

## **Using the open source TrueCrypt software system to provide on-the-fly encryption of storage media**

**Robin Snyder**  
**Savannah State University**  
**126 Jordan, P. O. Box 20359**  
**Savannah, GA 31404**  
**(912) 356-2716**  
**snyderr@savstate.edu**  
**<http://www.RobinSnyder.com>**

### **Abstract**

The need to encrypt information on hard drives, flash drives, laptops, etc., is becoming increasingly important for both personal and institutional use. After covering some basic security and encryption concepts, this paper (and talk) will discuss (and present) how the free and open source TrueCrypt software system can be used to provide encryption of information on either an entire drive partition or of a file that mounts as a virtual disk. TrueCrypt includes many advanced features such as plausible deniability, ways for the local computer staff to support users (who have lost their password access), easy installation, extensive documentation, etc. An add-on method that the author developed to make an encrypted drive useless without a corresponding USB drive key (and password) will be included.

### **Introduction**

It now happens often that some person or organization has lost information by the physical loss of an unencrypted storage device. Losing encrypted data on a storage drive is often not a problem. But losing information on a storage drive is definitely a problem. And that difference illustrates the difference between data and information.

The author defines data as "stuff". Anything can be data, but in the computer field data is usually represented as bits. Information, on the other hand, can be defined as data that is more valuable when interpreted with the proper insight, as a message sent by a sender to an intended receiver, etc. Thus data loss itself is not a problem. Data in the form of bits is an intangible entity that may take a tangible form. To see this, imagine a CD with 650MB of data bits. The bits could be arranged as information (e.g., Microsoft XP Pro, though some might argue about whether it is just data). Or, the bits could be in a form that is not recognized as anything useful (again, some people might view Microsoft XP Pro as such, but for the sake of argument, this line of thought will not be followed very far). An encrypted drive whose encryption cannot be broken is just data. However, with the proper insight (i.e., the decryption keys), the data can become valuable information. Thus, while information loss for which there is no backup can be catastrophic, data loss by itself is not a problem. But, data loss that can be interpreted as information might be catastrophic to the person or organization whose information has been compromised.

In previous decades, a floppy disk might hold a megabyte or so of data. In the mid-1990's the Zip disk moved portable storage to 100MB and beyond. Today, USB thumb drives are typically in the 1GB and beyond range. This leaves substantial room for a lot of valuable information. Lost

or stolen laptops, portable hard drives (100GB and up), flash memory drives (1GB and up), etc., provide additional avenues for information compromise. And humans are usually the weak link in the security chain. It should go without saying, but it will be said anyway, that information is today often more valuable than the computers and/or storage devices on which the information resides.

The key to information security, and security in general, is to make data appear as information to those to whom access is allowed and to make that same physical data appear as meaningless bits to others who might happen to get access to those bits (e.g., by stealing the storage device).

The obvious solution is to encrypt the data on the storage device. Most solutions to this problem are OTFE (On The Fly Encryption) that makes the encrypted drive appear as just another drive. Such access requires administrator privileges for installation but normal privileges for use. All standard tools, such as **chkdsk** (for Windows) can be used on such encrypted drives.

This paper discusses how the free and open source TrueCrypt software system can be used to provide encryption of information on either an entire drive partition or of a file that mounts as a virtual disk. Much of the material in this paper is from the author's previous work in this area [1].

### **USB devices**

USB memory devices range from flash memory, in the 1 GB range (i.e., 100MB to 10GB) while USB hard drive devices are in the 100GB range (i.e., 100MB to 1TB). Some, such as the Seagate 100GB 2.5 inch USB hard drive (8MB cache) are very compact and convenient, with no power brick, just a split USB cable to get the required power from the USB ports (about \$145 in Summer 2006). Other low-cost USB hard drives are larger capacity and larger size. 1GB USB memory devices are in the \$20 range (Summer 2006, Verbatim 1GB).

Many USB storage devices include encryption and decryption software. Most require administrator rights to run on a system. This might not be available (e.g., in student labs, Internet cafe's, etc.).

Lexar includes JumpDrive Secure.

*JumpDrive Secure is a software application that comes pre-loaded on the JumpDrive Secure allowing you to password protect files that are stored on the JumpDrive itself. It enables you to divide your JumpDrive into two different areas, or zones. The public zone has no password protection and is accessible by anyone using your JumpDrive. The private zone is password-protected so no one can open, copy or write files to it without entering the correct password. We go one step further than just password protecting your data, we also encrypt your data with 256-AES encryption.*

[http://www.buy.com/prod/Lexar\\_1GB\\_JumpDrive\\_Secure\\_USB\\_Flash\\_Drive/q/loc/101/10381529.html](http://www.buy.com/prod/Lexar_1GB_JumpDrive_Secure_USB_Flash_Drive/q/loc/101/10381529.html) [as of Fri, Jun 09, 2006]

Here are some comments.

*The secure partition feature is a nice feature, however it installs software on the computer to access the secure partition. If you put it on someone else's computer, you will see the software*

## 2007 ASCUE Proceedings

*running in the task manager and it will stay there even when you are done..not cool.*  
[http://www.buy.com/prod/Lexar\\_1GB\\_JumpDrive\\_Secure\\_USB\\_Flash\\_Drive/q/loc/101/10381529.html](http://www.buy.com/prod/Lexar_1GB_JumpDrive_Secure_USB_Flash_Drive/q/loc/101/10381529.html) [as of Fri, Jun 09, 2006]

*I get an error saying I must be an administrator to run the software whenever I plug it in. It has an auto-start program that wants to run every time you plug in the device. Included security and partitioning software uses over 4MB of the drive. I am using a company provided laptop, and have verified that I am logged in as a local administrator on the device, but perhaps I am not THE administrator. This seems to be an unnecessary feature of the device. After the error, I can access the non-encrypted drive, but have no access to the encrypted drive. I*  
[http://www.buy.com/prod/Lexar\\_1GB\\_JumpDrive\\_Secure\\_USB\\_Flash\\_Drive/q/loc/101/10381529.html](http://www.buy.com/prod/Lexar_1GB_JumpDrive_Secure_USB_Flash_Drive/q/loc/101/10381529.html) [as of Fri, Jun 09, 2006]

The problem seems to be that many are proprietary to company that produces the devices. Until a standard emerges, there will be quite a variety of such software systems. The author has reviewed a number of these and each seems to have many complaints from users. In general, many users avoid the supplied software and opt for some more general software solution.

### NTFS

The Microsoft NTFS file system offers an EFS (Encrypting File System). The author's experience has been that this is somewhat difficult to make work the way one wants it to work. The concept is simple enough. Just format a partition with NTFS file system and select the desired options.

However, moving a portable device from one location to another is, in general, not possible. Because the ID is based on the SID and not the login name, that SID may vary from machine to machine and system to system. And, a certificate in the root Windows directory is needed in addition to a password. In addition, portability is not easy. And anyone who has access to the machine can easily determine that there is an encrypted drive and may want to find out what is on it. There is no easy way to plausibly deny that there is encrypted information on the drive.

### PGP Disk

One popular drive encryption system has been PGPdisk. At one time, this was open source, and included with PHPfreeware, version 6.0.2.i. But, as of version 6.5, it is no longer included as it has since become commercial, available at <http://www.pgp.com/> [as of Fri, Jun 09, 2006]. It is now called "**PGP Whole Disk Encryption**". As such, the older open source version, though still available for noncommercial use, will gradually become out-of-date (i.e., through information entropy increase).

### TrueCrypt

A seemingly very well-engineered and free Open Source drive encryption solution that the author has successfully used is TrueCrypt, at <http://www.truecrypt.org/> [as of Fri, Jun 09, 2006]. The author's view is that confidence in the security of any encryption software should only be placed in open source software where the source code is available for everyone to inspect and study. Otherwise there is no easy way to be certain that there are no back doors or other hidden

features in the software. In addition, the nature of open source software means that, if the source is kept, in principle the binary software could be re-created and/or modified, if needed. This would not be the case for proprietary software.

From the TrueCrypt web site, here are some of the main features of TrueCrypt.

- Creates a virtual encrypted disk within a file and mounts it as a real disk.
- Encrypts an entire hard disk partition or a device, such as USB flash drive.
- Encryption is automatic, real-time (on-the-fly) and transparent.
- Provides two levels of plausible deniability, in case an adversary forces you to reveal the password:
  - 1) Hidden volume (steganography -- more information may be found here).
  - 2) No TrueCrypt volume can be identified (volumes cannot be distinguished from random data).
- Encryption algorithms: AES-256, Blowfish (448-bit key), CAST5, Serpent, Triple DES, and Twofish. Mode of operation: LRW (CBC supported as legacy).
- Conceived in 2003.

The web site, documentation, and performance of TrueCrypt is very good. There are very useful insights into security problems that can arise because of the way hardware and/or software works. For example encrypted data left in RAM, left in Windows paging files, available because of USB wear-leveling strategies, data left after drive fragmentation, etc. Each is beyond the scope of what the TrueCrypt software can control, but the information is useful for general security purposes.

As for wear-leveling, USB flash drive uses flash memory. It can only be written a certain number of times until it "wears out". The hardware in a USB flash drive moves the written sectors around so that the same sector is not written too many times. As the sectors "wear out", they are marked as not usable. For security, someone with low-level access to TrueCrypt's sectors (from the USB point of view) could get information that might compromise TrueCrypt's security.

Besides the plausible deniability features (i.e., hidden volumes, etc.), there are keyfiles. A key file can be (the first few thousand bytes of) any file that does not change. This is designed to hinder most keyloggers. There is automatic unmounting after a specified timeout, and locking the terminal or screen saver activation. There are command line options (discussed below) to automate the mounting and unmounting. And, the mouse movement can be used as part of the random number generation. A Linux version is available and work is underway on a Mac OS version.

A particularly useful feature, depending on need, is plausible deniability. Throughout the design, TrueCrypt has been engineered to make it difficult for an attacker to access. There is the pass-

## *2007 ASCUE Proceedings*

word, which can be up to 64 characters. Multiple keyfiles can be used to make it difficult for keyloggers. The file format appears entirely random. There are no markers, file directories, etc., that can be inferred from the encrypted file. The entire space for the file is filled with random data. There is no discernible file format. Nowhere is the encryption method stored. The software tries each of the allowable methods to see if any will work.

From a network administrative point of view, the password and keyfile(s) are used to find and decrypt the header file created when the volume was first created. This header file can be exported and saved in a secure place (e.g., by the IT support service). Changing the password and/or keyfile(s) by the user changes the access to the location of and decryption of the header file which decrypts the rest of the volume. Thus, it is possible to change the password and/or keyfiles for an encrypted volume without losing the original data, without going through a complete decrypt-with-old and encrypt-with-new process, and without the time requirements of an in-place changeover. If the user loses/forgets their password and/or keyfile(s), the administrator can recover the original header and decrypt the volume, setting it to the original header, password, and keyfile(s).

The primary disadvantage of TrueCrypt appears to be that administrator rights are needed for installation. Thus, the "**Traveler mode**" (e.g., for cafe use, school use, etc.) will not work if TrueCrypt is not already installed on the computer to be used. It should work in a student lab, however, if the IT staff has installed TrueCrypt on the lab computers. Traveler mode stores some configuration information in an XML file, but TrueCrypt does not use the Windows Registry.

### **Plausible deniability**

Plausible deniability means being able to say "I didn't know that" and not be able to be proven wrong. One might want "plausible deniability" so that one cannot be held accountable. For example:

- The government says: You have secret information on that drive.
- You want to say: No I don't.
- Government has to now "prove it".
- How will the government "prove it"?

The government now needs to find the "secret information" on the drive. Computer forensics looks at what is on the drive. The FBI can read a hard drive after many formats. But, if it was encrypted, it may not be possible.

The government wants you to provide the password. So, you have to provide the password, or you may be found guilty, or in "contempt of court". Once the password is provided, there is nothing on the drive that is found to be "illegal". With TrueCrypt, they can't determine if there is a hidden drive within the drive. TrueCrypt plausible deniability makes it practically impossible to determine if there is a "hidden" drive in an encrypted drive.

### **Using TrueCrypt**

Here are a few details on using TrueCrypt. For more information, see the TrueCrypt web site or do a Google search as there are a number of step-by-step tutorials with screen shots available.

Although the GUI (Graphical User Interface) can be used to mount and unmount volumes, this can also be done from the command line or from a batch file. Note that the initial creation of the volumes should be done from the GUI. Do not lose the password and/or keyfile(s) as there is no way to recover them. Recovery can also be done by backing up the header file information to a secure place.

By placing an **autorun.inf** file in the root of the drive, with autoplay enabled, TrueCrypt can be set to automatically run when the storage device is plugged into the USB port. Usually, part of the drive is saved for the TrueCrypt files (a few megabytes). Here is an example **autorun.inf**.

```
[autorun]
label=Seagate USB Drive
icon=truecrypt.exe
```

### Traveler mode

The TrueCrypt software has a traveler mode that includes an **autorun.inf** that makes mounting USB drives easier. For example, the following **autorun.inf** file might be created.

```
[autorun]
open=TrueCrypt\TrueCrypt.exe /q /a /e /m rm /v "\mydisk.tc"
shell=mount
action=Mount TrueCrypt Volume
shell\open\command=TrueCrypt\TrueCrypt.exe /e /m rm /v "\mydisk.tc"
shell\open=TrueCrypt Start
shell\mount\command=TrueCrypt\TrueCrypt.exe /q /a /e /m rm /v
"\mydisk.tc"
shell\mount=TrueCrypt Mount
shell\dismount\command=TrueCrypt\TrueCrypt.exe /q /d
shell\dismount=TrueCrypt Dismount All
```

Note again that administrator privileges are needed to install TrueCrypt, though administrator privileges are not needed to then run TrueCrypt. This is, in part, to allow for drive mounting as an encrypting file system. An encrypting file system then provides all of the features that any file system would have such as running ScanDisk, running the defragmentation program, sharing as a network drive, etc.

Here is a series of batch commands to mount file **H:\vol1.tc** as a volume using drive letter **Z:**. The password is **password** and the keyfile (more than one can be used) is **keyfile**.

```
truecrypt.exe /v H:\vol1.tc /l Z /p password /k keyfile /m rm /q
/s
```

Here is a series of batch commands to dismount a volume.

```
truecrypt.exe /q /d Z
```

## *2007 ASCUE Proceedings*

These commands can be put into batch files `mount.bat` and `umount.bat` to mount and unmount volumes. The author started using this method but soon decided to encapsulate the needed functionality in an executable program named `rmsinit.exe`.

One issue when using the command line is that the command line arguments contain the password. The command line is available in the current running services list via the Windows API. That could be a security problem. It seems that this information is available when TrueCrypt is first started but not when it is called once started. Thus, it seems better to first start TrueCrypt (i.e., automatically) then make the calls to TrueCrypt to automate from the command line. Note that this information is available via explicit programming but not necessarily from commonly used software such as the Windows "**Task Manager**".

### **Summary**

This paper has discussed how the free and open source TrueCrypt software system can be used to provide encryption of information on either an entire drive partition or of a file that mounts as a virtual disk.

### **References**

- [1] Snyder, R. (2006). Some security alternatives for encrypting information on storage devices 2006 Information Security Curriculum Development conference (September 22-23, 2006), Kennesaw, GA.