

Threat Hunting with VirusTotal

black belt edition

Gerardo Fernández, Vicente Díaz
VirusTotal
Jornadas STIC, 30 Noviembre 2020



JUAN INFANTES

A Graph Guy

VT Relationship Team Lead



BRANDON LEVENE

HEAD OF APPLIED INTELLIGENCE



JOSE MARTÍN

VT-Grep creator (Content-Search)

ML guy in VT



GERARDO FERNÁNDEZ

IDA Plugin creator

World-largest threat observatory

- Massive amounts of data, instantaneous searching
- Any kind of threat observable (files, URLs, domains, IPs)
- Multi-angular characterization (AVs, whitelists, sandboxes, etc.)
- Diverse, global, **crowdsourced**, real-time
- Unparalleled history, going back to 2004

15
YEARS
OBSERVATIONS
GOING BACK TO 2004

2.4B FILES
50B+ considering
compressed bundles

2M
analyses
per day

**600
M+**
sandbox reports



232
countries
submitting files



**1.9
M** users
per month

4.6B URLs
6M+ URL analyses per day

1.6B
DOMAINS

2.5B
pDNS
RESOLUTIONS



70+ Antivirus vendors
70+ URL blacklists
10+ Sandbox partners

Threat hunting - why

Get as much visibility as we can from known malicious activity.

Find new undetected threats.

Supercharge your security operations



VTAPI

Integrate VT into security tools

Automate workflows with VT

Programmatic enrichment of alerts

Any kind of threat observable



VTINTELLIGENCE

The “Google” of malware

PBs of data queryable in seconds

Download files for further scrutiny

Pivot to similar and related observables



VTHUNTING

Apply YARA rules to live uploads, get notifications

Run YARA rules back in time, track attackers

Generate automatic YARA rules for groups of files

Download matches for offline study



VTGRAPH

Explore VT's dataset visually, map campaigns

Automatic commonality and pattern discovery

Share and collaborate on investigations

Assisted investigation and expansion playbooks



VIRUSTOTAL

Free public service | 70+ Antiviruses | Vet suspicious files | 1M+ new files per day

01

Let's start with a
simple example

We got lucky!

And the community did the hard work for us!



VTGrep (aka Content Search)

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

VTGrep: Example 1, ASCII Strings

DETECTION

DETAILS

RELATIONS

BEHAVIOR

CONTENT

SUBMISSIONS

COMMUNITY

STRINGS

HEX

cmd.exe %s%s"

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

kernel32.dll

ADVAPI32.dll

KERNEL32.dll

kernel32.dll

mscoree.dll

VTGrep: Example 1, ASCII Strings

content:D:\\MyProjects\\secondWork\\Anchor\\Win32\\Release

Help



☐ FILES 6



AFD791CDD82B35AB187BDE91924F458A2FC109D8F15670A10CDE994990CA9



peexe

Detections

Size

First seen

Last seen

Submitters

49 / 70

394.50 KB

2019-04-15
22:19:30

2019-04-15
22:19:30

1



netxaufk.dll



pedll



46 / 70

136.00 KB

2019-06-20
14:17:19

2019-06-20
14:17:19

1



B288C3B3F5886B1CD7B6600DF2B8046F2C0FD17360FB188ECFBC8F6B7E552A5



b288c3b3f5886b1cd7b6600df2b8046f2c0fd17360fb188ecfbcc8f6b7e552a5.bin



peexe

54 / 68

112.50 KB

2019-07-21
14:59:03

2019-12-12
09:16:33

3



DA0D22FD7448E34BE162D40ECF18B23C1F4A76CD7EE125A8C8102C8358E84F04



peexe



overlay

52 / 73

112.62 KB

2019-08-07
12:20:13

2019-08-07
12:20:13

1



9B76C53A4C231933191524DDA8509B8DAEF8F0B2207F74D115DDFF65F56AAC2E



peexe



overlay

49 / 71

112.62 KB

2019-08-09
12:25:03

2019-08-09
12:25:03

1



FD27782F9C0FA6E74099738EFFD723AE17FE023AD0470A4B67D84E205B3F9149



peexe



overlay

49 / 71

112.62 KB

2019-08-10
12:26:33

2019-08-10
12:26:33

1



VTGrep: Example 2, Wildcards

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

content:{???3a5c4d7950726f6a656374735c7365636f6e64576f726b5c416e63686f725c57696e????5c52656c656173655c}

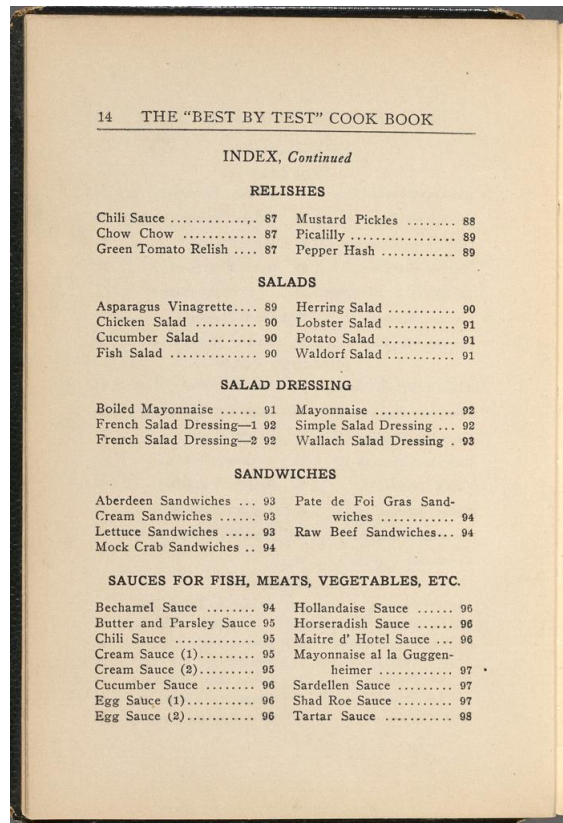
☐ FILES 6



		Detections	Size	First seen	Last seen	Submitters	
☐	AFD791CDD82B35A8187BDE91924F458A2FC109D8F15670A1DCDE3DE994990CA9 peexe	49 / 70	394.50 KB	2019-04-15 22:19:30	2019-04-15 22:19:30	1	
☐	44EA51D773DD10117EF76744A36C72F5F23D00C20C0A0F6A4F4436F40EEFAB80 netxaufk.dll pedll	46 / 70	136.00 KB	2019-06-20 14:17:19	2019-06-20 14:17:19	1	
☐	B288C3B3F5886B1CD7B6600DF2B8046F2C0FD17360FB188ECFBC8F6B7E552A5 b288c3b3f5886b1cd7b6600df2b8046f2c0fd17360fb188ecfbcc8f6b7e552a5.bin peexe	54 / 68	112.50 KB	2019-07-21 14:59:03	2019-12-12 09:16:33	3	
☐	DA0D22FD7448E348E162D4DEC18B23C1F4A76CD7EE125A8C8102C8358E84F04 ded204da5371a2b8118998d8e8b704e7.virus peexe overlay	52 / 73	112.62 KB	2019-08-07 12:20:13	2019-08-07 12:20:13	1	
☐	9B76C53A4C231933191524DDA850988DAEF8FB2207F74D115DDFF65F56AAC2E 8a8cc9552fcc974e843842e8a289f673.virus peexe overlay	49 / 71	112.62 KB	2019-08-09 12:25:03	2019-08-09 12:25:03	1	
☐	FD27782F9C0FA6E74099738EFD723AE17FE023AD0470A4867DB4E20583F9149 dc6aeedd38279559df27c0043b3cc5ea.virus peexe overlay	49 / 71	112.62 KB	2019-08-10 12:26:33	2019-08-10 12:26:33	1	

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 samples, OCR, macros, VBA code streams...



14 THE "BEST BY TEST" COOK BOOK

INDEX, *Continued*

RELISHES

Chili Sauce	87	Mustard Pickles	88
Chow Chow	87	Picalilly	89
Green Tomato Relish	87	Pepper Hash	89

SALADS

Asparagus Vinagrette....	89	Herring Salad	90
Chicken Salad	90	Lobster Salad	91
Cucumber Salad	90	Potato Salad	91
Fish Salad	90	Waldorf Salad	91

SALAD DRESSING

Boiled Mayonnaise	91	Mayonnaise	92
French Salad Dressing—1	92	Simple Salad Dressing ..	92
French Salad Dressing—2	92	Wallach Salad Dressing .	93

SANDWICHES

Aberdeen Sandwiches ...	93	Pate de Foi Gras Sand-	
Cream Sandwiches	93	wiches	94
Lettuce Sandwiches	93	Raw Beef Sandwiches...	94
Mock Crab Sandwiches ..	94		

SAUCES FOR FISH, MEATS, VEGETABLES, ETC.

Bechamel Sauce	94	Hollandaise Sauce	96
Butter and Parsley Sauce	95	Horseradish Sauce	96
Chili Sauce	95	Maitre d' Hotel Sauce ..	96
Cream Sauce (1).....	95	Mayonnaise al la Guggen-	
Cream Sauce (2).....	95	heimer	97
Cucumber Sauce	96	Sardellen Sauce	97
Egg Sauce (1).....	96	Shad Roe Sauce	97
Egg Sauce (2).....	96	Tartar Sauce	98

VTGrep: content-experimental

Allows several operators (positives; size; type; and more)

Available to all Intelligence users

content-experimental → content :)

02

I found nothing
unique, still I need
some context

What is similarity



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:

<https://securemail.citibank.com/privacy/securereader?id=64124813498052&brand=95642159>.

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

Sincerely,

Member Services

Email ID: 431

<https://www.Citibank.com/>

How to calculate similarity



How is this useful

- similar-to:x [have:itw](#)
- similar-to:x [tag:attachment](#)
- similar-to:x [have:compressed_parents](#)
- similar-to:x [have:comment](#)
- similar-to:x [have:behaviour_network](#)

03

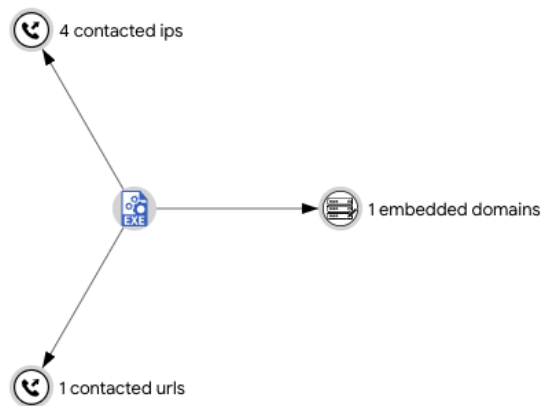
Finding related
artefacts and
infrastructure

Static analysis

Find artefacts inside the binary

Embedded Domains ⓘ				📄
Domain	Detections	Created	Registrar	
gmail.com	0 / 97	1995-08-13	MarkMonitor Inc.	

Graph Summary ⓘ	
-----------------	--



Dynamic analysis

Find relationships through sandbox detonation

Contacted URLs



Scanned	Detections	URL
2020-11-18	7 / 82	http://62.30.7.67:443/8ly7Q4G1EitRXhpS/l5Q31Eyh2hDq/H0edPIVkJ2rq0/

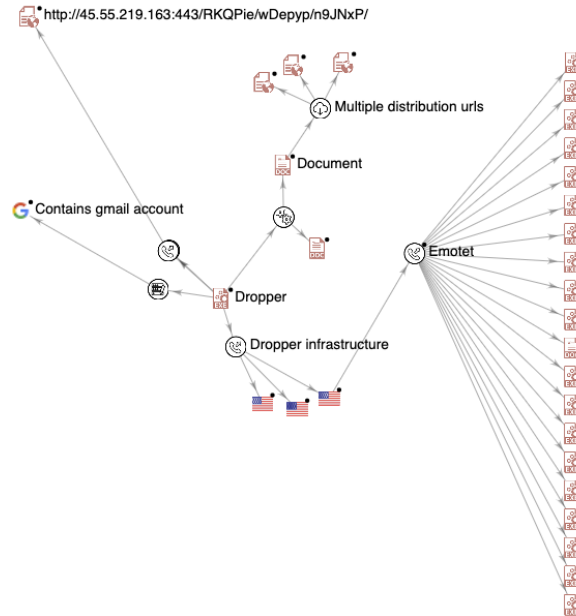
Contacted IPs



IP	Detections	Autonomous System	Country
107.5.122.110	9 / 85	7922	US
45.55.219.163	8 / 85	14061	US
199.101.86.6	6 / 96	26128	US
62.30.7.67	13 / 85	5089	GB

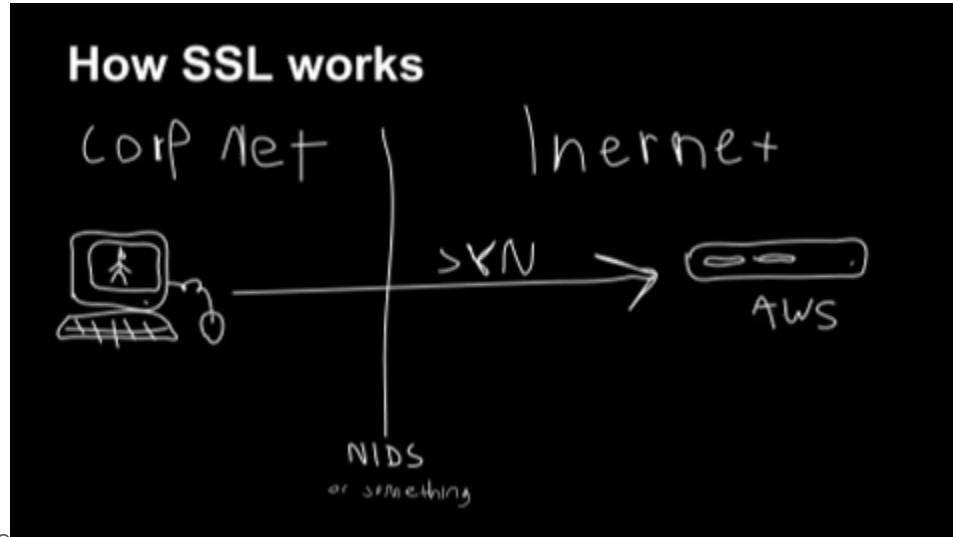
ITW infrastructure

We have a good visibility of how this was distributed



Advanced infrastructure pivoting

Certificates, Whois data, JA3



Source: <https://engineering.salesforce.com/tls-fingerprinting-with-ja3-and-ja3s-247362635967>

04

Once we have all the data, let's drop it into VTGraph

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

VTGraph

DEMO



LIVEHUNT NOTIFICATIONS



Search notifications



6ec9d3a2302a7cd6b170b155a9a2f0eed3c264e25a2c829a5d8f7df2e44ee9dc

log file.exe

45 / 65

3.58 MB

2019-03-22 16:17:17 date matched

1 submissions

2019-03-22 16:14:19 first seen

1 submitters

peexe assembly overlay apt_win_nedrat gazahackerteam



7b081379d83e7bcfac3619f4b274f5d5e073b81618803f8feded0127bb8f6918

OFXVKAAL.EXE

12 / 64

748.5 KB

2019-03-22 16:13:33 date matched

1 submissions

2019-03-22 16:07:10 first seen

1 submitters

peexe av_emotet av_trojan_win_emotet emotet



ba8aaca4dfb35315e502e66aede7be7aff535af42934024bd61391db8977f2ba

DISM.EXE

14 / 67

185.26 KB

2019-03-22 16:12:47 date matched

1 submissions

2019-03-22 16:08:55 first seen

1 submitters

peexe overlay av_emotet av_trojan_win_emotet emotet



aca9e4365bfe563fc21779c81d54ec6037be9c8a2202b3bef0428c388389da6

etohaknairinik.exe

38 / 66

6.31 MB

2019-03-22 16:11:10 date matched

1 submissions

2019-03-22 16:03:47 first seen

1 submitters

peexe av_emotet av_trojan_win_emotet emotet



05

TOOLS FOR
HUNTING

Name this Yara!

```
rule MW_neuron2_dotnet_strings : Turla APT
{
    meta:
        description = "Rule for detection of the .NET payload for Neuron2 based on strings used"
        author = "NCSC"
        family = "Turla"
        reference = "https://www.ncsc.gov.uk/alerts/turla-group-malware"
        date = "2018-01-18"
        hash1 = "83d8922e7a8212f1a2a9015973e668d7999b90e7000c31f57be83803747df015"
    strings:
        $dotnetMagic = "BSJB" ascii
        $s1 = "http://*:80/W3SVC/" wide
        $s2 = "https://*:443/W3SVC/" wide
        $s3 = "neuron2.exe" ascii
        $s4 = "D:\\Develop\\sps\\neuron2\\neuron2\\obj\\Release\\neuron2.pdb" ascii
    condition:
        (uint16(0) == 0x5A4D and uint16(uint32(0x3c)) == 0x4550) and $dotnetMagic and 2 of ($s*)
}
```

From similarity to Yara

DEMO

Retrohunt

+ New retrohunt job

100 %	Finished	blevene_Chron-1553198358 20 hours ago rule Office_Base64_BreakCatchWide :maldoc { meta: description = "Potential Emotet maldoc using base64 wide encoded break->catch" author ...	615 matches	↓
100 %	Finished	blevene_Chron-1553197869 20 hours ago import "pe" rule apt_win_cobint_dll : Cobalt_Group { meta: description = "Identify potential CobInt downloader DLL Trojan samples, unique to Co..."	1011 matches	↓
100 %	Finished	blevene_Chron-1553179894 1 day ago rule apt_win_EyeHawk : CN { meta: description = "Identify payload from RTF: 69f44ca082ed90c97d9c4ebaae589d7e41c69b02e582cc69886ebf..."	5 matches	↓
100 %	Finished	blevene_Chron-1553117311 1 day ago rule astra_docs { meta: description="ver1 of astra_docs" author="JDP" strings: \$header = {d0cf11e0a1b11ae1} \$a1 = "PROTECTED CONTENT"...	12 matches	↓
100 %	Finished	blevene_Chron-1553115402 1 day ago import "pe" rule LockerGogaRansomware { meta: description = "LockerGoga Ransomware" author = "Christiaan Beek - McAfee ATR team" date ...	7 matches	↓
100 %	Finished	blevene_Chron-1553023105 2 days ago rule ransomware_win_lockergoga : ransomware { meta: description = "Identify LockerGoga ransomware, mostly clustered around Dutch and Dan..."	15 matches	↓

```
import "vt"

rule yara_on_steroids_demo

{
  condition:

    vt.metadata.analysis_stats.malicious > 1
    and
    vt.metadata.file_type == vt.FileType.PE_EXE
    and
    vt.metadata.new_file
    and
    vt.metadata.submitter.country == "CN"
    and
    for any engine, signature in
vt.metadata.signatures :
      (
        signature contains "zbot"
      )
}
```

```
import "vt"

rule yara_on_steroids_demo_behaviour

{
  condition:

  for any file_dropped in vt.behaviour.files_dropped : (
    file_dropped.path contains
    "foo.exe"
  )
  or
  for any mutex in vt.behaviour.mutexes_created : (
    mutex == "HGL345"
  )
  or
  for any trait in vt.behaviour.traits : (
    trait == vt.BehaviourTrait.PERSISTENCE
  )
}
```

type:docx and p:10+ and s:3+ and tag:macros and fs:2020-09-15T16:59:22+

```
import "vt"

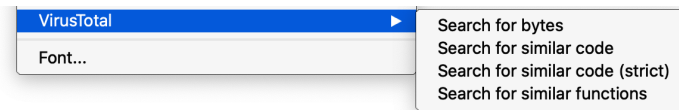
rule new_potential_droppers
{
    condition:

    vt.metadata.analysis_stats.malicious> 10          and
    vt.metadata.times_submitted>3                      and
    for any t in vt.behaviour.traits : (
        t == vt.BehaviourTrait.MACRO_POWERSHELL
    )
        and
    vt.metadata.file_type == vt.FileType.DOCX          and
    for any tag in vt.metadata.tags : (
        tag == "macros"
    )
        and
    vt.metadata.first_submission_date > 1600191396
}
```

06

IDA PLUGIN

VT-IDA plugin



- Search for a sequence of bytes or similar code, directly from the IDA Pro interface.
 - Disassembly and Debugging windows.
 - Strings window.
 - Follow VTGrep syntax when creating queries.
- v0.10 includes IDA Pro 7.5 support.
 - <https://github.com/VirusTotal/vt-ida-plugin>

Search for similar code

55		push	ebp	
8B EC		mov	ebp, esp	
68 04 01 00 00		push	104h	
6A 00		push	0	
68 B0 EC 24 01		push	offset byte_124ECB0	
E8 FC F7 FF FF		call	sub_1193080	
83 C4 0C		add	esp, 0Ch	
A1 E8 E0 24 01		mov	eax, off_124E0E8	
8B 48 04		mov	ecx, [eax+4]	
51		push	ecx	; lpString2 = 0 --> Genera excepción
68 B0 EC 24 01		push	offset byte_124ECB0	; lpString1
FF 15 28 50 23 01		call	ds:lstrcpyA	; LPSTR lstrcpyA(; LPSTR lpString1, ; LPCSTR lpString2 ;);
68 10 27 00 00		push	10000	; dwMilliseconds = 10 segundos
FF 15 44 50 23 01		call	ds:GetCurrentThread	
50		push	eax	; hHandle
FF 15 40 50 23 01		call	ds: imp WaitForSingleObject	
E8 4E D7 FF FF		call	Create_New_IAT	
E8 89 F7 FF FF		call	Collect_Info_From_Registry	
E8 F4 FA FF FF		call	Create_Report	

content:{558bec68040100006a0068[9]83c40ca1[4]8b48045168[10]6810270000ff15[26]33c05dc21000}

VT-IDA plugin

DEMO

07

Summarizing

Thank you.

Vicente Diaz

[@trompi](#)

Gerardo Fernández

[@gerardofn](#)

virustotal.com/learn