Free Open source disk encryption with strong security for the Paranoid

22-27 minutes

Hidden Operating System

If your system partition or system drive is encrypted using VeraCrypt, you need to enter your pre-boot authentication password in the VeraCrypt Boot Loader screen after you turn on or restart your computer. It may happen that you are forced by somebody to decrypt the operating system or to reveal the pre-boot authentication password. There are many situations where you cannot refuse to do so (for example, due to extortion). VeraCrypt allows you to create a hidden operating system whose existence should be impossible to prove (provided that certain guidelines are followed — see below). Thus, you will not have to decrypt or reveal the password for the hidden operating system.

Before you continue reading this section, make sure you have read the section **Hidden Volume** and that you understand what a hidden VeraCrypt volume is.

A **hidden operating system** is a system (for example, Windows 7 or Windows XP) that is installed in a hidden VeraCrypt volume. It should be impossible to prove that a hidden VeraCrypt volume exists (provided that certain guidelines are followed; for more information, see the section Hidden Volume) and, therefore, it should be impossible to prove that a hidden operating system exists.

However, in order to boot a system encrypted by VeraCrypt, an unencrypted copy of the VeraCrypt Boot Loader has to be stored on the system drive or on a VeraCrypt Rescue Disk. Hence, the mere presence of the VeraCrypt Boot Loader can indicate that there is a system encrypted by VeraCrypt on the computer. Therefore, to provide a plausible explanation for the presence of the VeraCrypt Boot Loader, the VeraCrypt wizard helps you create a second encrypted operating system, so-called **decoy operating system**, during the process of creation of a hidden operating system. A decoy operating system must not contain any sensitive files. Its existence is not secret (it is *not* installed in a hidden volume). The password for the decoy operating system can be safely revealed to anyone forcing you to disclose your pre-boot authentication password.*

You should use the decoy operating system as frequently as you use your computer. Ideally, you should use it for all activities that do not involve sensitive data. Otherwise, plausible deniability of the hidden operating system might be adversely affected (if you revealed the password for the decoy operating system to an adversary, he could find out that the system is not used very often, which might indicate the existence of a hidden operating system on your computer). Note that you can save data to the decoy system partition anytime without any risk that the hidden volume will get damaged (because the decoy system is *not* installed in the outer volume — see below).

There will be two pre-boot authentication passwords — one for the hidden system and the other for the decoy system. If you want to start the hidden system, you simply enter the password for the hidden system in the VeraCrypt Boot Loader screen (which appears after you turn on or restart your computer). Likewise, if you want to start the decoy system (for example, when asked to do so by an adversary), you just enter the password for the decoy system in the VeraCrypt Boot Loader screen.

Note: When you enter a pre-boot authentication password, the VeraCrypt Boot Loader first attempts to decrypt (using the entered password) the last 512 bytes of the first logical track of the system drive (where encrypted master key data for non-hidden encrypted system partitions/drives are normally stored). If it fails and if there is a partition behind the active partition, the VeraCrypt Boot Loader (even if there is actually no hidden volume on the drive) automatically tries to decrypt (using the same entered password again) the area of the first partition behind the active partition where the encrypted header of a possible hidden volume might be stored (however, if the size of the active partition is less than 256 MB, then the data is read from the *second* partition behind the active one, because Windows 7 and later, by default, do not boot from the partition on which they are installed). Note that VeraCrypt never knows if there is a hidden volume in advance (the hidden volume header cannot be identified, as it appears to consist entirely of random data). If the header is successfully decrypted (for information on how VeraCrypt determines that it was successfully decrypted, see the section Encryption Scheme), the information about the size of the hidden volume is retrieved from the decrypted header (which is still stored in RAM), and the hidden volume is mounted (its size also determines its offset). For further technical details, see the section Encryption Scheme in the chapter Technical Details.

When running, the hidden operating system appears to be installed on the same partition as the original operating system (the decoy system). However, in reality, it is installed within the partition behind it (in a hidden volume). All read/write operations are transparently redirected from the system partition to the hidden volume. Neither the operating system nor applications will know that data written to and read from the system partition is actually written to and read from the partition behind it (from/to a hidden volume). Any such data is encrypted and decrypted on the fly as usual (with an encryption key different from the one that is used for the decoy operating system).

Note that there will also be a third password — the one for the **outer volume**. It is not a pre-boot authentication password, but a regular VeraCrypt volume password. It can be safely disclosed to anyone forcing you to reveal the password for the encrypted partition where the hidden volume (containing the hidden operating system) resides. Thus, the existence of the hidden volume (and of the hidden operating system) will remain secret. If you are not sure you understand how this is possible, or what an outer volume is, please read the section Hidden Volume. The outer volume should contain some sensitive-looking files that you actually do *not* want to hide.

To summarize, there will be three passwords in total. Two of them can be revealed to an attacker (for the decoy system and for the outer volume). The third password, for the hidden system, must remain secret.

Example Layout of System Drive Containing Hidden Operating System

Process of Creation of Hidden Operating System

To start the process of creation of a hidden operating system, select *System* > *Create Hidden Operating System* and then follow the instructions in the wizard.

Initially, the wizard verifies that there is a suitable partition for a hidden operating system on the system drive. Note that before you can create a hidden operating system, you need to create a partition for it on the system drive. It must be the first partition behind the system partition and it must be at least 5% larger than the system partition (the system partition is the one where the currently running operating system is installed). However, if the outer volume (not to be confused with the system partition) is formatted as NTFS, the partition for the hidden operating system must be at least 110% (2.1 times) larger than the system partition (the reason is that the NTFS file system always stores internal data exactly in the middle of the volume and, therefore, the hidden volume, which is to contain a clone of the system partition, can reside only in the second half of the partition).

In the next steps, the wizard will create two VeraCrypt volumes (outer and hidden) within the first partition behind the system partition. The hidden volume will contain the hidden operating system. The size of the hidden volume is always the same as the size of the system partition. The reason is that the hidden volume will need to contain a clone of the content of the system partition (see below). Note that the clone will be encrypted using a different encryption key than the original. Before you start copying some sensitive-looking files to the outer volume, the wizard tells you the maximum recommended size of space that the files should occupy, so that there is enough free space on the outer volume for the hidden volume.

Remark: After you copy some sensitive-looking files to the outer volume, the cluster bitmap of the volume will be scanned in order to determine the size of uninterrupted area of free space whose end is aligned with the end of the outer volume. This area will accommodate the hidden volume, so it limits its maximum possible size. The maximum possible size of the hidden volume will be determined and it will be verified that it is greater than the size of the system partition (which is required, because the entire content of the system partition will need to be copied to the hidden volume — see below). This ensures that no data stored on the outer volume will be overwritten by data written to the area of the hidden volume (e.g. when the system is being copied to it). The size of the hidden volume is always the same as the size of the system partition.

Then, VeraCrypt will create the hidden operating system by copying the content of the system partition to the hidden volume. Data being copied will be encrypted on the fly with an encryption key different from the one that will be used for the decoy operating system. The process of copying the system is performed in the pre-boot environment (before Windows starts) and it may take a long time to complete; several hours or even several days (depending on the size of the system partition and on the performance of the computer). You will be able to interrupt the process, shut down your computer, start the operating system and then resume the process. However, if you interrupt it, the entire process of copying the system will have to start from the beginning (because the content of the system partition must not change during cloning). The hidden operating system will initially be a clone of the operating system under which you started the wizard.

Windows creates (typically, without your knowledge or consent) various log files, temporary files, etc., on the system partition. It also saves the content of RAM to hibernation and paging files located on the system partition. Therefore, if an adversary analyzed files stored on the partition where the original system (of which the hidden system is a clone) resides, he might find out, for example, that you used the VeraCrypt wizard in the hidden-system-creation mode (which might indicate the existence of a hidden operating system on your computer). To prevent such issues, VeraCrypt will securely erase the entire content of the partition where the original system resides after the hidden system has been created. Afterwards, in order to achieve plausible deniability, VeraCrypt will prompt you to install a new system on the partition and encrypt it using VeraCrypt. Thus, you will create the decoy system and the whole process of creation of the hidden operating system will be completed.

Note: VeraCrypt will erase the content of the partition where the original system resides by filling it with random data entirely. If you revealed the password for the decoy system to an adversary and he asked you why the free space of the (decoy) system partition contains random data, you could answer, for example: "The partition previously contained a system encrypted by VeraCrypt, but I forgot the pre-boot authentication password (or the system was damaged and stopped booting), so I had to reinstall Windows and encrypt the partition again."

Plausible Deniability and Data Leak Protection

For security reasons, when a hidden operating system is running, VeraCrypt ensures that all local unencrypted filesystems and non-hidden VeraCrypt volumes are read-only (i.e. no files can be written to such filesystems or VeraCrypt volumes).† Data is allowed to be written to any filesystem that resides within a hidden VeraCrypt volume (provided that the hidden volume is not located in a container stored on an unencrypted filesystem or on any other read-only filesystem).

There are three main reasons why such countermeasures have been implemented:

1. It enables the creation of a secure platform for mounting of hidden VeraCrypt volumes. Note that we officially recommend that hidden volumes are mounted only when a hidden operating system is running. For more information, see the subsection Security Requirements and Precautions Pertaining to Hidden Volumes.
2. In some cases, it is possible to determine that, at a certain time, a particular filesystem was not mounted under (or that a particular file on the filesystem was not saved or accessed from within) a particular instance of an operating system (e.g. by analyzing and comparing filesystem journals, file timestamps, application logs, error logs, etc). This might indicate that a hidden operating system is installed on the computer. The countermeasures prevent these issues.
3. It prevents data corruption and allows safe hibernation. When Windows resumes from hibernation, it assumes that all mounted filesystems are in the same state as when the system entered hibernation. VeraCrypt ensures this by write-protecting any filesystem accessible both from within the decoy and hidden systems. Without such protection, the filesystem could become corrupted when mounted by one system while the other system is hibernated.

If you need to securely transfer files from the decoy system to the hidden system, follow these steps:

1. Start the decoy system.
2. Save the files to an unencrypted volume or to an outer/normal VeraCrypt volume.
3. Start the hidden system

4. If you saved the files to a VeraCrypt volume, mount it (it will be automatically mounted as read-only).
5. Copy the files to the hidden system partition or to another hidden volume.

Possible Explanations for Existence of Two VeraCrypt Partitions on Single Drive

An adversary might ask why you created two VeraCrypt-encrypted partitions on a single drive (a system partition and a non-system partition) rather than encrypting the entire disk with a single encryption key. There are many possible reasons to do that. However, if you do not know any (other than creating a hidden operating system), you can provide, for example, one of the following explanations:

- If there are more than two partitions on a system drive and you want to encrypt only two of them (the system partition and the one behind it) and to leave the other partitions unencrypted (for example, to achieve the best possible performance when reading and writing data, which is not sensitive, to such unencrypted partitions), the only way to do that is to encrypt both partitions separately (note that, with a single encryption key, VeraCrypt could encrypt the entire system drive and *all* partitions on it, but it cannot encrypt only two of them — only one or all of the partitions can be encrypted with a single key). As a result, there will be two adjacent VeraCrypt partitions on the system drive (the first will be a system partition, the second will be a non-system one), each encrypted with a different key (which is also the case when you create a hidden operating system, and therefore it can be explained this way).

  If you do not know any good reason why there should be more than one partition on a system drive at all:

  It is generally recommended to separate non-system files (documents) from system files. One of the easiest and most reliable ways to do that is to create two partitions on the system drive; one for the operating system and the other for documents (non-system files). The reasons why this practice is recommended include:

  - If the filesystem on one of the partitions is damaged, files on the partition may get corrupted or lost, whereas files on the other partition are not affected.
  - It is easier to reinstall the system without losing your documents (reinstallation of an operating system involves formatting the system partition, after which all files stored on it are lost). If the system is damaged, full reinstallation is often the only option.
- A cascade encryption algorithm (e.g. AES-Twofish-Serpent) can be many times slower than a non-cascade one (e.g. AES). However, a cascade encryption algorithm may be more secure than a non-cascade one (for example, the probability that three distinct encryption algorithms will be broken, e.g. due to advances in cryptanalysis, is significantly lower than the probability that only one of them will be broken). Therefore, if you encrypt the outer volume with a cascade encryption algorithm and the decoy system with a non-cascade encryption algorithm, you can answer that you wanted the best performance (and adequate security) for the system partition, and the highest possible security (but worse performance) for the non-system partition (i.e. the outer volume), where you store the most sensitive data, which you do not need to access very often (unlike the operating system, which you use very often, and therefore you need it to have the best possible performance). On the system partition, you store data

that is less sensitive (but which you need to access very often) than data you store on the non-system partition (i.e. on the outer volume).

- Provided that you encrypt the outer volume with a cascade encryption algorithm (e.g. AES-Twofish-Serpent) and the decoy system with a non-cascade encryption algorithm (e.g. AES), you can also answer that you wanted to prevent the problems about which VeraCrypt warns when the user attempts to choose a cascade encryption algorithm for system encryption (see below for a list of the problems). Therefore, to prevent those problems, you decided to encrypt the system partition with a non-cascade encryption algorithm. However, you still wanted to use a cascade encryption algorithm (because it is more secure than a non-cascade encryption algorithm) for the most sensitive data, so you decided to create a second partition, which those problems do *not* affect (because it is non-system) and to encrypt it with a cascade encryption algorithm. On the system partition, you store data that is less sensitive than data you store on the non-system partition (i.e. on the outer volume).

  Note: When the user attempts to encrypt the system partition with a cascade encryption algorithm, VeraCrypt warns him or her that it can cause the following problems (and implicitly recommends to choose a non-cascade encryption algorithm instead):

  - For cascade encryption algorithms, the VeraCrypt Boot Loader is larger than normal and, therefore, there is not enough space in the first drive track for a backup of the VeraCrypt Boot Loader. Hence, *whenever* it gets damaged (which often happens, for example, during inappropriately designed anti-piracy activation procedures of certain programs), the user must use the VeraCrypt Rescue Disk to repair the VeraCrypt Boot Loader or to boot.
  - On some computers, resuming from hibernation takes longer.
- In contrast to a password for a non-system VeraCrypt volume, a pre-boot authentication password needs to be typed each time the computer is turned on or restarted. Therefore, if the pre-boot authentication password is long (which is required for security purposes), it may be very tiresome to type it so frequently. Hence, you can answer that it was more convenient for you to use a short (and therefore weaker) password for the system partition (i.e. the decoy system) and that it is more convenient for you to store the most sensitive data (which you do not need to access as often) in the non-system VeraCrypt partition (i.e. in the outer volume) for which you chose a very long password.

  As the password for the system partition is not very strong (because it is short), you do not intentionally store sensitive data on the system partition. However, you still prefer the system partition to be encrypted, because potentially sensitive or mildly sensitive data is stored on it as a result of your everyday use of the computer (for example, passwords to online forums you visit, which can be automatically remembered by your browser, browsing history, applications you run, etc.)

- When an attacker gets hold of your computer when a VeraCrypt volume is mounted (for example, when you use a laptop outside), he can, in most cases, read any data stored on the volume (data is decrypted on the fly as he reads it). Therefore, it may be wise to limit the time the volume is mounted to a minimum. Obviously, this may be impossible or difficult if the sensitive data is stored on an encrypted system partition or on an entirely encrypted system drive (because you would also have to limit the time you work with the computer to a minimum). Hence, you can answer that you created a separate partition (encrypted with a different key than your system partition) for your most sensitive data and that you mount it only when necessary and dismount it as soon as possible (so as to limit the time the volume is mounted to a minimum). On the system partition, you store data that is less sensitive (but which you need to access often) than data you store on the non-system partition (i.e. on the outer volume).

Safety/Security Precautions and Requirements Pertaining to Hidden Operating Systems

As a hidden operating system resides in a hidden VeraCrypt volume, a user of a hidden operating system must follow all of the security requirements and precautions that apply to normal hidden VeraCrypt volumes. These requirements and precautions, as well as additional requirements and precautions pertaining specifically to hidden operating systems, are listed in the subsection Security Requirements and Precautions Pertaining to Hidden Volumes.

WARNING: If you do not protect the hidden volume (for information on how to do so, refer to the section Protection of Hidden Volumes Against Damage), do *not* write to the outer volume (note that the decoy operating system is *not* installed in the outer volume). Otherwise, you may overwrite and damage the hidden volume (and the hidden operating system within it)!

If all the instructions in the wizard have been followed and if the security requirements and precautions listed in the subsection Security Requirements and Precautions Pertaining to Hidden Volumes are followed, it should be impossible to prove that the hidden volume and hidden operating system exist, even when the outer volume is mounted or when the decoy operating system is decrypted or started.

---

* It is not practical (and therefore is not supported) to install operating systems in two VeraCrypt volumes that are embedded within a single partition, because using the outer operating system would often require data to be written to the area of the hidden operating system (and if such write operations were prevented using the hidden volume protection feature, it would inherently cause system crashes, i.e. 'Blue Screen' errors).
† This does not apply to filesystems on CD/DVD-like media and on custom, atypical, or non-standard devices/media.

See also: **System Encryption**, **Hidden Volume**