

virustotal



context

dit document beschrijft de installatie en werking van virustotal op ubuntu.

algemeen

- de goto place voor malware research
- vt-cli is de CLI waarmee je ahv API-key de website kunt bevragen.
Of zoals ze het zelf zeggen: *a tool designed for those who love both VirusTotal and command-line interfaces*

Important: de Public API heeft een quotum van 4queries/minuut. LET DUS op als je loops gebruikt om veel bestanden te bevragen.

installatie

- maak een account aan op Virustotal.
In je profielpagina vind je de API-key die je nodig hebt om VirusTotal te bevragen via je CLI
- download de tool van de [Github pagina](https://github.com/VirusTotal/vt-cli/releases/download/0.10.0/Linux64.zip):
`cd /tmp && wget https://github.com/VirusTotal/vt-cli/releases/download/0.10.0/Linux64.zip`
- pak het bestand uit:
`unzip -x /tmp/Linux64.zip -d /usr/local/bin`
- stel Bash completion in:
`mkdir /etc/bash_completion.d && vt completion bash > /etc/bash_completion.d/vt`
- start vt en geef je API-key op:
`vt init`



werking

- elk type zoekactie heeft een eigen **trefwoord**: file, url, domain, ip, scan, search
- daarop zijn een aantal **commando's** mogelijk: vt file contacted_ips <hash>, bv
- je kan je resultaten **filteren** ahv:
 - -x: exclude pattern
 - -i: include pattern
 - voorbeeld: vt file d7bb40e4858211167b8e18c41f85fea02fe643f1211903b283b829db9c9f4b92 -i last_analysis_results: geeft enkel resultaten laatste analyse weer.
 - voorbeeld: vt file d7bb40e4858211167b8e18c41f85fea02fe643f1211903b283b829db9c9f4b92 -i names: geeft de namen weer waaronder het bestand nog bekend is.
 - waarden uit **sub-velden** worden gescheiden door een 'punt'
 - voorbeeld: vt file d7bb40e4858211167b8e18c41f85fea02fe643f1211903b283b829db9c9f4b92 -i last_analysis_results.Kaspersky: geeft enkel resultaten van Kaspersky's laatste analyse weer.
 - voorbeeld: vt file d7bb40e4858211167b8e18c41f85fea02fe643f1211903b283b829db9c9f4b92 -i signature_info.signers: geeft de ondertekenaars van certificaat weer.
 - **meerdere includes** worden gescheiden door een 'komma' (zonder trailing spatie!)
 - voorbeeld: vt file d7bb40e4858211167b8e18c41f85fea02fe643f1211903b283b829db9c9f4b92 -i last_analysis_stats,last_modification_date: laatste analyse en laatste wijziging
 - **interessante** filter, lijkt me:
vt file <hash> -i creation_date,first_seen_itw_date,first_submission_date,last_analysis_date,last_analysis_stats,last_submission_date,meaningful_names,sha256,signature_info.verified,trid

bestand

bestand opzoeken ahv hash: vt file <hash>

vb: vt file `sha256sum setup.exe|awk '{print \$1}'` -i

last_analysis_date,last_analysis_stats

vb: vt file contacted_ips|contacted_domains|contacted_urls <hash>

standaard

```
vt file `sha256sum <file>` -i  
creation_date,first_seen_itw_date,first_submission_date,last_analysis_date,la  
st_analysis_stats,last_submission_date,meaningful_name,names,sha256,signature  
_info.verified,trid
```

groep bestanden

```
for file in `ls *.exe`; do echo $file && vt file `sha256sum $file` -i  
_id,sha256,last_analysis_date,last_analysis_stats.malicious,last_analysis_sta  
ts.suspicious; done| tee -a VTAnalysis.txt
```

scan

bestand

```
bestand uploaden voor analyse: vt scan file <bestand>  
vb:vt scan file /tmp/elex_setup.exe  
/tmp/elex_setup.exe  
ODUwYmNiYmViZTJiOGQwMDM0Nzg3NDhmYjEwZDQwNmI6MTY0MTE1ODgwNQ==  
daarna kan je de analyse opvragen: vt analyse <_id>  
vb: vt analysis ODUwYmNiYmViZTJiOGQwMDM0Nzg3NDhmYjEwZDQwNmI6MTY0MTE1ODgwNQ==
```

groep bestanden

```
for file in `ls *.exe`; do echo $file && vt scan file $file; done| tee -a  
VTUpload.txt
```

domain

```
domain info opzoeken:  
vt domain <domeinnaam>
```

standaard

```
vt domain <domeinnaam> -i  
last_analysis_stats,last_modification_date,last_dns_records,last_https_certif  
icate.extensions.subject_alternative_name,last_https_certificate.issuer,last_  
https_certificate.validity,whois
```

Important: last_modification_date is laatste scantime van VT.

Zorg dat die recent is, anders kijk je naar outdated info (bv. rond certificataat vernieuwing, enzo)

subdomeinen

vt domain subdomains _id:
geeft alle gekende subdomeinen weer.

IP

vt ip <ip>

standaard

vt ip <ip> -i as_owner,continent,country,network,reputation,whois

URL

vt url <url>
indien deze niet gekend is: vt scan url <url>
vb: vt scan url koe.net/stier/os.html
koe.net/stier/os.html
u-02bd49629925820204837b7db6303e7f6180c7a7d586de9ee3aaede9ac046f24-1641223372

standaard

vt url <url> -i
categories,first_submission_date,html_meta,last_analysis_date,last_analysis_s
tats,title

analysis

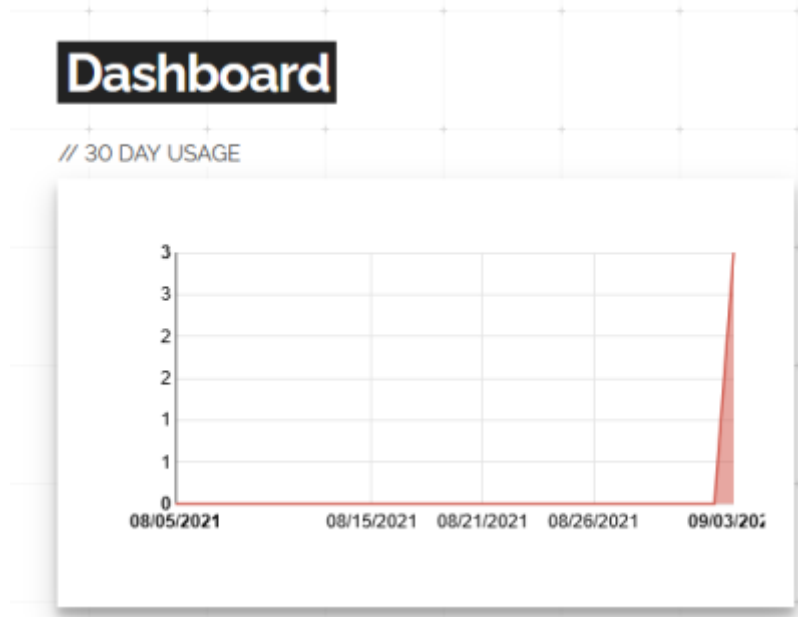
vraag analyse op ahv unieke id, meestal nadat je vt scan file/url hebt uitgevoerd.
vt analysis <_id>:

standaard

watch -d vt analysis <_id>

audit

- elke API call die je maakt wordt gelogd: **Account > DEVELOPER DASHBOARD**



meer info

[vt-cli](#)

[Linux](#), [virustotal](#), [digital forensics](#)

From:

<https://louslab.be/> - **Lou's lab**

Permanent link:

<https://louslab.be/doku.php?id=linux:virustotal>

Last update: **2024/11/16 18:14**

