Evaluation of TrueCrypt
as a
Mobile Data Encryption Solution

Tom Olzak
April 2008

Introduction

Protecting data on mobile devices is not an option. Every security manager knows this can be a hole in an organization's security framework. The best way to protect data on the move is to encrypt them. However, providing the right tools is not an easy task—especially when cost is an issue. Any tool must be easy to use and one most if not all users are willing to integrate into their daily routines. TrueCrypt, on the surface, seemed to meet these criteria.

I installed and tested TrueCrypt from the perspectives of user and security manager. The results of that test, and my conclusions about the value of TrueCrypt as a mobile data encryption solution, are contained in this paper.

What is TrueCrypt?

TrueCrypt (truecrypt.org) is an open-source encryption solution provided by the TrueCrypt Foundation. It isn't new to the market. Version 1 was released in February of 2004, with version 5.1a released in March of 2008. According to the TrueCrypt Web site, this free encryption product provides the following:

- Creates a **virtual encrypted disk** within a file and mounts it as a real disk.
- Encrypts an **entire partition or storage device** such as USB flash drive or hard drive.
- Encrypts a **partition or drive where Windows is installed**.
- 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:
    - **Hidden volume** (steganography).

- No TrueCrypt volume can be identified (volumes cannot be distinguished from random data).

➢ Encryption algorithms: AES-256, Serpent, and Twofish. Mode of operation: XTS.

The well-written, 110 page user guide contains many more features and functions. For the purpose of this paper, I focus on functionality that can protect laptops, flash drives, iPods, and other personal mobile storage devices, i.e., basic containers and volume encryption using password protection.

Creating an Encrypted Volume

The first test I conducted was TrueCrypt's ability to encrypt sensitive information on a laptop's local drive. After downloading and installing it on my Windows XP laptop, I opened TrueCrypt. Figure 1 shows the main management window.

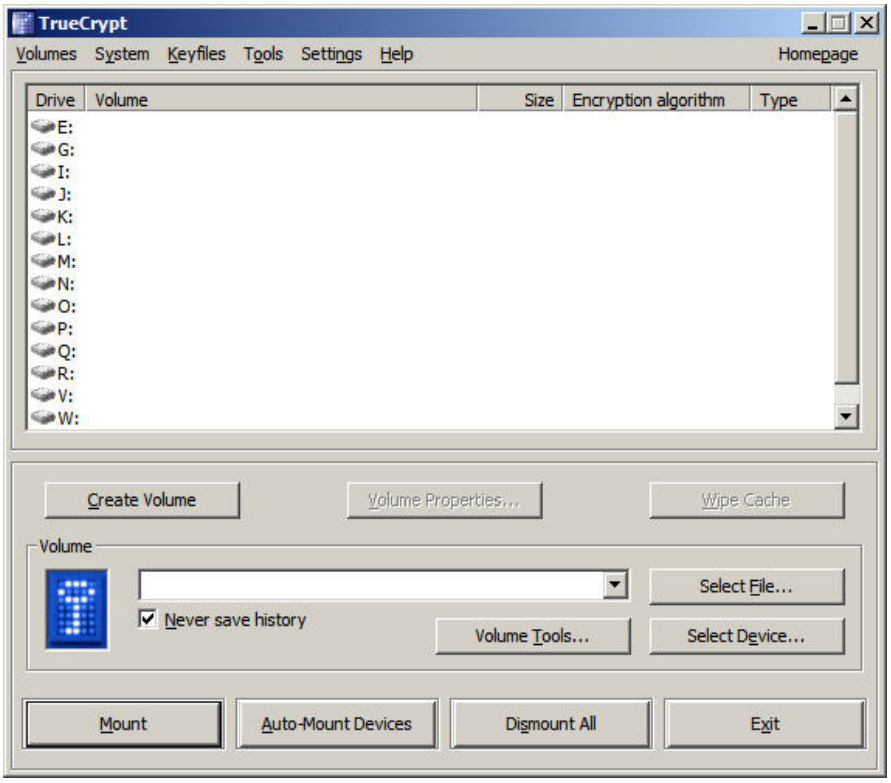

Figure 1: TrueCrypt Management Window

Available drives are listed at the top with TrueCrypt volume management buttons at the bottom.

The first step in encrypting information with this solution is creation of a TrueCrypt volume. I clicked the *Create Volume* button to start the wizard. The window in Figure 2 appeared.

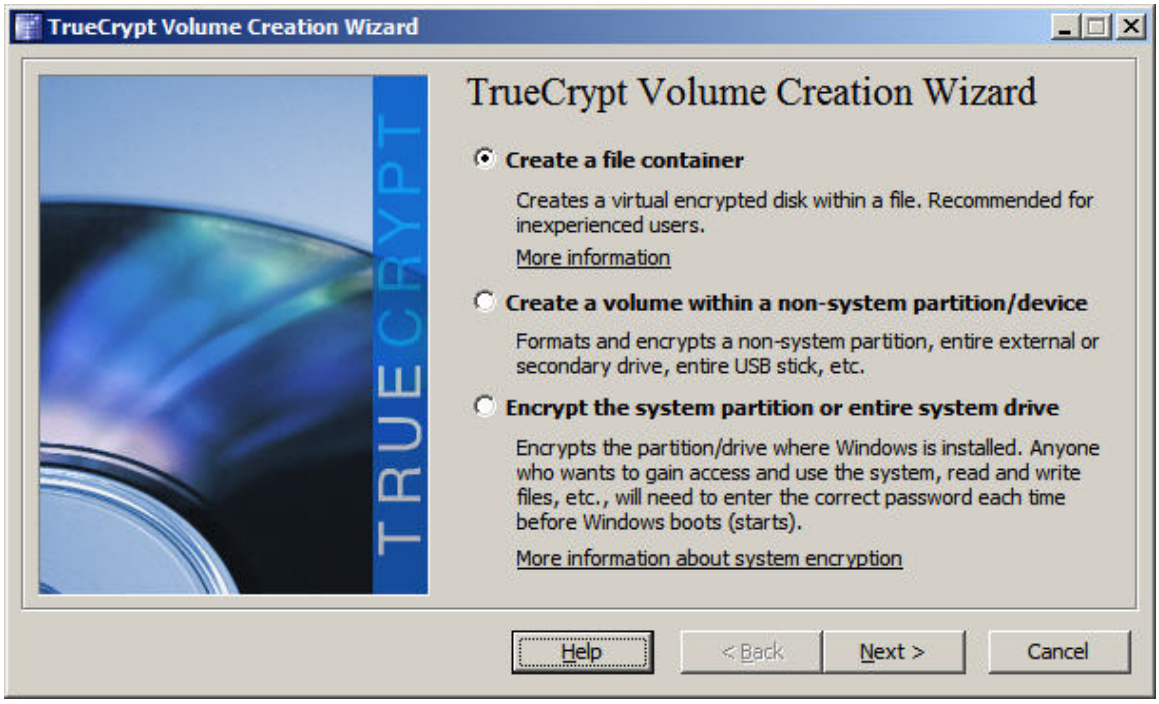

Figure 2: Create Volume, Step 1

The three options allow significant flexibility. Before continuing the process, let's take a moment and look at the functional, pros, and cons of each.

Create a File Container

A TrueCrypt file container looks like any other file when viewed via Windows Explorer. It also acts like a file, capable of being copied, deleted, and moved. The difference is that when you mount it to a drive letter, it looks like a normal storage volume. Anything you place into the mounted volume is encrypted and stored in the container. If the container is moved or copied, the files stay encrypted.

The upside of file containers is flexibility. You can create them on almost any media (optical disk has some exceptions) and open them on any supported platform (Windows, Mac OS, and Linux). The downside is the need for users to actually write the sensitive files to a mounted container. This disadvantage is resolved by using one or both of the next two options.

Create a Volume within a Non-system Partition/Device

The non-system partition/device option allows you to encrypt an entire storage device. For example, you could encrypt a non-system laptop volume or an entire flash drive. One big caveat, don't do this unless you've backed up all files on the volume. They will be deleted during the encryption process.

The advantage of this solution is information written anywhere in the volume is encrypted. Users do not have to be relied upon to do the right thing. The biggest disadvantage is caused by potential configuration decisions, i.e., not encrypting the

system partition containing the paging file, hibernation file, and folders into which users drop stuff when in a hurry. The desktop is a popular catchall.

Encrypt the System Partition or Entire System Drive

Unlike the non-system partition encryption process, the third option, encrypting the system partition, does not wipe the disk clean. The TrueCrypt manual recommends it as the best way to secure laptop data. The following is from the TrueCrypt manual:

> "System encryption provides the highest level of security and privacy, because all files, including any temporary files that Windows and applications create on the system partition (typically, without your knowledge or consent), hibernation files, swap files, etc., are always permanently encrypted (even when power supply is suddenly interrupted). Windows also records large amounts of potentially sensitive data, such as the names and locations of files you open, applications you run, etc. All such log files and registry entries are always permanently encrypted as well."

So to get the best results, encrypt both the system and non-system partitions. Encrypting the system partition also lets you force pre-boot authentication. Also from the manual,

> "System encryption involves pre-boot authentication, which means that anyone who wants to gain access and use the encrypted system, read and write files stored on the system drive, etc., will need to enter the correct password each time before Windows boots (starts). Pre-boot authentication is handled by the TrueCrypt Boot Loader, which resides in the first cylinder of the boot drive and on the TrueCrypt Rescue Disk…"

TrueCrypt requires the creation of a rescue disk during the system partition encryption process. We'll look at the importance of header backups later.

The Volume Creation Wizard

Now let's continue with the wizard. Since I wanted to verify the functionality of a container, I selected *Create a file container* and clicked *Next*. I was asked if I wanted a standard or hidden volume. Hidden volumes are interesting, but outside the scope of this paper. I accepted the default Standard TrueCrypt Volume and once again clicked *Next.*

This brought up a window asking which encryption and hash algorithms I wanted to use. See Figure 3. I clicked *Next*, accepting the defaults, and moving to a prompt for container size. See Figure 4. You can specify a container size up to 1 PB (1,048,576 GB). I decided on a 1 GB container, and moved to the next and final step in the container creation process, depicted in Figures 5 and 6.

I accepted the default FAT file system, and followed the screen instructions to move my mouse as "randomly as possible." As I moved my mouse, the Random Pool value continuously changed. Clicking *Format* resulted in the assigning of a header key and a master key, and the formatting of the container, as shown in Figure 6.

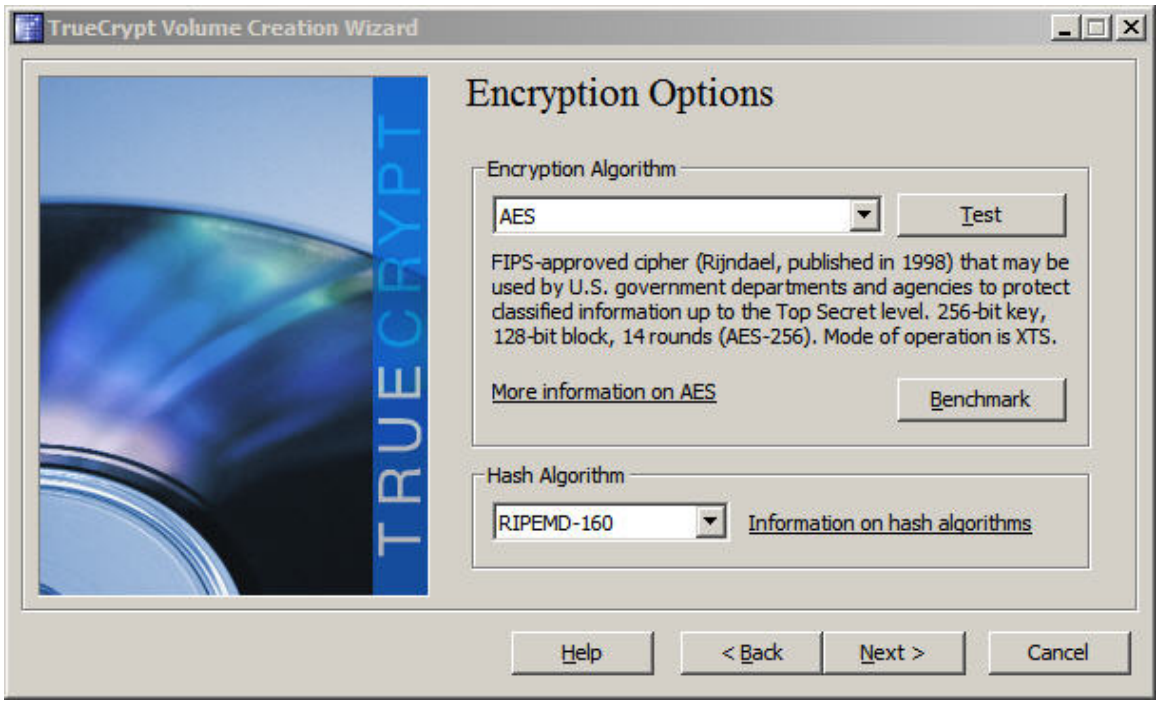

Figure 3: Select Encryption Algorithm

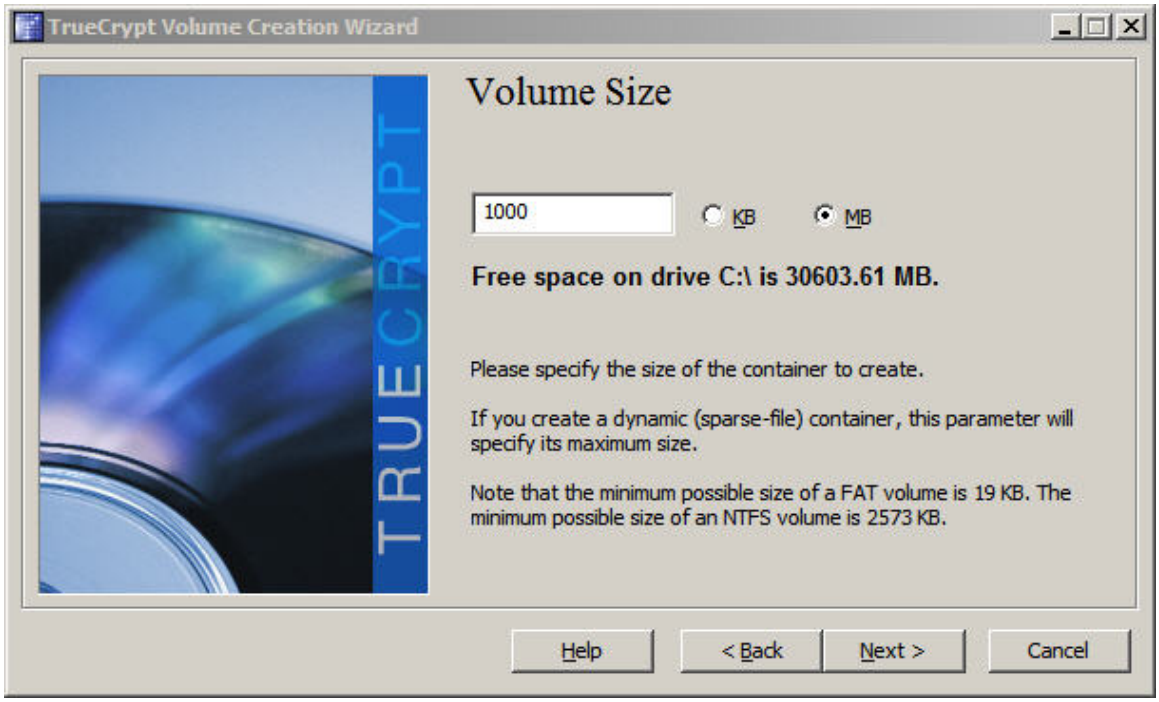

Figure 4: Specify Volume Size

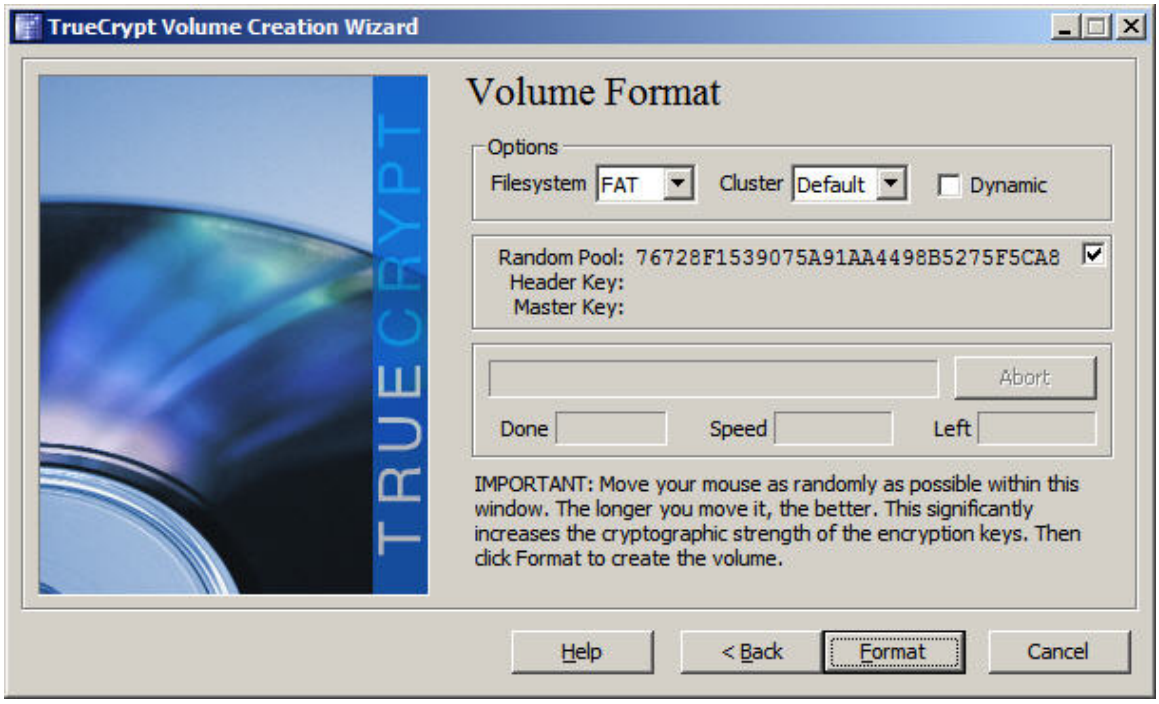

Figure 5: Volume Format -- Key Selection

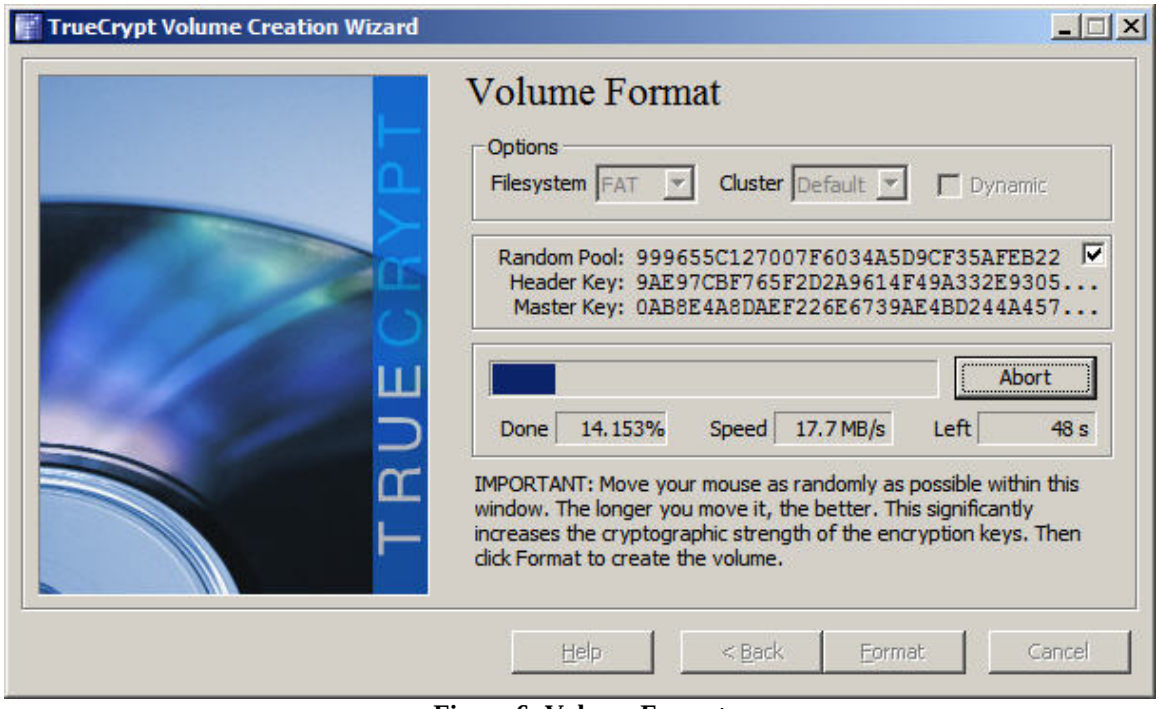

Figure 6: Volume Format

Before you can use a container, you must mount it and assign a drive letter. This is also done from the TrueCrypt management window. I'll describe the mounting process as part of the full partition/device encryption process, described in the next section.

Full Partition/Device Encryption

The same wizard used to create a container is used to encrypt an entire partition or flash drive. There are a couple of differences once you get past the decision whether to create a standard or hidden volume. You're asked to select a device to encrypt instead of a file. A list of devices is provided when clicking the Select Device button. The list in Figure 7 shows my system volume and a flash drive, highlighted in blue.

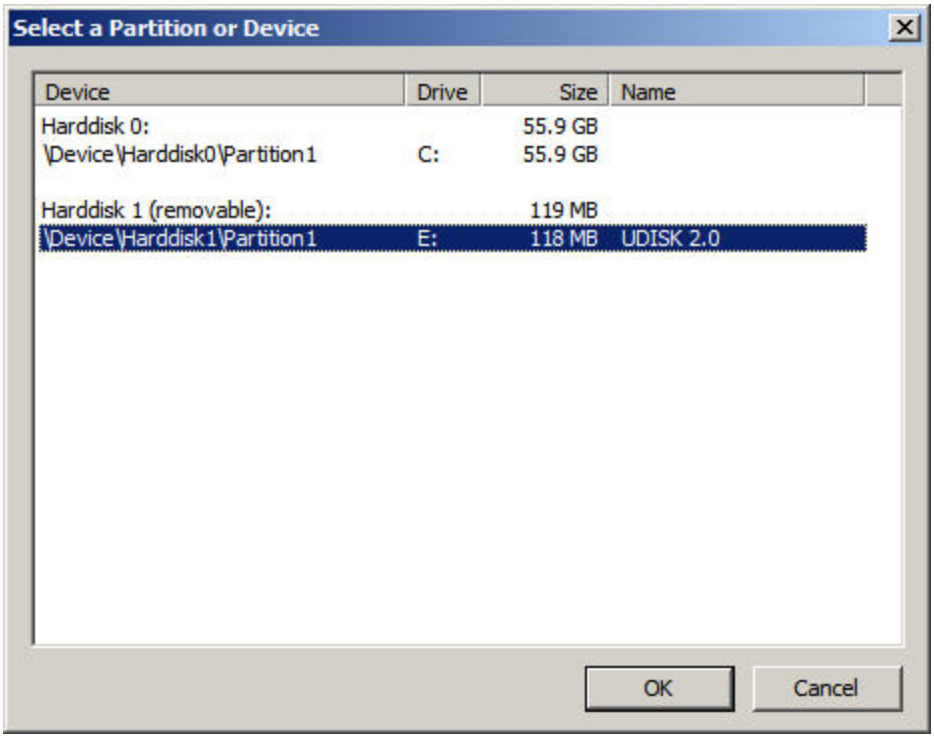

Figure 7: Select Device to Encrypt

This is a 128 MB flash drive I inserted into one of my docking station USB ports. TrueCrypt sees it has a hard drive. After clicking OK, I was prompted to select my encryption method and format the volume.

Now it was time to put the volumes to work.

Using TrueCrypt Volumes

Manually Mounting Volumes

To use my file container and full encrypted partition/device, I had to mount them. Volume mounting is also done via the TrueCrypt management window. See Figure 8.

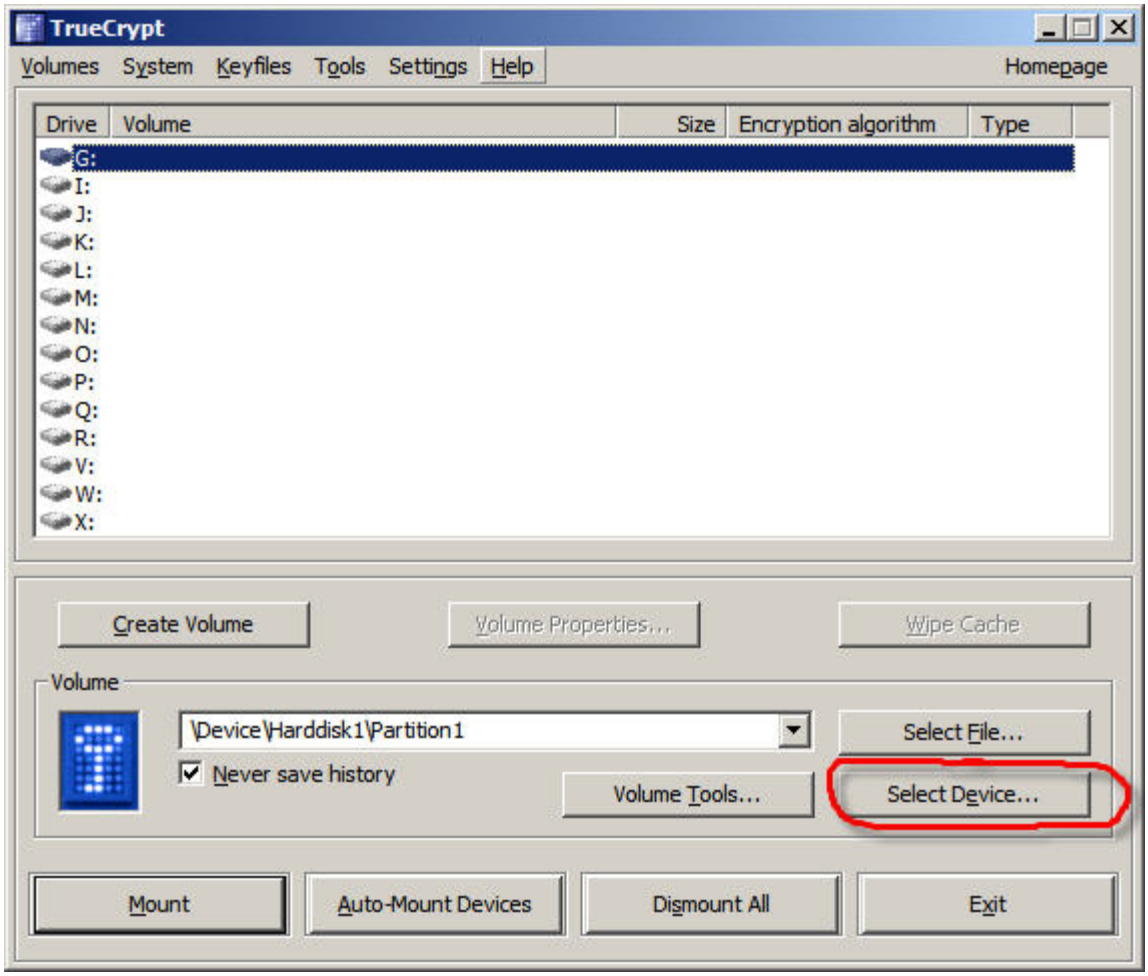

Figure 8: Mount Device

After selecting the drive letter I wanted to use (in this case G), I had the option of selecting a file container or selecting a device. I chose *Select Device* and clicked on my flash drive. It appeared in the Volume field as shown above. I then clicked *Mount*, and TrueCrypt displayed a password prompt, as depicted in Figure 9. Within a second or two, the flash drive was mounted and the results displayed, as in Figure 10.

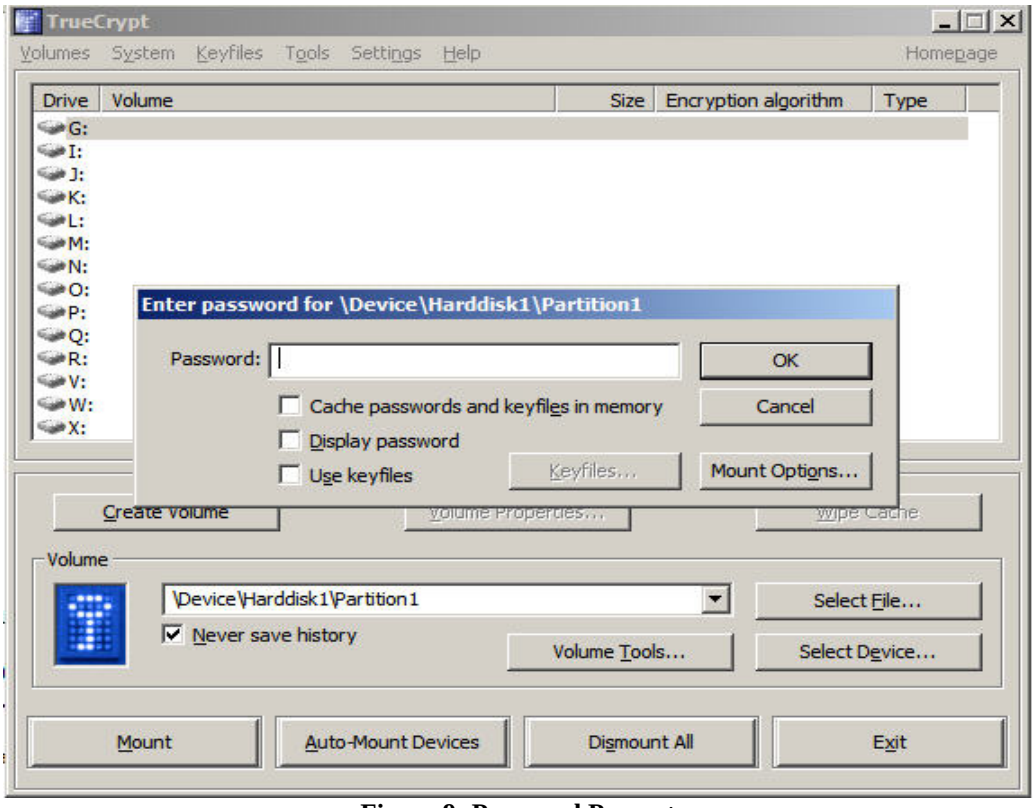

Figure 9: Password Prompt

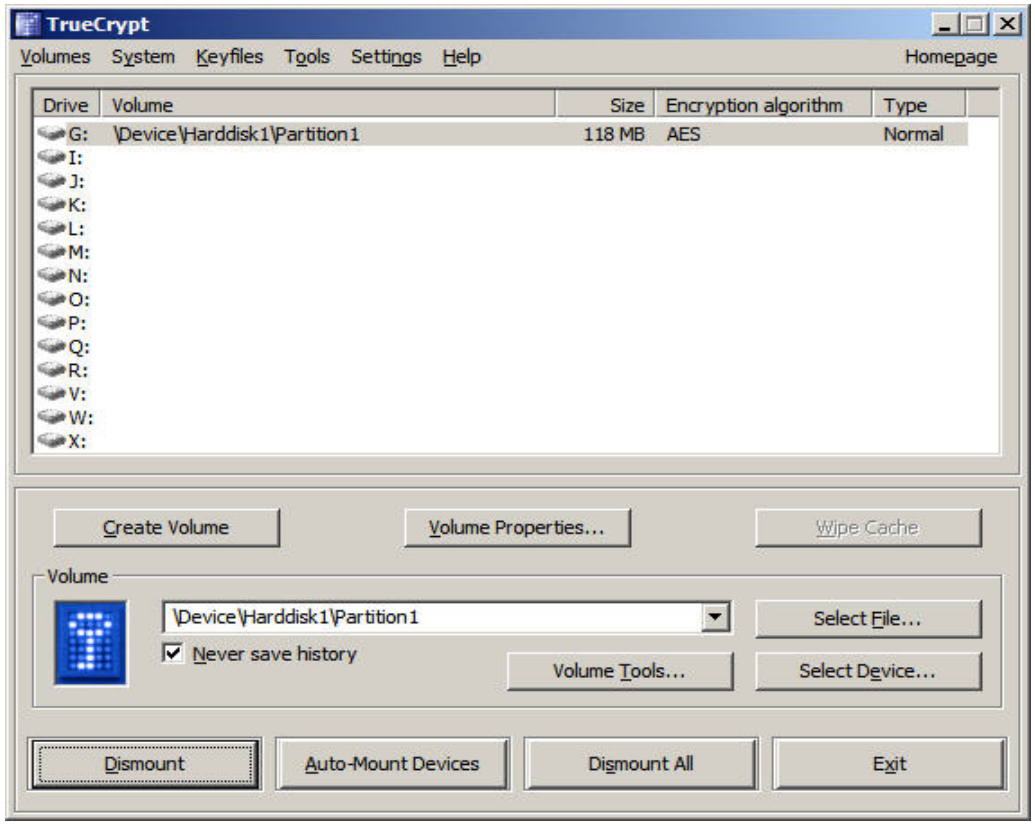

Figure 10: Mounted Volume

To test TrueCrypt's ability to nest encrypted containers within encrypted device volumes, I proceeded to create a file container on the flash drive, drive G. Nested containers allow users to create encrypted content that they can move or copy to various locations without losing encryption protection. I mounted it as drive E. Figure 11 shows how these mounted TrueCrypt volumes appeared in Windows Explorer. I was able to manage these volumes, and any new volumes I might want to create, via the TrueCrypt icon that appeared in the system tray.

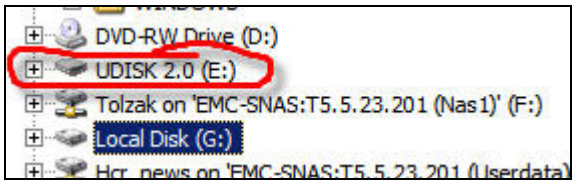

Figure 11: TrueCrypt Volumes in Windows Explorer

Mounting volumes manually is no problem for us technical types, but business users might elect not to use encryption if they have to remember to mount drives (assuming they understand what mounting a drive means).

Auto-mounting Volumes

TrueCrypt provides three methods of auto-mounting volumes. One is to auto-mount devices previously identified as "favorites." I identified my favorite volumes by bringing up the TrueCrypt management window. As shown in Figure 12, both my flash drive (G) and the container on the flash drive (N) were mounted.

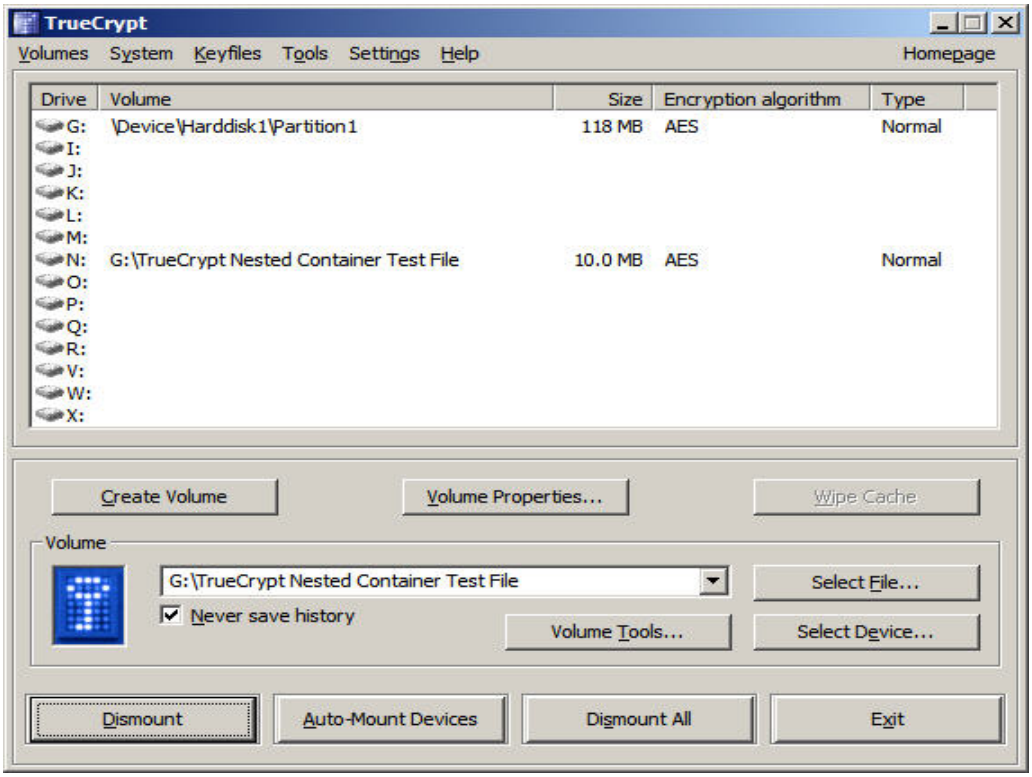

Figure 12: Mounted Volumes

From the menu bar, I selected *Volumes-->Save currently mounted volumes as favorite.* I then dismounted the drives and selected *Volumes-->Mount Favorite Volumes*. After I entered my password, both volumes were remounted within a few seconds. I could also right-click on the TrueCrypt icon in the system tray and mount my favorites. However, this still seemed to be too much to ask of my users.

The second way to mount volumes is via the command line. This provides for scripted volume mounting that is automatic and invisible to the user, except for the password prompt. Figure 13 shows the contents of the .BAT file I created to test this approach. Note that I mounted the flash drive first, as drive G (/lG). This worked great and is my preferred method of providing standardized encryption functionality with TrueCrypt.

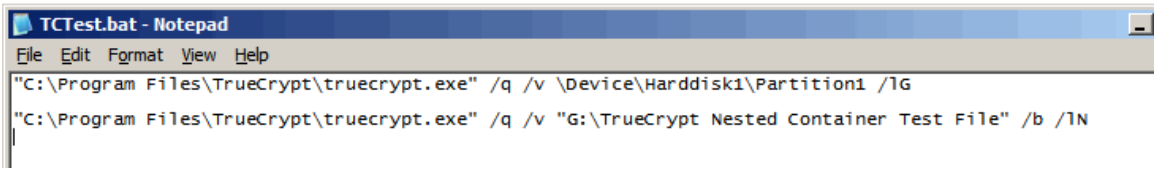

Figure 13: Scripted Volume Mount

Finally, I could have simply requested TrueCrypt to auto-mount all devices. It looks for all volumes, tests whether they're encrypted with TrueCrypt, and applies the password entered when the process begins. I couldn't get this to work for nested containers, and it takes a long time to complete. This doesn't appear to be a good option for most users.

Other Features

TrueCrypt Mobile Installation

When transporting data on a flash drive or other handheld storage device, it's often not convenient, or possible, to install TrueCrypt on the available desktop machines. This is not a problem if the complete handheld device is not a TrueCrypt volume. Using TrueCrypt's Traveler Disk Setup, TrueCrypt.exe is installed on the mobile device, enabling you to mount and access encrypted containers, without installing TrueCrypt on the personal computer used to access the protected information.

If desired, you can configure TrueCrypt to run automatically with, for example, a flash drive containing protected data, inserted into a PC running a supported OS.

Key Files

TrueCrypt supports the use of keyfiles to, according to the manual,

- Provide protection against keystroke loggers (even if an adversary captures your password using a keystroke logger, he will not be able to mount the volume without your keyfile).
- Potentially improve protection against brute force attacks (significant particularly if the volume password is weak).

- Allow for managing multi-user *shared* access (all keyfile holders must present their keyfiles before a volume can be mounted).

I didn't test keyfiles, because I don't believe they provide a manageable solution for mobile encryption. Keeping track of passwords is bad enough without having to help users figure out what they did with their keyfiles. If you disagree, the TrueCrypt manual provides excellent instructions for their use.

Encrypted Data Recovery

Users can't always remember their passwords, and terminated employees might conveniently misplace theirs. However, TrueCrypt provides the means to recover the encrypted information—from both local disk and mobile storage devices. I didn't test recovery of local disk. But the process for is similar.

In order to recover a container when a password is lost or the container/volume header is damaged, you must perform a volume header backup. Since you can't rely on most users to perform a header backup when creating a new container or volume, this might not be an effective recovery option. If most mobile devices contain information copied from local or network disk, this might not be an issue. However, telling someone who took a file home on a flash drive, performed extensive modifications, and now can't access it, that there's nothing you can do is career-limiting if expectations are not properly set.

To enable system volume recovery, TrueCrypt adds an additional dimension—the rescue disk. Creating a rescue disk is required when preparing the encryption of a system partition or drive. According to the manual, a rescue disk can save you from catastrophic failures of the encryption/decryption process, including,

- Failure of the TrueCrypt Boot Loader at system startup;
- Master key or other critical data corruption; and
- Malware infection in the TrueCrypt Boot Loader.

Booting the failing system from the rescue disk bypasses the contents of the first drive cylinder, which contains the original boot loader or boot manager overwritten by TrueCrypt. The TrueCrypt password is still required.

Conclusions

TrueCrypt is an outstanding encryption solution for anyone familiar with managing volumes and a slight knowledge of encryption technology. For the rest, it can be a bit daunting. Any organization planning to deploy TrueCrypt as a mobile data protection solution must consider the cost and logistics of training and supporting users, managing versions, and recovering damaged volumes. Like any security solution, assessing risk associated with how TrueCrypt will be used is necessary before rolling it out to your mobile-worker population.

In any case, I use TrueCrypt to protect data on my iPod and my flash drives. I find that it's easy to use and works as advertised. The documentation is better than I expected for an open-source encryption product. Overall, I encourage you to take a look.

Tom Olzak, MBA, CISSP, MCSE, is President and CEO of Erudio Security, LLC.

He can be reached at tom.olzak@erudiosecurity.com
Check out Tom's book, Just Enough Security
Additional security management resources are available at http://adventuresinsecurity.com
Free security training available at http://adventuresinsecurity.com/SCourses

---