

Dual-booting Bitlocker Windows 7 and Ubuntu 16.04 with NeoGRUB and EasyBCD

As part of offering tools to do the job efficiently, I was tasked to set up a second operating system on an existing Windows 7 laptop with Bitlocker-encrypted NTFS partition. I tried an older version of VMWare but vmware-tools there supported upstart, not systemd. That made using the mouse too difficult with misalignment issues, switching Aero off and setting scaling to 100% didn't help. Virtualbox didn't support the 64bit os.

Steps to create the dualboot:

1. Shrink the existing NTFS partition to make room for the Ubuntu installation. Rightclick Computer - Manage - Disk Management - rightclick partition - Shrink Volume. About 20-30GB should do.
2. Pause bitlocker while the configuration is being updated. Control panel - Bitlocker Drive Encryption - Suspend Protection
3. Create a Live-USB of the Ubuntu or whatever Linux you're about to install. Used [LiLi](#) myself.
4. Boot the computer with the Live USB stick, Install Ubuntu with advanced settings.
 1. Add two logical partitions to the empty space: a swap partition the same size as the machines RAM, and an ext4 partition with mountpoint / (root)
 2. **Make sure to set the bootloader to the logical partition, not MBR.** Example: /dev/sda6 instead of /dev/sda
5. After Ubuntu has finished installing, boot to windows and download [EasyBCD](#)
6. Using EasyBCD, Add a new entry to the bootmenu from the NeoGRUB tab (install)
7. Click configure on the NeoGRUB tab and write this entry, referencing /dev/sda6:
title Ubuntu
root (hd0,5)
chainloader (hd0,5)+1
8. Save changes from the Edit boot menu -tab
9. Reveal the unencrypted windows partition from Computer - Manage - Disk Management - rightclick system partition - Change drive letter and paths. I assigned the letter Q: for the system drive.
10. Copy C:\NeoGrub and C:\NST folder to the Q: drive as Q:\NeoGrub and Q:\NST
11. Boot, verify that both Windows and Ubuntu start, resume bitlocker protection
 1. Bonus cleanup: C:\NST\NeoGrub.mbr is required, as are Q:\NeoGrub and Q:\NST\menu.lst. You can remove the extra copies.

This method doesn't flinch when the kernel is updated, unlike the traditional method of copying 512 bytes of the bootloader to a separate file and using bcdedit to boot from that file.

By: Joni Nevalainen / CSC