Fedora Build ACS Override Patch Kernel

Add RPM Fusion

```
sudo dnf install https://download1.rpmfusion.org/free/fedora/rpmfusion-free-release-$(rpm -E %fedora).noarch.rpm https://download1.rpmfusion.org/nonfree/fedora/rpmfusion-nonfree-release-$(rpm -E %fedora).noarch.rpm
```

Install the dependencies to start building your own kernel.

```
sudo dnf install fedpkg fedora-packager rpmdevtools ncurses-devel pesign fedora-packager fedora-review rpmdevtools numactl-devel pesign
sudo dnf groupinstall "Development Tools"
sudo dnf build-dep kernel
```

Set up your home build directory (if you haven't ever built any RPMs before)

```
rpmdev-setuptree
```

Install the kernel source and finish installing dependencies.

```
cd ~/rpmbuild/SOURCES
sudo dnf download --source kernel
rpm2cpio kernel-* | cpio -i --make-directories
mv kernel-*.src.rpm ../SRPMS
cd ../SRPMS
rpm -Uvh kernel-*.src.rpm
vim ~/rpmbuild/SPECS/kernel.spec
```

Add the two lines near the top of the spec file.

> # Set buildid
> %define buildid .acs

```
# ACS override patch
Patch1000: add-acs-override.patch
```

Download ACS path

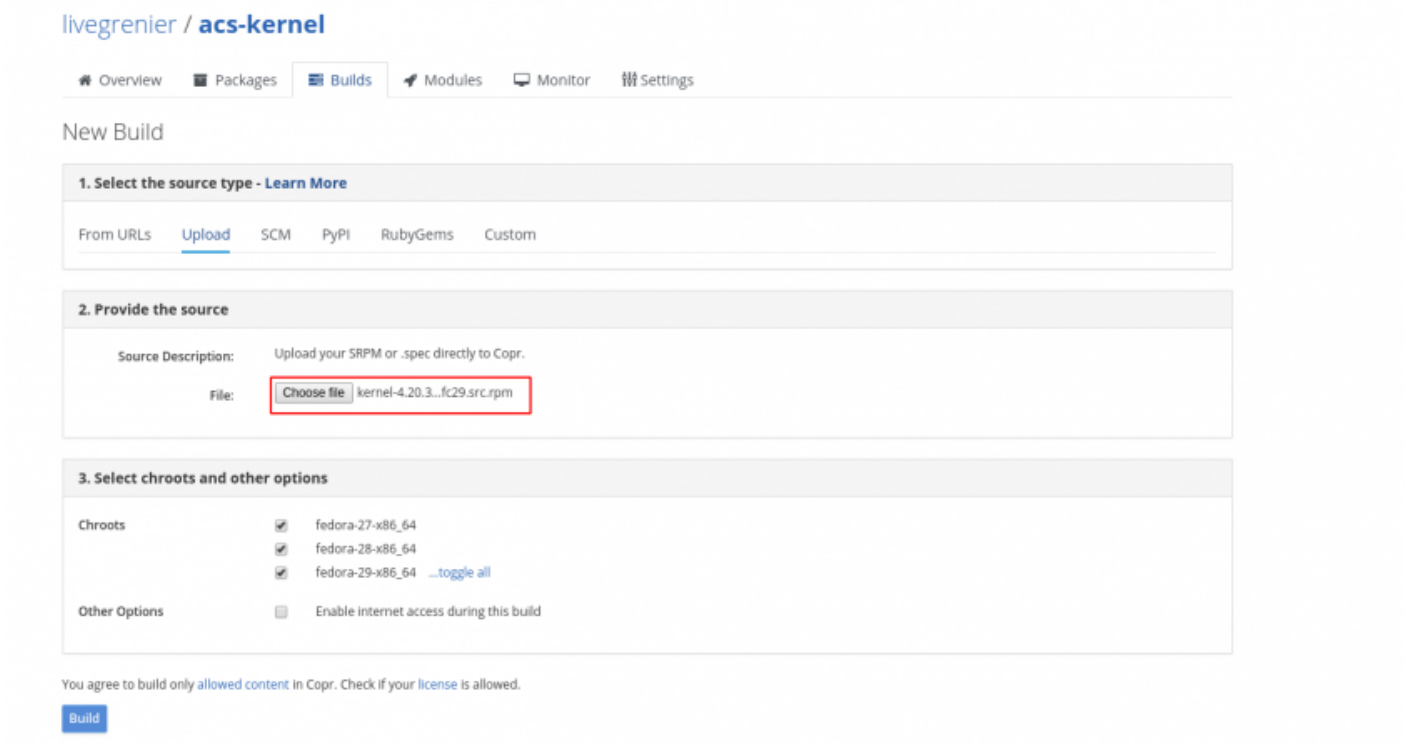
```
cd ~/rpmbuild/SOURCES/
wget https://git.myhypervisor.ca/dave/fedora-acs-override/raw/master/acs/add-acs-override.patch
```

Re-create new SRC RPM

```
rpmbuild -bs ~/rpmbuild/SPECS/kernel.spec
```

Upload to Copr (need account)

> **New file located in ~/rpmbuild/SRPMS/kernel-4.20.3-200.acs.fc29.src.rpm**



(Wait 10hours for RPM to build)

Enable new repo

```
dnf copr enable user/pkg-name
```

Install new kernel

```
sudo dnf update kernel-4.20.6-200.acs.fc29 kernel-devel-4.20.6-200.acs.fc29 --disableexcludes all --refresh
```

> **Update and reboot**

Update GRUB file /etc/default/grub, change GRUB_CMDLINE to bellow

```
GRUB_CMDLINE_LINUX="rd.driver.pre=vfio-pci rd.driver.blacklist=nouveau modprobe.blacklist=nouveau rhgb quiet intel_iommu=on iommu=pt pcie_acs_override=downstream"
```

Rebuild GRUB's configuration

```
sudo grub2-mkconfig -o /boot/efi/EFI/fedora/grub.cfg
```

Create or edit /etc/modprobe.d/local.conf, adding the line below:

```
install vfio-pci /sbin/vfio-pci-override.sh
```

Create or edit /etc/dracut.conf.d/local.conf, adding the line below:

```
add_drivers+="vfio vfio_iommu_type1 vfio_pci vfio_virqfd"
install_items+="/sbin/vfio-pci-override.sh /usr/bin/find /usr/bin/dirname"
```

Create a file /sbin/vfio-pci-override.sh with permissions 755 (file in this directory of the repo).

> #!/bin/sh
>
> # This script overrides the default driver to be the vfio-pci driver (similar
> # to the pci-stub driver) for the devices listed. In this case, it only uses
> # two devices that both belong to one nVidia graphics card (graphics, audio).
>
> # Located at /sbin/vfio-pci-override.sh
>
> DEVS="0000:02:00.0 0000:02:00.1"
>
> if [! -z "$(ls -A /sys/class/iommu)"] ; then
> for DEV in $DEVS; do
> echo "vfio-pci" > /sys/bus/pci/devices/$DEV/driver_override
> done
> fi

> modprobe -i vfio-pci

Rebuild using dracut

```
sudo dracut -f --kver `uname -r`
```

> **IF YOU HAVE 2 NVIDIA CARDS**

Install (proprietary) nVidia drivers and remove/blacklist (open source) nouveau drivers.

```
sudo su -
dnf install xorg-x11-drv-nvidia akmod-nvidia "kernel-devel-uname-r == $(uname -r)" xorg-x11-drv-nvidia-cuda vulkan vdpauinfo libva-vdpau-driver libva-utils
dnf remove *nouveau*
echo "blacklist nouveau" >> /etc/modprobe.d/blacklist.conf
```

> **Reboot**

Revision #19
Created 2018-12-07 05:53:23 UTC by Dave
Updated 2019-02-12 08:31:39 UTC by Dave