This session will demonstrate methods for using VirusTotal data to deep dive into malware campaigns. We will begin by exploring the design and implementation of the newest tools introduced to the VirusTotal arsenal: VTGrep and Graph. The workshop will then progress into discussion around how best to leverage the data available to VT users. By better understanding the breadth and depth of malicious campaigns, researchers and law enforcement can better investigate and mitigate impact. Recently introduced improved relational metadata as well as expanded retroactive and proactive hunting capabilities allow investigators to dive deep into malware within a global data source.

**Objective:**
Users will learn:

1. How to use VirusTotal Graph to visualize malware campaigns.
2. How to use VirusTotal Intelligence to identify interesting malware metadata.
3. How to use Yara for proactive and retroactive visibility.
Goals

- Discussion and Practical Application of Tools
  - Static Data Pivots
  - Faceted Search
  - VTGrep
  - VTGraph
  - YARA + VirusTotal Externals
- Highlight APIv3 Functionality

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

1) One Click Pivots and Visual Similarity Pivoting
2) VTGrep technical dive and functionality
3) VTGraph Technical dive and functionality
4) Yara guided rule dev, key modules, and practical examples
5) APIv3 New Go Tool (commandline) and functions
6) TEASERS
www.virustotal.com/gui/

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

https://www.virustotal.com/subscription/event/klas19/join
VirusTotal Introduction

Introduction to VTE Functionality
What is VirusTotal Intelligence?

- VirusTotal Intelligence has been called the “Google of malware”
- VTI provides the ability to search through VT’s dataset using:
  - Binary properties
  - Detection verdicts/signatures
  - Static properties
  - Behavior patterns
  - Metadata
- Access via web interface or APIs
Faceted Search
Existing Search Modifiers:
https://support.virustotal.com/hc/en-us/articles/360000347157-VT-Intelligence

Madlibs style guided query builder

Assumes all terms are AND

Faceted SEarches support OR queries as well!
Tips

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

Practice: https://www.virustotal.com/wargame/

Supports logical “AND” “OR” “NOT”

Respects order of operations: ()

Ranges can be denoted with + or -

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

Practice makes perfect: https://www.virustotal.com/wargame/
Example 1

Find Windows Executables that communicate over HTTP

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

- Find Windows Executables that exhibit HTTP behaviors
  - (type:peexe OR type:pedll) behavior:http
Example 1

Key Search Terms:
- type
- behavior

Find Windows Executables that communicate over HTTP

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

- Find Windows Executables that exhibit HTTP behaviors
  - (type:peexe OR type:pedll) behavior:http
Example 1

Key Search Terms:
- type
- behavior

Find **Windows Executables** that communicate over HTTP

(type:peexe OR type:pedll) behavior:http

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

https://www.virustotal.com/gui/search/(type%253Apeexe%2520OR%2520type%253Apedll)%2520behavior%253Ahttp

- Find Windows Executables that exhibit HTTP behaviors
  - (type:peexe OR type:pedll) behavior:http
Example 2

Find poorly detected Executables that use “fre.php” in their URI

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence
Example 2

Find poorly detected Executables that use “fre.php” in their URI

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

Key Terms:
Poorly Detected => Positives Ratio
Executables => Type
Uses => Behavior

- type:peexe behavior:fre.php p:10-
Example 2: Solution [Lokibot]

Find poorly detected Executables that use “fre.php” in their URI

p:10- type:peexe behavior:fre.php

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

Show Results:
https://www.virustotal.com/gui/search/p%253A10-%2520type%253Apeexe%2520behavior%253Afre.php/files
Example 3

Files named “invoice” from the US with macros and greater than 5 detections

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

Show Results:
https://www.virustotal.com/gui/search/p%253A10-%2520type%253Apeexe%2520behavior%253Afre.php/files
Example 3

Files named “invoice” from the US with macros and greater than 5 detections

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

Key Terms:
Named => name:
With Macros => tag (static facet of file analysis)
From: => Submitter
Detections => p (or positives)

- name:"invoice" tag:macros p:5+ submitter:US
Example 3: Solution

Files named “invoice” from the US with macros and greater than 5 detections

name:"invoice" tag:macros p:5+ submitter:US

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

https://www.virustotal.com/gui/search/name%253A%2522invoice%2522%2520tag%2520%255Bname\%0A%255D%0A%255Btag\%0A%255D\%255Bp%255D%255Bsubmitter%255D/files

Key Terms:
Named => name:
With Macros => tag (static facet of file analysis)
From: => Submitter
Detections => p (or positives)

- name:"invoice" tag:macros p:5+ submitter:US
Example 4

Find files from domains containing “dropbox.com” or “box.com” with 10 or more detections

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

Key Terms:
From Domains => itw
Detections => p
Example 4: Solution

Find files from domains containing “dropbox.com” or “box.com” with 10 or more detections

(itw:dropbox.com or itw:box.com) p:10+

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

https://www.virustotal.com/gui/search/(itw%253Adropbox.com%2520or%2520itw%2520box.com)%2520p%252010/files

Key Terms:
From Domains => itw
Detections => p

■ (itw:dropbox.com or itw:box.com) p:10+

Notice this shows all files, what if we want Windows Executables only?
Find **Windows Executables** from domains containing “dropbox.com” or “box.com” with 10 or more detections.

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

Key Terms:
- Windows Executables => type
- From Domains => itw
- Detections => p

- (type:peexe OR type:pedll) (itw:dropbox.com or itw:box.com)
  p:10+
Example 4a: Solution

Find Windows Executables from domains containing “dropbox.com” or “box.com” with 10 or more detections
(type:peexe OR type:pedll) (itw:dropbox.com OR itw:box.com) p:10+

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

https://www.virustotal.com/gui/search/(type%253Apeexe%2520OR%2520type%253Apedll)%2520(itw%253Adropbox.com%2520or%2520itw%253Abox.com)%2520p%253A10%252B/files

Key Terms:
Windows Executables => type
From Domains => itw
Detections => p

- (type:peexe OR type:pedll) (itw:dropbox.com or itw:box.com) p:10+
Example 4b

Find all files OTHER than Windows Executables from domains containing “dropbox.com” or “box.com” with 10 or more detections

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

Key Terms:
Windows Executables => type
From Domains => itw
Detections => p

- (NOT type:peexe OR NOT type:pedll) (itw:dropbox.com or itw:box.com) p:10+
Example 4b: Solution

Find Windows Executables from domains containing “dropbox.com” or “box.com” with 10 or more detections
(NOT type:peexe AND NOT type:pedll) (itw:dropbox.com or itw:box.com) p:10+

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

https://www.virustotal.com/gui/search/(NOT%20type%253Apeexe%2520OR%2520NOT%20type%253Apedll)%2520(itw%253Adropbox.com%2520or%2520itw%253Abox.com)%2520p%253A10%252B/files

Key Terms:
Windows Executables => type
From Domains => itw
Detections => p
  ■ (NOT type:peexe AND NOT type:pedll) (itw:dropbox.com or itw:box.com) p:10+
  ● Technically the OR between the types is superfluous, because logic
Example 5

Find DLLs with Turla signature hits first seen after March 1st, 2019

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

Note we want to evaluate ALL engines for this signature string

Key Terms:
DLLs => type
signature hits => engines
First Seen = > fs

- engines:Turla type:pedll fs:2019-03-01+

If seeking from a specific AV scanner partner refer to the Full Vendor List:
  a_squared
  ad_aware
  aegislab
  agnitum
  ahnlab
  ahlab_v3
  alibaba
  alyac
  antivir
  antivir7
Example 5

Find DLLs with Turla signature hits first seen after March 1st, 2019

engines:Turla type:pedll fs:2019-03-01+

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

https://www.virustotal.com/gui/search/engines%3ATurla%2520type%253Apedll%2520fs%253A2019-03-01%252B/files

Key Terms:
DLLs => type
signature hits => engines
First Seen = > fs

- engines:Turla type:pedll fs:2019-03-01+

Full Vendor List:
a_squared
ad_aware
aegislab
agnitum
ahnlab
ahnlab_v3
alibaba
alyac
antivir
nod32
norman
nprotect
paloalto
panda
pctools
prevx
prevx1
qihoo_360
rising
sentinelone
sophos
sunbelt
superantispyware
symantec
symantecmobileinsight
tencent
thehacker
totaldefense
trendmicro
trendmicro_housecall
trustlook
vba32
vipre
virobot
webroot
whitearmor
yandex
zillya
zonealarm
zoner
Find all RTF files with metadata containing “Windows User” that are using a known/identified CVE and NOT macros

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

Key Terms:
files => type
metadata => metadata
known/identified => engines or tag

- metadata:"Windows User" type:rtf (tag:cve or engines:exploit or engines:cve) NOT tag:macros
Example 6

Find all RTF files with metadata containing “Windows User” that are using a known/identified CVE and NOT macros

metadata:"Windows User" type:rtf (tag:cve or engines:exploit or engines:cve) NOT tag:macros

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

https://www.virustotal.com/gui/search/metadata%3A%22Windows%20User%22%20type%3Artf%20(tag%3Acve%20or%20engines%3Aexploit%20or%20engines%3Acve)%20NOT%20tag%3Amacros/files

Key Terms:
files => type
metadata => metadata
known/identified => engines or tag

- metadata:"Windows User" type:rtf (tag:cve or engines:exploit or engines:cve) NOT tag:macros
Example 7

Find all document types using an exploit/or known CVE with RU lang encoding

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

Key Terms:
files => type
known/identified => engines or tag
lang encoding => lang

- type:document (tag:cve or engines:exploit or engines:cve)
  lang:ru
- Note: Lang is available for peexe/pedll and office file formats ONLY
Example 7

Find all document types using an exploit/or known CVE with RU lang encoding

type:document (tag:cve or engines:exploit or engines:cve) lang:ru

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

https://www.virustotal.com/gui/search/type%253Adocument%2520(tag%253Acve%2520or%2520engines%253Aexploit%2520or%2520engines%253Acve)%2520lang%253Aru%2520fs%253A2019-03-01%252B%2520ls%253A2019-03-15-/files

Key Terms:
files => type
known/identified => engines or tag
lang encoding => lang
first seen => fs

- type:document (tag:cve or engines:exploit or engines:cve) lang:ru
  - Note: Lang is available for peexe/pedll and office file formats ONLY
Example 8

Find all document types using an exploit/or known CVE with RU lang encoding first seen between March 1st and March 15th

type:document (tag:cve or engines:exploit or engines:cve) lang:ru

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

Key Terms:
files => type
known/identified => engines or tag
lang encoding => lang

- type:document (tag:cve or engines:exploit or engines:cve) lang:ru fs:2019-03-01+ fs:2019-03-15-
  - Note: Lang is available for peexe/pedll and office file formats ONLY
Example 8

Find all document types using an exploit/or known CVE with RU lang encoding first seen between March 1st and March 15th

type:document (tag:cve or engines:exploit or engines:cve) lang:ru fs:2019-03-01+ fs:2019-03-15-

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

https://www.virustotal.com/gui/search/type%253Adocument%2520(tag%2520cve%2520or%2520engines%2520exploit%2520or%2520engines%2520cve)%2520lang%253Aru%2520fs%253A2019-03-01%2520%2520fs%253A2019-03-15-

Key Terms:
files => type
known/identified => engines or tag
lang encoding => lang

- type:document (tag:cve or engines:exploit or engines:cve) lang:ru fs:2019-03-01+ fs:2019-03-15-
  - Note: Lang is available for peexe/pedll and office file formats ONLY
Static Data Pivoting
Assessing the “Details” Tab

Let’s look at a Trickbot Sample:
f579682f1be62564aab114b2cb1dc06e7ced7740f61b1b8a11eb92f5ed88fdf

https://www.virustotal.com/gui/file/f579682f1be62564aab114b2cb1dc06e7ced7740f61b1b8a11eb92f5ed88fdf/detection
Example: Pivot on Imphash

imphash:"d0472d140aa0003beaf55821a63a5b03" (Anchor Trickbot)

https://www.virustotal.com/gui/search/imphash%253A%2522d0472d140aa0003beaf55821a63a5b03%2522/files
Example: Pivot on Imphash

imphash: "d0472d140aa0003beaf55821a63a5b03" (Anchor Trickbot)

https://www.virustotal.com/gui/search/imphash%253A%2522d0472d140aa0003beaf55821a63a5b03%2522/files
Caveat: Imphash is NOT Always Reliable

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

- Example UPX packed files: section:upx1 section:upx0
- imphash:"7326001be3ced77b153640be93a8dff6"

Example:
175bafbcd5218e062619b16dd4c18279635ae3d621daa7aa559a3ca5882ebf1c
https://www.virustotal.com/gui/search/imphash%253A%25227326001be3ced77b153640be93a8dff6%2522/files

TLDR: You’re actually matching the packer.
Similar-To (vhash)

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

https://www.virustotal.com/gui/search/similar-to%253Af579682f1be62564aab114b2cb1dc06e7ced77406f61b1b8a11eb92f5ed88f/Files

Lets check out our trickbot sample that we explored imphash with:
similar-to:f579682f1be62564aab114b2cb1dc06e7ced77406f61b1b8a11eb92f5ed88f

Notice how many more results there are?
Visual Similarity

Supports:
● Windows Executables (embedded)
● PDF
● Office Documents

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

Julio To Talk about how it works

Trickbot EXE:
https://www.virustotal.com/gui/file/01c299e4895eb222d24ad9c6bbefe2a389bc3d54e37de8c7a8dc73a95a7f093b/detection
-alternative mechanism:
resource:"77073160cc8d0c6443a55cf6514f3606d979ca8ce78a1a9cc20ec71c57e392d6"

Emotet PDF:
https://www.virustotal.com/gui/file/8c15b770e32ff70527a4e17e1173d3f2ff91f7f27be17268f2a814c72d863859/detection

ADP Lure Doc:
https://www.virustotal.com/gui/file/2db7425c7c9eff9c87fb45719ac4a9c7b24722f2cf19de1ba7f1b9d1f59de45/detection
Visual Similarity: EXE

main_icon_dhash:b168c6a98ee460b2

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

Trickbot EXE:
https://www.virustotal.com/gui/file/01c299e4895eb222d24ad9c6b6e62e2a389bc3d54e37de8c7a8dc73a95a7f093b/detection
-alternative mechanism:
resource:"77073160cc8d0c6443a55cf6514f3606d979ca8ce78a1a9cc20ec71c57e392d6"

Emotet PDF:
https://www.virustotal.com/gui/file/8c15b770e32ff70527a4e17e1173d3f2ff91f7f27be17268f2a814c72d863859/detection

ADP Lure Doc:
https://www.virustotal.com/gui/file/2db7425c7c9eff9c87ff45719ac4a9c7b24722f2cf19de1ba7f1b9d1f59de45/detection
Visual Similarity: PDF

Your Citibank, N.A. Account Has Been Suspended

We have temporarily suspended your Citibank, N.A. account for the funds transfer service.

Here are your account details:

Please contact Member Services to re-activate your suspended account.

Sincerely,
Member Services

Email ID: 431

https://www.Citibank.com/
https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

Pivoting on the PDF Visual Similarity

**Trickbot EXE:**
https://www.virustotal.com/gui/file/01c299e4895eb222d24ad9c6bbefe2a389bc3d54e37de8c7a8dc73a95a7f093b/detection

**Emotet PDF:**
https://www.virustotal.com/gui/file/8c15b770e32ff70527a4e17e1173d3f2ff91f27be17268f2a814c72d863859/detection

**ADP Lure Doc:**
https://www.virustotal.com/gui/file/2db7425c7c9eff9c87fb45719ac4a9c7b24722f2cf19deb7a7f1b9d1f59de45/detection
Visual Similarity: Office Documents

[Image: Protected Document]

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

Trickbot EXE:
https://www.virustotal.com/gui/file/01c299e4895eb222d24ad9c6bbefe2a389bc3d54e37de8c7a8dc73a95a7f093b/detection

Emotet PDF:
https://www.virustotal.com/gui/file/8c15b770e32ff70527a4e17e1173d3f2ff91ff27be17268f2a814c72d863859/detection

ADP Lure Doc:
https://www.virustotal.com/gui/file/2db7425c7c9ef99c87fb45719ac4a9c7b24722f2cf19de1ba7f1b9d1f59de45/detection
Visual Similarity: Office Documents

main_icon_dhash:0103001101000301

Files 2

89f7ab110807cf0352ada55582611e6094af8e13ad8f832561d3e2c2763a70659c3
33 / 56
46.02 KB
2019-03-12 14:10:29 first seen
2019-03-12 14:10:29 last seen

2db7425c7cdef8bc87f6d45719ac4aeccf2c2763a70659c3
43 / 60
46 KB
2018-07-19 16:31:07 first seen
2018-07-30 04:48:49 last seen

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

Trickbot EXE:
https://www.virustotal.com/gui/file/01c299e4895eb222d24ad9c6bbef2a389bc3d54e37de8c7a8dc73a95a7f093b/detection

Emotet PDF:
https://www.virustotal.com/gui/file/8c15b770e32ff70527a4e17e1173d3f2ff91f7f27be17268f2a814c72d863859/detection

ADP Lure Doc:
https://www.virustotal.com/gui/file/2db7425c7c9eff9c87fb45719ac4a9c7b24722f2cf19de1ba7f1b9d1f59de45/detection
<table>
<thead>
<tr>
<th>Signers</th>
<th>ALISA LTD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>ALISA LTD</td>
</tr>
<tr>
<td>Status</td>
<td>Trust for this certificate or one of the certificates in the certificate chain has been revoked.</td>
</tr>
<tr>
<td>Valid From</td>
<td>12:00 AM 02/22/2019</td>
</tr>
<tr>
<td>Valid To</td>
<td>11:59 PM 02/21/2020</td>
</tr>
<tr>
<td>Valid</td>
<td>Code Signing</td>
</tr>
<tr>
<td>Usage</td>
<td></td>
</tr>
<tr>
<td>Algorithm</td>
<td>sha256RSA</td>
</tr>
<tr>
<td>Serial Number</td>
<td>5D A1 73 EB 1A C7 63 40 AC 05 8E 1F F4 BF 5E 1B</td>
</tr>
</tbody>
</table>

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

LockerGoga:
c97d9bbc80b573bdeeda3812f4d00e5183493dd0d5805e2508728f659777dda15
https://www.virustotal.com/gui/file/c97d9bbc80b573bdeeda3812f4d00e5183493dd0d5805e2508728f659777dda15/detection
**Signature Data**

**Signature Verification**

⚠️ File signature could not be verified

**File Version Information**

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Copyright</td>
<td>Copyright (C) ALISA LTD 2019</td>
</tr>
<tr>
<td>Product</td>
<td>Service zzbdrimp</td>
</tr>
<tr>
<td>Description</td>
<td>Background Tasks Host</td>
</tr>
<tr>
<td>Original Name</td>
<td>zzbdrimp</td>
</tr>
<tr>
<td>Internal Name</td>
<td>zzbdrimp</td>
</tr>
<tr>
<td>File Version</td>
<td>1.4.4.0</td>
</tr>
<tr>
<td>Date signed</td>
<td>8:40 PM 3/20/2019</td>
</tr>
</tbody>
</table>

[https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence](https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence)

LockerGoga:
c97d9bbc80b573bdeeda3812f4d00e5183493dd0d5805e2508728f65977dda15
[https://www.virustotal.com/gui/file/c97d9bbc80b573bdeeda3812f4d00e5183493dd0d5805e2508728f65977dda15/detection](https://www.virustotal.com/gui/file/c97d9bbc80b573bdeeda3812f4d00e5183493dd0d5805e2508728f65977dda15/detection)
Signature Data

Pivot Options:
- Broad = signature: "ALISA LTD"
- Narrow = signature: "5D A1 73 EB 1A C7 63 40 AC 05 8E 1F F4 BF 5E 1B"

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

LockerGoga:
c97d9bbc80b573bdeeda3812f4d00e5183493dd0d5805e2508728f65977dda15
https://www.virustotal.com/gui/file/c97d9bbc80b573bdeeda3812f4d00e5183493dd0d5805e2508728f65977dda15/detection

By Name of Signer:
Broad => signature: "ALISA LTD"
BY Cert Hash
Narrow => signature: "5D A1 73 EB 1A C7 63 40 AC 05 8E 1F F4 BF 5E 1B"
Signature Data

signature:"5D A1 73 EB 1A C7 63 40 AC 05 8E 1F F4 BF 5E 1B" OR signature:"ALISA LTD"

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

LockerGoga:
c97d9bbc80b573bdeeda3812f4d00e5183493dd0d5805e2508728f659777dda15
https://www.virustotal.com/gui/file/c97d9bbc80b573bdeeda3812f4d00e5183493dd0d5805e2508728f659777dda15/detection

By Name of Signer:
Broad => signature:"ALISA LTD"
BY Cert Hash
Narrow => signature:"5D A1 73 EB 1A C7 63 40 AC 05 8E 1F F4 BF 5E 1B"

Combine the Two:
https://www.virustotal.com/gui/search/signature%253A%25225D%2520A1%252073%2520EB%25201A%2520C7%252063%2520AC%252005%2520BF%25205E%25201B%2522/files

Note: In this instance the signer and hash were only used together, so you’ll get the same results with either query.
Signature Data

signature: "Copyright (C) ALISA LTD 2019"

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

LockerGoga:
c97d9bbc80b573bdeeda3812f4d00e5183493dd0d5805e2508728f65977dda15
https://www.virustotal.com/gui/file/c97d9bbc80b573bdeeda3812f4d00e5183493dd0d5805e2508728f65977dda15/detection

There is another option as well that leverages the structure within the PE itself with contains additional signature information:

signature: "Copyright (C) ALISA LTD 2019"

https://www.virustotal.com/gui/search/signature%253A%2522Copyright%2520(C)%2520ALISA%2520LTD%25202019%2522/files
I have no idea what the convention is for this word, ya’l know what I mean.

This is data generated from Exiftool.

EmpireMonkey:
d57f128af8443b6f0072fadda8d14046b31703098e365bc5a226e117090d44
https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

I have no idea what the convention is for this word, ya’ll know what I mean.

This is data generated from Exiftool.

EmpireMonkey:
d57f128afbf4843b6f0072fadda8dd14046b31703098e365bc5a226e117090d44

metadata:"Normal.dotm" and metadata:"cobalt"
https://www.virustotal.com/gui/search/metadata%253A%2522Normal.dotm%2522%2520and%2520metadata%253A%2522cobalt%2522/files
I have no idea what the convention is for this word, ya’ll know what I mean.

This is data generated from Exiftool.

Example 2: LockerGoga,
https://www.virustotal.com/gui/file/88d149f3e47dc337695d76da52b25660e3a454768af0d7e59c913995af496a0f/detection

metadata:"Background Tasks Host"

https://www.virustotal.com/gui/search/metadata%253A%2522Background%2520Tasks%2522/files
### Imports
- ADVAPI32.dll
- KERNEL32.dll
- RPCRT4.dll
- ntdll.dll

### Exports
- DhcpNewPktHook
- DhcpServerCalloutEntry
- DnsPluginCleanup
- DnsPluginInitialize
- DnsPluginQuery
- ExtensionApiVersion
- InitializeChangeNotify
- PasswordChangeNotify
- SpLsidModelInitialize
- WinDbgExtensionDllInit

---

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

Example: Mimikatz
b04e58327191222e27405c2dc4871cb4c81e3ea732d70c67ad744088619c97e4

https://www.virustotal.com/gui/file/b04e58327191222e27405c2dc4871cb4c81e3ea732d70c67ad744088619c97e4/detection
Imports and Exports

Example: Mimikatz
b04e58327191222e27405c2dc4871cb4c81e3ea732d70c67ad744088619c97e4

https://www.virustotal.com/gui/file/b04e58327191222e27405c2dc4871cb4c81e3ea732d70c67ad744088619c97e4/detection

Imports=> imports:NdrMesTypeFree2 imports:MesHandleFree imports:RtlStringFromGUID imports:GetOEMCP
Exports => exports:"InitializeChangeNotify" exports:"PasswordChangeNotify" exports:"SpLsaModeInitialize"
Imports and Exports

Example: Mimikatz
b04e58327191222e27405c2dc4871cb4c81e3ea732d70c67ad744088619c97e4

https://www.virustotal.com/gui/file/b04e58327191222e27405c2dc4871cb4c81e3ea732d70c67ad744088619c97e4/detection

Imports=> imports:NdrMesTypeFree2 imports:MesHandleFree imports:RtlStringFromGUID imports:GetOEMCP

Exports => exports:"InitializeChangeNotify" exports:"PasswordChangeNotify" exports:"SpLsaModeInitialize"
https://support.virustotal.com/hc/en-us/articles/360000347157-VT-Intelligence#content-search
VTGrep (aka Content Search)

Use the “content:” search modifier to search for arbitrary hex or string patterns within files on VirusTotal

https://support.virustotal.com/hc/en-us/articles/360000347157-VT-Intelligence#content-search
VTGrep (aka Content Search)

- VTGrep is an index of 32bit substrings to sample IDs (SHA256)
- It returns all the samples with the given content in less than 60 seconds
- It supports most YARA's string conditions
  - Wildcards, UTF-8, HEX, offsets, AND, OR, ...
  - No regexps, though :-(
- Great for prototyping Retrohunts
- It uses Google infrastructure to serve 1PB of compressed data (all samples uploaded to VT in a year)
  - Includes unpacked, OCR, macros, VBA code streams...

https://support.virustotal.com/hc/en-us/articles/36000347157-VT-Intelligence#content-search
VTGrep: Example 1, ASCII Strings

<table>
<thead>
<tr>
<th>DETECTION</th>
<th>DETAILS</th>
<th>RELATIONS</th>
<th>BEHAVIOR</th>
<th>CONTENT</th>
<th>SUBMISSIONS</th>
<th>COMMUNITY</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>STRINGS</strong></td>
<td><strong>HEX</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>cmd.exe &amp;s&amp;s&quot;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D:\MyProjects\secondWork\Anchor\Win32\Release\anchorInstaller_x86.pdb</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>kernel32.dll</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADVAPI32.dll</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>KERNEL32.dll</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>kernel32.dll</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>mscoree.dll</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

https://support.virustotal.com/hc/en-us/articles/360000347157-VT-Intelligence#content-search

**Trickbot**: 5739549850fe635fc0ac5de81ce1fd495669fcabc1b8ede35b82a22093c86399

https://www.virustotal.com/gui/file/5739549850fe635fc0ac5de81ce1fd495669fcabc1b8ede35b82a22093c86399/detection

There appears to be a PDB! I love PDBs.

“Program database (**PDB**) is a proprietary file format (developed by Microsoft) for storing debugging information about a program (or, commonly, program modules such as a DLL or **EXE**). **PDB** files commonly have a **.pdb** extension. A **PDB** file is typically created from source files during compilation.”

Wouldn’t it be cool if we could find more samples that contain this PDB string? Maybe the attackers left something behind!
## VTGrep: Example 1, ASCII Strings

<table>
<thead>
<tr>
<th>File Path</th>
<th>MD5 Hash</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>content:D:\MyProjects\secondWork\Anchor\Win32\Release\anchorInstaller_x86.pdb</td>
<td>96e484009a93c796c2a405b5375b5b054abd3b5579d9f493d4f97a31aeced88b6b</td>
<td>698 KB</td>
</tr>
<tr>
<td>testtab.png</td>
<td>5739549850fe635fc0ac5de81ce1fd495669fcabc1b8ede35b82a22093c86399</td>
<td>390 KB</td>
</tr>
<tr>
<td></td>
<td>3e0b63b51d79e2f40b7a775d1d353d68594eef3617af66804ac00775754d89</td>
<td>394 KB</td>
</tr>
</tbody>
</table>

https://support.virustotal.com/hc/en-us/articles/360000347157-VT-Intelligence#content-search

**Trickbot**: 5739549850fe635fc0ac5de81ce1fd495669fcabc1b8ede35b82a22093c86399

https://www.virustotal.com/gui/file/5739549850fe635fc0ac5de81ce1fd495669fcabc1b8ede35b82a22093c86399/detection

Oho there appears to be a PDB! I love PDBs.

“Program database (**PDB**) is a proprietary file format (developed by Microsoft) for storing debugging information about a program (or, commonly, program modules such as a DLL or **EXE**). **PDB** files commonly have a **.pdb** extension. A **PDB** file is typically created from source files during compilation.”

Wouldn’t it be cool if we could find more samples that contain this PDB string? Maybe the attackers left something behind!

We can!

Two ways:

1) Click on the string in the “Content” tab to generate a query in Hex
   a) `content:{443a5c4d7950726f6a656374735c7365636f66e64576f726b5c416e636866725c57696e33325c52656c656173655c616e64}

2) Use VTGrep to search for PDB strings in the file(s) you've identified.
a) 3686f72496e7374616c65725f7838362e706462}

2) For an ASCII string search
   a) content:D:\MyProjects\secondWork\Anchor\Win32\Release\anchorInstaller_x86.pdb

These methods are functionally identical.

Bonus:
we can actually find more of this by shortening the string to not be SO specific:
   ■ content:D:\MyProjects\secondWork\Anchor\Win32\Release
VTGrep: Example 2 Wildcards

content:D:\MyProjects\secondWork\Anchor\Win32\Release\anchorInstaller_x86.pdb

https://support.virustotal.com/hc/en-us/articles/360000347157-VT-Intelligence#content-search

Wildcarding
Trickbot: 5739549850fe635fc0ac5de81ce1fd495669fcabc1b8ede35b82a22093c86399

https://www.virustotal.com/gui/file/5739549850fe635fc0ac5de81ce1fd495669fcabc1b8ede35b82a22093c86399/detection

D:\MyProjects\secondWork\Anchor\Win32\Release\anchorInstaller_x86.pdb

What if we weren’t sure about that D:\ path or the target compile platform

{??3a5c4d7950726f6a656347735c7365636f6e64576f726b5c416e63686f725c57696e5c52656c656173655c}

We’ll truncate it a bit more to remove the very specific build path:
?:\MyProjects\secondWork\Anchor\Win??\Release\

where ? represents a single character value
VTGrep: Example 2, Wildcards

https://support.virustotal.com/hc/en-us/articles/360000347157-VT-Intelligence#content-search

Wildcarding

Trickbot: 5739549850fe635fc0ac5de81ce1fd495669fcabc1b8ede35b82a22093c86399

https://www.virustotal.com/gui/file/5739549850fe635fc0ac5de81ce1fd495669fcabc1b8ede35b82a22093c86399/detection

D:\MyProjects\secondWork\Anchor\Win32\Release\anchorInstaller_x86.pdb

What if we weren’t sure about that D:\ path or the target compile platform

{??3a5c4d7950726f6a656374735c7365636f6e64576f726b5c416e63686f725c57696e????5c52656c65617365c}

We’ll truncate it a bit more to remove the very specific build path:

?:\MyProjects\secondWork\Anchor\Win??\Release\_

where ? represents a single character value
VTGrep: Example 3, Unicode

You need to read the OpenSSL FAQ, http://www.openssl.org/support/faq.html

Unicode

Bookworm: b2737192ea1c912daa3ca4c43224fb6afcc878c5e3303e86a459de06df7af33f

“%s”, AfxGetHttpRequestMgr %s => string of interest... but its in Unicode

Full String

https://www.virustotal.com/gui/search/content:%7B22002500730022002c0041006600780047006500740048007400700052006500710075006100730074004d0067007200200025007300

Common strings at the beginning and end will typically be skipped:
Fixed content:{410066007800470065007400480074007400700052006500710075006100730074004d00670072}
VTGrep: Example 3, Unicode

content:{22002500730022002c004100660078004700650074004800740070005200650071005006500730074004d0067007200}

https://support.virustotal.com/hc/en-us/articles/360000347157-VT-Intelligence#content-search

Unicode

Bookworm: b2737192ea1c912daa3ca4c43224fb6afcc878c5e3303e86a459de06df7af33f

https://www.virustotal.com/gui/file/5739549850fe635fc0ac5de81ce1fd495669fcabc1b8ede35b82a22903c86399/detection

"%s",AfxGetHttpRequestMgr %s => string of interest... but its in Unicode

We can just click on the string :-(

Full String

content:{22002500730022002c004100660078004700650074004800740070005200650071005006500730074004d0067007200200025007300}

https://www.virustotal.com/gui/search/content:%7B22002500730022002c004100660078004700650074004800740070005200650071005006500730074004d0067007200200025007300%7D/files
Fixed
content:{410066007800470065007400480074007400700052006500710075006100730074004d00670072}

https://www.virustotal.com/gui/search/content%253A%257B410066007800470065007400480074007400700052006500710075006100730074004d00670072%257D
VTGrep: Example 4, Logical “AND”

logout.log  CPU: %s(%d)  **RAM:** %lld Mb
WS2_32.dll  cmd /c %s
logout.log  %s %s: %s
ntdll.dll  %s %s: %s
cbomb.dat  GET /%s
data.dat  Host:%s

https://support.virustotal.com/hc/en-us/articles/360000347157-VT-Intelligence#content-search

Example:
Rietspoof:
f5c4782591675cd51ac3cdefd1bc719d576b7b98d529cf281b706d94fd1916c96 (bot)

These strings seem to be relatively common on their own… Can we combine them somehow?
VTGrep: Example 4, Logical “AND”

content:"logout.log" AND content:"data.dat" AND content:"RAM: "

Note: The “AND” is implied by spaces and is present by convention in this example. It is not necessary to specific AND between terms.

Example:
Rietspoof:
f5c4782591675cd51ac3cfd1bc719d576b7b98d529cf281b706d94fd1916cd96 (bot)

These strings seem to be relatively common on their own… Can we combine them somehow?

Yup we can!
content:"logout.log" AND content:"data.dat" AND content:"RAM: "
(content:{52 41 4d 3a 20} AND content:{64 61 74 61 2e 64 61 74} AND content:{6c 6f 67 6f 75 74 2e 6c 6f 67})
VTGrep: Example 5, Logical “OR”

FirstStage Dropper.dll is responsible for injecting SecondStageDropper.dll into another process to execute it. While the shellcode payload only contains code to search for and bypass EMET, FirstStage Dropper.dll also contains code for Kaspersky and Bitdefender. In case of EMET, it searches the loaded modules for emet.dll and emet64.dll, for Kaspersky it searches for klsihk.dll, and for Bitdefender it searches for avcuf32.dll and avcuf64.dll. It also collects and sends encrypted user system and process information data together with a unique hardcoded ID to the attacker's server. The data is sent to URLs that contain “/home/” and “/log/” directories and for encryption it uses the Rijndael algorithm. As the attacker server did not respond at the time of our analysis, we guess a command is sent back to execute the SecondStageDropper.dll.

https://support.virustotal.com/hc/en-us/articles/360000347157-VT-Intelligence#content-search

Chainshot:

Can we use the data from this report to find samples of Chainshot?

Yup we can!
Can we use the data from this report to find samples of Chainshot?

Yup we can!

- **CHAINSHOT Dropper Stages**
  - content:{46697273453461676544726F707065722E646C6C} OR content:{5365636F6E6453461676544726F707065722E646C6C}
  - content:FirstStageDropper.dll OR content:SecondStageDropper.dll
VTGrep: Example 6, Combine Logical Operators

https://support.virustotal.com/hc/en-us/articles/360000347157-VT-Intelligence#content-search

Rietspoof (loader)
No PDB: f5c4782591675cd51ac3cddf1bc719d576b7b98d529cf281b706d94fd1916c96
PDB: 9dd4032902d83367286ebc453e440a423625a3cb7b3191a55811a2d51b222986

https://www.virustotal.com/gui/search/f5c4782591675cd51ac3cddf1bc719d576b7b98d529cf281b706d94fd1916c96%250A9dd4032902d83367286ebc453e440a423625a3cb7b3191a55811a2d51b222986/files

It looks like Rietspoof’s loader occasionally has a PDB left in it… Sometimes it doesn’t. Can we account for that?
VTGrep: Example 6, Combine Logical Operators

https://support.virustotal.com/hc/en-us/articles/360000347157-VT-Intelligence#content-search

Rietspoof (loader)

No PDB: f5c4782591675cd51ac3cddf1bc719d576b7b98d529cf281b706d94fd1916c96
PDB: 9dd4032902d83367286ebc453e440a423625a3cb7b3191a55811a2d51b222986

It looks like Rietspoof’s loader occasionally has a PDB left in it… Sometimes it doesn’t. Can we account for that?

By combining logical operators, we can!

(content:{52 41 4d 3a 20} AND content:{64 61 74 61 2e 64 61 74}) AND content:{6c 6f 67 75 74 2e 6c 6f 67}) OR content:"G:\Work\Dr.Dre\hivez\new\loader\Release\loader.pdb"
VTGrep: Example 7, Search at offset with range
"MZP" {00} [0-10000] "virus" @0

content:{4d5a5000 [0-10000] 7669727573}@0
VTGrep: Example 8, Unpacked, OCR, macros, ...

content:"M4BUBc_ / Asc"
content:"Dim AWYbM5aiAlIClynkC7xb4gRO"
content:"JACKY ROSEN"
VTGrep: Pro tips

Prefer rare substrings.
Avoid long common substrings.
Particularly at the extremes.

Aaaah! Something went wrong here...

No results found due to unselective query.
Try avoiding extremely common substrings (tips):

content:{00 00 00 00} content:{CAFE 00 00 00 00 CAFE}
content:{CAFE 00 00 00 00 CAFE} content:{CAFE 00 00 ?? 00 00 CAFE}
content:"http://www.virustotal.com" content:"virustotal.co"
https://support.virustotal.com hc/en-us/articles/360000298637-VirusTotal-Graph
https://www.virustotal.com/graph/
VTGraph

A visualization tool built on top of VirusTotal’s data set. It understands the relationship between files, URLs, domains and IP addresses and it provides an easy interface to pivot and navigate over them.

https://support.virustotal.com/hc/en-us/articles/360000347157-VT-Intelligence#content-search

Start with f887e50af1c99ba73f280e28c7b0581b392782dba0bf2effc72d1719d039152b

### T1/SOC/ANALYST POV

Or

EmotetDoc: f887e50af1c99ba73f280e28c7b0581b392782dba0bf2effc72d1719d039152b
T1/SOC/ANALYST POV

Or

EmotetDoc: f887e50af1c99ba73f280e28c7b0581b392782db0bf2effc72d1719d039152b

#### T1/SOC/ANALYST POV

Or

**EmotetDoc:** f887e50af1c99ba73f280e28c7b0581b392782dba0bf2effc72d1719d039152b
Central Pivot, First Document Observed:
f887e50af1c99ba73f280e28c7b0581b392782dba0bf2effc72d1719d039152b

Full context around campaigns and indicators
IR POV

search for the Emotet C2: itw: www.xianjiaopi.com (expand graph too?) [Graph is a different perspective]
virustotal.com/graph

These campaigns are typically pretty broad, let's figure out the full scope, you can miss stuff when relying on just one view!

- Private graph for internal investigations
Central Pivot, First Document Observed: 
f887e50af1c99ba73f280e28c7b0581b392782dba0bf2effc72d1719d039152b

Full context around campaigns and indicators
IR POV

search for the Emotet C2: itw: www.xianjiaopi.com (expand graph too?) [Graph is a different perspective]
virustotal.com/graph

These campaigns are typically pretty broad, let's figure out the full scope, you can miss stuff when relying on just one view!

- Private graph for internal investigations
**Central Pivot, First Document Observed:**
`f887e50af1c99ba73f280e28c7b0581b392782dba0bf2efcc72d1719d039152b`

**Full context around campaigns and indicators**
**IR POV**

search for the Emotet C2: `itw: www.xianjiaopi.com` (expand graph too?) [Graph is a different perspective]
`virustotal.com/graph`

These campaigns are typically pretty broad, let's figure out the full scope, you can miss stuff when relying on just one view!

- Private graph for internal investigations
Central Pivot, First Document Observed:
f887e50af1c99ba73f280e28c7b0581b392782dba0bf2effc72d1719d039152b

Full context around campaigns and indicators
IR POV

search for the Emotet C2: itw: www.xianjiaopi.com (expand graph too?) [Graph is a different perspective]
virustotal.com/graph

These campaigns are typically pretty broad, let's figure out the full scope, you can miss stuff when relying on just one view!

- Private graph for internal investigations
Central Pivot, First Document Observed:
f887e50af1c99ba73f280e28c7b0581b392782dba0bf2effc72d1719d039152b

Full context around campaigns and indicators
IR POV

search for the Emotet C2: itw: www.xianjiaopi.com (expand graph too?) [Graph is a different perspective]
virustotal.com/graph

These campaigns are typically pretty broad, let's figure out the full scope, you can miss stuff when relying on just one view!

- Private graph for internal investigations
Try it Yourself!

Let’s escape from slide hell for a little bit!

https://www.virustotal.com/graph/

Image source: https://www.youtube.com/watch?v=Iug0OAUDz7M

---

- Single Point Expansion: Gameradon [*Power of Visual Pivot*]
  - 195.[.]62.53.126
  - https://www.virustotal.com/graph/g17f21463fdf54396a0bf05ff61385788cb4951a3b694c1c895fc1105b725dab

- OSINT Reporting
  - Option 1: Farseer,
    https://unit42.paloaltonetworks.com/farseer-previously-unknown-malware-family-bolsters-the-chinese-armoury/
  - Option 2: Babyshark,
    https://unit42.paloaltonetworks.com/new-babyshark-malware-targets-u-s-national-security-think-tanks/
    - https://www.virustotal.com/graph/g1766a5d086d84a6d859f1c598620f704925844d218584cdb81e6c57363485283

Option 4: Bronze Union (APT27/LuckyMouse), https://www.secureworks.com/research/a-peek-into-bronze-unions-toolbox
  ● https://www.virustotal.com/graph/gde1e0f5c8a3d41ef8f456d40df09c3b978e1e5f380034dfcad8adf4b9be2ca1a

  ● https://www.virustotal.com/graph/g7781518d548f4726a4d70d9189b5a6d12d01c88fc494fc78d187056974da545

Messy/Large Campaigns - Emotet
  ● 459397a134b2b4a201c2855bb2ed4d1eeda9cc7637d7c65201e0a78217a8780
  ● C060ca7e926c137d2a9b90d0182b288b86117430f8a7614a1bff92b722ee1fa6
  ● Source: https://paste.cryptolaemus.com/emotet/2019/03/14/emotet-malware-loCs_03-14-19.html

Emotet dropping Trickbot, https://www.malware-traffic-analysis.net/2019/03/13/index.html

Manual Expansion Using VTI Queries
  ● VTI Query - content:FirstStageDropper.dll OR content:SecondStageDropper.dll
https://yara.readthedocs.io/
https://github.com/InQuest/awesome-yara
https://support.virustotal.com/hc/en-us/articles/360000347157-VT-Intelligence#search-modifiers
What is YARA?

YARA is an acronym for: YARA: Another Recursive Acronym, or Yet Another Ridiculous Acronym. Pick your choice.

-- Victor M. Alvarez (@plusvic)

- Tool to assist malware researchers identify and classify malware
- Identify malware in string or binary patterns
- YARA rule = strings + condition
- Useful to catalog threat actors and associated IOCs

YARA is a tool aimed at (but not limited to) helping malware researchers to identify and classify malware samples. With YARA you can create descriptions of malware families (or whatever you want to describe) based on textual or binary patterns.

https://yara.readthedocs.io/
https://github.com/InQuest/awesome-yara
https://support.virustotal.com/hc/en-us/articles/36000347157-VT-Intelligence#search-modifiers
What is a YARA Rule?

```
sample-rule {
    strings:
        $a = "malicious_string"
        $b = {56 54 59}

    condition:
        $a or $b
}
```

By default strings are considered ASCII
Crafting a Custom YARA Rule

Malware family: **CobInt**

- PE file
- Typically < 30kb in size
- Specifically Named for an embedded DLL string
- *OPTIONAL:* Imphash Might be shared
- *OPTIONAL:* Interesting Function Calls
Crafting a Custom YARA Rule (2)

Sample #1

<table>
<thead>
<tr>
<th>STRINGS</th>
<th>HEX</th>
</tr>
</thead>
<tbody>
<tr>
<td>KERNEL32.dll</td>
<td>9540c062e1aefdb78e1f3f0b40c7f9d7f1a7c7f90f2748e369a7d2e6fe4a6bb</td>
</tr>
<tr>
<td>SHLPI.dll</td>
<td>2f7b5219193541ae993f5cf87a1f6c07705aaa907354a6292bc5c8d8585e8bd1</td>
</tr>
<tr>
<td>ADVAPI32.dll</td>
<td>int.dll</td>
</tr>
<tr>
<td>WININET.dll</td>
<td>int.dll</td>
</tr>
<tr>
<td>urlmon.dll</td>
<td>int.dll</td>
</tr>
</tbody>
</table>

Sample #2

<table>
<thead>
<tr>
<th>STRINGS</th>
<th>HEX</th>
</tr>
</thead>
<tbody>
<tr>
<td>KERNEL32.dll</td>
<td>9540c062e1aefdb78e1f3f0b40c7f9d7f1a7c7f90f2748e369a7d2e6fe4a6bb</td>
</tr>
<tr>
<td>SHLPI.dll</td>
<td>2f7b5219193541ae993f5cf87a1f6c07705aaa907354a6292bc5c8d8585e8bd1</td>
</tr>
<tr>
<td>ADVAPI32.dll</td>
<td>int.dll</td>
</tr>
<tr>
<td>WININET.dll</td>
<td>int.dll</td>
</tr>
<tr>
<td>urlmon.dll</td>
<td>int.dll</td>
</tr>
</tbody>
</table>

Sample 1: 9540c062e1aefdb78e1f3f0b40c7f9d7f1a7c7f90f2748e369a7d2e6fe4a6bb
Sample 2: 2f7b5219193541ae993f5cf87a1f6c07705aaa907354a6292bc5c8d8585e8bd1
Crafting a Custom YARA Rule (4)

Both Samples

- ADVAPI32.dll
- IPHLPAPI.DLL
- KERNEL32.dll
- WININET.dll
- urlmon.dll

<table>
<thead>
<tr>
<th>Basic Properties</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>MDS</td>
<td>616799072a11d95373b3c38626adae93</td>
</tr>
<tr>
<td>SHA-1</td>
<td>572010d3d3a0b1185f5b5967d39275c2c81b1494</td>
</tr>
<tr>
<td>Authenthash</td>
<td>46a3defc73709160dbcf01aad0b02c2946c71641504229304244a1bc8bd6d32c0</td>
</tr>
<tr>
<td>Imphash</td>
<td>0f476779121b8ccabebe0b20953054bcb</td>
</tr>
<tr>
<td>SSDEEP</td>
<td>192/jdu6/08f/04f/01t/0egpp/27qawnd/3d/fv85p6rzhf/pdq/A8b4pP</td>
</tr>
<tr>
<td>File type</td>
<td>Win32 DLL</td>
</tr>
<tr>
<td>Magic</td>
<td>PE32 executable for MS Windows (DLL) (GUI) Intel 80386 32-bit</td>
</tr>
<tr>
<td>File size</td>
<td>10.5 KB</td>
</tr>
</tbody>
</table>
Crafting a Custom YARA Rule (5)

strings:

    // interesting strings
$s1 = "int.dll"
$s2 = "ReflectiveLoader"
$s3 = "ObtainUserAgentString"
Crafting a Custom YARA Rule (8)

```yara
import "pe"
rule apt_win_cobint_dll : Cobalt_Group
{
    strings:
        // interesting strings
        $s1 = "int.dll"
        $s2 = "ReflectiveLoader"
        $s3 = "ObtainUserAgentString"

    condition:
        uint16(0)==0x5a4d
        and (all of them or
            pe.imphash()== "9fd476779121c8ccabe0e029935bcbcb"
        )
        and filesize < 30KB
}
```

Strings are assumed to be ASCII by default.

if you want to indicate strings are unicode, use the wide modifier

If you want to include both ASCII and Unicode strings, use the two modifiers, ascii unicode, after your closing quote.
Retrohunt Results

Hunter/Researcher POV

Demo hunt UI = forward looking https://www.virustotal.com/intelligence/hunting/
Retrohunt = retrospective

Search old Notifications UI for “Cobalt_Group” to get CobInt notifications. Retrohunt
should be done as well.
-----

If people are interested show rule for pivoting on:
DHS Cosmos Backdoor:
820ca1903a30516263d630c7c08f2b95f7b65dffceb21129c51c9e21cf9551c6

-----

Embedded Resource Pivoting:
a76c79a4146cdf5cc1fb99ee7fce96da94d2dca00c029056bc1b7683058c02e3 (ursnif)
=> rsrsrc image pivot => Yara Rule
### Retrohunt Notifications

<table>
<thead>
<tr>
<th>Hash</th>
<th>File Size</th>
<th>Date Seen</th>
<th>Submitters</th>
</tr>
</thead>
<tbody>
<tr>
<td>d386ba740153b645937c32b64e69d04102134008e7673c69068e3ef903c</td>
<td>10.5 KB</td>
<td>2018-07-06 15:33:08</td>
<td>3 submissions</td>
</tr>
<tr>
<td>9767c1803394a5e85a49d9e59cd3c525b1d8c7f174zae5c15585344040e4d03502c</td>
<td>10.5 KB</td>
<td>2018-09-19 18:15:57</td>
<td>1 submission</td>
</tr>
<tr>
<td>a328078d3529d03c4a40e8a59532d5073b47005d79edc4e9e5064a380e35f6b</td>
<td>11.01 KB</td>
<td>2018-01-25 05:59:40</td>
<td>5 submissions</td>
</tr>
<tr>
<td>44c15f1ebd8a5fd8e1e727144525a0e2f1986d9565683650a6e35d80e1958b087</td>
<td>8.12 KB</td>
<td>2018-08-14 15:33:16</td>
<td>1 submission</td>
</tr>
<tr>
<td>1b9b9e35b5e0f78f152e3a2a64931fae8b2e6f95f62e9e1960347</td>
<td>24 KB</td>
<td>2017-12-26 12:48:24</td>
<td>1 submission</td>
</tr>
<tr>
<td>b3a0c6b9e0d62c3e20d4024f1e0573277badd75e4772b893d9f777f0d03504</td>
<td>10.9 KB</td>
<td>2018-09-19 15:44:40</td>
<td>2 submissions</td>
</tr>
<tr>
<td>68d5c6d2d61e88b989f8d45c7637774707478e3d8a7d0d6e4d8b8f6956bc033a5e401106b1ebe</td>
<td>28 KB</td>
<td>2018-09-04 19:45:49</td>
<td>2 submissions</td>
</tr>
<tr>
<td>0964b2a0d2a771v5c40f99e3v3f771712569837898d00c334a1e798422dz2</td>
<td>10.5 KB</td>
<td>2018-07-05 14:10:53</td>
<td>3 submissions</td>
</tr>
</tbody>
</table>
# LiveHunt Results

## LIVEHUNT NOTIFICATIONS

<table>
<thead>
<tr>
<th>hash</th>
<th>name</th>
<th>size</th>
<th>date matched</th>
<th>first seen</th>
<th>submissions</th>
<th>submitters</th>
</tr>
</thead>
<tbody>
<tr>
<td>44b5685527babba721f81ca4ce14e4e8da5f79c6d5668146d2e9145f1ff1d</td>
<td>int.dl</td>
<td>11 KB</td>
<td>2018-11-20 10:58:12</td>
<td>2018-03-26 14:10:37</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>44b5685527babba721f81ca4ce14e4e8da5f79c6d5668146d2e9145f1ff1d</td>
<td>int.dl</td>
<td>11 KB</td>
<td>2018-11-20 10:58:10</td>
<td>2018-03-26 14:10:37</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>44b5685527babba721f81ca4ce14e4e8da5f79c6d5668146d2e9145f1ff1d</td>
<td>int.dl</td>
<td>11 KB</td>
<td>2018-11-20 10:58:10</td>
<td>2018-03-26 14:10:37</td>
<td>3</td>
<td>2</td>
</tr>
</tbody>
</table>

*Disclaimer: The provided hash is for demonstration purposes only.*

---

https://www.virustotal.com/gui/hunting/notifications/cobalt_group
But, CobInt doesn’t use int.dll anymore!

```python
import "pe"
rule apt_win_cobint_dll : Cobalt_Group
{
    strings:
        // interesting strings
        $s1 = "/[a-zA-Z]{3}\.dll/ ascii"
        $s2 = "ReflectiveLoader"
        $s3 = "ObtainUserAgentString"

        condition:
            uint16(0)==0x5a4d
            and (all of them or
            pe.imphash()== "9fd476779121c8ccabe0e029935bcbc"
            )
            and filesize < 30KB
}
```

Regex!
Basic Rule Dev 1:
I want to use all the strings

https://yara.readthedocs.io/
https://github.com/InQuest/awesome-yara
https://support.virustotal.com/hc/en-us/articles/360000347157-VT-Intelligence#search-modifiers

Rietspoof (bot):
8ea856534561e1fbc8c13c8901c9c8f7eb6139e76ef56ea8f9137c2295199
https://www.virustotal.com/gui/file/8ea856534561e1fbc8c13c8901c9c8f7eb6139e76ef56ea8f9137c2295199/submissions

---I’m going to have to do this in a notepad I think, the rule is too big to be visible in slides---

rule trojan_win_rietspoof_bot : commodity
{
    meta:
        description = "Identify Reitspoof Bot"
author = "blevene@chronicle.security"
date = "20-02-2019" //dd-mm-yyyy
hash01 = "8ea856534561e1fbfe8c13c8901c0c9c8f7eb6139e76ef5eea8f9137c2295199"

strings:

$ = "cbomb.dat" wide
$ = "Secur32.dll" wide
$ = "CreatePipe"
$ = "PeekNamedPipe"
$ = "WS2_32.dll"

condition:
  uint16(0)==0x5a4d
  and all of them
Basic Rule Dev 2: 
I need to employ more selective logic for my strings

https://yara.readthedocs.io/ 
https://github.com/InQuest/awesome-yara 
https://support.virustotal.com/hc/en-us/articles/360000347157-VT-Intelligence#search-modifiers

Rietspoof (loader) 
No PDB: f5c4782591675cd51ac3cddf1bc719d576b7b98d529cf281b706d94fd1916c96 
PDB: 9dd4032902d83376286ebc453e440a423625a3cb7b3191a55811a2d51b222986 

---Again, going to have to do this in a Notepad---

rule trojan_win_rietspoof_loader : commodity 
{ 
   meta: 
       description = "Identify Reitspoof Loader Phase" 
       author = "blevene@chronicle.security" 
       date = "20-02-2019" //dd-mm-yyyy 
       reference =
hash01 = "f5c4782591675cd51ac3cddf1bc719d576b7b98d529cf281b706d94fd1916c96"
hash02 = "d7a15001a45c6157f0b2ed728a88cc9db09ed39e733310e76bd906ccdf527a4e"

strings:

$s1 = "CPU: %s(%d)"
$s2 = "data.dat"
$s3 = "Host:%s"
$s4 = "logout.log"
$s5 = "RAM: "
$s6 = "WScript"

//old PDB
$pdb = "G:\Work\Dr.Dre\hivez\new\loader\Release\loader.pdb"

condition:
  uint16(0)==0x5a4d
  and ( all of ($s*) or $pdb)
Modules

Modules are the method YARA provides for extending its features. They allow you to define data structures and functions which can be used in your rules to express more complex conditions.

YARA is a tool aimed at (but not limited to) helping malware researchers to identify and classify malware samples. With YARA you can create descriptions of malware families (or whatever you want to describe) based on textual or binary patterns.

https://yara.readthedocs.io/
https://github.com/InQuest/awesome-yara
https://support.virustotal.com/hc/en-us/articles/360000347157-VT-Intelligence#search-modifiers

Important => https://yara.readthedocs.io/en/v3.9.0/modules.html
Writing your own modules =>
https://yara.readthedocs.io/en/v3.9.0/writingmodules.html#writing-modules (Its in C)
YARA is a tool aimed at (but not limited to) helping malware researchers to identify and classify malware samples. With YARA you can create descriptions of malware families (or whatever you want to describe) based on textual or binary patterns.

https://yara.readthedocs.io/
https://github.com/InQuest/awesome-yara
https://support.virustotal.com/hc/en-us/articles/360000347157-VT-Intelligence#search-modifiers

https://yara.readthedocs.io/en/v3.9.0/modules/pe.html

“The PE module allows you to create more fine-grained rules for PE files by using attributes and features of the PE file format. This module exposes most of the fields present in a PE header and provides functions which can be used to write more expressive and targeted rules.”
import "pe"
import "hash"

rule trojan_win_ursnif_resource : Commodity
{
    meta:
        description = "Identify Ursnif/Gozi/ISFB samples seen on 10/26/2018"
        author = "blevene@chronicle.security"
        hash01 = "a76c79a4146cf5cc1fb99ee7f8ce96da94d2dca00c029056bc1b7683058c02e3"

    condition:
        uint16(0)==0x5a4d
        and filesize < 600KB
        and for any i in (0..pe.number_of_resources - 1):
            (hash.sha256(pe.resources[i].offset, pe.resources[i].length) == "059f9bf1ced9a989daedcde2df32db54318347d3975f343aaf8d123d0ca517d")
}

YARA is a tool aimed at (but not limited to) helping malware researchers to identify and classify malware samples. With YARA you can create descriptions of malware families (or whatever you want to describe) based on textual or binary patterns.

https://yara.readthedocs.io/
https://github.com/InQuest/awesome-yara
https://support.virustotal.com/hc/en-us/articles/360000347157-VT-Intelligence#search-modifiers


The Hash module allows you to calculate hashes (MD5, SHA1, SHA256) from portions of your file and create signatures based on those hashes.

Bonus, we are a looping!
https://yara.readthedocs.io/en/v3.9.0/writingrules.html#iterating-over-string-occurrences

Sample:
https://www.virustotal.com/gui/file/a76c79a4146cfd5cc1fb99ee7fce96da94d2dca00c029056bc1b7683058c02e3/detection
YARA is a tool aimed at (but not limited to) helping malware researchers to identify and classify malware samples. With YARA you can create descriptions of malware families (or whatever you want to describe) based on textual or binary patterns.

https://yara.readthedocs.io/
https://github.com/InQuest/awesome-yara
https://support.virustotal.com/hc/en-us/articles/360000347157-VT-Intelligence#search-modifiers

math.entropy(offset, filesize) => Returns the entropy for size bytes starting at offset

“The Math module allows you to calculate certain values from portions of your file and create signatures based on those results.”
Bonus: Looping in Yara!
https://yara.readthedocs.io/en/v3.9.0/writingrules.html#iterating-over-string-occurrences

atmos = Zeus Variant
https://www.virustotal.com/gui/file/16ce22397e8261714a272d82627bb3a55b65d7f4e65e0d54acfb3c5ed37e68cc/detection
Practical Applications 1

Mimikatz

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

Example:
b04e58327191222e27405c2dc4871cb4c81e3ea732d70c67ad744088619c97e4 (mimikatz)
https://github.com/gentilkiwi/mimikatz
Mimikatz, Option 1
Strings Only

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

Example:
b04e58327191222e27405c2dc4871cb4c81e3ea732d70c67ad744088619c97e4
(mimikatz)
https://www.virustotal.com/gui/file/b04e58327191222e27405c2dc4871cb4c81e3ea732d70c67ad744088619c97e4/detection

Ref: https://github.com/gentilkiwi/mimikatz

---Will do in a notepad---

Option 1, strings only

rule hacktool_win_mimikatz_dll_option1_strings : hacktool
{
  meta:
    description = "Identify stock Mimikatz DLL. Example rule."
    author = "blevene@chronicle.security"
    date = "03/15/2019" //mm/dd/yyyy
    hash =
strings:

//exports
$e1 = "InitializeChangeNotify" ascii fullword
$e2 = "PasswordChangeNotify" ascii fullword
$e3 = "SpLsaModeInitialize" ascii fullword

//imports
$i1 = "NdrMesTypeFree2" ascii fullword
$i2 = "MesHandleFree" ascii fullword
$i3 = "RtlStringFromGUID" ascii fullword
$i4 = "GetOEMCP" ascii fullword

condition:
/* option 1:
   all of them
*/
/*option 2:
   all of ($e*) and all of ($i*)
*/
}
Practical Applications 1

Mimikatz, Option 2
PE Exports

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

Example:
b04e58327191222e27405c2dc4871cb4c81e3ea732d70c67ad744088619c97e4
(mimikatz)
https://www.virustotal.com/gui/file/b04e58327191222e27405c2dc4871cb4c81e3ea732d70c67ad744088619c97e4/detection

Ref: https://github.com/gentilkiwi/mimikatz

---Will do in a notepad---

import "pe"
rule hacktool_win_mimikatz_dll_option2_exportsonly : hacktool
{
    meta:
        description = "Identify stock Mimikatz DLL. Example rule."
        author = "blevene@chronicle.security"
        date = "03/15/2019" /mm/dd/yyyy
        hash = "b04e58327191222e27405c2dc4871cb4c81e3ea732d70c67ad744088619c97e4"
condition:
  pe.exports("InitializeChangeNotify")
  and pe.exports("PasswordChangeNotify")
  and pe.exports("SpLsaModelInitialize")
}
Practical Applications 1

Mimikatz, Option 3
PE Imports

Example:
b04e58327191222e27405c2dc4871cb4c81e3ea732d70c67ad744088619c97e4
(mimikatz)
https://www.virustotal.com/gui/file/b04e58327191222e27405c2dc4871cb4c81e3ea732d70c67ad744088619c97e4/detection

Ref: https://github.com/gentilkiwi/mimikatz

---Will do in a notepad---

import "pe"
rule hacktool_win_mimikatz_dll_option3_importsonly : hacktool
{
  meta:
    description = "Identify stock Mimikatz DLL. Example rule."
    author = "blevene@chronicle.security"
    date = "03/15/2019" //mm/dd/yyyy
    hash =
        "b04e58327191222e27405c2dc4871cb4c81e3ea732d70c67ad744088619c97e4"
condition:
    pe.imports("RPCRT4.dll", "MesHandleFree")
and pe.imports("RPCRT4.dll", "NdrMesTypeFree2")
and pe.imports("ntdll.dll", "RtlStringFromGUID")
and pe.imports("KERNEL32.dll", "GetOEMCP")
}
Practical Applications 2

DustSquad
“Octopus” Implant

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

2d5f3edc4132f463cb6efe6379fda46e00fb7225f51a9fb69d2b11161c43faa6
2af44715d4f0655bd50d30d46b01336b7f7743ade6b78e2e7650a8d60dc35858
ciaaf10e6f65d630130c04453160596eada9a5b78167c934e9ea3e8baffa2c345

https://www.virustotal.com/gui/search/2d5f3edc4132f463cb6efe6379fda46e00fb72
25f51a9fb69d2b11161c43faa6%250A2af44715d4f0655bd50d30d46b01336b7f7743a
de6b78e2e7650a8d60dc35858%250Acaaf10e6f65d630130c04453160596eada9a5b7
8167c934e9ea3e8baffa2c345/files

---Will do in a notepad---

import "pe"
rule apt_win_octopus : DustSquad {
    meta:
        description = "Identify potential DustSquad 'octopus' implants"
        graph = "https://www.virustotal.com/graph/g4c327ce3e88e43f99191d7618b1b74e4eefee8ccd2e44451ae8aa49ac1a36e47"
date = "10-16-2018"
author = "blevene@chronicle.security"
hash01 = "2d5f3edc4132f463cb6efe6379fda46e00fb7225f51a9fb69d2b11161c43faa6"
hash02 = "2af44715d4f0655bd50d30d46b01336b7f7743ade6b78e2e7650a8d60dc35858"
hash03 = "caaf10e665d630130c04453160596eada9a5b78167c934e9ea3e8baffa2c345"

strings:

// unicode strings
$u1 = "Download:" wide
$u2 = "Remove:" wide
$u3 = "Embaracdero" wide
$u4 = "php?check" wide

condition:
uint16(0)==0x5a4d
    and ( all of them
or
        pe.imphash()== "65ffe87ad21cc53609d3db7bc15603b0"
    or
        vhash == "0360b6666d5c0d5d151c003232z5e002c025z8035z23z303cz1"
    )
    and filesize < 5MB
}
LuckyCat
“ExileRat” Implant

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

3eb026d8b778716231a07b3dbbdc99e2d3a635b1956de8a1e6efc659330e52de
https://www.virustotal.com/gui/file/3eb026d8b778716231a07b3dbbdc99e2d3a635b1956de8a1e6efc659330e52de/detection

---Will do in a notepad---

rule apt_win_exilerat : LuckyCat
{
    meta:
        description = "Identify ExileRat as described by Talos"
        author = "blevene@chronicle.security"
        date = "04-02-2019" //dd-mm-yyyy
        hash01 = "3eb026d8b778716231a07b3dbbdc99e2d3a635b1956de8a1e6efc659330e52de"

    strings:

//filewrites
$f1 = "prepare to write file %s, %d, %d"
$f2 = "end write file %s, %d"

//pdb
$pdb = "D:\proj\proj vs2015\scout\Release\scout.pdb"

//hardcoded IPs
$ip1 = "27.126.188.212"
$ip2 = "192.100.106.207"

condition:
    uint16(0)==0x5a4d
    and (all of ($f*)
or $pdb
    or 1 of ($ip*)
    )
Practical Applications 4

Lazarus
“RisingSun” Implant

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

37b04dcdfdcba885f0f392524db7ae7b73806ad8a8e76fbc6a2df4db064e71
https://www.virustotal.com/gui/file/37b04dcdfdcba885f0f392524db7ae7b73806ad8a8e76fbc6a2df4db064e71/detection

---Will do in a notepad---

rule apt_win_RisingSun : Lazarus
{
  meta:
    description = "Identify RisingSun Implant which Mcafee Aleeges is related to Lazarus' Duuzer"
    author = "blevene@chronicle.security"
    date = "12-12-2018" //dd-mm-yyyy
  hash01 =
strings:

$ = "Accept-Language: en-us;q=0.8;q=0.6,en-us;q=0.4,en;q=0.2" wide
$ = "charset={A-Za-z0-9\-\_\}+\)" wide
$ = "Content-Length: \{[0-9]+\}" wide
$ = "Location: \{[0-9]+\}" wide
$ = "q\"[^\"]*\"\)\([^"]*\)" wide
$ = "Set-Cookie:\b*{.+?}\n" wide
$ = "%s%d&page=result%s%d" ascii

$uri = "%s%d&page=" ascii

condition:

    uint16(0)==0x5a4d
    and #uri > 3
    and all of them
CobaltGang
“ShapesMacro” Dropper

https://support.virustotal.com/hc/en-us/articles/360000347157-VirusTotal-Intelligence

2a8c62c4e167f9f52c2c5a4fe5be96df53d1f6015dd793747391775e34d16fbf
https://www.virustotal.com/gui/file/2a8c62c4e167f9f52c2c5a4fe5be96df53d1f6015dd793747391775e34d16fbf/detection

---Will do in a notepad---

rule apt_win_shapesmacro_cobaltdropper : Cobalt_Gang
{
    meta:
        description = "Identify malicious office documents which abuse the Shapes function"
        author = "blevene@chronicle.security"
        date = "20-12-2018" //dd-mm-yyyy
        reference = "https://twitter.com/dissectmalware/status/1064977287915950080?lang=en"
        hash01 = "2a8c62c4e167f9f52c2c5a4fe5be96df53d1f6015dd793747391775e34d16fbf"
strings:

$office = { D0 CF 11 E0 A1 B1 1A E1 }

$s1 = "ThisProject.ThisDocument.AutoOpen" wide nocase
$s2 = "Shapes" ascii nocase
$s3 = "Shell" ascii nocase
$s4 = "TextFrame" ascii
$s5 = "TextRange" ascii
$s4 = "ThisProject" ascii

$var = "var" ascii nocase

condition:
  $office at 0
  and #var > 8
  and all of ($s*)
YARA Tools on VT

https://www.virustotal.com/gui/hunting/retrohunt
Beside hunting for files in real time as they arrive to VirusTotal, you can also apply your YARA rules to files sent in the past with the Retrohunt feature. The concept is plain simple: just put your YARA rules in the provided text box, launch your Retrohunt job and you'll get a list of files matching your rules. The process can take a few hours, as it scans multiple terabytes of data, but you can provide an email address in order to be notified when the scanning finishes.

However, notice that none of the Malware Hunting-specific features will work with Retrohunt, including rules based on the number of positives, antivirus signatures, tags, file type and Cuckoo's behaviour reports. Only pure YARA rules will work.
LiveHunt allows you to hook into the stream of files submitted to VirusTotal and get notified whenever one of them matches a certain rule written in the YARA language. Applying YARA rules to the files submitted to VirusTotal you should be able to get a constant flow of malware files classified by family, discover new malware files not detected by antivirus engines, collect files written in a given language or packed with a specific run-time packer, create heuristic rules to detect suspicious files, and, in general, enjoy the benefits of YARA's versatility acting on the huge amount of files processed by VirusTotal every day.

For more information, visit the following links:

- [https://www.virustotal.com/gui/hunting/notifications](https://www.virustotal.com/gui/hunting/notifications)
- [https://www.virustotal.com/gui/hunting/rulesets](https://www.virustotal.com/gui/hunting/rulesets)
LiveHunt: VirusTotal Externals

AntiVirus Detection Externals

rule av_externals_example
{
    condition:
        signatures contains “Trojan”
or
        eset_nod32 contains “Backdoor”
}

https://www.virustotal.com/gui/hunting/notifications
https://www.virustotal.com/gui/hunting/rulesets


In malware hunting your rules can take into account not only the contents of the file itself, but also the signatures generated by the different antivirus engines that scanned the file, which means that you can construct rules stating: "give me the files containing the strings 'foo' and 'bar', and detected by more than two antivirus vendors" or "give me the files detected by antivirus X" or "give me new files that antivirus X detects as 'baz'".
In malware hunting your rules can take into account not only the contents of the file itself, but also the signatures generated by the different antivirus engines that scanned the file, which means that you can construct rules stating: "give me the files containing the strings 'foo' and 'bar', and detected by more than two antivirus vendors" or "give me the files detected by antivirus X" or "give me new files that antivirus X detects as 'baz'".

https://www.virustotal.com/gui/hunting/notifications
https://www.virustotal.com/gui/hunting/rulesets

LiveHunt: VirusTotal Externals

VirusTotal Externals

rule VT_Tags_example
{
    condition:
        file_type contains "document"
        and file_name contains "invoice"
        and new_file
}

https://www.virustotal.com/gui/hunting/notifications
https://www.virustotal.com/gui/hunting/rulesets

In malware hunting your rules can take into account not only the contents of the file itself, but also the signatures generated by the different antivirus engines that scanned the file, which means that you can construct rules stating: "give me the files containing the strings 'foo' and 'bar', and detected by more than two antivirus vendors" or "give me the files detected by antivirus X" or "give me new files that antivirus X detects as 'baz'".
LiveHunt: VirusTotal Externals Example 2

rule mueller_report_lures : current_event
{
    meta:
        description = "Identify mueller report lures"

    condition:
        (file_type contains "document" or file_type contains "email") and new_file and
        (file_name contains "mueller"
        or file_name contains "mueller report"
        or (file_name contains "mueller" and file_name contains "report")
        )

}


https://www.virustotal.com/gui/hunting/notifications
https://www.virustotal.com/gui/hunting/rulesets


In malware hunting your rules can take into account not only the contents of the file itself, but also the signatures generated by the different antivirus engines that scanned the file, which means that you can construct rules stating: "give me the files containing the strings 'foo' and 'bar', and detected by more than two antivirus vendors" or "give me the files detected by antivirus X" or "give me new files that antivirus X detects as 'baz'".
APIv3 - VTCLI

https://asciinema.org/a/179696

https://developers.virustotal.com/v3.0/reference

https://github.com/VirusTotal/vt-cli
Helpful Tool for APIv3

https://github.com/VirusTotal/vt-cli

https://asciinema.org/a/179696

https://developers.virustotal.com/v3.0/reference

https://github.com/VirusTotal/vt-cli
Displayed output is abbreviated in the slide (it doesn’t all fit)

Truncated the results in the raw output as well, ya’ll get the point, hopefully.

blevene@blevene-imacpro ~ vt file
b331ae16014d6219f1e2e3a2e2d568e7836bfbb0e6b40ec081a3c71edd508a37
 - file <b331ae16014d6219f1e2e3a2e2d568e7836bfbb0e6b40ec081a3c71edd508a37>:  
  bundle_info:
  extensions:
    bin: 1
    jpg: 1
    xml: 14
  file_types:
    JPG: 1
    Microsoft Office: 1
    XML: 18
  highest_datetime: "1980-01-01 00:00:00"
  lowest_datetime: "1980-01-01 00:00:00"
  num_children: 20
  type: "DOCX"
  uncompressed_size: 149149
vhash: "7905a57c5a030c4ad3a9e777cc1bd352"
creation_date: 1553645460  # 2019-03-26 20:11:00 -0400 EDT
downloadable: true
exiftool:
  AppVersion: "16.0"
  Application: "Microsoft Office Word"
  Characters: "3"
  CharactersWithSpaces: "3"
  Company: "VPS2day"
  CreateDate: "2019:03:27 00:11:00Z"
  Creator: "VPS2day"
  DocSecurity: "None"
  FileType: "DOCM"
  FileTypeExtension: "docm"
  HeadingPairs: "Title, 1"
  HyperlinksChanged: "No"
  LastModifiedBy: "VPS2day"
  Lines: "1"
  LinksUpToDate: "No"
  MIMEType: "application/vnd.ms-word.document.macroEnabled"
  ModifyDate: "2019:03:31 19:23:00Z"
  Pages: "1"
  Paragraphs: "1"
  RevisionNumber: "31"
  ScaleCrop: "No"
  SharedDoc: "No"
  Template: "Normal.dotm"
  TotalEditTime: "1.8 hours"
  Words: "0"
  ZipBitFlag: "0x0006"
  ZipCRC: "0x2a6675f6"
  ZipCompressedSize: "445"
  ZipCompression: "Deflated"
  ZipFileName: "[Content_Types].xml"
  ZipModifyDate: "1980:01:01 00:00:00"
  ZipRequiredVersion: "20"
  ZipUncompressedSize: "1900"
first_submission_date: 1554194217  # 2019-04-02 04:36:57 -0400 EDT
last_analysis_date: 1554194217  # 2019-04-02 04:36:57 -0400 EDT
last_analysis_results:
  <---TRUNCATED--->
blevene@blevene-imacpro ~/Malware_Stuff vt url -i first_submission_date virustotal.com
url <a354494a73382ea0b4bc47f4c9e8d6c578027cd4598196dc88f05a22b5817293>:
first_submission_date: 1276681241 # 2010-06-16 05:40:41 -0400 EDT
Where -l is identifiers only (hashes) and -n is number of results

✗ bleepne@blevene-imacpro ~ vt search "metadata:VPS2day size:75KB+ size:100KB-" -l -n 25
b331ae16014d6219f1e2e3a2e2d568e7836bfb0b0e6b40ec081a3c71edd508a37
3b8f3dd364e1ae8222b4cf53052eb1a2ed8eca3c18b3ee36a2e0da94fc20023a
7564707aff8b852b6dd91ba86876965aa6f9c78ed910b95a157c022c142cb4765
7430193891a295771ef76047d6e5a79656adb3f8edac0631f012111e4d8587aa
20150fb9e6d1245c1c4be577dfae1e198a474ab39595cd19163bea6da6645ceb
d3f153e3ee3c3d42d12b41a4c74c64c7e9f2de297861d59f0e71940c786996
b742caee4f394368f62d61f499466e20085106166e21b2d3aefb3c8f18708f17
2589053addc1f8350d7c04cf72eb87e94d5ac5b4d4d79f3b1664e913739
f57fe3d626d52360d9b53b8e39d626543d07272eae4f21b7d417f715d9
93f9abac3c60f33cedc3c1e2a37644c47e9f2de26c97861d504fa71940c706096
b0f42caee82f2493068f26df1af49946e620058106166be21b63baefcbeb0180f7810
2589053adcf1803c667dfd5ac4f4df0c8161b8a48ae2c30200cf5e3b8b79579
9f1c60dfe106ec7c0d6a86319b60af6dc727a7e4d429baf1f64b4d417f715d9
93fa8abcafb60fd3ebd7ec31cce252605a225e1f5302ba8c19dadb9cc92229
62247624c068970cd4cf255cb07e8c94cdac59bd44de7e99f3db1664cede0913789
f57fe3d62c6052360d9a0d52b486ce8a93a0f518a1eb32bbc8ea52675254d777
f2e55acc8c59b0a0d9a5f6d4cde720a44f56e0918603b81520b3709b2ab
8445b6e4cc719aadbd734be767a6790e6cd3654b84d15f404bca9a8e64794383b37
f46ab59d7c9aee2d76f156fddec348affcc8d16dfba47f8aa85a8d2e31168a89b8
6354de716654177ebff8705971ee33e732708734eb2f3f3d0a9cdbab95ab22bb
4134c7ed5bd2f8894e8311031959ff6a367e324e127366dafa133f538096f215a
984362a42eb6c372f22a0c90cece3df3ab71169eaa7853d23c4228e5d8caacb
APIv3 - Search and Download VTI

Where -d is download and -n is number of results

✘ blevene@blevene-imacpro ~/Malware_Stuff vt search "metadata:VPS2day size:75KB+ size:100KB-" -n 25 -d
20150fb9e6d1245c1c4be577daee198a474ab39595cd191963bea6da6645ceb [ok]
b331ae16014d6219f1e2e3a2e2d568e7836bfbb0e6b40e081a3c71edd508a37 [ok]
3b8f3df364e1ae822b4cf53052eb1a2ed8eca3c18b3ee36a2e0da94fc20023a [ok]
7430193891a295771ef76047dede5a7965aadb23f86dac0631f10211e4d8578a [ok]
7564707affb852bb6dd91ba86876965a6f9c7e8ded109b9a157c022c1402cb4765 [ok]
9f1c60dfe106ec7cbad8a68319b60af6dc727a4e4d429baf1f64b4d417f715d [ok]
2589053adcf1830c667df5ac4afdf0c8161b8a48ae2c30200cfe5e3b8bb7957 [ok]
d3f153e3ee3cc3d14212e14a37644c47e9f2d26c97861d504efa7e191600c79506 [ok]
93fa8abca6f0d33edbc7ee3c1c25620a5a225e1f5032ba8c19dcbca9cc92229 [ok]
bof42ca8e8f2493068f26df14f946e620058106166e21bd63baefcb180f7810 [ok]
62247624oc68970cd4cfc25cb07e8c94cdac59bd44de799f3bd1664ce0913789 [ok]
f57fe3ed626c052360dad052b486ce8a93af0518aa1eb32b8ca5a526752f41d77 [ok]
f2e5acc8c59b0ae0d9af5d6fc4cde720ab44f56e0918603b81520b3709b2ab [ok]
8445b6e4cc719abadd7d3428e7f67aecc63d654b84d15f404bca9a8e6474983b37 [ok]
f4e6ab59d79acee2d76f156fdec348aafccc8d16dbf47f8a85a8e231168a89b8 [ok]
6354d671654177ebff8705971ee33e373280734eb2f3d0a9cdaab95ab22bb [ok]
984362a42eb6c372f22a0c90cece3d3f3ab71169eaa7853d23c4b228e5dcaacb [ok]
4134c7edf5bd2f8894e83110319596f637e324e172366dafa1335f38096f215a [ok]
Where -I is identifiers only (hashes)

blevene@blevene-imacpro ~ vt search "content:FirstStageDropper.dll OR content:SecondStageDropper.dll" -I
feaa627fa65c452b75522ea3633e51f1842fc7577a523d43c5ea529c8aa08713 3485c9b79dfe88af9347326b9ccf588018a608f89ecd6597da552e3872f
a09273b4cc08c39afe0c964f14ce98e532ae530eb60b93aeec669731c185ea23
a09273b4cc08c39afe0c964f14ce98e532ae530eb60b93aeec669731c185ea23
43f7a8e8e5471917178430f3842561d333b736974f4b2784ca543e3093204b
a260d222dfc94b91a09485647c21a4a426469528ec4b1b49469db3b283eb9a
a260d222dfc94b91a09485647c21a4a426469528ec4b1b49469db3b283eb9a
2d7cb5ff4a449fa284721f83e352098c2fdea125f756322c903a40ad3ebc5e40d
a63437a044d3ad01c52b0b18016bfdb8af2338067a4216be2dcaaa04ec8ceee97
bf38bea3f89a697b0be13413b0fb1db2154b3dc79fffbee238014e4adeb0b880
Full Console Output

blevene@blevene-imacpro  ~  vt hunting rulesets add TestBotSet

~/Documents/YaraRules/trojan_win_psixbot.yar

- hunting_ruleset <5529739319771136>:
  creation_date: 1554219815  # 2019-04-02 11:43:35 -0400 EDT
  enabled: false
  limit: 100
  modification_date: 1554219815  # 2019-04-02 11:43:35 -0400 EDT
  name: "TestBotSet"
  notification_emails: []
  number_of_rules: 1
  rules: |
    import "pe"
    rule trojan_win_psixbot : commodity
      { 
        meta: 
          description = "Identify PsiXBot dropped from Splevo Exploit Kit"
          author = "blevene@chronicle.security"
          date = "01-04-2019" //dd-mm-yyyy
          hash01 =
hash02 =
"1b213a457a9d1949feb5aaca7402ee6a200cb871c6c03e22e86f862007404ec5"

strings:
// not actually contained in import table
$s1 = "acledit.dll"

// ekjynhadefrderatafrhnamioppliynhaioplhaterafdertayunm
$u1 =
{656b6a796e686164656f7265616e64696669636f6e6e656465726179756e6d}

condition:
  uint16(0)==0x5a4d
  and
  (
  (pe.imports("authz.dll")
     and pe.imports("clbcatq.dll")
  )
  or for any i in (0..pe.number_of_sections -1):
    (pe.sections[i].name == ".relok")
  )
  and 1 of them

blevene@blevene-imacpro ~ vt hunting rulesets enable 5529739319771136
Full Console Output

✘ blevene@blevene-imacpro ~ vt retrohunt start
~/Documents/YaraRules/trojan_win_psixbot.yar
blesvene_chron-1554220302
blesvene@blevene-imacpro ~ vt rh list
- retrohunt_job <blesvene_chron-1554220302>:
  creation_date: 1554220302 # 2019-04-02 11:51:42 -0400 EDT
  eta_seconds: 21876
  num_matches: 0
  num_matches_outside_time_range: 0
  progress: 0.027419341
  rules: |
    import "pe"
    rule trojan_win_psixbot : commodity
    {
    meta:
      description = "Identify PsiXBot dropped from Splevo Exploit Kit"
      author = "blesvene@chronicle.security"
      date = "01-04-2019" //dd-mm-yyyy
      hash01 = 149
    }
strings:
    // not actually contained in import table
    $s1 = "acledit.dll"

    // ekjynhaderfdreratnhlamkioplpmhnaioplhaterafdertyunm
    $u1 = {656b6a796e6861646566726465726174616672686e616d61696f706c68617465726166646572746179756e686196f706c68617465726166646572746179756e6d}

    condition:
        uint16(0)==0x5a4d
        and
(  
    (  
        pe.imports("authz.dll")
        and pe.imports("clbcatq.dll")
    )
    or for any i in (0..pe.number_of_sections -1):
    (pe.sections[i].name == ".relok")
)
        and 1 of them
        and filesize < 500KB
    }

scanned_bytes: 2579372141
start_date: 1554220307  # 2019-04-02 11:51:47 -0400 EDT
status: "running"
APIv3 - Retrohunt Matches

Where -I is the file identifier (sha256 only) and -n is number of results (to auto iterate over pages)

blevene@blevene-imacpro ~ vt retrohunt matches blevene_Chron-1554220302 -I -n 50

2644cd4a843e35b807271aeff9ec17a48a7b3214fc052b625dba89796822a240
a7490c93e8423909cf1c53571697b260c17e0b0661bfa538754d72daf965a156
1b213a457a9d1949febf5aaac7402ee6a200cb871c603e22e86f862007404ec5
6810234dddf132808aa231e8884b3426bc430136ac5119228217edd2adc0d83b
c8d2a9f6a9e6974ac28c3f1c81f7522715ed69c673892858c47f876777c2807
0252e1e82ce399e4886dabc90f33188183ac5029b8d5049f47d087bdfb5cabc
6fb4b6a4e5367ce4bfff82fda3b3e6fa3939b9ee86e677eba6cc658606310bd0
eff363c3be4fe16411dcb52599c9242ef0377daa8742c1e672ace00553c88b8d
15049d944375f2022a5bf3f13dd0e9a0c02f517a6153d9cfc4d8c841d7e2248
fd35b23d2c82bd57d54ac1f0979b16bfaa25a910839b8b089f8f2ab5c1d4
3736b350803dbac86b67967556a35216c318dd5e10722c52791181812e71c0
e6a2a24796480401d75b1f524f13f14081c737cdfdca39559a76d808e6b7
8e13b9d9a0746203d2e676b0975dfc6621409fec560c4a240aa
21c4efedfb9017b6de7967649696e95968283eba8315c0ceef6523bb16d11e72
Full Console Output

blevene@blevene-imacpro ~ vt hunting notifications list
--cursor=JM3PTlsWHADgu4_RxJvSysQFEmlmKGQCMxDDYDEh3dbRn-va2T9b1Pjuhrh
9t-8xBUA1aLaEc52dYTmWv6gH46TjslRqVvMoSB9gL3IuxJC8p4rfo6F0QoLrxyoy1ipBz
WUzIFW9tIwZWAzs5rKpDEzJ-Kr2x2zsiz5nFNd5rmqyXJ98lrvOv1UpO5Wu5j-nn4osf0v
mrl3VMSRHaRBfKbREcmhkiDRVpUmUTwrkJem-nhGV5dxovCcumfpXEUziG67DbbRY
vlyiJ6G2wuNYSWnDZgL0FtdW-7QjPzd_AcAAP_

- hunting_notification <5223863694327808>:
  body: ""
  date: 1554217750  # 2019-04-02 11:09:10 -0400 EDT
  file: "1a7a99b970fdbe2c8321398be15c30a0367fc66d7d36c72f7e3e3cca332c556f"
  subject: "AV_trojan_win_Emotet: Emotet"
  tags:
  - "emotet"
  - "av_trojan_win_emotet"
  - "1a7a99b970fdbe2c8321398be15c30a0367fc66d7d36c72f7e3e3cca332c556f"
  - "av_emotet"
APIv3 - ASCII Cinema

https://asciinema.org/a/179696
Thank you.

info@virustotal.com
virustotal.com/learn