

#IntelCon2020



IntelCon
by Ginseg

Scrapeando Foros Hacking para CTI

sarvmetal

Congreso Online de Ciberinteligencia
Julio 2020



IntelCon
by Ginseg



@sarvmetal



<https://github.com/santiagorocha>

Índice

- 1) Definición de Scraping y Crawling
- 2) Análisis de estructura típica de un Foro Hacking
- 3) Técnicas y herramientas de Scraping
- 4) Medidas anti-bots implementadas en Foros y cómo evitarlas
- 5) Ejemplos de generación de CTI proactivo

Disclaimer

Las técnicas y herramientas usadas en esta ponencia tienen fines informativos y educativos. No me hago responsable de un uso indebido de estas técnicas o herramientas ya que pueden llegar a ser ilegales si se usan para atacar, dañar, penetrar o perjudicar de alguna forma sistemas de terceros y puede conllevar a sanciones o violaciones de la ley. No se mostrará explícitamente los foros analizados por cuestiones de anonimato.

Scraping y Crawling

Web scraping, web harvesting, or web data extraction is [data scraping](#) used for [extracting data](#) from [websites](#). Web scraping software may access the [World Wide Web](#) directly using the [Hypertext Transfer Protocol](#), or through a web browser. While web scraping can be done manually by a software user, the term typically refers to automated processes implemented using a [bot](#) or [web crawler](#). It is a form of copying, in which specific data is gathered and copied from the web, typically into a central local [database](#) or spreadsheet, for later [retrieval](#) or [analysis](#).

A **Web crawler**, sometimes called a **spider** or **spiderbot** and often shortened to **crawler**, is an [Internet bot](#) that systematically browses the [World Wide Web](#), typically for the purpose of [Web indexing](#) (*web spidering*).

[Web search engines](#) and some other sites use Web crawling or spidering software to update their [web content](#) or indices of others sites' web content. Web crawlers copy pages for processing by a search engine which [indexes](#) the downloaded pages so users can search more efficiently.

Scraping vs Crawling



¿Por qué un Foro Hacking?

Dharma ransomware | Source code | Sale!

Yesterday at 01:20 · dharma ransomware source code

Go to New Track

NO AVATAR

Yesterday at 01:20

Topic Author New

Good day. I want to sell the source of this product. The reason for the sale, went to other topics, the code has been lying idle for 3 months, completely repulsed itself and worked. The other day I came across him and decided to sell.

Included is a completely ready-made code for C, payload, decryptor, a simple console keygen as a bonus. There is no builder, but you can work out of the box, collect the outcomes in C, change the mail extension keys and go. Who needs a bilder will make for itself either php, or C, or any other option. There was a complete reverse of the original software. Sale in 3 hands. You can agree in 1 hand. Work only through the guarantor of the forum or exploit forum.

Price \$ 2000

First contact PM

registration: 03/28/2020
Messages: 1
Reactions: 0
Points: 1

* <https://www.zdnet.com/article/source-code-of-dharma-ransomware-pops-up-for-sale-on-hacking-forums/>

PULLING EMAIL FOR ANY TWITTER / TAKING REQUESTS

chaewon 8 HOURS AGO (This post was last modified: 1 hour ago by chaewon)

Price: \$250

you heard me, 250\$ per email to any twit acc

will sell multiple for less ie 2 for 420 3 for 675

btc only

u go first or @lol can hold funds idc

taking requests 2.5k-3k per @

usernames claimed done so far:

anx***s
dr*g
**
ob*nna
d**k
*

people who have used this service: @maxwell @pawdit

ever so anxious#0001 - dont message saying hey say the twit youre interested in

this is NOT a method, you will be given a full refund if for any reason you arent given the email/@, however if it is reverted/suspended i will not be held accountable

851 Rep 116 Vouches

vibing

Posts: 2,583
Threads: 502
Joined: Apr 2017
Credits: 902

3 YEARS OF SERVICE

* <https://krebsonsecurity.com/2020/07/whos-behind-wednesdays-epic-twitter-hack/>

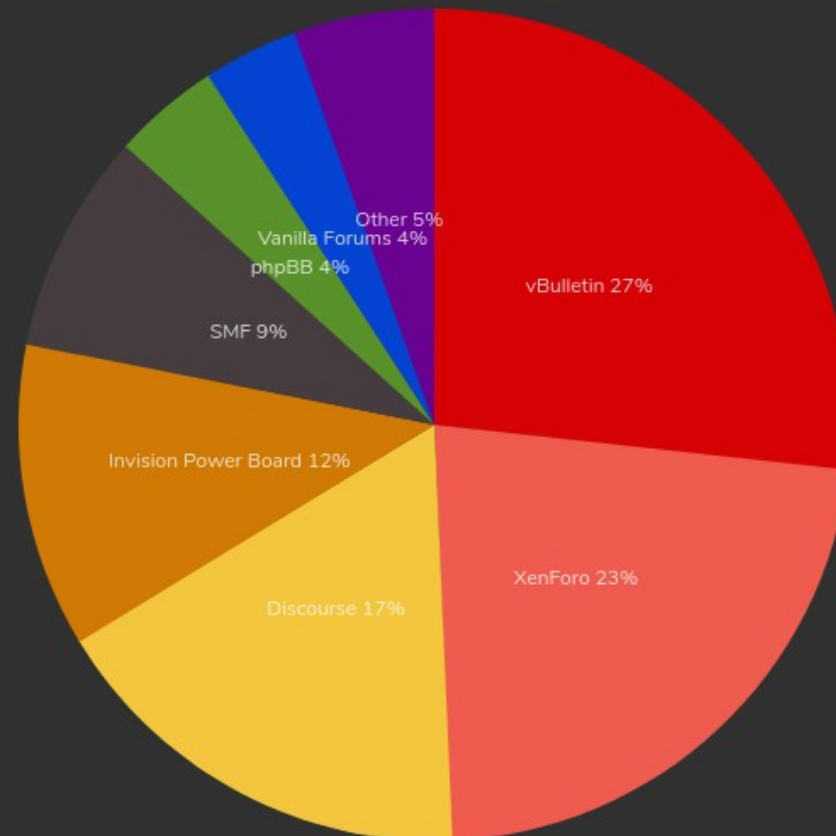
1

Análisis de la estructura de los foros

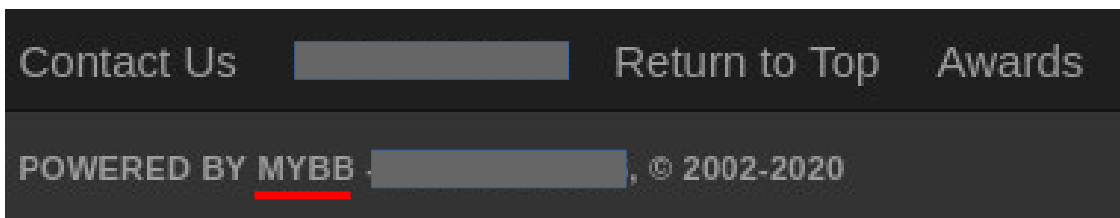
Softwares de foros de internet

Forum Software Usage Distribution in the Top 1 Million Sites

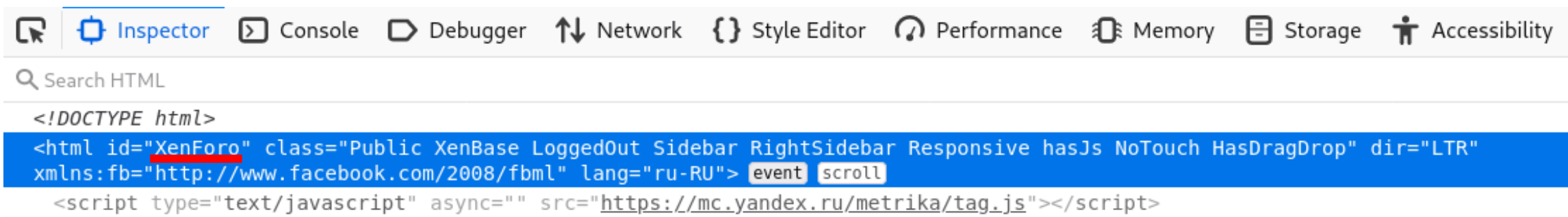
Statistics for websites using Forum Software technologies



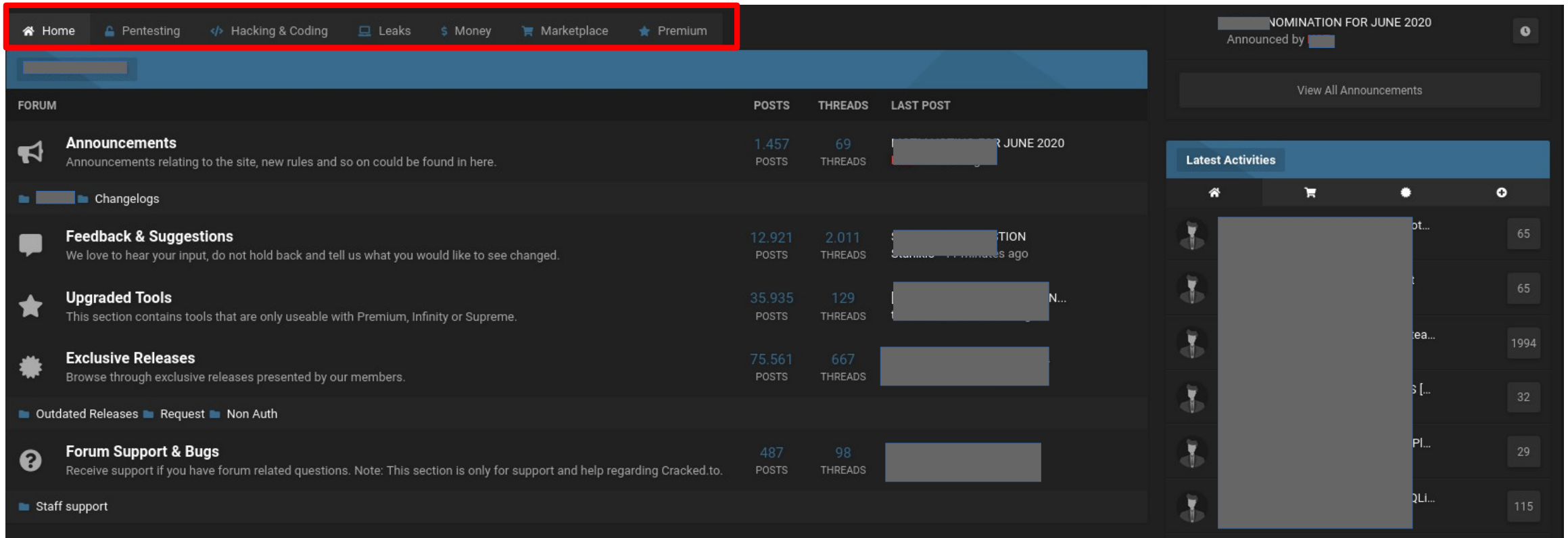
Identificando el software usado



```
▼ <head>
  <link rel="preconnect" href="https://www.birdiecdn.com">
  <link rel="preconnect" href="https://www.google-analytics.com">
  <meta http-equiv="Content-Type" content="text/html; charset=ISO-
  <meta id="e_vb_meta_bburl" name="vb_meta_bburl" content="https:,
  <base href="https://www.[redacted]/forum/">
  <!--[if IE]></base><![endif]-->
  <meta name="generator" content="vBadvanced, vBulletin 4.2.3">
```



Estructura de un Foro



The screenshot displays a forum interface with a dark theme. A red rectangular box highlights the top navigation bar, which contains the following links: Home, Pentesting, Hacking & Coding, Leaks, Money, Marketplace, and Premium. Below the navigation bar, the forum is organized into several sections:

- FORUM**: A header section with columns for POSTS, THREADS, and LAST POST.
- Announcements**: A section with 1,457 posts and 69 threads. It includes a sub-section for Changelogs.
- Feedback & Suggestions**: A section with 12,921 posts and 2,011 threads.
- Upgraded Tools**: A section with 35,935 posts and 129 threads.
- Exclusive Releases**: A section with 75,561 posts and 667 threads. It includes sub-sections for Outdated Releases, Request, and Non Auth.
- Forum Support & Bugs**: A section with 487 posts and 98 threads.
- Staff support**: A sub-section for staff support.

On the right side of the forum, there is a sidebar with the following elements:

- NOMINATION FOR JUNE 2020**: A section for announcing nominees.
- View All Announcements**: A button to view all announcements.
- Latest Activities**: A section showing recent forum activity, including user avatars and post counts.

Threads

Moderated By: [redacted]

1 2 3 4 5 ... 33 Next ↓

Sort by: Last Post ▼ Order: Descending ▼ From: The Beginning ▼ Prefix: Any/No Prefix ▼ Go

Latest Threads

Threads

Mark as Read

Important Threads

Posting Rules , Follow it or Your threads will be ignored

[redacted] 11 months ago 0 REPLIES 4,564 VIEWS 16 Last Post: 11 months ago

Normal Threads

SUPREME [NEW] [redacted] (Pages: 1 2) 3 days ago 13 REPLIES 207 VIEWS 2 Last Post: 3 minutes ago

[redacted] (Pages: 1 2 3 4 ... 8) 4 days ago 61 REPLIES 108,741 VIEWS 20 Last Post: 4 minutes ago

[redacted] (Pages: 1 2 3 4 ... 70) 1 year ago 552 REPLIES 21,257 VIEWS 146 Last Post: 21 minutes ago

SUPREME [redacted] + Source Code (Pages: 1 2 3 4 ... 32) 4 months ago 251 REPLIES 10,273 VIEWS 51 Last Post: 26 minutes ago



SUPREME [redacted] + Source (Pages: 1 2 3 4 ... 10) 5 months ago 76 REPLIES 3,571 VIEWS 17 Last Post: 28 minutes ago



[redacted] attack tools (Pages: 1 2 3 4 ... 112) 1 year ago 888 REPLIES 30,550 VIEWS 132 Last Post: 36 minutes ago

ALL IN ONE [redacted] HQ TOOL] (Pages: 1 2 3 4 ... 7) 3 days ago 48 REPLIES 526 VIEWS 12 Last Post: 41 minutes ago

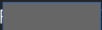
Post

123456Next

 (REMOTE ACCESS TOOL)
by  3 weeks ago 1118


☆☆☆


OP 3 weeks ago

Credits to 

Like and +rep for more HQ stuff 🐸


Leeching will be reported 🐸

virustotal included

19
REP

542
LIKES

Contributor

 Contributor

POSTS: 210

THREADS: 162

JOINED: MAY 2020

VOUCHES: 1


CREDITS: 19.349

Hidden Content

You must register or login to view this content.

BUY MY SIGNATURE SPACE

start earning \$500/day immediately

By @  ends 8/11/20

2

Escogiendo las tecnologías/herramientas necesarias

Web-Crawling Framework



Scrapy

```
from scrapy.spiders import CrawlSpider, Rule
from wikiSpider.items import Article
from scrapy.linkextractors import LinkExtractor

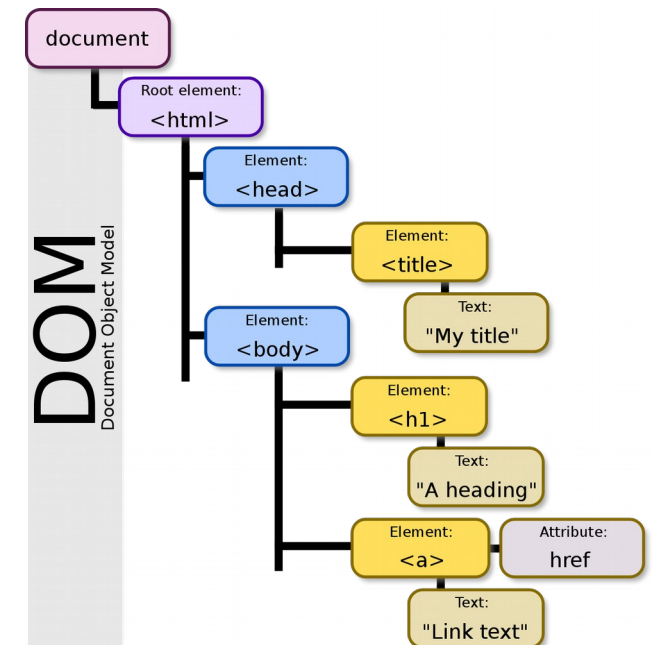
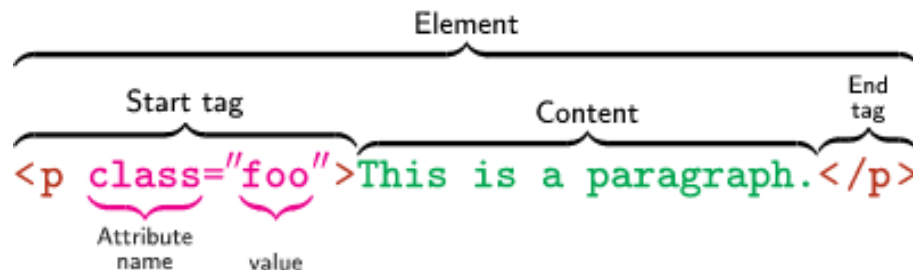
class ArticleSpider(CrawlSpider):
    name = "article"
    allowed_domains = ["en.wikipedia.org"]
    start_urls = ["https://en.wikipedia.org/wiki/Cyber_threat_intelligence"]
    rules = [
        Rule(LinkExtractor(allow=('/wiki/)((?!:).)*$'),, callback="parse_item", follow=True)
    ]

    def parse_item(self, response):
        item = Article()
        title = response.xpath('//h1/text()')[0].extract()
        print("Title is: "+title)
        item['title'] = title
        return item
```

HTML DOM (Document Object Model)

The screenshot shows a web browser displaying the Wikipedia page for "Cyber threat intelligence". The Chrome DevTools DOM inspector is open, showing the HTML structure. The selected element is the `<h1 id="firstHeading" class="firstHeading" lang="en">` tag, which contains the text "Cyber threat intelligence". The breadcrumb at the bottom of the DOM tree reads: `html.client-js.ve-available > body.mediawiki.ltr.sitedir-ltr.mw-hide-e... > div#content.mw-body > h1#firstHeading.firstHeading`.

XPath uses path expressions to select nodes or node-sets in an XML document.



Usando Librerías de Python

Selenium

- Originalmente diseñado para testeo de paginas web
- Necesario descargar el driver del navegador a usar
- Múltiples posibilidades para navegar en la página a analizar
- Opción headless
- Tomar capturas de pantalla
- Manejador de cookies, JavaScript, cabeceras, etc.



Usando Librerías de Python

Beautiful Soup

- Forma fácil de encontrar la información necesaria
- Diferentes opciones de parsers para una mejor extracción del HTML DOM
- Efectivo en Foros con una estructura HTML que no esté perfectamente formada
- Depende de alguna librería que se encargue de hacer las peticiones al Foro



BeautifulSoup

3

“Scrapeando” información



VS



* <https://selenium-python.readthedocs.io/locating-elements.html>

```
title = firefox.find_element_by_id("firstHeading").text
title = firefox.find_element_by_class_name("firstHeading").text

title = firefox.find_elements_by_id("firstHeading")[0].text
title = firefox.find_elements_by_class_name("firstHeading")[0].text
```

* <https://www.crummy.com/software/BeautifulSoup/bs4/doc/>

```
title = bs_obj.find("h1", {"id":"firstHeading"}).get_text()
title = bs_obj.find("h1", {"class":"firstHeading"}).get_text()

title = bs_obj.findAll("h1", {"id":"firstHeading"})[0].get_text()
title = bs_obj.findAll("h1", {"class":"firstHeading"})[0].get_text()
```

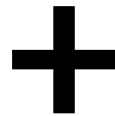
```
<!DOCTYPE html>
<html class="client-js ve-available" dir="ltr" lang="en"> event scroll
  <head> ... </head>
  <body class="mediawiki ltr sitedir-ltr mw-hide-empty-elt ns-0 ns-subject ...view :
page-action-enabled"> event
    <div id="mw-page-base" class="noprint"></div>
    <div id="mw-head-base" class="noprint"></div>
    <div id="content" class="mw-body" role="main">
      <a id="top"></a>
      <div id="siteNotice" class="mw-body-content"> ... </div>
      <div class="mw-indicators mw-body-content"> ... </div>
      <h1 id="firstHeading" class="firstHeading" lang="en">
        ::before
        Cyber threat intelligence
      </h1>
```

```
title = firefox.find_element_by_xpath("/html/body/div[@id='content']/h1[@id='firstHeading']").text
```

Combinando el poder de interacción de Selenium y el parser de BeautifulSoup



```
firefox = webdriver.Firefox()
firefox.get("https://en.wikipedia.org/wiki/Cyber_threat_intelligence")
page_source = firefox.page_source
bs_obj = BeautifulSoup(page_source, "html.parser")
title = bs_obj.find("h1", {"id": "firstHeading"}).get_text()
print(title)
firefox.close()
```



- Esperas explícitas e implícitas
- Rellenado de formularios
- Drag and Drop
- Moveirse entre ventanas
- Popup
- Historial
- Cookies
- Acciones con el cursor:
 - click()
 - click_and_hold()
 - release()
 - double_click()

- Manejo de diferentes Parsers
 - html.parser
 - xml
 - lxml
 - Html5lib
- Múltiples opciones para navegar por el HTML DOM
 - .children
 - .descendants
 - .parent
 - .next_sibling
 - Etc...

4

Medidas anti-bots

Verificación del Browser



Checking your browser before accessing 

This process is automatic. Your browser will redirect to your requested content shortly.

This should only take a moment

- * Bloqueo de User-agent

```
WebDriverWait(firefox, 10).until(EC.presence_of_element_located((By.ID, "id_name"))):
```

- ✓ Espera explícita e implícita
- ✓ Ajuste de Cabeceras
- ✓ Límite de peticiones

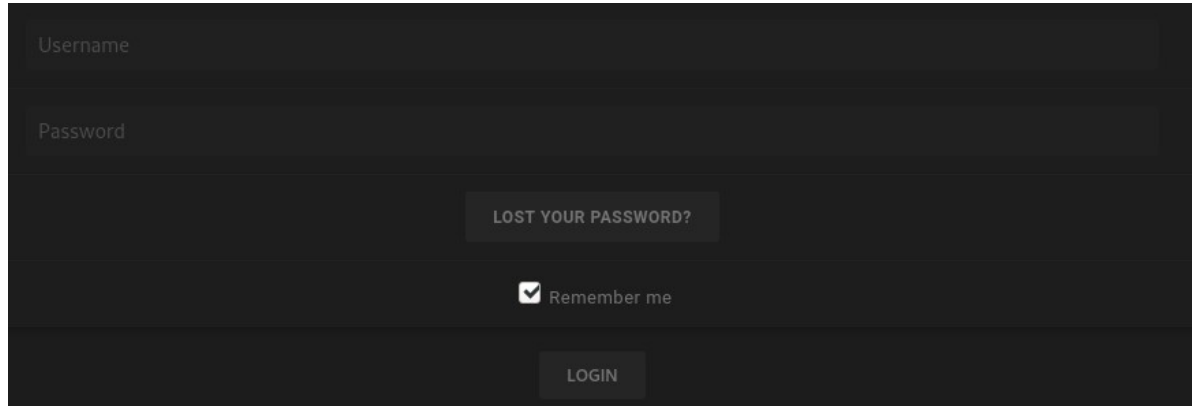
- ✓ Ajustando las cabeceras con requests

```
import requests

TOR_SESSION = requests.session()
HEADER = {"User-Agent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:78.0) Gecko/20100101 Firefox/78.0",
          "Accept": "text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8",
          "Accept-language": "es-ES,es;q=0.8,en-US;q=0.5,en;q=0.3",
          "Connection": "keep-alive"}

req = TOR_SESSION.get("http://www.10000000000.com", headers = HEADER)
bs_obj = BeautifulSoup(req.text, "html.parser")
```

Autenticación de usuarios (Selenium)



Username

Password

LOST YOUR PASSWORD?

☒ Remember me

LOGIN

```
if WebDriverWait(firefox, 25).until(EC.presence_of_element_located((By.ID, "id_name"))):  
  
    username = firefox.find_element_by_name("username")  
    password = firefox.find_element_by_name("password")  
    log_button = firefox.find_element_by_name("submit")  
  
    username.send_keys(config.USERNAME)  
    firefox.implicitly_wait(5)  
    password.send_keys(config.PASSWORD)  
    log_button.click()
```

Autenticación de usuarios (BS4)

```
fields = firefox.find_elements_by_tag_name("input")
for field in fields:
    if not field.is_displayed():
        HIDDEN_INPUT[field.get_attribute("name")] = field.get_attribute("value")
    elif field.is_displayed():
        INPUTS[field.get_attribute("name")] = field.get_attribute("value")
```

St	Met...	D	File	Ca...	Ty	Tra...	Size	0 ms		13	Headers	Cookies	Params	Response
30	POST		member.php	doc...	htr	37.6...	275....	425 ms						
26	GET		/	doc...	htr	37.2...	275....	1001 ms						
26	GET		b52916f59f.js	script	js	cac...	5.62...							
26	GET		get_updates.js	script	js	cac...	1.61...							
26	GET		icon?family=...	styl...	cs	cac...	564 B							
26	GET		css.php?sty...	styl...	cs	6.14...	25.3...	306 ms						
26	GET		font-awesome...	styl...	cs	cac...	30.2...							

1.62 MB / 506.48 KB transferred | Finish: 6.80 min

Filter request parameters

Form data

my_post_key: [REDACTED]

username: [REDACTED]

password: [REDACTED]

remember: yes

submit: Login

action: do_login

url: [REDACTED]

```
PARAMS = {
    "username": [REDACTED],
    "password": [REDACTED],
    "remember": "yes",
    "submit": "Login",
    "action": "do_login",
    "url": "[REDACTED]"
}

SESSION.post("https://[REDACTED]", data=PARAMS, headers=HEADER)

res = SESSION.get([REDACTED])
```

Captcha

☐ Remember me?

Please validate the following expression:

4 + 3 =

- Reconocimiento OCR

→ Pytesseract

- ML

→ Tesseract

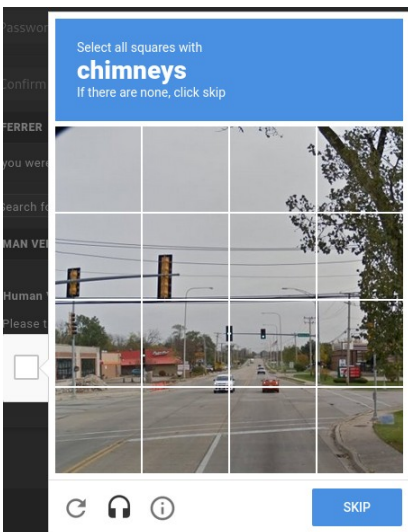
→ Tensorflow

```
import pytesseract, PIL

value = pytesseract.image_to_string(PIL.Image.open("captcha.jpeg"))
print("The Capcha value is --> " + value)
```

NQK 7 a

Captcha



- NonoCAPTCHA

- Basado en resolver CAPTCHA usando
- Mozilla DeepSpeech, PocketSphinx,
- Microsoft Azure y Amazon Transcribe

* <https://github.com/mikeyy/nonoCAPTCHA>



- Image Captcha
- Google Recaptcha
- FunCaptcha
- GeeTest/Distil
- hCaptcha

* <https://anti-captcha.com/apidoc/>

Evadiendo Captchas

Registro

- ✓ Registrarse manualmente
- ✓ Reusar credenciales

Autenticación

- ✓ Autenticación manual
- ✓ Reuso de Cookies

Carga de páginas

- ✓ Captcha Solver
- ✓ Look like human

Evitar invocar una petición de CAPTCHA

Prueba y error

! COOKIES !

Honeypots

Links que no pueden ser vistos por una persona navegando en el foro

“display:none”, “visibility:hidden”, “color:#fff”

Producen:

- Redireccionamiento a otras páginas
- Ejecución de Scripts sobre el navegador
- Banning

```
links = firefox.find_elements_by_tag_name("a")
print("[!] Possible trap links: ")
for link in links:
    if not link.is_displayed():
        if link.get_attribute("href") is not None:
            print(link.get_attribute("href"))
```

Anonimato y red Tor

- ✓ Evitar IP banning
- ✓ Scraping paralelamente

Beautiful Soup

```
TOR_SESSION = requests.session()

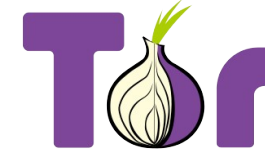
TOR_SESSION.proxies = {'http': 'socks5://127.0.0.1:9050',
                       'https': 'socks5://127.0.0.1:9050'}

res = TOR_SESSION.get("https://www.facebookcorewwi.onion/")
```



Stem

+



Selenium

```
profile = webdriver.FirefoxProfile()
profile.set_preference("network.proxy.type", 1)
profile.set_preference("network.proxy.socks", "127.0.0.1")
profile.set_preference("network.proxy.socks_port", 9150)

firefox = webdriver.Firefox(options=options, firefox_profile=profile)
```



- Acceso a todos los contenidos
- Sin registro o autenticación
- Sin límite de peticiones
- Sin CAPTCHA



- Acceso a contenido mediante registro y autenticación
- CAPTCHA en registro o autenticación
- Baneo Usuarios bot
- Ofuscación de contenido

- Contenido restringido a invitación
- Captcha tanto en registro como en autenticación
- Analiza comportamiento usuarios
 - Acceso diferentes IP
 - Acceso a contenido antiguo
- Baneo Usuarios bot
- Cambio tags y atributos en el DOM HTML

5

Cyber Threat Intelligence

Análisis a partir de foros underground

CrimeBB: Enabling Cybercrime Research on Underground Forums at Scale

Sergio Pastrana
Cambridge Cybercrime Centre, Computer Laboratory
University of Cambridge
Sergio.Pastrana@cl.cam.ac.uk

Alice Hutchings
Cambridge Cybercrime Centre, Computer Laboratory
University of Cambridge
Alice.Hutchings@cl.cam.ac.uk

Daniel R. Thomas
Cambridge Cybercrime Centre, Computer Laboratory
University of Cambridge
Daniel.Thomas@cl.cam.ac.uk

Richard Clayton
Cambridge Cybercrime Centre, Computer Laboratory
University of Cambridge
Richard.Clayton@cl.cam.ac.uk



[*https://www.cl.cam.ac.uk/~drt24/papers/2018-crimebb.pdf](https://www.cl.cam.ac.uk/~drt24/papers/2018-crimebb.pdf)

Let Me Cheat!

An analysis of anti-cheat bypass techniques on videogames

Author: David Rodríguez Regueira
Universidad Carlos III de Madrid
Leganés, Spain
100434170@alumnos.uc3m.es



Supervisor: Sergio Pastrana
Universidad Carlos III de Madrid
Leganés, Spain
Sergio.Pastrana@uc3m.es

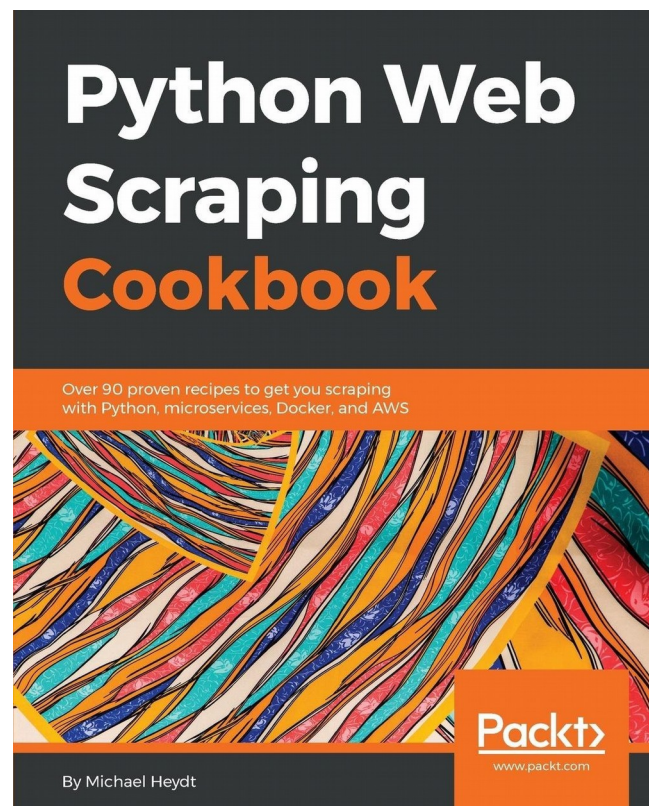
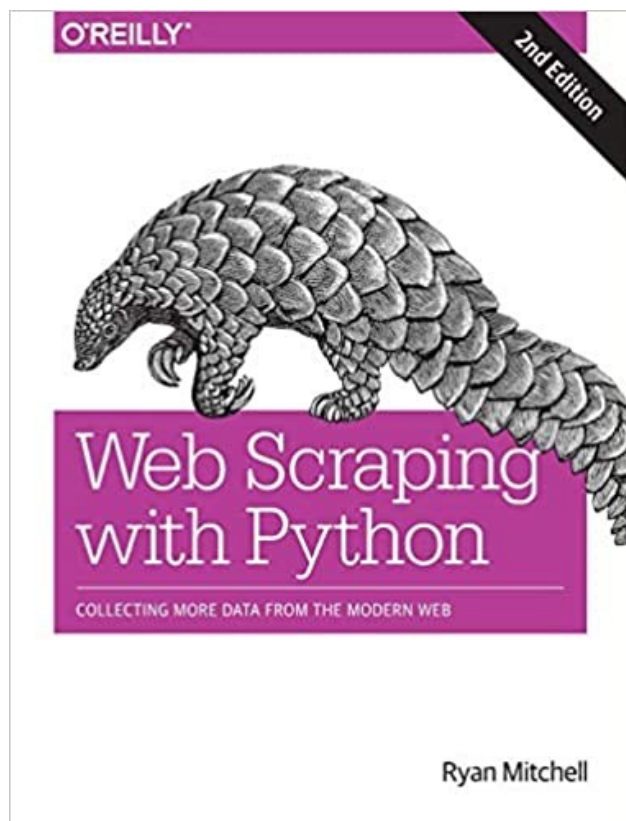
[*https://github.com/90n20/LetMeCheat](https://github.com/90n20/LetMeCheat)

SOC4IoCs: Scraping Online Community for Indicators of Compromise



Identifying Mobile Malware and Key Threat Actors in Online Hacker Forums for Proactive Cyber Threat Intelligence

John Grisham, Sagar Samtani, Mark Patton, Hsinchun Chen
Department of Management Information Systems
The University of Arizona
Tucson, AZ 85721
{johngrisham93, sagars, mpatton}@email.arizona.edu, hchen@eller.arizona.edu



- Tianjun Fu, Ahmed Abbasi, and Hsinchun Chen. A focused crawler for dark web forums. Journal of the American Society for Information Science and Technology, 61(6):1213–1231, 2010.
- Kieron Turk, Sergio Pastrana and Ben Collier. A tight scrape: methodological approaches to cybercrime research data collection in adversarial environments. https://www.cl.cam.ac.uk/~bjc63/tight_scrape.pdf
- Richard Frank, Mitch Macdonalds, and Bryan Monk. Location, Location, Location: Mapping Potential Canadian Targets in Online Hacker Discussion Forums. 2016 European Intelligence and Security Informatics Conference

#IntelCon2020



IntelCon
by Ginseg

Gracias por la atención



@sarvmetal



<https://github.com/santiagorocha>

Congreso Online de Ciberinteligencia | Julio 2020