



Enabling User Provided Kernels in Amazon EC2

Version 1.2 – Document update 2010-09-20

AWS is pleased to announce that customers can now run user provided kernels in EC2.

Enablement

With this feature Amazon allows you to load a para-virtual Linux kernel within an Amazon Machine Image (AMI) or Amazon EBS volume. Thus, you have the option to create images that contain a kernel and initrd, and behave closer to traditional virtual or physical Linux installations. By enabling you to boot from the kernel within volumes, you can now seamlessly upgrade the kernel on Amazon EBS-backed instances. It is our expectation that AMI providers will update their AMIs to use this new feature, and most EC2 users will be able to begin managing their own kernels when these updated AMIs become available. Your AMI provider will be able to tell you when they plan to support this feature. However, if you would like to begin managing your own kernel now, this document will provide you with the necessary steps to do so. This process assumes general knowledge of EC2 AMI bundling and registration, as well as knowledge of how to install kernel packages and configure GRUB on your Linux systems.

PVGRUB: A New Amazon Kernel Image (AKI)

To enable user provided kernels, Amazon has published AKIs that use a system called PVGRUB. PVGRUB is a para-virtual “mini-OS” that runs a version of GNU GRUB, the standard Linux boot loader. PVGRUB selects the kernel to boot by reading `/boot/grub/menu.lst` from your image. It will load the kernel specified by your image and then shut down the “mini-OS”, so that it no longer consumes any resources. One of the advantages of this solution is that PVGRUB understands standard `grub.conf` or `menu.lst` commands, allowing it to work with most existing Linux distributions.

Instruction Overview

To get started, you will need to perform the following high-level steps:

1. Install an EC2 compatible kernel.
2. Generate an initrd.
3. Populate `/boot/grub/menu.lst` referencing your kernel.
4. Select an appropriate AKI ID from the “Amazon Kernel Image IDs” section below.
5. For new AMIs or Amazon EBS volumes, bundle the AMI and set the default to your chosen AKI.
6. Upload and register your new AMI.
7. For existing AMIs, you can simply specify the appropriate AKI ID when you call `RunInstances` or from within the AWS Management console.



Detailed Instructions

The following instructions give an example of how to enable an openSUSE AMI to use the PVGRUB AKI to run a user provided kernel by rebundling from a running instance. **IMPORTANT:** The specific details of configuring your AMI to use PVGRUB will vary depending on your exact Linux environment. The following example is for openSUSE 11.2.

1. Install an EC2 compatible kernel from the command line on your running Linux instance.

```
# rpm -ivh /tmp/kernel-ec2-2.6.35-rc4.8.1.x86_64.rpm

warning: /tmp/kernel-ec2-2.6.35-rc4.8.1.x86_64.rpm: Header V3 DSA signature:
NOKEY, key ID a29f6635
Preparing...                               #####
[100%]
   1:kernel-ec2                             #####
[100%]

Kernel image:   /boot/vmlinuz-2.6.35-rc4-8-ec2
Initrd image:   /boot/initrd-2.6.35-rc4-8-ec2
Root device:    /dev/sda1 (mounted on / as ext3)

Features:       block
14807 blocks
```

2. Generate an initrd on your running Linux instance.

```
# mkinitrd
Kernel image:   /boot/vmlinuz-2.6.35-rc4-8-ec2
Initrd image:   /boot/initrd-2.6.35-rc4-8-ec2
Root device:    /dev/sda1 (mounted on / as ext3)
Features:       block
14806 blocks
```

3. Populate `/boot/grub/menu.lst` referencing your kernel on your running Linux instance. **IMPORTANT:** You must modify your own menu.lst for your specific environment.

```
default 0
timeout 3

title EC2
    root (hd0)
    kernel /boot/vmlinuz-ec2 root=/dev/sda1
    initrd /boot/initrd-ec2
```



4. Select an appropriate AKI ID from the “Amazon Kernel Image IDs” section below. For this host, we’ve chosen aki-427d952b, because we are bundling an AMI.
5. For new AMIs or Amazon EBS volumes, bundle the AMI and set the default to your chosen AKI from your running Linux instance.

```
# ec2-bundle-vol -r x86_64 -d /mnt -p openSUSE-11.2-PVGRUB -u [AWS-ID] -k /mnt/pkey.pem -c /mnt/cert.pem -s 10240 -e /mnt,/root/.ssh --kernel aki-427d952b
```

6. Upload the bundle from your running Linux instance to Amazon S3.

```
# ec2-upload-bundle -b MyReallyCoolBucketLocation -m /mnt/oeprnSUSE-11.2-PVGRUB.manifest.xml -a MyAccessKey -s MySecretKey
```

7. Register the AMI with the AKI chosen in step 4 from your desktop using the EC2 command line tools.

```
$ ec2-register --name openSUSE-11.2-PVGRUB MyReallyCoolBucketLocation/oeprnSUSE-11.2-PVGRUB.manifest.xml
```

Amazon Kernel Image (AKI) IDs

Several PVGRUB AKIs are available depending on the type and location of your instance. There are AKIs for 32-bit and 64-bit architecture types, and there are AKIs for Amazon S3-Backed instances and Amazon EBS-backed instances. For S3-Backed instances, choose an AKI with “hd0” in the name, and for EBS-backed instances chose an AKI with “hd00” in the name. These names refer to different disk layouts within the AKI that are appropriate for each use case.

Each AKI type is available in all four regions:

US-East-1

| | |
|--------------|---|
| aki-4c7d9525 | ec2-public-images/pv-grub-hd00-V1.01-i386.gz.manifest.xml |
| aki-4e7d9527 | ec2-public-images/pv-grub-hd00-V1.01-x86_64.gz.manifest.xml |
| aki-407d9529 | ec2-public-images/pv-grub-hd0-V1.01-i386.gz.manifest.xml |
| aki-427d952b | ec2-public-images/pv-grub-hd0-V1.01-x86_64.gz.manifest.xml |

US-West-1

| | |
|--------------|---|
| aki-9da0f1d8 | ec2-public-images-us-west-1/pv-grub-hd00-V1.01-i386.gz.manifest.xml |
| aki-9fa0f1da | ec2-public-images-us-west-1/pv-grub-hd00-V1.01-x86_64.gz.manifest.xml |
| aki-99a0f1dc | ec2-public-images-us-west-1/pv-grub-hd0-V1.01-i386.gz.manifest.xml |
| aki-9ba0f1de | ec2-public-images-us-west-1/pv-grub-hd0-V1.01-x86_64.gz.manifest.xml |

EU-West-1

| | |
|--------------|--|
| aki-47eec433 | ec2-public-images-eu/pv-grub-hd00-V1.01-i386.gz.manifest.xml |
| aki-41eec435 | ec2-public-images-eu/pv-grub-hd00-V1.01-x86_64.gz.manifest.xml |
| aki-4deec439 | ec2-public-images-eu/pv-grub-hd0-V1.01-i386.gz.manifest.xml |
| aki-4feec43b | ec2-public-images-eu/pv-grub-hd0-V1.01-x86_64.gz.manifest.xml |



AP-SouthEast-1

| | |
|--------------|--|
| aki-6fd5aa3d | ec2-public-images-ap-southeast-1/pv-grub-hd00-V1.01-i386.gz.manifest.xml |
| aki-6dd5aa3f | ec2-public-images-ap-southeast-1/pv-grub-hd00-V1.01-x86_64.gz.manifest.xml |
| aki-13d5aa41 | ec2-public-images-ap-southeast-1/pv-grub-hd0-V1.01-i386.gz.manifest.xml |
| aki-11d5aa43 | ec2-public-images-ap-southeast-1/pv-grub-hd0-V1.01-x86_64.gz.manifest.xml |

Compatible PVGRUB Kernels

There are a number of Linux distributions that have compatible EC2 kernels. The following is a *brief, non-comprehensive list* of kernels that we have worked with the maintainers to test:

- Fedora 8-9 Xen kernels
- Fedora 13 (2.6.33.6-147 and higher)
- SLES/openSUSE 10.x, 11.0, 11.1 Xen
- SLES/openSUSE 11.x EC2 Variant
- Ubuntu EC2 Variant kernels
- RHEL 5.x kernels
- RHEL 6.x kernels
- CentOS 5.x kernels
- Gentoo (see FAQ)

It is possible that your specific Linux kernel will not boot using the new PVGRUB method. If you find that to be the case, you should select a different kernel or use a non-PVGRUB AKI to boot your instance.

Frequently Asked Questions

How do I create an AMI that uses PVGRUB?

When you create your AMI disk image, simply install the appropriate kernel and GRUB packages in your Linux operating system. Some Linux distributions have a specific EC2 kernel. If an EC2 kernel is indicated, you should install that kernel. More detail on the GRUB configuration is provided below.

How do I update an AMI to use PVGRUB?

Updating an existing AMI consists of the following steps. These steps are more fully documented in the Amazon EC2 Documentation:

1. Re-launch your AMI.
2. Install GRUB and your desired kernel in the running instance.
3. Rebundle the AMI.
4. Register the new AMI setting your chosen PVGRUB AKI as the default.



How do I configure GRUB?

PVGRUB contains a full-featured version of GRUB. In order for PVGRUB to boot, a GRUB menu.lst file must exist in the image.

For most distributions, you have two options for the GRUB configuration:

- Option 1: Install GRUB and allow the default kernel installation scripts to handle the installation and updating of the GRUB configuration. The steps necessary to install GRUB will vary depending on your Linux distribution, but typically GRUB will be available as a package you can install online.
- Option 2: Populate a general /boot/grub/menu.lst. An example of a menu.lst configuration file for booting an AMI with a PVGRUB AKI follows. **IMPORTANT:** You must modify your own menu.lst for your specific environment.

```
default 0
timeout 3
fallback 1

title Vanilla EC2 Kernel 2.6.32.10
    root (hd0)
    kernel /boot/vmlinuz-2.6.32.10-ACME_SYS_EC2 root=/dev/sda1
    initrd /boot/initrd-2.6.32.10-ACME_SYS_EC2

title Ubuntu EC2 2.6.32.302-EC
    root (hd0)
    kernel /boot/ubuntu-ec2 root=/dev/sda1
    initrd /boot/initrd-ec2
```

We recommend Option 2 to control the kernel booting for two reasons. First, EC2 users don't have interactive control over the boot process; since there is no keyboard access, GRUB will proceed without user interaction. Secondly, and most importantly for Amazon EBS-backed images, you want to protect against distributions that auto-update the kernel from breaking your image. By not relying on the auto-update mechanism and explicitly choosing which kernel you run, you reduce the risk of an incompatible kernel becoming the default kernel.

A fallback kernel does not have to be specified in your menu.lst, but is recommended when testing new kernels. GRUB can fall back to another kernel in the event that the new kernel fails. Having a fallback kernel allows the instance to boot even if the new kernel is not found.

What kernels will work with PVGRUB?

Some Linux distributions provide kernels that are not compatible with EC2. We are working with vendors to ensure that the most popular AMIs provide kernels that work with EC2, and we have tested a number of these AMIs and found them to be compatible with PVGRUB. Unfortunately, it is not possible to support every kernel that is or can be compiled. In the event that a kernel does not work consistently or at all, we recommend you use a known good kernel, select a non-PVGRUB AKI, or seek support from your AMI vendor.



Kernels that disable the pv-ops XSAVE hypercall are known to work on all instance types, whereas those that enable this hypercall will fail to launch in some cases. Similarly, non-pv-ots kernels that do not adhere to the Xen 3.0.2 interface might fail to launch in some cases.

What are the limitations of PVGRUB?

Since PVGRUB is a para-virtual version of GRUB 0.97, it has all the limitations of GRUB. Most importantly, this means that it will not work properly for certain disk layouts or filesystem types.

Supported /boot file systems that PVGRUB can boot from are:

- EXT2/3/4
- XFS
- ReiserFS

Note: These are the /boot file systems that we have tested and verified. Others could boot from PVGRUB, but haven't been tested.

I am an expert. How do I compile my own EC2 capable kernel?

Amazon is unable to offer support for compiling your own kernel. The following information is provided to assist those who are familiar with compiling kernels. Many Linux distributions provide documentation on how to compile a kernel in kernel source packages.

openSUSE and Gentoo kernels

Both openSUSE and Gentoo provide native source code that enable compiling an EC2 capable kernel. For openSUSE, the configuration options are part of the mainline 11.2 and higher, while Gentoo provides a Xen-compatible in the "xen-sources" portage package. For both distributions, please use the following kernel configuration options:

```
CONFIG_XEN_COMPAT_00002_AND_LATER=y
CONFIG_XEN_COMPAT=0x030002
CONFIG_HOTPLUG_CPU=y"
```

Patching vanilla or other distribution kernels (apply Xen DomU patchset)

The process of patching any kernel (vanilla or distribution) is an advanced topic. For instructions, see "kernel/Documentation/applying-patches.txt" under the kernel source tree for instructions.

The same patchset that is used with openSUSE and Gentoo can be applied to a vanilla kernel.org kernel.

<http://code.google.com/p/gentoo-xen-kernel/downloads/list>

2.6.29: <http://lists.xensource.com/archives/html/xen-users/2009-04/msg00004.html>

<http://lists.xensource.com/archives/html/xen-devel/2009-06/msg00127.html>

2.6.30: <http://lists.xensource.com/archives/html/xen-devel/2009-07/msg00027.html>

2.6.31: <http://lists.xensource.com/archives/html/xen-users/2009-10/msg00342.html>



- 2.6.31-9: <http://lists.xensource.com/archives/html/xen-users/2009-12/msg00411.html>
- 2.6.31-10: <http://lists.xensource.com/archives/html/xen-users/2010-01/msg00032.html>
- 2.6.31-14 and 2.6.32-1 : <http://lists.xensource.com/archives/html/xen-users/2010-04/msg00091.html>

Patching vanilla or other distribution kernels (PVOps)

For PVOps kernels, you can elect to disable the XSAVE hypercall in the guest. The following patch works against 2.6.32 through 2.6.35 kernels.

```
---
arch/x86/xen/enlighten.c |      1 +
1 files changed, 1 insertions(+), 0 deletions(-)

diff --git a/arch/x86/xen/enlighten.c b/arch/x86/xen/enlighten.c
index 52f8e19..6db3d67 100644
--- a/arch/x86/xen/enlighten.c
+++ b/arch/x86/xen/enlighten.c
@@ -802,6 +802,7 @@ static void xen_write_cr4(unsigned long cr4)
 {
     cr4 &= ~X86_CR4_PGE;
     cr4 &= ~X86_CR4_PSE;
+    cr4 &= ~X86_CR4_OSXSAVE;

     native_write_cr4(cr4);
 }
--
1.6.6.1
```

How do I use EBS Volumes with a PVGRUB enabled image?

There are two special considerations when using a PVGRUB enabled image to mount EBS volumes. First, for EBS volumes the first partition must be a boot partition. Second, if you plan to use LVM with Amazon EBS volumes, you need a separate boot partition outside of LVM and can then create logical volumes with LVM. Please note that PVGRUB expects to find the menu.lst in /boot/grub. As a result, if the boot partition is mounted in at /boot, menu.lst will be found in /boot/boot/grub.

How is PVGRUB supported by Amazon?

Amazon supports the use of PVGRUB to load a kernel for your AMI. However, we cannot provide support for your kernel itself or failures caused by the use of a kernel that does not meet the requirements of PVGRUB. If you have trouble loading a kernel, you should consult with your AMI vendor or the Amazon EC2 forums. Unfortunately, due to the wide and varied kernel landscape, it is impossible for Amazon to provide support for all kernel varieties.

Whoops.....

In the event that using the PVGRUB AKI fails, you can boot the instance using a non-PVGRUB AKI/ARI.