Name: Cristian Barreno

Encoding vs Hashing vs Encryption

EXERCISE 2 - Hashing

Task 1

```
echo hello world > file1.txt
echo hello world > file2.txt
```

Please note that the MD5 hash of the outputs are identical to each other.
That means when you created the files, they were identical, so were their hashes.
This proves the deterministic nature of using the same hash function, which in this case was md5.

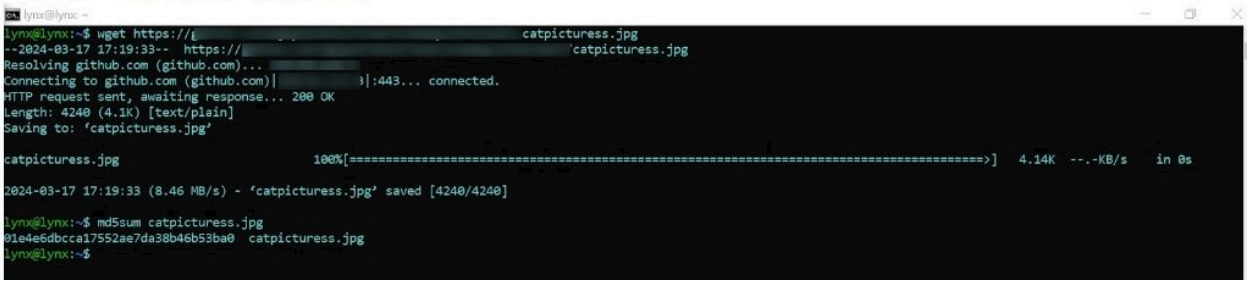
Task 2

```
echo hello world! > file3.txt
md5sum file3.txt
```

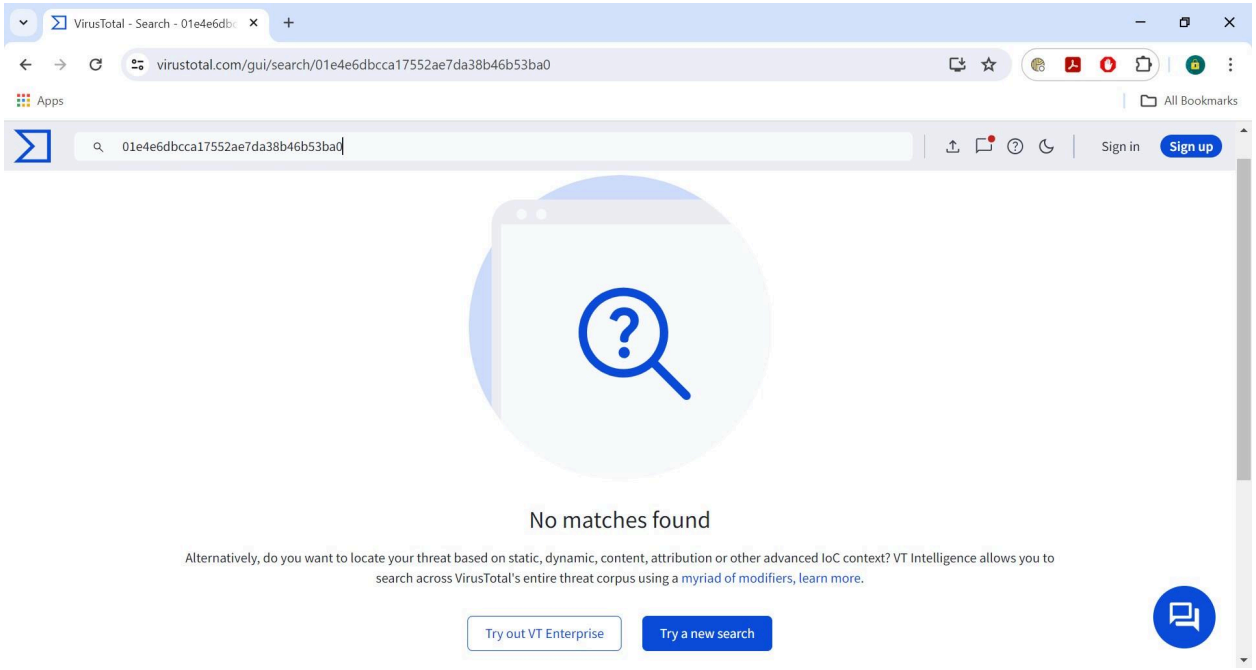
Please note that by inputting different text into file3, that the hash is wildly different that the files in Task 1. This is due to the Avalanche effect in Hashing.

Task 3

```
wget https://[redacted]catpicturess.jpg
md5sum catpicturess.jpg
```



Let's briefly check virustotal
Browse to virus total: `https://www.virustotal.com/gui/home/search`
Copy the above has for catpicturess.jpg and paste it into the search field.
Enter it into the system.
It should return with "No Matches Found" which is GREAT NEWS!

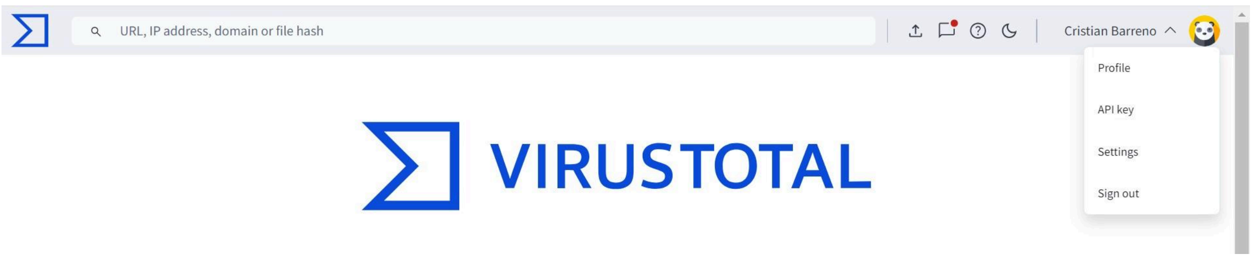


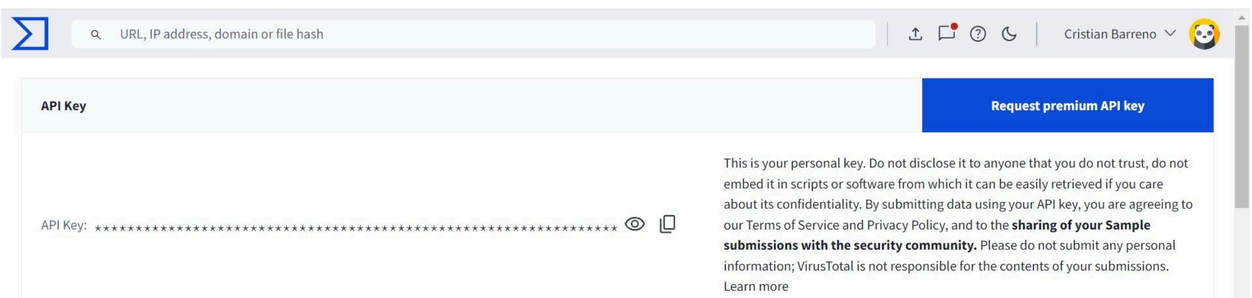
Task 4

Let's take this up a couple of notches.
Start in the home folder of your Linux Server

Step 1

Open a browser to https://www.virustotal.com
Create an Account
Once you create an account, navigate to the API key



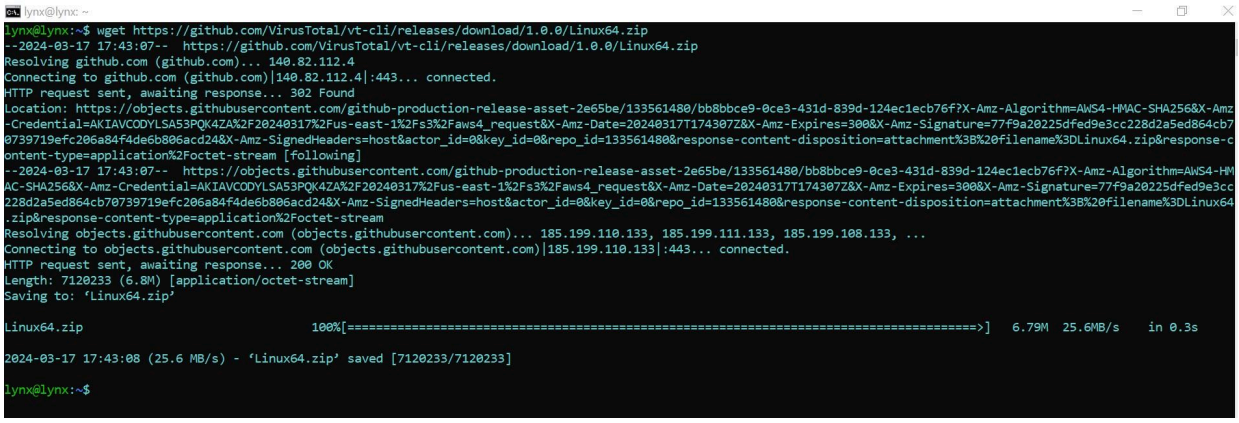


You'll need to copy the API key into the VirusTotal software later, don't navigate away from this page.

Step 2

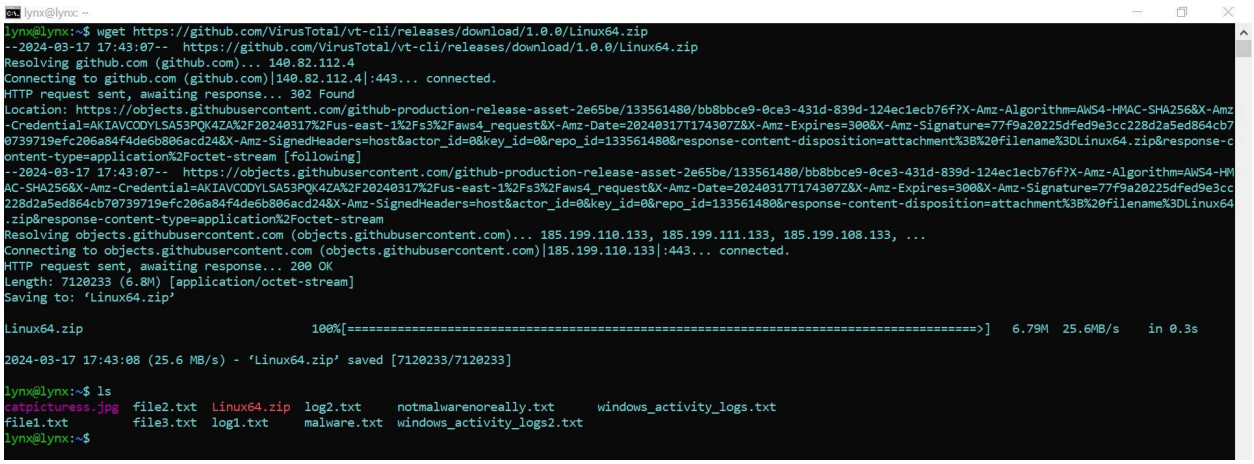
In your Linux Server
We have to download the pre-compiled VirusTotal Application.
Use the following command to download the zipped file.

```
wget https://github.com/VirusTotal/vt-cli/releases/download/1.0.0/Linux64.zip
```



List the directory to check.

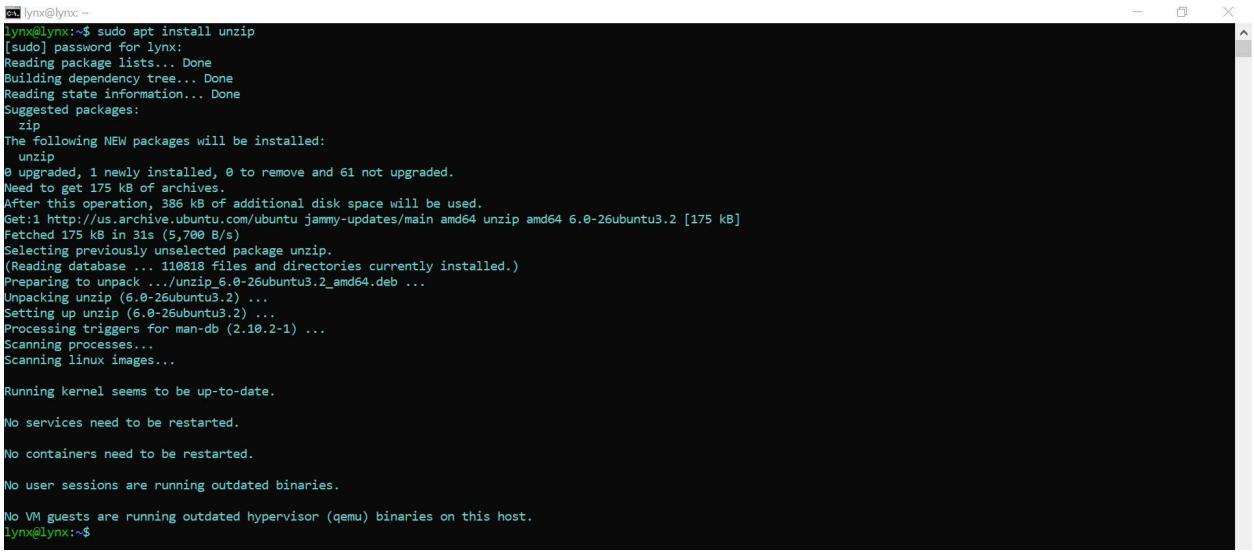
ls



and Linux 64.zip should be there.

Now, to unzip this zipped file, we'll need to install unzip.

sudo apt install unzip -y

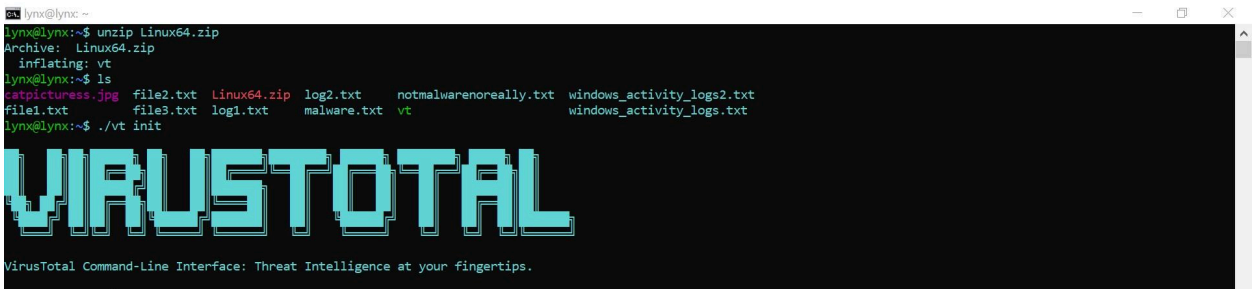


Time to unzip the VirusTotal application.

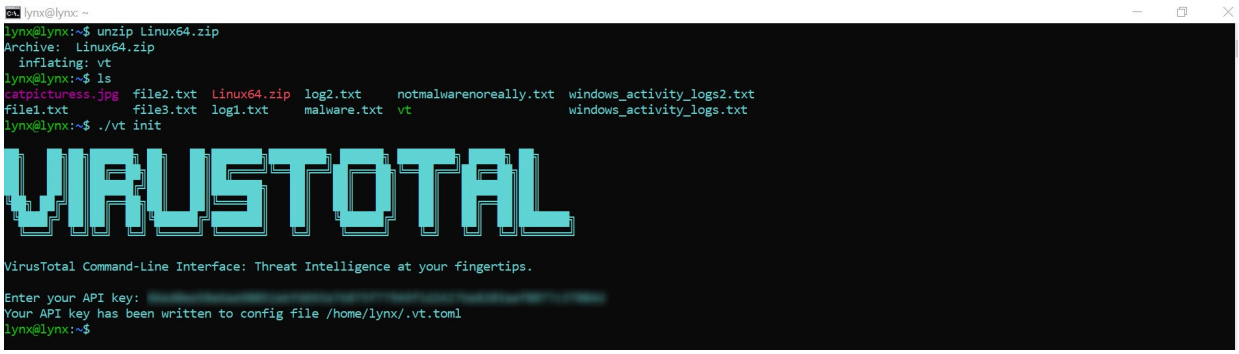
`unzip Linux64.zip`

Now run ls and you should see the vt application in your folder. (Hopefully your home folder)
Time to initialize the vt or VirusTotal Application. Please note you'll need to use dot slash before the application to get it to work.

`./vt init`



Enter your API key:



NOW! We are ready to start using Virus Total from the CLI!

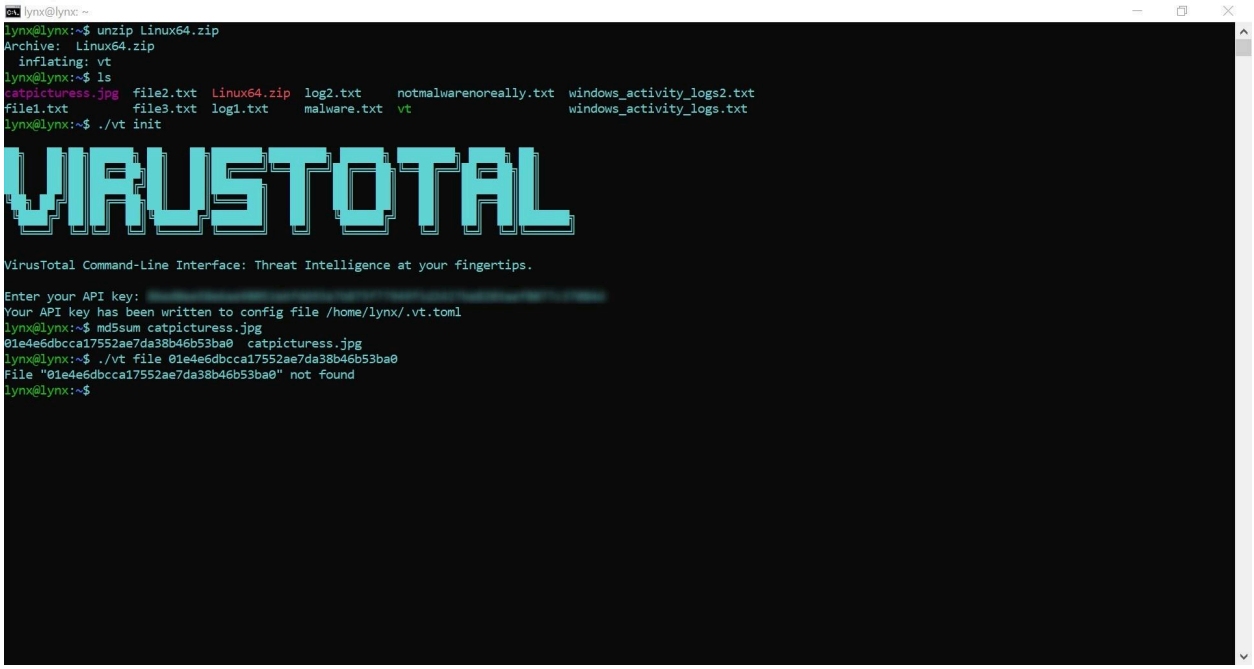
Step 3

`md5sum catpicturess.jpg`

Now we are going to check this hash in virustotal from the CLI. Please note you'll need to use dot slash before the application to get it to work.

Understand the syntax **`./`** then **`vt`** then **`file`** and then **`the hash.`**

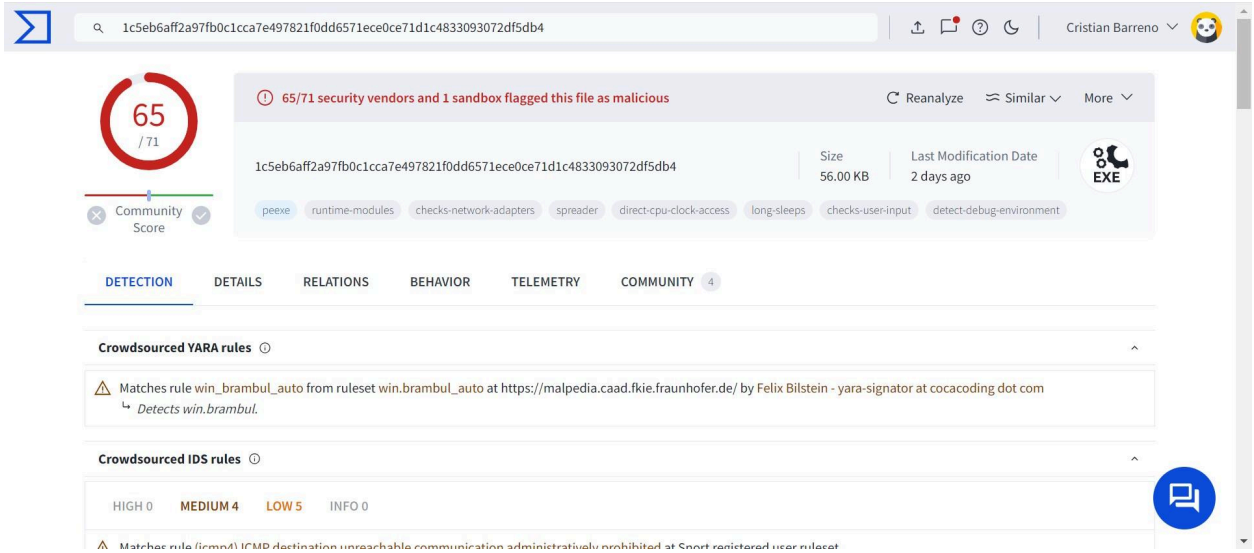
`./vt file 01e4e6dbcca17552ae7da38b46b53ba0`



HURRAY! The cat picture is still not a virus, according to Virus Total.

Step 4

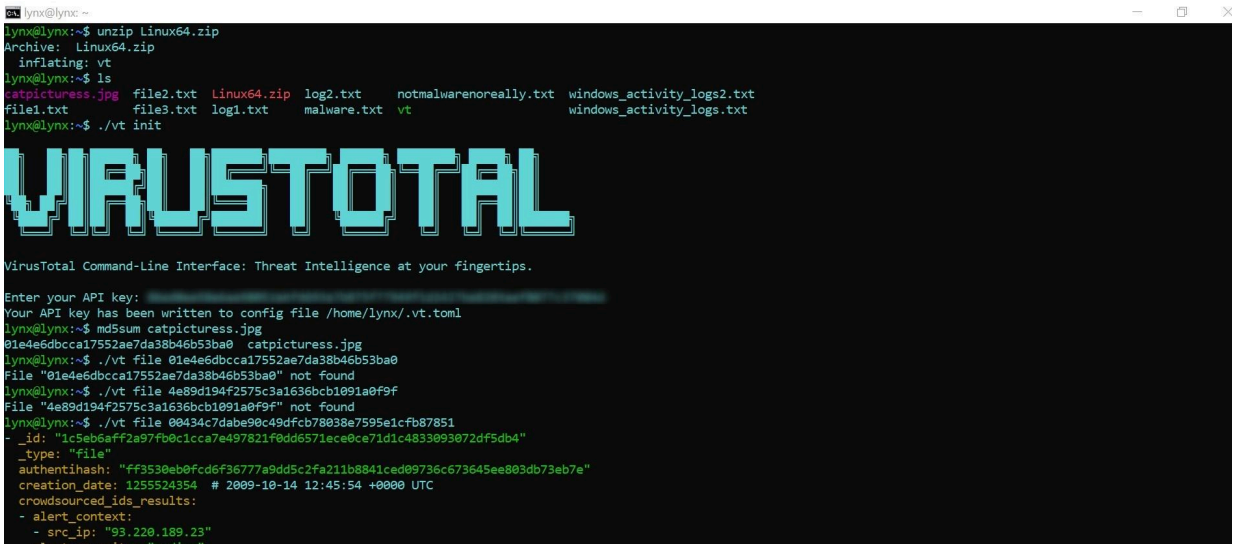
Let's use a known bad hash.
Let's pull up our Virus Total Web page, and paste the following known bad hash:
00434c7dabe90c49dfcb78038e7595e1cfb87851
And let's press the search icon on the right.
Oh my!



So… that's really bad.
Lots of detections

But, let's see how this looks like doing it via the command line.

```
./vt file 4e89d194f2575c3a1636bcb1091a0f9f
```



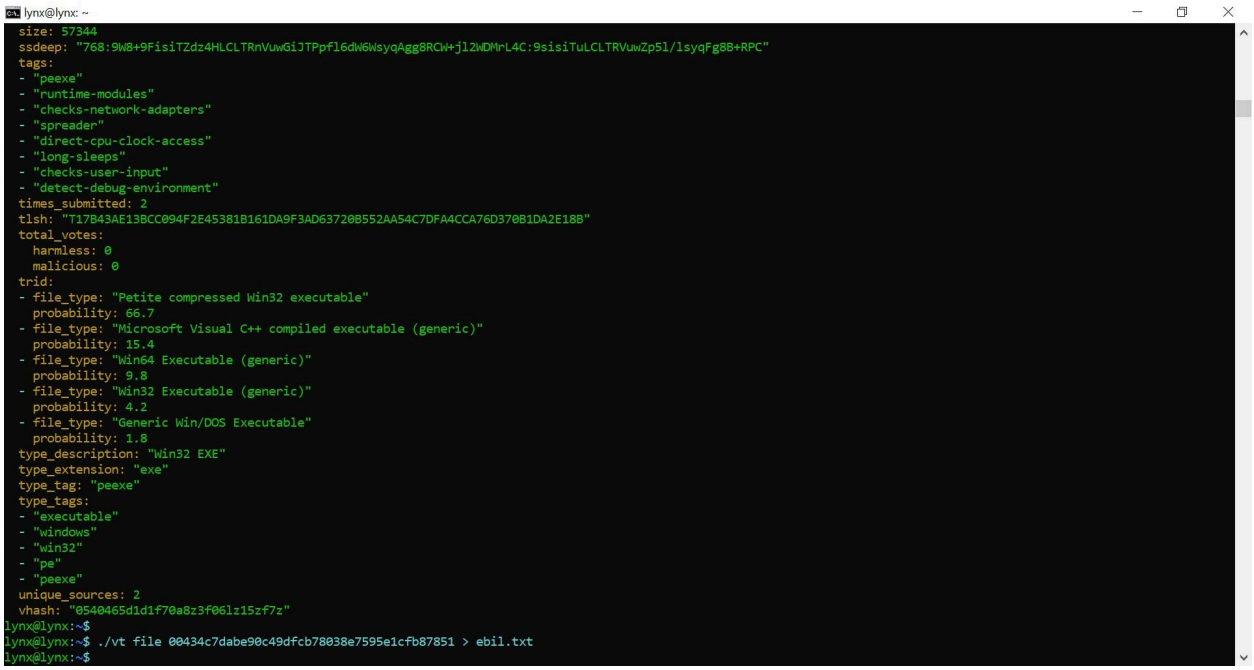
And it goes on and on and on, wow! That is a huge output! How can we control or manage this?

Step 5

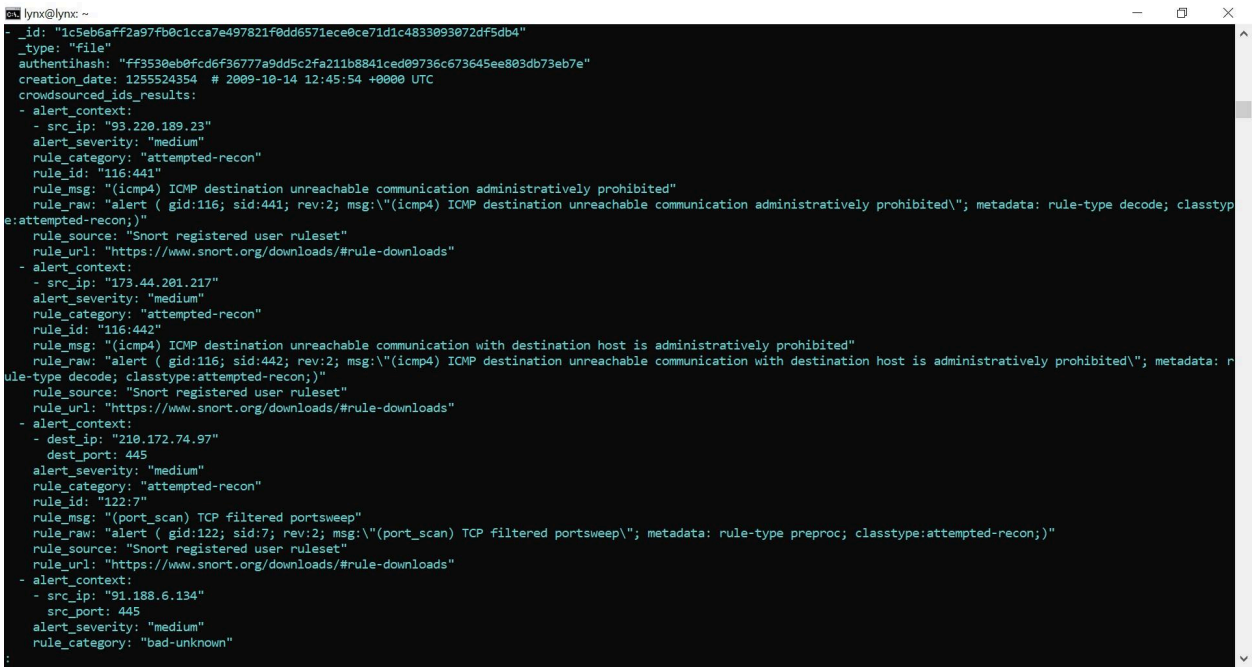
Let’s redirect the output to a file, so we can control what we are seeing.

`./vt file 00434c7dabe90c49dfcb78038e7595e1cfb87851 > ebil.txt`

Now, we can use any of the Linux Text READERS to check out the file.
Less, More, Cat, Head, Tail? What seems like it would be the most useful? The author is going to use less.



`less ebil.txt`



Now by using the command line interface text connection to virus total, I can bring in automatable log enrichment and information that can be used with other code to provide good threat intelligence and a way to populate cybersecurity systems.