How I Found the Honeypot: Dark Web OSINT and Image Processing

12-02-2017

This post is for Honeypot Advent Calendar 2017.

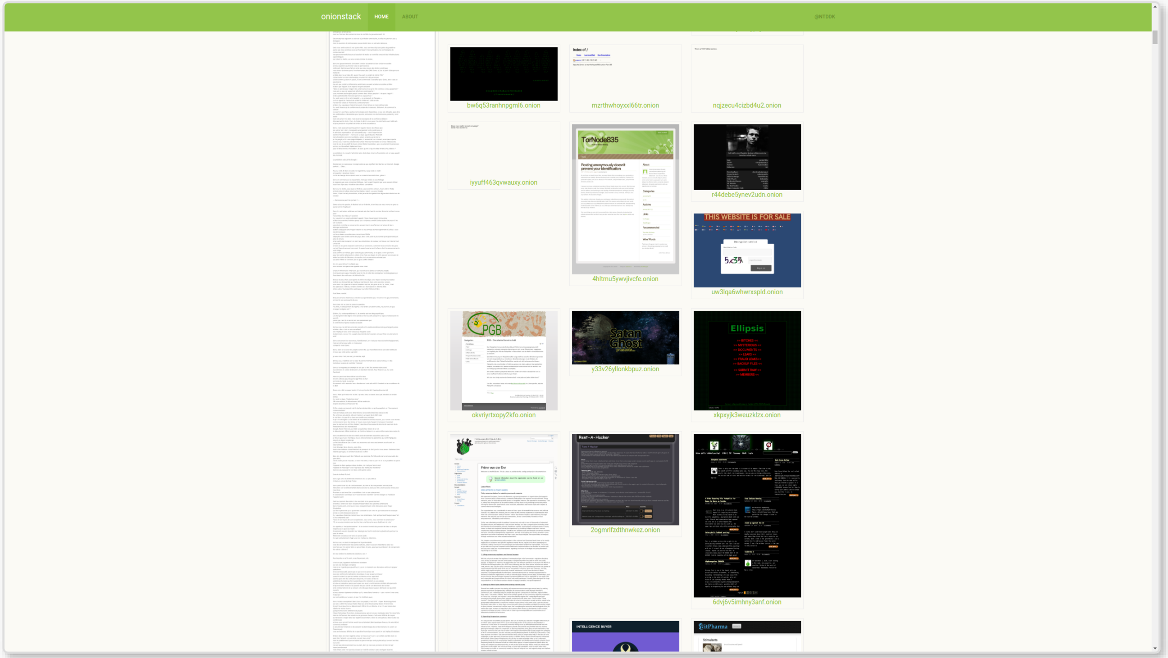
Introduction

In May this year, Trend Micro researchers have announced interesting research results in the article titled Red on Red: The Attack Landscape of the Dark Web - TrendLabs Security Intelligence Blog. They had deployed a honeypot on the dark web and monitored attack activities. They've done great work, indeed.

Well then, with the aid of the screenshot in that article, probably I found the honeypot. May I introduce the how and why?

Dark Web OSINT

In March this year, when they would have created an presentation slide, I have been running the crawler for the dark web. My purpose was to create a pictorial book of Tor hidden services below:



The crawler is simple, just like saying “Hello, world!” to PhantomJS. It has only capability of taking a screenshot.

```
service_args = [
            # Do not insert blank to each of args.
            '--proxy=127.0.0.1:9050',
            '--proxy-type=socks5'
        ]
```

```
dcap = {
        'phantomjs.page.settings.userAgent': 'Mozilla/5.0 (Windows NT 6.1; rv:
}

def get_title_with_screenshot(url):
    driver = webdriver.PhantomJS(service_args = service_args, desired_capabili
    driver.set_window_size(1024, 512)
    driver.get('http://' + url + '.onion') # 'http://' is required.
    driver.save_screenshot(url + '.png')
    title = driver.title
    driver.close()
    return title
```

I have discovered 40,208 onion domains and confirmed 1,797 domains were active.

Image Processing

Thanks to collecting screenshots by chance, I was able to find a site similar to the screenshot in their article–with histgram calculation:

```
list = glob.glob('./images/*.png')

```

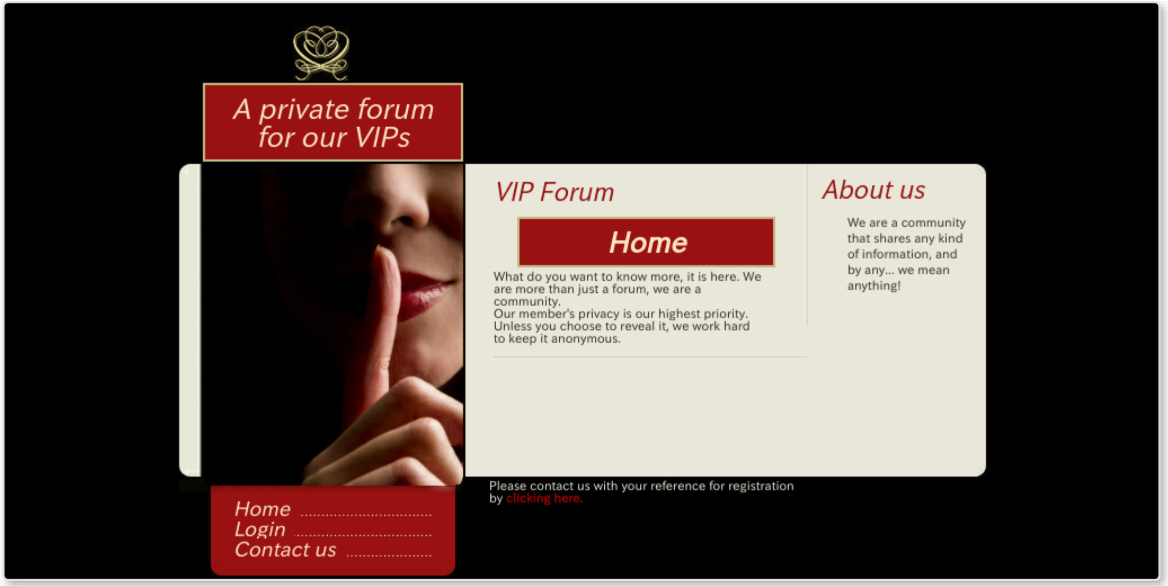
```
def cbir():
    target_im = cv2.imread(sys.argv[1])
    target_hist = cv2.calcHist([target_im], [0], None, [256], [0, 256])

    for i in list:
        comparing_im = cv2.imread(i)
        comparing_hist = cv2.calcHist([comparing_im], [0], None, [256], [0, 25
        diff = cv2.compareHist(target_hist, comparing_hist, 0)
        if diff > float(sys.argv[2]): # Threshold
            print i, diff
```

Yet domain names are not posted in their articles, I believe this is it.

```
$ wget http://blog.trendmicro.com/trendlabs-security-intelligence/files/2017/05
$ cbir.py red-on-red-1.jpg
$ python cbir.py red-on-red-1.jpg 0.9
./images/s5**********jlp2.png 0.979927357262
```

The screenshot:



A private forum for our VIPs

VIP Forum

Home

What do you want to know more, it is here. We are more than just a forum, we are a community.
Our member's privacy is our highest priority. Unless you choose to reveal it, we work hard to keep it anonymous.

About us

We are a community that shares any kind of information, and by any... we mean anything!

Home
Login
Contact us

Please contact us with your reference for registration by clicking here.

There are many clone sites in the dark web–for backup, or even for spying? `s5**********jlp2.onion` might be cloned and have maintained by others. Even in that case, the site is likely to be a honeypot.

Interestingly, I also found some posts like to induce to the site at r/onions. I believe these are done by researcher.

Last Words

We got a glimpse of deep in abyss. This is just an accidental case study. Needless to say, no insult intended.

My crawler and image processing scripts are available at ntddk/onionstack. Crawling of the dark web is accompanied by risk. After all, with ethical considerations, I've deleted screenshots I'd captured except for the honeypot.

If you interested in the dark web OSINT, Dark Web | Automating OSINT Blog will be a good starting point.

Anyway, keep safety.

#dark web #honeypot

Share

NEWER

Backward Program Slicing with IDAPython and Triton

OLDER

DEF CON CTF Qualifier 2017 Pepperidge Farm Write-up

JAthena by ntddk | Powered by Hexo