Detect TrueCrypt and Veracrypt volumes - Raedts.BIZ | IT SECURITY & FORENSICS

Nick ⋮ 8-10 minutes ⋮ 2/12/2018

-
-
-

Some time ago I talked about BitLocker forensics and the decryption of BitLocker encrypted volumes. As a result, I received a few questions and a request regarding TrueCrypt encrypted volumes.
Encryption, in general, is quite the challenge for forensic investigators. When well executed, an encrypted volume is inaccessible without the proper keys. TrueCrypt is no exception. However, while a BitLocker volume has a nice header indicating it's encrypted, a TrueCrypt volume appears to be random data.
In this post I try to answer the question: "*how do you detect a TrueCrypt or Veracrypt volume*"?

Table of Contents

Detection methods

I suggest reading my post about TrueCrypt and Veracrypt (Link) before reading this article, it explains the basics about the software and why it's so hard to detect.

When you create an encrypted volume using TrueCrypt or VeraCrypt it is stored as a file (container) on your hard drive. The problem is that these files are designed to be hidden, and won't have an identifiable signature (header or footer). Therefore unless the encrypted volume is named "MyEncryptedVolume.tc" you won't be able to quickly identify these files. There are some telltale signs of encrypted volumes, a single 200GB file is quite suspicious, but unless you are able to decrypt this file and access it's volume it's nothing more than a 200GB file. If the person creating the encrypted volume knows what he or she is doing they might want to store their encrypted volume along with some other large files, let's say, the level files of a computer game. Unless you know what you are looking for you might overlook this file.

Luckily investigators have come up with several ways to identify possible encrypted volumes. The most common 3 things to look out for are:

– File size

– File signature

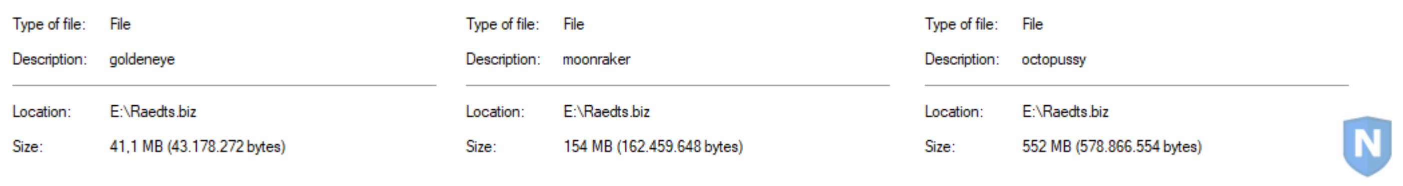
– File entropy

File size

The first and most common technique to find encrypted volumes is by sorting all files on a system by size. Commonly an encrypted volume is a single large file stored on some random location. Finding a large 50GB file called "backup.bak" or "Archive.pst" might be a reason to suspect an encrypted volume, checking the file signature will confirm if the pst file is actually a pst file or something else.

When creating an encrypted volume, users are asked to enter the size of the new volume. The user is offered to enter the size in GB, MB or KB. Most users take a nice round number like 200MB, this means a large file of exactly 200MB is suspicious. Both TrueCrypt and VeraCrypt create containers that always are nice blocks of 512 bytes in size. Because of this, the size of all containers is dividable by 512.

Example:



Type of file:	File
Description:	goldeneye
Location:	E:\Raedts.biz
Size:	41,1 MB (43.178.272 bytes)

Type of file:	File
Description:	moonraker
Location:	E:\Raedts.biz
Size:	154 MB (162.459.648 bytes)

Type of file:	File
Description:	octopussy
Location:	E:\Raedts.biz
Size:	552 MB (578.866.554 bytes)

Here I have taken 3 random files to test the file size check on. If one of them is actually an encrypted volume created with TrueCrypt or VeraCrypt we should be able to divide it by 512.

goldeneye

43.178.272 bytes / 512 = **84.332,5625**

Probably not an encrypted volume

moonraker

162.459.648 bytes / 512 = **317.304**

Possibly an encrypted volume

octopussy

578.866.554 bytes / 512 = **1.130.598,73828125**

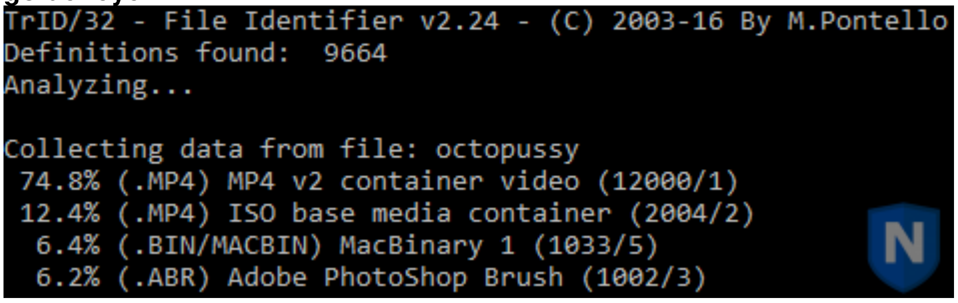
Probably not an encrypted volume

Here we see that moonraker, while it's size look rather random, is dividable by 512, and might be an encrypted volume. Using this test alone wil leave you with a lot of false-positives, a lot of normal files will be dividable by 512.

File signature

Since encrypted volumes seem to contain random data, they don't have an identifiable file signature. This means when we try to identify the file it won't match any known header.To try to identify what kind of files the 3 previously mentioned files are we will use Marco Pontello's excellent TrID tool (homepage).

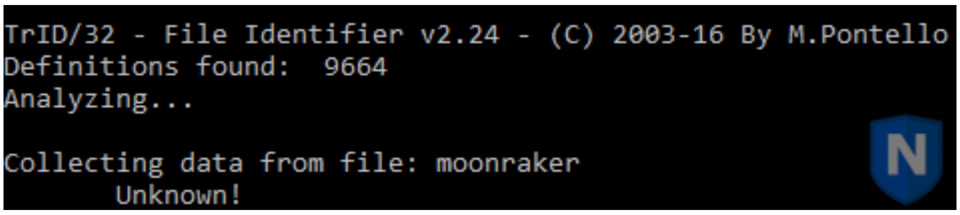
goldeneye



```
TrID/32 - File Identifier v2.24 - (C) 2003-16 By M.Pontello
Definitions found:  9664
Analyzing...

Collecting data from file: octopussy
 74.8% (.MP4) MP4 v2 container video (12000/1)
 12.4% (.MP4) ISO base media container (2004/2)
  6.4% (.BIN/MACBIN) MacBinary 1 (1033/5)
  6.2% (.ABR) Adobe PhotoShop Brush (1002/3)
```

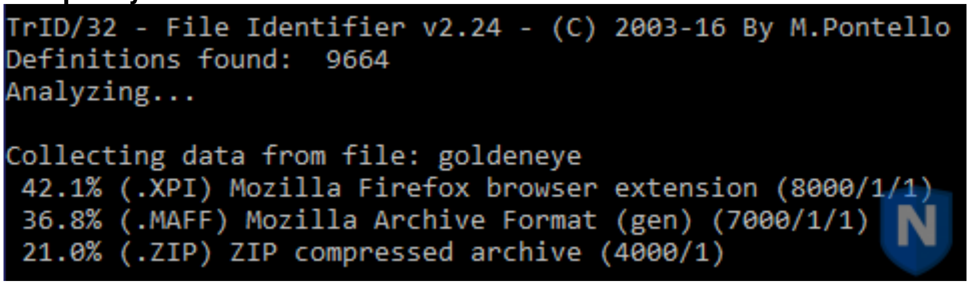
moonraker



```
TrID/32 - File Identifier v2.24 - (C) 2003-16 By M.Pontello
Definitions found:  9664
Analyzing...

Collecting data from file: moonraker
       Unknown!
```

octopussy



```
TrID/32 - File Identifier v2.24 - (C) 2003-16 By M.Pontello
Definitions found:  9664
Analyzing...

Collecting data from file: goldeneye
 42.1% (.XPI) Mozilla Firefox browser extension (8000/1/1)
 36.8% (.MAFF) Mozilla Archive Format (gen) (7000/1/1)
 21.0% (.ZIP) ZIP compressed archive (4000/1)
```

As you can see TrID was unable to identify the moonraker file adding to the suspicion that this file might be encrypted volume. Let's move on to the next and final test.

Entropy

An encrypted volume will look like random data, so random that it's identifiable by doing an entropy test. An entropy test essentially measures the predictability of any specific character in the file (details: http://rosettacode.org/wiki/Entropy). While lots of files can get a high entropy or "randomness" score, encrypted volumes will always get the highest score. Most forensic tools like FTK have an entropy test build-in, enabling this during evidence processing will allow you to quickly sort the files by their entropy score and easily identify suspicious files. We will perform the entropy test using the free sigcheck tool from Sysinternals.

goldeneye: 7.999

moonraker: 8

octopussy: 7.999

As you can see, all files got a high entropy score, indicating they all are quite random. If we look at the file signature results we see that goldeneye might be an MP4 file and octopussy a zip archive. Both are compressed formats and will result in a high entropy score. Our suspected encrypted volume moonraker got the highest score indicating this might, indeed, be an encrypted volume.

If we combine the results of all three of the tests we get the following result:

goldeneye
File Size: Failed
File Signature: Failed
File Entropy: Failed

Conclusion: This file probably isn't an encrypted volume

moonraker
File Size: Passed
File Signature: Passed
File Entropy: Passed

Conclusion: This file might be an encrypted volume!

octopussy
File Size: Failed
File Signature: Failed
File Entropy: Failed

Conclusion: This file probably isn't an encrypted volume

Python script

When performing a forensic investigation it's important to know where to look for. I have found that during my investigations I was able to reliably identify TrueCrypt volumes using the method described above.
To speed up the process I have created two python scripts that automatically perform these 3 tests.

Script 1: TC-Detective
This script scans the entire hard drive and performs the tests on all files, it generates a list of potential TrueCrypt volumes.

Click here to go to the download page for this script

Script 2: TC-FileDetective
This script performs the 3 tests on a file of your choice, a quick way to check if the file you suspect to be an encrypted volume passes the size, signature and entropy test.

Click here to go to the download page for this script

About these scripts:
The Entropy test is based on this post by : http://blog.dkbza.org/2007/05/scanning-data-for-entropy-anomalies.html It might not be perfect, but it does its job rather well, all files scoring above 7.9 are considered to be a potential encrypted volume.
These scripts also perform a very rudimentary signature check by checking if one of the 137 included headers appear within the first 32 bytes of the file. It's a good way to filter out most false-positives but it might also lead to a false-negative since it doesn't check the offset.

Testing the scripts against the samples

I have tested my scripts against the samples in this article, here are the results:

TC-Detective output:

```
TC-Detective Logfile generated on 11-02-2018 20:26:35

Possible encrypted volume: "E:\Raedts.biz\moonraker"

TC-Detective completed scan on 11-02-2018 20:26:48
```

It took the script 13 seconds to scan an entire drive (204GB of data, 62.919 files, SSD).
It correctly identified the moonraker file as a encrypted volume, this was the only VeraCrypt volume stored on that drive.

TC-FileDetective output:

Analyzing the 3 samples with this script yields the following results:

goldeneye

```
=================================
TC-FileDetective
=================================
file: goldeneyeResults
Header check: Failed (pages,key,numbers,epub,zip,jar)
Division check: Failed (288)
Entropy test: Failed (7.7258588463331055)
```

moonraker

```
=================================
TC-FileDetective
=================================
file: moonrakerResults
Header check: Passed (unknown format)
Division check: Passed (0)
Entropy test: Passed (7.910181626323565)
```

octopussy

```
=================================
TC-FileDetective
=================================
file: octopussyResults
Header check: Failed (m4v)
Division check: Failed (378)
Entropy test: Failed (7.221603803290186)
```

Decryption

Using the script we were able to confirm that moonraker might indeed, be an encrypted volume. In my next post, we will be looking at ways to decrypt this volume.

Post Views: 38,963