please wait...

WARNING: Couldn't open directory /lib/modules/2.6.34-4-virtual: No
such file or directory

FATAL: Could not open /lib/modules/2.6.34-4-virtual/modules.dep.temp
for writing: No such file or directory

FATAL: Could not load /lib/modules/2.6.34-4-virtual/modules.dep: No
such file or directory

Couldn't get a file descriptor referring to the console

Begin: Loading essential drivers... ...
```

```
FATAL: Could not load /lib/modules/2.6.34-4-virtual/modules.dep: No
such file or directory

FATAL: Could not load /lib/modules/2.6.34-4-virtual/modules.dep: No
such file or directory

Done.

Begin: Running /scripts/init-premount ...

Done.

Begin: Mounting root file system... ...

Begin: Running /scripts/local-top ...

Done.

Begin: Waiting for root file system... ...

Done.

Gave up waiting for root device.  Common problems:

 - Boot args (cat /proc/cmdline)

   - Check rootdelay= (did the system wait long enough?)

   - Check root= (did the system wait for the right device?)

 - Missing modules (cat /proc/modules; ls /dev)

FATAL: Could not load /lib/modules/2.6.34-4-virtual/modules.dep: No
such file or directory

FATAL: Could not load /lib/modules/2.6.34-4-virtual/modules.dep: No
such file or directory

ALERT! /dev/sda1 does not exist. Dropping to a shell!




BusyBox v1.13.3 (Ubuntu 1:1.13.3-1ubuntu5) built-in shell (ash)
```

```
Enter 'help' for a list of built-in commands.



(initramfs)
```

Potential causes

One or more of the following conditions can cause this problem:

- Missing ramdisk
- Missing correct modules from ramdisk
- Amazon EBS root volume not correctly attached as `/dev/sda1`

Suggested actions

For this instance type	**Do this**
Amazon EBS-backed	Use the following procedure: 1. Select corrected ramdisk for the Amazon EBS volume. 2. Stop the instance. 3. Detach the volume and repair it. 4. Attach the volume to the instance. 5. Start the instance. 6. Modify the AMI to use the corrected ramdisk.
Instance store-backed	Use the following procedure: 1. Terminate the instance and launch a new instance with the correct ramdisk. 2. Create a new AMI with the correct ramdisk.

ERROR Invalid kernel (EC2 incompatible kernel)

This condition is indicated by a system log similar to the one shown below.

```
...
```

```
root (hd0)

 Filesystem type is ext2fs, using whole disk

kernel /vmlinuz root=/dev/sda1 ro

initrd /initrd.img

ERROR Invalid kernel: elf_xen_note_check: ERROR: Will only load
images

built for the generic loader or Linux images

xc_dom_parse_image returned -1

Error 9: Unknown boot failure

  Booting 'Fallback'

root (hd0)

 Filesystem type is ext2fs, using whole disk

kernel /vmlinuz.old root=/dev/sda1 ro

Error 15: File not found
```

Potential causes

One or both of the following conditions can cause this problem:

- Supplied kernel is not supported by GRUB
- Fallback kernel does not exist

Suggested actions

For this instance type	Do this
Amazon EBS-backed	Use the following procedure: 1. Stop the instance. 2. Replace with working kernel. 3. Install a fallback kernel. 4. Modify the AMI by correcting the kernel.
Instance store-backed	Use the following procedure: 1. Terminate the instance and launch a new instance with the correct kernel. 2. Create an AMI with the correct kernel. 3. (Optional) Seek technical assistance for data recovery using AWS Support.

File System Errors

- fsck: No such file or directory while trying to open... (File system not found)
- General error mounting filesystems (failed mount)
- VFS: Unable to mount root fs on unknown-block (Root filesystem mismatch)
- Error: Unable to determine major/minor number of root device... (Root file system/device mismatch)
- XENBUS: Device with no driver...
- ... days without being checked, check forced (File system check required)
- fsck died with exit status... (Missing device)

fsck: No such file or directory while trying to open... (File system not found)

This condition is indicated by a system log similar to the one shown below.

```
               Welcome to Fedora

               Press 'I' to enter interactive startup.

Setting clock : Wed Oct 26 05:52:05 EDT 2011 [  OK  ]


Starting udev: [  OK  ]


Setting hostname localhost:  [  OK  ]


No devices found

Setting up Logical Volume Management: File descriptor 7 left open

  No volume groups found

[  OK  ]


Checking filesystems

Checking all file systems.

[/sbin/fsck.ext3 (1) -- /] fsck.ext3 -a /dev/sda1

/dev/sda1: clean, 82081/1310720 files, 2141116/2621440 blocks

[/sbin/fsck.ext3 (1) -- /mnt/dbbackups] fsck.ext3 -a /dev/sdh

fsck.ext3: No such file or directory while trying to open /dev/sdh
```

```
/dev/sdh:

The superblock could not be read or does not describe a correct ext2

filesystem.  If the device is valid and it really contains an ext2

filesystem (and not swap or ufs or something else), then the superblock

is corrupt, and you might try running e2fsck with an alternate superblock:

    e2fsck -b 8193 <device>



[FAILED]




*** An error occurred during the file system check.

*** Dropping you to a shell; the system will reboot

*** when you leave the shell.

Give root password for maintenance

(or type Control-D to continue):
```

Potential causes

- A bug exists in ramdisk filesystem definitions /etc/fstab
- Misconfigured filesystem definitions in /etc/fstab
- Missing/failed drive

Suggested actions

For this instance type	Do this
Amazon EBS-backed	Use the following procedure: 1. Stop the instance, detach the root volume, repair/modify /etc/fstab the volume, attach the volume to the instance, and start the instance. 2. Fix ramdisk to include modified /etc/fstab (if applicable). 3. Modify the AMI to use a newer ramdisk. The sixth field in the fstab defines availability requirements of the mount – a nonzero value implies that an fsck will be done on that volume and *must* succeed. Using this field can be problematic in Amazon EC2 because a failure typically results in an interactive console prompt that is not currently available in Amazon EC2. Use care with this feature and read the Linux man page for fstab.
Instance store-backed	Use the following procedure: 1. Terminate the instance and launch a new instance. 2. Detach any errant Amazon EBS volumes and the reboot instance. 3. (Optional) Seek technical assistance for data recovery using AWS Support.

General error mounting filesystems (failed mount)

This condition is indicated by a system log similar to the one shown below.

```
Loading xenblk.ko module

xen-vbd: registered block device major 8



Loading ehci-hcd.ko module
```

```
Loading ohci-hcd.ko module

Loading uhci-hcd.ko module

USB Universal Host Controller Interface driver v3.0


Loading mbcache.ko module

Loading jbd.ko module

Loading ext3.ko module

Creating root device.

Mounting root filesystem.

kjournald starting.  Commit interval 5 seconds


EXT3-fs: mounted filesystem with ordered data mode.


Setting up other filesystems.

Setting up new root fs

no fstab.sys, mounting internal defaults

Switching to new root and running init.

unmounting old /dev

unmounting old /proc

unmounting old /sys

mountall:/proc: unable to mount: Device or resource busy

mountall:/proc/self/mountinfo: No such file or directory

mountall: root filesystem isn't mounted
```

```
init: mountall main process (221) terminated with status 1


General error mounting filesystems.

A maintenance shell will now be started.

CONTROL-D will terminate this shell and re-try.

Press enter for maintenance

(or type Control-D to continue):
```

Potential causes

Instance type	Potential cause
Amazon EBS-backed	• Detached or failed Amazon EBS volume. • Corrupted filesystem. • Mismatched ramdisk and AMI combination (such as Debian ramdisk with a SUSE AMI).
Instance store-backed	• A failed drive. • A corrupted file system. • A mismatched ramdisk and combination (for example, a Debian ramdisk with a SUSE AMI).

Suggested actions

For this instance type	Do this
Amazon EBS-backed	Use the following procedure: 1. Stop the instance. 2. Detach the root volume. 3. Attach the root volume to a known working instance. 4. Run filesystem check (fsck -a /dev/...).

For this instance type	Do this
	5. Fix any errors. 6. Detach the volume from the known working instance. 7. Attach the volume to the stopped instance. 8. Start the instance. 9. Recheck the instance status.
Instance store-backed	Try one of the following: • Start a new instance. • (Optional) Seek technical assistance for data recovery using AWS Support.

VFS: Unable to mount root fs on unknown-block (Root filesystem mismatch)

This condition is indicated by a system log similar to the one shown below.

```
Linux version 2.6.16-xenU (builder@xenbat.amazonsa) (gcc version
4.0.1

 20050727 (Red Hat 4.0.1-5)) #1 SMP Mon May 28 03:41:49 SAST 2007

...

Kernel command line:  root=/dev/sda1 ro 4

...

Registering block device major 8

...

Kernel panic - not syncing: VFS: Unable to mount root fs on unknown-
block(8,1)
```

Potential causes

Instance type	Potential cause
Amazon EBS-backed	• Device not attached correctly. • Root device not attached at correct device point. • Filesystem not in expected format. • Use of legacy kernel (such as 2.6.16-XenU). • A recent kernel update on your instance (faulty update, or an update bug)
Instance store-backed	Hardware device failure.

Suggested actions

For this instance type	Do this
Amazon EBS-backed	Do one of the following: • Stop and then restart the instance. • Modify root volume to attach at the correct device point, possible /dev/sda1 instead of /dev/sda. • Stop and modify to use modern kernel. • Refer to the documentation for your Linux distribution to check for known update bugs. Change or reinstall the kernel.
Instance store-backed	Terminate the instance and launch a new instance using a modern kernel.

Error: Unable to determine major/minor number of root device... (Root file system/device mismatch)

This condition is indicated by a system log similar to the one shown below.

```
. . .
```

```
XENBUS: Device with no driver: device/vif/0

XENBUS: Device with no driver: device/vbd/2048

drivers/rtc/hctosys.c: unable to open rtc device (rtc0)

Initializing network drop monitor service

Freeing unused kernel memory: 508k freed

:: Starting udevd...

done.

:: Running Hook [udev]

:: Triggering uevents...<30>udevd[65]: starting version 173

done.

Waiting 10 seconds for device /dev/xvda1 ...

Root device '/dev/xvda1' doesn't exist. Attempting to create it.

ERROR: Unable to determine major/minor number of root device
'/dev/xvda1'.

You are being dropped to a recovery shell

    Type 'exit' to try and continue booting

sh: can't access tty; job control turned off

[ramfs /]#
```

Potential causes

- Missing or incorrectly configured virtual block device driver
- Device enumeration clash (sda versus xvda or sda instead of sda1)
- Incorrect choice of instance kernel

Suggested actions

For this instance type	Do this
Amazon EBS-backed	Use the following procedure: 1. Stop the instance. 2. Detach the volume. 3. Fix the device mapping problem. 4. Start the instance. 5. Modify the AMI to address device mapping issues.
Instance store-backed	Use the following procedure: 1. Create a new AMI with the appropriate fix (map block device correctly). 2. Terminate the instance and launch a new instance from the AMI you created.

XENBUS: Device with no driver...

This condition is indicated by a system log similar to the one shown below.

```
XENBUS: Device with no driver: device/vbd/2048

drivers/rtc/hctosys.c: unable to open rtc device (rtc0)

Initializing network drop monitor service

Freeing unused kernel memory: 508k freed

:: Starting udevd...

done.

:: Running Hook [udev]

:: Triggering uevents...<30>udevd[65]: starting version 173

done.
```

```
Waiting 10 seconds for device /dev/xvda1 ...

Root device '/dev/xvda1' doesn't exist. Attempting to create it.

ERROR: Unable to determine major/minor number of root device
'/dev/xvda1'.

You are being dropped to a recovery shell

    Type 'exit' to try and continue booting

sh: can't access tty; job control turned off

[ramfs /]#
```

Potential causes

- Missing or incorrectly configured virtual block device driver
- Device enumeration clash (sda versus xvda)
- Incorrect choice of instance kernel

Suggested actions

For this instance type	Do this
Amazon EBS-backed	Use the following procedure: 1. Stop the instance. 2. Detach the volume. 3. Fix the device mapping problem. 4. Start the instance. 5. Modify the AMI to address device mapping issues.
Instance store-backed	Use the following procedure: 1. Create an AMI with the appropriate fix (map block device correctly). 2. Terminate the instance and launch a new instance using the AMI you created.

... days without being checked, check forced (File system check required)

This condition is indicated by a system log similar to the one shown below.

```
...

Checking filesystems

Checking all file systems.

[/sbin/fsck.ext3 (1) -- /] fsck.ext3 -a /dev/sda1

/dev/sda1 has gone 361 days without being checked, check forced
```

Potential causes

Filesystem check time passed; a filesystem check is being forced.

Suggested actions

- Wait until the filesystem check completes. A filesystem check can take a long time depending on the size of the root filesystem.
- Modify your filesystems to remove the filesystem check (fsck) enforcement using tune2fs or tools appropriate for your filesystem.

---

fsck died with exit status... (Missing device)

This condition is indicated by a system log similar to the one shown below.

```
Cleaning up ifupdown....

Loading kernel modules...done.

...

Activating lvm and md swap...done.

Checking file systems...fsck from util-linux-ng 2.16.2

/sbin/fsck.xfs: /dev/sdh does not exist
```

```
fsck died with exit status 8

[31mfailed (code 8).[39;49m
```

Potential causes

- Ramdisk looking for missing drive
- Filesystem consistency check forced
- Drive failed or detached

Suggested actions

For this instance type	Do this
Amazon EBS-backed	Try one or more of the following to resolve the issue: • Stop the instance, attach the volume to an existing running instance. • Manually run consistency checks. • Fix ramdisk to include relevant utilities. • Modify filesystem tuning parameters to remove consistency requirements (not recommended).
Instance store-backed	Try one or more of the following to resolve the issue: • Rebundle ramdisk with correct tooling. • Modify file system tuning parameters to remove consistency requirements (not recommended). • Terminate the instance and launch a new instance. • (Optional) Seek technical assistance for data recovery using AWS Support

Operating System Errors

- GRUB prompt (grubdom>)
- Bringing up interface eth0: Device eth0 has different MAC address than expected, ignoring. (Hard-coded MAC address)

- Unable to load SELinux Policy. Machine is in enforcing mode. Halting now. (SELinux misconfiguration)
- XENBUS: Timeout connecting to devices (Xenbus timeout)

GRUB prompt (grubdom>)

This condition is indicated by a system log similar to the one shown below.

```
   GNU GRUB  version 0.97  (629760K lower / 0K upper memory)

      [ Minimal BASH-like line editing is supported.   For

        the   first   word,  TAB  lists  possible  command

        completions.  Anywhere else TAB lists the possible

        completions of a device/filename. ]

grubdom>
```

Potential causes

Instance type	Potential causes
Amazon EBS-backed	• Missing GRUB configuration file. • Incorrect GRUB image used, expecting GRUB configuration file at a different location. • Unsupported filesystem used to store your GRUB configuration file (for example, converting your root file

Instance type	Potential causes
	system to a type that is not supported by an earlier version of GRUB).
Instance store-backed	• Missing GRUB configuration file. • Incorrect GRUB image used, expecting GRUB configuration file at a different location. • Unsupported filesystem used to store your GRUB configuration file (for example, converting your root file system to a type that is not supported by an earlier version of GRUB).

Suggested actions

For this instance type	Do this
Amazon EBS-backed	Option 1: Modify the AMI and relaunch the instance: 1. Modify the source AMI to create a GRUB configuration file at the standard location (/boot/grub/menu.lst). 2. Verify that your version of GRUB supports the underlying file system type and upgrade GRUB if necessary. 3. Pick the appropriate GRUB image, (hd0-1st drive or hd00 – 1st drive, 1st partition). 4. Terminate the instance and launch a new one using the AMI that you created. Option 2: Fix the existing instance: 1. Stop the instance. 2. Detach the root filesystem. 3. Attach the root filesystem to a known working instance. 4. Mount filesystem. 5. Create a GRUB configuration file.

For this instance type	Do this
	6. Verify that your version of GRUB supports the underlying file system type and upgrade GRUB if necessary. 7. Detach filesystem. 8. Attach to the original instance. 9. Modify kernel attribute to use the appropriate GRUB image (1st disk or 1st partition on 1st disk). 10. Start the instance.
Instance store-backed	Option 1: Modify the AMI and relaunch the instance: 1. Create the new AMI with a GRUB configuration file at the standard location (/boot/grub/menu.lst). 2. Pick the appropriate GRUB image, (hd0-1st drive or hd00 – 1st drive, 1st partition). 3. Verify that your version of GRUB supports the underlying file system type and upgrade GRUB if necessary. 4. Terminate the instance and launch a new instance using the AMI you created. Option 2: Terminate the instance and launch a new instance, specifying the correct kernel. Note To recover data from the existing instance, contact AWS Support.

Bringing up interface eth0: Device eth0 has different MAC address than expected, ignoring. (Hard-coded MAC address)

This condition is indicated by a system log similar to the one shown below.

```
. . .
```

```
Bringing up loopback interface:  [  OK  ]


Bringing up interface eth0:  Device eth0 has different MAC address
than expected, ignoring.

[FAILED]


Starting auditd: [  OK  ]
```

Potential causes

There is a hardcoded interface MAC in the AMI configuration

Suggested actions

For this instance type	Do this
Amazon EBS-backed	Do one of the following: • Modify the AMI to remove the hardcoding and relaunch the instance. • Modify the instance to remove the hardcoded MAC address. OR Use the following procedure: 1. Stop the instance. 2. Detach the root volume. 3. Attach the volume to another instance and modify the volume to remove the hardcoded MAC address. 4. Attach the volume to the original instance. 5. Start the instance.

For this instance type	Do this
Instance store-backed	Do one of the following: • Modify the instance to remove the hardcoded MAC address. • Terminate the instance and launch a new instance.

Unable to load SELinux Policy. Machine is in enforcing mode. Halting now. (SELinux misconfiguration)

This condition is indicated by a system log similar to the one shown below.

```
audit(1313445102.626:2): enforcing=1 old_enforcing=0 auid=4294967295

Unable to load SELinux Policy. Machine is in enforcing mode. Halting now.

Kernel panic - not syncing: Attempted to kill init!
```

Potential causes

SELinux has been enabled in error:

- Supplied kernel is not supported by GRUB
- Fallback kernel does not exist

Suggested actions

For this instance type	Do this
Amazon EBS-backed	Use the following procedure: 1. Stop the failed instance. 2. Detach the failed instance's root volume. 3. Attach the root volume to another running Linux instance (later referred to as a recovery instance). 4. Connect to the recovery instance and mount the failed instance's root volume. 5. Disable SELinux on the mounted root volume. This process varies across Linux distributions; for more information, consult your OS-specific documentation. Note On some systems, you disable SELinux by setting `SELINUX=disabled` in the `/mount_point/etc/sysconfig/selinux` file, where *`mount_point`* is the location that you mounted the volume on your recovery instance. 6. Unmount and detach the root volume from the recovery instance and reattach it to the original instance. 7. Start the instance.
Instance store-backed	Use the following procedure: 1. Terminate the instance and launch a new instance. 2. (Optional) Seek technical assistance for data recovery using AWS Support.

XENBUS: Timeout connecting to devices (Xenbus timeout)

This condition is indicated by a system log similar to the one shown below.

```
Linux version 2.6.16-xenU (builder@xenbat.amazonsa) (gcc version
4.0.1

20050727 (Red Hat 4.0.1-5)) #1 SMP Mon May 28 03:41:49 SAST 2007

...

XENBUS: Timeout connecting to devices!

...

Kernel panic - not syncing: No init found.  Try passing init= option
to kernel.
```

Potential causes

- The block device is not connected to the instance
- This instance is using an old instance kernel

Suggested actions

For this instance type	Do this
Amazon EBS-backed	Do one of the following: • Modify the AMI and instance to use a modern kernel and relaunch the instance. • Reboot the instance.
Instance store-backed	Do one of the following: • Terminate the instance. • Modify the AMI to use a modern kernel, and launch a new instance using this AMI.

Error-Common

Boot from the wrong volume

In some situations, you may find that a volume other than the volume attached to `/dev/xvda` or `/dev/sda` has become the root volume of your instance. This can happen when you have attached the root volume of another instance, or a volume created from the snapshot of a root volume, to an instance with an existing root volume.

This is due to how the initial ramdisk in Linux works. It chooses the volume defined as `/` in the `/etc/fstab`, and in some distributions, this is determined by the label attached to the volume partition. Specifically, you find that your `/etc/fstab` looks something like the following:

```
LABEL=/ / ext4 defaults,noatime 1 1

tmpfs /dev/shm tmpfs defaults 0 0

devpts /dev/pts devpts gid=5,mode=620 0 0

sysfs /sys sysfs defaults 0 0

proc /proc proc defaults 0 0
```

If you check the label of both volumes, you see that they both contain the `/` label:

```
[ec2-user ~]$ sudo e2label /dev/xvda1

/

[ec2-user ~]$ sudo e2label /dev/xvdf1

/
```

In this example, you could end up having `/dev/xvdf1` become the root device that your instance boots to after the initial ramdisk runs, instead of the `/dev/xvda1` volume from which you had intended to boot. To solve this, use the same **e2label** command to change the label of the attached volume that you do not want to boot from.

In some cases, specifying a UUID in `/etc/fstab` can resolve this. However, if both volumes come from the same snapshot, or the secondary is created from a snapshot of the primary volume, they share a UUID.

```
[ec2-user ~]$ sudo blkid

/dev/xvda1: LABEL="/" UUID=73947a77-ddbe-4dc7-bd8f-3fe0bc840778
TYPE="ext4" PARTLABEL="Linux" PARTUUID=d55925ee-72c8-41e7-b514-
7084e28f7334

/dev/xvdf1: LABEL="old/" UUID=73947a77-ddbe-4dc7-bd8f-3fe0bc840778
TYPE="ext4" PARTLABEL="Linux" PARTUUID=d55925ee-72c8-41e7-b514-
7084e28f7334
```

To change the label of an attached ext4 volume

1. Use the **e2label** command to change the label of the volume to something other than `/`.

```
2. [ec2-user ~]$ sudo e2label /dev/xvdf1 old/
```

3. Verify that the volume has the new label.

```
4. [ec2-user ~]$ sudo e2label /dev/xvdf1

   old/
```

To change the label of an attached xfs volume

- Use the **xfs_admin** command to change the label of the volume to something other than `/`.

```
- [ec2-user ~]$ sudo xfs_admin -L old/ /dev/xvdf1
- writing all SBs

  new label = "old/"
```

After changing the volume label as shown, you should be able to reboot the instance and have the proper volume selected by the initial ramdisk when the instance boots.

Important

If you intend to detach the volume with the new label and return it to another instance to use as the root volume, you must perform the above procedure again

and change the volume label back to its original value. Otherwise, the other instance does not boot because the ramdisk is unable to find the volume with the label `/`.