

Creación de tools de Inteligencia

Desde localizar, analizar y enriquecer el dato hasta automatizar el proceso



The better the question. The better the answer.
The better the world works.

JORNADAS DE
SEGURIDAD Y
CIBERDEFENSA



EY

Building a better
working world



Contenido ...

- ▶ ¿Quiénes somos?
- ▶ Definiciones
- ▶ Búsqueda de información en OSINT
- ▶ Automatización de scripts de Inteligencia
- ▶ Arquitectura de una plataforma de Inteligencia



25 de Enero del 2017 / Universidad de Alcalá

EY

Building a better
working world



¿Quiénes somos?

- ▶ ¿Quiénes somos?
- ▶ Definiciones
- ▶ Búsqueda de información en OSINT
- ▶ Automatización de scripts de Inteligencia
- ▶ Arquitectura de una plataforma de Inteligencia

JORNADAS DE
SEGURIDAD Y
CIBERDEFENSA



EY

Building a better
working world



Wiktor Nykiel

Cyber Intelligence & Security Expert
Senior Manager - EY

 [wiktonykiel](#)

 [wiktonykiel](#)

 wiktor.nykiel@es.ey.com



Iván Portillo Morales

Cyber Intelligence Expert
Senior Analyst - EY

 [ivanportillomorales](#)

 [ivanPorMor](#)

 ivan.portillomorales@es.ey.com



Definiciones

- ▶ ¿Quiénes somos?
- ▶ Definiciones
- ▶ Búsqueda de información en OSINT
- ▶ Automatización de scripts de Inteligencia
- ▶ Arquitectura de una plataforma de Inteligencia

	OSINT	Inteligencia de Fuentes Abiertas. Información cuya procedencia se origina en fuentes públicas.
	RISP	Reutilización de la Información del Sector Publico. Iniciativa de Datos Abiertos promovido por red.es, la cual permite obtener información relacionada con dominios de manera pública.
	RIPE	Es el registro regional de internet encargado de supervisar y registrar los recursos de internet (Direcciones IP y AS) en las zonas de Europa, Oriente Medio y Asia Central.
	TLD	Dominio de nivel superior: com, net, cn, info, es, jp, gov, edu, mil, uk, de ...
	Docker	Proyecto de código abierto que permite automatizar el despliegue de aplicaciones dentro de contenedores virtualizados, proporcionando una abstracción con el resto de contenedores.
	Dockerfile	Archivo de configuración con los parámetros necesarios para crear una imagen en Docker.



Búsqueda de información en OSINT

JORNADAS DE
SEGURIDAD Y
CIBERDEFENSA



- ▶ ¿Quiénes somos?
- ▶ Definiciones
- ▶ Búsqueda de información en OSINT
- ▶ Automatización de scripts de Inteligencia
- ▶ Arquitectura de una plataforma de Inteligencia



<https://www.robtx.com>

Herramienta de footprinting pasiva que proporciona información relacionada con direcciones IP, dominios, subdominios, DNS, servidores de correo y blacklist.

The screenshot shows the Namech_k website interface. At the top, the search term 'wiktornykiel' is entered. Below the search bar, there is a legend with colored dots: green for 'Available', grey for 'Unavailable', red for 'Error', and yellow for 'Invalid'. The main content is divided into two sections: 'Domains' and 'Usernames'.

Domains: A grid of domain names is displayed, each with a green background indicating availability. The domains include: wiktornykiel.com, wiktornykiel.net, wiktornykiel.org, wiktornykiel.me, wiktornykiel.us, wiktornykiel.co, wiktornykiel.io, wiktornykiel.biz, wiktornykiel.co.uk, wiktornykiel.place, wiktornykiel.auto, wiktornykiel.xyz, wiktornykiel.tv, wiktornykiel.ninja, wiktornykiel.ink, wiktornykiel.bar, wiktornykiel.work, wiktornykiel.cars, wiktornykiel.eu, wiktornykiel.be, wiktornykiel.am, wiktornykiel.it, wiktornykiel.info, wiktornykiel.so, wiktornykiel.at, wiktornykiel.wtf, wiktornykiel.guru, wiktornykiel.pro, wiktornykiel.ms, wiktornykiel.ca, vegas, and .social.

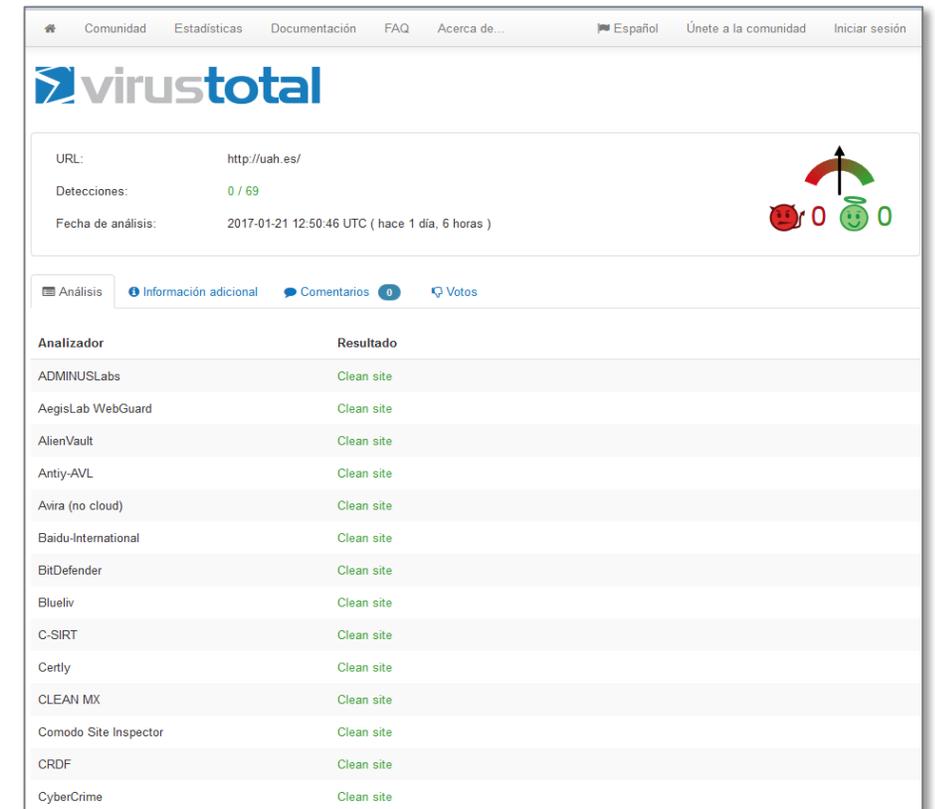
Usernames: A grid of social media platforms is shown, each with a green background and a checkmark indicating the presence of the username. The platforms include: Facebook, YouTube, Twitter, Instagram, Blogger, GooglePlus, Twitch, Reddit, ebay, Wordpress, Pinterest, Yelp, PayPal, Slack, Github, Vine, Basecamp, Tumblr, Flickr, Spotify, and Pandora.



	A	B	C	D	E	F
1	username,status,url					
2	wiktornykiel,available,http://www.google.com/					
3	wiktornykiel,unavailable,https://facebook.com/wiktornykiel					
4	wiktornykiel,unavailable,https://www.youtube.com/wiktornykiel					
5	wiktornykiel,available,https://plus.google.com/					
6	wiktornykiel,available,http://instagram.com					
7	wiktornykiel,available,http://www.twitch.tv/					
8	wiktornykiel,available,http://www.ebay.com/					
9	wiktornykiel,available,http://www.yelp.com/					
10	wiktornykiel,unavailable,http://pinterest.com/wiktornykiel/					
11	wiktornykiel,available,https://slack.com/					
12	wiktornykiel,unavailable,https://github.com/wiktornykiel					
13	wiktornykiel,available,https://www.paypal.me/					
14	wiktornykiel,available,https://vine.co/					
15	wiktornykiel,available,http://www.tumblr.com/					
16	wiktornykiel,available,https://basecamp.com					
17	wiktornykiel,available,http://www.reddit.com/					
18	wiktornykiel,available,https://www.spotify.com/					
19	wiktornykiel,unavailable,https://www.producthunt.com/@wiktornykiel					
20	wiktornykiel,available,http://www.flickr.com/					
21	wiktornykiel,available,http://www.pandora.com					
22	wiktornykiel,available,http://www.myspace.com/					
23	wiktornykiel,unavailable,https://foursquare.com/wiktornykiel					
24	wiktornykiel,available,http://steamcommunity.com/					
25	wiktornykiel,available,https://www.okcupid.com/					
26	wiktornykiel,available,http://vimeo.com/					
27	wiktornykiel,unavailable,https://flipboard.com/@wiktornykiel					

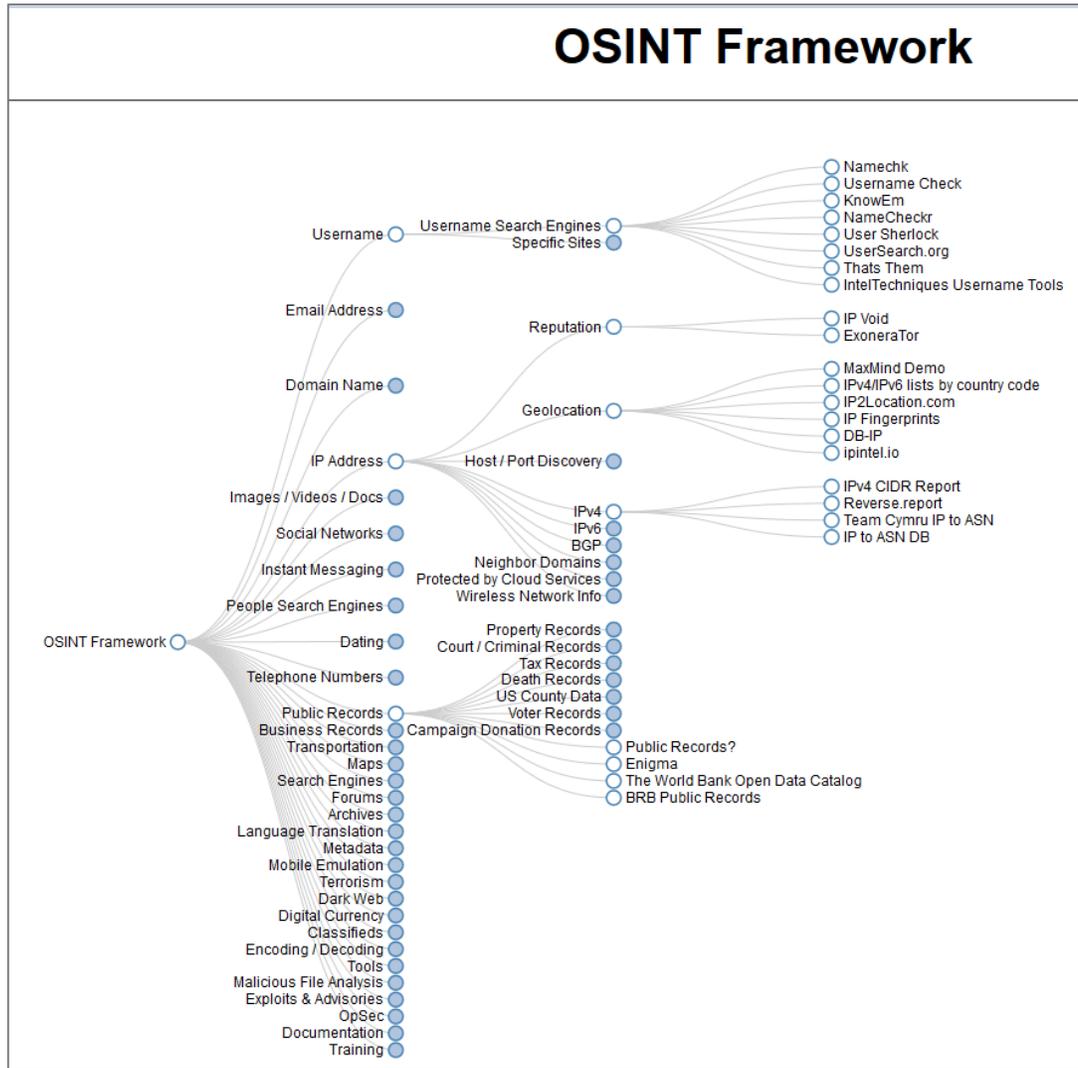
<https://namechk.com>

Herramienta que permite realizar búsquedas de un usuario en todas las redes sociales y comprobar coincidencias.



<https://www.virustotal.com>

Herramienta que posee una base de datos de todo tipo de malware y que permite comprobar si un fichero o una web está infectada y presenta amenazas.



<http://osintframework.com>

Web que posee una base de datos con enlaces URL de herramientas OSINT agrupadas por categorías y subcategorías en formato árbol.



<http://spiderfoot.net/download/>

Sistema Operativo
Linux y Windows

Dependencias Linux
pip install lxml netaddr M2Crypto cherrypy mako

Ejecución en Linux
python ./sf.py 0.0.0.0:5001

Ejecución en Windows ejecutar fichero sf.exe

Acceso web a herramienta localhost:5001

Herramienta de inteligencia de código abierto, que permite recopilar información sobre un objetivo concreto de manera automatizada.

Buscador de libre acceso, de desarrollo propio, para localizar amenazas en dominios con TLD .es relacionado con CyberSquatting y TypoSquatting.

The screenshot shows the web interface for CyberSquatting.es. At the top left, there is a dark red header with the text "Cybersquatting [es]". To the right of this header is a dark blue navigation bar with the text "SOBRE EL PROYECTO". Below the header, the main content area has a light gray background. At the top of this area, it says "Introduce el nombre dominio a analizar (sin TLD .es)". Below this text is a white input field and a "BUSCAR" button. Underneath the input field, there are two dropdown menus: "Opciones de Typos" (with "Distancia entre letras" selected) and "Porcentaje de similitud". Below these dropdowns is a slider control with two red dots, labeled "De 1 a 10 caracteres similares". At the bottom left of the interface is the "cyber camp" logo. At the bottom right, there is a copyright notice: "Copyright © 2016 Cyber.squatting.es - Servicio gratuito de análisis de Cybersquatting en TLD .es".

<http://cyber.squatting.es>



Automatización de scripts de Inteligencia

- ▶ ¿Quiénes somos?
- ▶ Definiciones
- ▶ Búsqueda de información en OSINT
- ▶ Automatización de scripts de Inteligencia
- ▶ Arquitectura de una plataforma de Inteligencia

JORNADAS DE
SEGURIDAD Y
CIBERDEFENSA



Dominios_es

<https://www.dominios.es/>

- Pdf – wget – PdfExtract
- RISP

RIPE

<https://www.ripe.net/>

domi
ni●s es

datos.gob.es
reutiliza la información pública



Extracción Estadísticas

Se obtienen listados del alta de dominios .es registrados mensualmente desde el año 2007.

<http://dominios.es/dominios/es/todo-lo-que-necesitas-saber/estadisticas/>

Dominio	Fecha Alta Dominio
gold-art.es	01/11/2016
diebold-nadorf.com.es	01/11/2016
asturleonaventura.es	01/11/2016
mitcho.es	01/11/2016
becasbarcelona.es	01/11/2016
ohmakeup.es	01/11/2016
markediem.es	01/11/2016
forstrun.es	01/11/2016
aviariotrescolores.es	01/11/2016
wwwparklink.es	01/11/2016
laposadadelmomo.es	01/11/2016
megadance.es	01/11/2016
beepear.es	01/11/2016
rickyspark.es	01/11/2016
elcamnodelanillo.es	01/11/2016
camsetasba2017.es	01/11/2016
uniockers.com.es	01/11/2016
isabelvalero.es	01/11/2016
saboreaponteareas.es	01/11/2016
arist.es	01/11/2016
e-moto.es	01/11/2016
jantoki.es	01/11/2016
redreando.es	01/11/2016
equipe.es	01/11/2016
hunafalikandy.es	01/11/2016
crazytraction.es	01/11/2016
carrerasguamavarro.es	01/11/2016
beststay.es	01/11/2016
caravanalanzarote.es	01/11/2016
mouababy.es	01/11/2016
bioemp.es	01/11/2016
meganor.es	01/11/2016
inagroup.es	01/11/2016
rati.es	01/11/2016
jugueteriamadrid.es	01/11/2016
barcelonaclick.es	01/11/2016
grupodager.es	01/11/2016
ramocoronadojavier.es	01/11/2016
yeclatoday.es	01/11/2016
protegermicasa.es	01/11/2016
rojodevelop.es	01/11/2016
lenguaguru.es	01/11/2016
agvisision.es	01/11/2016
ticapod.es	01/11/2016
glittercompany.es	01/11/2016
protegermihogar.es	01/11/2016
elcan.es	01/11/2016
cygeek.es	01/11/2016
centroesteticotharma.es	01/11/2016
alugolshouse.es	01/11/2016
hackr.es	01/11/2016
mentideroliterario.es	01/11/2016
pelisfix.es	01/11/2016
modamotera.es	01/11/2016
skyweb.es	01/11/2016
canperiquet.es	01/11/2016

landrivas.es	30/11/2016
tiendaderelegon.es	30/11/2016
welaw.es	30/11/2016
barriete.es	30/11/2016
startweekend.es	30/11/2016
bak.es	30/11/2016
tienda-ugg.es	30/11/2016
matocorilla.es	30/11/2016
santamarala.es	30/11/2016
encastellana.es	30/11/2016
estoposki.es	30/11/2016
agenciaseconcom.es	30/11/2016
herenciasmalaga.es	30/11/2016
rua7.es	30/11/2016
laltheenciamicabncn.es	30/11/2016
empresadrones.es	30/11/2016
dionand.es	30/11/2016
dronshops.es	30/11/2016
reviverd.es	30/11/2016
benepublicas.es	30/11/2016
robinhoodrestaurant.es	30/11/2016
horse-concept.es	30/11/2016
tablermecanocogitautogas.es	30/11/2016
expiracioncarmona.es	30/11/2016
onebooking2.es	30/11/2016
poisgurus.es	30/11/2016
academiologica.es	30/11/2016
helgafalics.es	30/11/2016
reyesdelasi.es	30/11/2016
e-moto.es	30/11/2016
visualfreeurope.es	30/11/2016
circular.es	30/11/2016
desayunoscanarias.es	30/11/2016
canciondelverano-ladival.es	30/11/2016
caballosespafoles.es	30/11/2016
mapasrespcanarias.es	30/11/2016
fototerreno.es	30/11/2016
larsanz.es	30/11/2016
namastonline.es	30/11/2016
labolsadewallstreet.es	30/11/2016
trisolpan.es	30/11/2016
manco.es	30/11/2016
longtermlet.es	30/11/2016
sentidamusica.es	30/11/2016
taulawconsulting.es	30/11/2016
podemosretremadura.es	30/11/2016
algobal.es	30/11/2016
canariagastro.es	30/11/2016
nanocoatings.es	30/11/2016
estrategnia.es	30/11/2016
soyborque.es	30/11/2016
adrianacometicanatural.es	30/11/2016
motorelax.es	30/11/2016
euromahua.es	30/11/2016
legomalaga.es	30/11/2016
superarcereentender.es	30/11/2016
suantos.es	30/11/2016
ruli.es	30/11/2016
teclhinoarecife.es	30/11/2016
occora.es	30/11/2016
centroesteticaluna.es	30/11/2016
seasons2.com.es	30/11/2016
alucinnet.es	30/11/2016
camisadateepcalm.es	30/11/2016
republicadeespaña.es	30/11/2016
ferrarcenrodebellezaavanzado.es	30/11/2016

Extracción Estadísticas

Se utiliza wget para descargar todos los PDFs de dominios registrados mensualmente

wget -r -A .pdf <http://www.dominios.es/dominios/es/todo-lo-que-necesitas-saber/estadisticas>

```
(ui) /public_html/seguridad/dominiosES/dominios/cybercamp$ wget -A .pdf -r http://www.dominios.es/dominios/es/todo-lo-que-necesitas-saber/estadisticas
converted 'http://www.dominios.es/dominios/es/todo-lo-que-necesitas-saber/estadisticas' (ANSI_X3.4-1968) -> 'http://www.dominios.es/dominios/es/todo-lo-que-necesitas-saber/estadisticas' (UTF-8)
converted 'http://www.dominios.es/dominios/es/todo-lo-que-necesitas-saber/estadisticas' (ANSI_X3.4-1968) -> 'http://www.dominios.es/dominios/es/todo-lo-que-necesitas-saber/estadisticas' (UTF-8)
--2016-12-01 23:51:35-- http://www.dominios.es/dominios/es/todo-lo-que-necesitas-saber/estadisticas
Resolving www.dominios.es (www.dominios.es)... 194.69.254.120, 2001:67c:21cc:2000::64:120
Connecting to www.dominios.es (www.dominios.es)|194.69.254.120|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [text/html]
Saving to: 'www.dominios.es/dominios/es/todo-lo-que-necesitas-saber/estadisticas'

www.dominios.es/dominios/es/todo-lo-que-n [ <=> ] 71.44K --.-KB/s in 0.1s

2016-12-01 23:51:37 (695 KB/s) - 'www.dominios.es/dominios/es/todo-lo-que-necesitas-saber/estadisticas' saved [73150]

converted 'http://www.dominios.es/dominios/sites/dominios/themes/redesms/favicon.ico' (ANSI_X3.4-1968) -> 'http://www.dominios.es/dominios/sites/dominios/themes/redesms/favicon.ico' (UTF-8)
```

```
./www.dominios.es/dominios/sites/dominios/files/files:
total 67524
5381780802 12 drwx---r-x 2 8192 Dec 2 00:01 .
4296657645 8 drwx---r-x 4 4096 Dec 1 23:59 ..
5369132768 632 -rw----r-- 1 645965 Nov 13 2014 1-15_10_2007.pdf
5408502494 400 -rw----r-- 1 406412 Mar 11 2013 Agentes Registradores Febrero 2013(1).pdf
5408513564 348 -rw----r-- 1 355015 Dec 18 2012 Agentes Registradores Noviembre 2012.pdf
5408513568 352 -rw----r-- 1 360047 Nov 16 2012 Agentes Registradores Octubre 2012.pdf
5408513559 348 -rw----r-- 1 354482 Jan 15 2013 Agentes Registradores diciembre 2012.pdf
5408513552 348 -rw----r-- 1 354053 Feb 11 2013 Agentes Registradores enero 2013.pdf
5383930970 896 -rw----r-- 1 915420 May 5 2014 Altas Abril 2014 (espa??ol).pdf
5383610266 1220 -rw----r-- 1 1245354 May 4 2015 Altas Abril 2015 (espa??ol).pdf
5408502476 704 -rw----r-- 1 717582 Sep 4 2013 Altas Agosto 2013.pdf
5383930959 1452 -rw----r-- 1 1483120 Sep 4 2014 Altas Agosto 2014(espa??ol)(1).pdf
5382381106 952 -rw----r-- 1 972497 Sep 1 2015 Altas Agosto 2015 (espa??ol).pdf
5408513560 1492 -rw----r-- 1 1524158 Jan 15 2013 Altas Diciembre 2012.pdf
5384162067 780 -rw----r-- 1 795228 Jan 9 2014 Altas Diciembre 2013 (espa??ol)(1).pdf
5383930947 620 -rw----r-- 1 634235 Jan 8 2015 Altas Diciembre 2014 (espa??ol).pdf
5408513556 1404 -rw----r-- 1 1435928 Feb 4 2013 Altas Enero 2013 (espa??ol).pdf
5384162055 924 -rw----r-- 1 942937 Feb 3 2014 Altas Enero 2014 (espa??ol).pdf
5383610286 1376 -rw----r-- 1 1405625 Feb 4 2015 Altas Enero 2015 (espa??ol)(1).pdf
5408502496 1208 -rw----r-- 1 1233310 Mar 4 2013 Altas Febrero 2013 (espa??ol).pdf
```

Is -lisaR

Extracción Estadísticas

Se parsea cada documento PDF a texto con *PDF Parser*. <https://github.com/smalot/pdfparser>

PdfParser

Pdf Parser, a standalone PHP library, provides various tools to extract data from a PDF file.

build passing downloads 164.11 k Current Version HHVM Partial

Website : <http://www.pdfparser.org>

Test the API on our [demo page](#).

This project is supported by [Actualys](#).

Features

Features included :

- Load/parse objects and headers
- Extract meta data (author, description, ...)
- Extract text from ordered pages
- Support of compressed pdf
- Support of MAC OS Roman charset encoding
- Handling of hexa and octal encoding in text sections
- PSR-0 compliant (autoloader)
- PSR-1 compliant (code styling)

Currently, secured documents are not supported.

This Library is still under active development. As a result, users must expect BC breaks when using the master version.

PDF Parser

PHP library to parse PDF files and extract elements like text.

build passing downloads 164.11 k stable v0.9.26

[Home](#) / [Demo](#)

Demo

Demo form

Select a file to be parsed with the PDF Parser API. Text will be extracted from each page and rendered below.

No se ha seleccionado ningún archivo.

Extracción Estadísticas

Para convertir PDF a Texto por CLI se utilizará la librería software *Apache PDFBox 2.0.3* (o superior).

<https://pdfbox.apache.org/>

<http://pdfbox.apache.org/download.cgi>



The screenshot shows the Apache PDFBox website. The left sidebar contains navigation links categorized into APACHE PDFBOX, COMMUNITY, DOCUMENTATION, and DEVELOPMENT. The main content area features the title 'Apache PDFBox® - A Java PDF Library', a description of the library, a news item about the 2.0.3 release, and a 'Getting Help' section. Below these are 'Features' listed in a grid, including Extract Text, Split & Merge, Fill Forms, Preflight, Print, Save as Image, Create PDFs, and Signing.

Apache PDFBox® - A Java PDF Library

The Apache PDFBox® library is an open source Java tool for working with PDF documents. This project allows creation of new PDF documents, manipulation of existing documents and the ability to extract content from documents. Apache PDFBox also includes several command line utilities. Apache PDFBox is published under the Apache License v2.0.

Apache PDFBox 2.0.3 released (2016-09-17)

The Apache PDFBox community is pleased to announce the release of Apache PDFBox version 2.0.3. It is available for download at:

<http://pdfbox.apache.org/download.cgi>

See the [full release notes](#) for details about this release.

Getting Help

To get help on using PDFBox, please [Subscribe to the Users Mailing List](#) and post your questions there. We're happy to help.

The project is a volunteer effort and we're always looking for interested people to help us improve PDFBox. There are a multitude of ways that you can help us depending on your skills. Subscribe to the [Mailing Lists](#) and find out how you can help.

Features

 Extract Text Extract Unicode text from PDF files.	 Split & Merge Split a single PDF into many files or merge multiple PDF files.	 Fill Forms Extract data from PDF forms or fill a PDF form.	 Preflight Validate PDF files against the PDF/A-1b standard.
 Print Print a PDF file using the standard Java printing API.	 Save as Image Save PDFs as image files, such as PNG or JPEG.	 Create PDFs Create a PDF from scratch, with embedded fonts and	 Signing Digitally sign PDF files.

Extracción Estadísticas

pdfbox-app-2.0.3.jar

java -jar pdfbox-app-2.0.3.jar ExtractText Altas-Noviembre-2016.pdf AltasNoviembre.txt

Altas-Noviembre-2016.pdf

Altas - Dominios 01-11-2016 al 30-11-2016	
Dominio	Fecha Alta Dominio
gold-ant.es	01/11/2016
diebold-nixdorf.com.es	01/11/2016
asturleonaventura.es	01/11/2016
mitecho.es	01/11/2016
becasbarcelona.es	01/11/2016
ohmakeup.es	01/11/2016
markediem.es	01/11/2016
ferratun.es	01/11/2016
aviariotrescolores.es	01/11/2016
wwwparklink.es	01/11/2016
laposadadelmomo.es	01/11/2016
megadance.es	01/11/2016
beepcar.es	01/11/2016
rickysarkany.es	01/11/2016
elcaminodelanillo.es	01/11/2016
camisetasnba2017.es	01/11/2016
unlockers.com.es	01/11/2016
isabelyalvaro.es	01/11/2016
saboreaponteareas.es	01/11/2016
asist.es	01/11/2016
e-moto.es	01/11/2016
jantoki.es	01/11/2016
redcreando.es	01/11/2016
expate.es	01/11/2016
hunasfallskandy.es	01/11/2016
crazytraction.es	01/11/2016
cerrajerijuannavarro.es	01/11/2016
beststay.es	01/11/2016
caravanaslanzarote.es	01/11/2016
mousebaby.es	01/11/2016
biohemp.es	01/11/2016
meganor.es	01/11/2016
inasgroup.es	01/11/2016
ratz.es	01/11/2016
jugueteriamadrid.es	01/11/2016
barcelonaclick.es	01/11/2016
grupodager.es	01/11/2016
ramoscoronadojavier.es	01/11/2016
yeclatoday.es	01/11/2016
protegermicasa.es	01/11/2016
riojadevelop.es	01/11/2016
lenguaguru.es	01/11/2016
agvision.es	01/11/2016
any.es	01/11/2016
hogar.es	01/11/2016
centrosoteriodharma.es	01/11/2016
abogadomhouse.es	01/11/2016
hacir.es	01/11/2016
mentideroliterario.es	01/11/2016
pelisflv.es	01/11/2016
modamotera.es	01/11/2016



```

1 Dominio Fecha Alta Dominio
2 gold-ant.es 01/11/2016
3 diebold-nixdorf.com.es 01/11/2016
4 asturleonaventura.es 01/11/2016
5 mitecho.es 01/11/2016
6 becasbarcelona.es 01/11/2016
7 ohmakeup.es 01/11/2016
8 markediem.es 01/11/2016
9 ferratun.es 01/11/2016
10 aviariotrescolores.es 01/11/2016
11 wwwparklink.es 01/11/2016
12 laposadadelmomo.es 01/11/2016
13 megadance.es 01/11/2016
14 beepcar.es 01/11/2016
15 rickysarkany.es 01/11/2016
16 elcaminodelanillo.es 01/11/2016
17 camisetasnba2017.es 01/11/2016
18 unlockers.com.es 01/11/2016
19 isabelyalvaro.es 01/11/2016
20 saboreaponteareas.es 01/11/2016
21 asist.es 01/11/2016
22 e-moto.es 01/11/2016
23 jantoki.es 01/11/2016
24 redcreando.es 01/11/2016
25 expate.es 01/11/2016
26 hunasfallskandy.es 01/11/2016
27 crazytraction.es 01/11/2016
28 cerrajerijuannavarro.es 01/11/2016
29 beststay.es 01/11/2016
30 caravanaslanzarote.es 01/11/2016
31 mousebaby.es 01/11/2016
32 biohemp.es 01/11/2016
33 meganor.es 01/11/2016
34 inasgroup.es 01/11/2016
35 ratz.es 01/11/2016
36 jugueteriamadrid.es 01/11/2016
37 barcelonaclick.es 01/11/2016
38 grupodager.es 01/11/2016
39 ramoscoronadojavier.es 01/11/2016
40 yeclatoday.es 01/11/2016
41 protegermicasa.es 01/11/2016
42 riojadevelop.es 01/11/2016
43 lenguaguru.es 01/11/2016

```

AltasNoviembre.txt

RISP

Se accede a *Solicitud acceso listado dominios (RISP)* dentro del sitio oficial de red.es y se solicita la última Base de Datos con todos los dominios registrados con TLD .es.

Se necesita:

- URL <https://sede.red.gob.es>
- Certificado Digital

Solicitud acceso listado dominios (RISP) - Solicitud / Expediente: BORRADOR/2016/RISP

Información del solicitante

Documento identificativo	NIF	Documento identificativo	
Nombre	IVAN	Primer apellido	PORTILLO
Segundo apellido		Email	
Indique la finalidad de la solicitud	<input type="radio"/> Comercial <input type="radio"/> No comercial <input type="radio"/> Investigación de mercados <input type="radio"/> Desarrollo de aplicaciones <input checked="" type="radio"/> Estadísticas <input type="radio"/> Servicios de seguridad <input type="radio"/> Otros		
Indique el motivo de su solicitud	<input checked="" type="checkbox"/> Propósito solicitud Investigación sobre estadísticas de dominios.es		

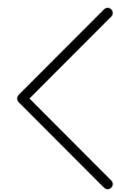
Anterior Siguiente Salir

RISP

Se accede al último registro recibido en la solicitud abierta y se descarga el fichero CSV.

```

RISP_OTROS.csv
1 "NOMBRE_DOMINIO"|"NOMBRE_TITULAR"|"IDENTIFICACION"
2 "idealia.es"|"Idealia Comunicacion Y Marketing S.L.L."|"B-82554056"
3 "andresca.es"|"Indresca S.A."|"A-08116808"
4 "Fundacionpatrimoniocyl.es"|"Fundacion del Patrimonio Historico de Castilla y Leon"|"G-47392618"
5 "ecf.es"|"ECF Escriptori Comptable i Fiscal S.L."|"B-59927962"
6 "alimotor.es"|"Alimotor S.A."|"A-03303609"
7 "cofiber.es"|"Cofiber Financiera Establecimiento Financiero de Credito S.A."|"A-79352274"
8 "agroland.es"|"Alvaro Rodriguez Eiras S.L."|"B-27015304"
9 "vilalba.es"|"Ayuntamiento de Vilalba"|"P-2706500-B"
10 "powerline.es"|"Fower Line Marketing Telefonico S.L."|"B-81072407"
11 "down21.es"|"Fundacion Down 21"|"G-82737024"
12 "dimac.es"|"Comercial Dimac S.L."|"B-58130626"
13 "arkhitekton.es"|"Arkhitekton S.A."|"A-39225859"
14 "cofeserbit.es"|"Cofeserbit S.L."|"B-60387859"
15 "microcad.es"|"Micro Cad Informatica S.L."|"B-29627510"
16 "neomania.es"|"Ibermaison S.L."|"B81049892"
17 "mfe.es"|"Mondial Forni Espan-a S.A."|"A-08396152"
18 "microtech.es"|"Microtech Sistemas S.L."|"B-61022943"
19 "carnesfrescassa.es"|"Carnes Frescas S.A."|"A-12027686"
20 "productosseco.es"|"Productos Seco S.L."|"B-24340382"
21 "fut.es"|"ORGANISME AUTONOM PER A LA SOCIETAT DE LA INFORMACIO"|"P4300025F"
22 "urv.es"|"Universitat Rovira i Virgili"|"Q-9350003-A"
23 "calvoymunarsa.es"|"Calvo y Munar S.A."|"A-28027068"
24 "ras.es"|"Grabados Electroquimicos Ras S.L."|"B-28004968"
25 "rafer.es"|"Comercial Rafer S.L."|"B-50045988"
26 "pmp.es"|"PMP Management Factory, S.L."|"B95590865"
27 "comtech.es"|"Computer Technology Catalunya S.L."|"B-60960200"
28 "ceona.es"|"Tecnologia de Aislamientos y Climatizacion S.L."|"B-81725970"
29 "albura.es"|"Asociacion Cultural Recreativo Deportiva Albura"|"G-79133245"
30 "belgicast.es"|"Belgicast Internacional S.L."|"B-48750806"
31 "arancia.es"|"Arancia S.L."|"B-20602264"
32 "coyzer.es"|"Coyzer S.A."|"A-58107053"
33 "queubia.es"|"Calizados Fal S.A."|"A-26004978"
34 "telecal.es"|"Telecal Instalaciones de Telefonía S.L."|"B-04264974"
35 "reteal.es"|"Red de Telefonía Almeriense S.L."|"B-04292082"
36 "rovschach.es"|"Sociedad Andaluza de Rorschach y Metodos Projectivos"|"G-41830670"
37 "juarez.es"|"Graficas Juarez S.L."|"B-03422334"
38 "menta.es"|"Auna Telecomunicaciones S.A."|"A-60774429"
39 "indatronic.es"|"Indatronic S.L."|"B-04271193"
40 "movinet.es"|"Movinet Valencia S.L."|"B-96852421"
41 "irta.es"|"Institut de Recerca i Tecnologia Agroalimentaries"|"Q-5855049-B"
42 "consejo-estado.es"|"Consejo de Estado"|"S-2811013-B"
43 "bingosreunidos.es"|"Bingos Reunidos S.A."|"A-03193323"
44 "fripanel.es"|"Paneles Frigoríficos S.A."|"A-08563009"
45 "lafincosomasaguas.es"|"Urbanizadora Somosaguas S.A.U."|"A-28069441"
46 "imathia.es"|"Imathia S.L."|"B-79158325"
47 "castillosasociacion.es"|"Asociacion Espan-ola de Amigos de los Castillos"|"G-28230746"
48 "scpsa.es"|"Servicios de la Comarca de Pamplona S.A."|"A-31118441"
49 "alpine.es"|"Alpine Electronics de Espan-a S.A."|"A-01102995"
50 "ums.es"|"United Metal Supply Spain S.A."|"A-82302886"
51 "kronospan.es"|"Kronospan Spain S.L."|"B-83020081"
52 "pgmedia.es"|"Promocion y Gestion Multimedia S.L."|"B-48832109"
53 "clubporsche.es"|"Club Porsche Espan-a"|"G-28786515"
54 "lccsa.es"|"Inmobiliaria Comercial Cerdena S.A."|"A-08100570"
55 "maper.es"|"Recubrimientos Maper S.L."|"B-96541099"
56 "fpa.es"|"Fundacion Principe de Asturias"|"G-33039348"
57 "uam.es"|"Universidad Autonoma de Madrid"|"Q-2818013-A"
    
```



Detalle de la solicitud / expediente

Descripción

Solicitud / Expediente: 2016/RISP
 Título: IVAN PORTILLO
 Tipo de solicitud: Tipo de Expediente acceso listado dominios (RISP)
 Fecha de alta: 05/11/2016
 Estado actual: DESCARGA DE DOMINIOS
 Organismo: DOMINIOS

Historia

Situación	Fecha
DESCARGAR LISTADO DE DOMINIOS (DESCARGA DE DOMINIOS)	05/11/2016
PRESENTACIÓN EN CURSO	05/11/2016

Documentación

LISTADO DE DOMINIOS .ES Nombre: RISP_OTROS.csv.bz2	Estado: «No vigente»	Fecha: 05/11/2016	Vigencia: 10/11/2016	
RECIBI SOLICITUD Nombre: recib.pdf	Estado: «Firmado»	Fecha: 05/11/2016		
SOLICITUD RISP Nombre: SOLICITUD RISP.pdf	Estado: «Firmado»	Fecha: 05/11/2016		

Pasos a seguir

- Acceso CSV durante 5 días.
- Control de Base de Datos de 1.842.568 de dominios .es.

Descargando fichero

Index of /ripe/dbase

Name	Last modified	Size	Description
Parent Directory		-	
RIPE.CURRENTSERIAL	20-Jan-2017 00:00	8	
ripe.db.gz	20-Jan-2017 01:20	336M	
split/	20-Jan-2017 00:00	-	

Index of /ripe/dbase/split

Name	Last modified	Size	Description
Parent Directory		-	
ripe.db.as-block.gz	20-Jan-2017 01:20	9.4K	
ripe.db.as-set.gz	20-Jan-2017 01:20	2.0M	
ripe.db.aut-num.gz	20-Jan-2017 01:20	7.5M	
ripe.db.domain.gz	20-Jan-2017 01:20	19M	
ripe.db.filter-set.gz	20-Jan-2017 01:20	395K	
ripe.db.inet-rtr.gz	20-Jan-2017 01:20	11K	
ripe.db.inet6num.gz	20-Jan-2017 01:20	23M	
ripe.db.inetnum.gz	20-Jan-2017 01:20	228M	
ripe.db.irt.gz	20-Jan-2017 01:20	27K	
ripe.db.key-cert.gz	20-Jan-2017 01:20	15M	
ripe.db.mntner.gz	20-Jan-2017 01:20	1.6M	
ripe.db.organisation.gz	20-Jan-2017 01:20	4.5M	
ripe.db.peering-set.gz	20-Jan-2017 01:20	72K	
ripe.db.person.gz	20-Jan-2017 01:20	409	
ripe.db.poem.gz	20-Jan-2017 01:20	29K	
ripe.db.poetic-form.gz	20-Jan-2017 01:20	1.3K	
ripe.db.role.gz	20-Jan-2017 01:20	2.7M	
ripe.db.route-set.gz	20-Jan-2017 01:20	229K	
ripe.db.route.gz	20-Jan-2017 01:20	8.2M	
ripe.db.route6.gz	20-Jan-2017 01:20	713K	
ripe.db.rtr-set.gz	20-Jan-2017 01:20	2.4K	

<http://ftp.ripe.net/ripe/dbase/>

<http://ftp.ripe.net/ripe/dbase/split/>

```

root@prometheus002:~/taller_ciberseg# cat ripe.db.inetnum
#
# The contents of this file are subject to
# RIPE Database Terms and Conditions
#
# http://www.ripe.net/db/support/db-terms-conditions.pdf
#
inetnum:      80.16.151.184 - 80.16.151.191
netname:      NETECONOMY-MG41731
descr:        TELECOM ITALIA LAB SPA
country:      IT
admin-c:      DUMY-RIPE
tech-c:       DUMY-RIPE
status:       ASSIGNED PA
notify:       neteconomy.rete@telecomitalia.it
mnt-by:       INTERB-MNT
created:      1970-01-01T00:00:00Z
last-modified: 2001-09-21T22:08:01Z
source:       RIPE
remarks:      *****
remarks:      * THIS OBJECT IS MODIFIED
remarks:      * Please note that all data that is generally regarded as personal
remarks:      * data has been removed from this object.
remarks:      * To view the original object, please query the RIPE Database at:
remarks:      * http://www.ripe.net/whois
remarks:      *****

% Tags relating to '80.16.151.184 - 80.16.151.191'
% RIPE-USER-RESOURCE

inetnum:      80.16.151.180 - 80.16.151.183
netname:      NETECONOMY-MG41731
descr:        TELECOM ITALIA LAB SPA
country:      IT
admin-c:      DUMY-RIPE
tech-c:       DUMY-RIPE
status:       ASSIGNED PA
notify:       neteconomy.rete@telecomitalia.it
mnt-by:       INTERB-MNT
created:      1970-01-01T00:00:00Z
last-modified: 2001-09-21T22:08:01Z
source:       RIPE
remarks:      *****
remarks:      * THIS OBJECT IS MODIFIED
remarks:      * Please note that all data that is generally regarded as personal
remarks:      * data has been removed from this object.
remarks:      * To view the original object, please query the RIPE Database at:
remarks:      * http://www.ripe.net/whois
    
```

Contenido de archivo parcial de ripe.db.inetnum

API

Application Programming Interface

<http://rest.db.ripe.net/search.json?query-string=EY>

Resultados en formato JSON



```
rest.db.ripe.net/search.json?query-string=ey
{"service": {
  "name": "search"
},
"parameters": {
  "inverse-lookup": { },
  "type-filters": { },
  "flags": { },
  "query-strings": {
    "query-string": [ {
      "value": "ey"
    } ]
  },
  "sources": { }
},
"objects": {
  "object": [ {
    "type": "inetnum",
    "link": {
      "type": "locator",
      "href": "http://rest.db.ripe.net/ripe/inetnum/194.165.144.48 - 194.165.144.55"
    },
    "source": {
      "id": "ripe"
    },
    "primary-key": {
      "attribute": [ {
        "name": "inetnum",
        "value": "194.165.144.48 - 194.165.144.55"
      } ]
    },
    "attributes": {
      "attribute": [ {
        "name": "inetnum",
        "value": "194.165.144.48 - 194.165.144.55"
      }, {
        "name": "netname",
        "value": "EY"
      }, {
        "name": "descr",
        "value": "E&Y"
      }, {
        "name": "descr",
        "value": "L.L customer for Orange"
      }, {
        "name": "remarks",
        "value": "for hacking, spamming or security problems send mail to"
      }, {
        "name": "remarks",
        "value": "====abuse@go.com.jo===="
      }, {
        "name": "country",
        "value": "jo"
      }
    ]
  }
]
}
```

API

script_ripe_api.py

```
#!/usr/bin/python
# -*- coding: utf-8 -*-
from urllib2 import Request, urlopen
import json

def copy_csv(final_result):
    csv = open('ripe_csv.csv','a+')
    print final_result

    csv.write('inetnum;netname;descr;descr2;descr3\n')

    for dict1 in final_result:
        for dict2 in dict1:
            for key,value in dict2.items():
                csv.write(value+';')
            csv.write('\n')

    csv.close()

def search_ripe():
    inetnum=[]
    request = Request("http://rest.db.ripe.net/search.json?query-string=EY")
    response_body = urlopen(request).read()
    tmp = json.loads(response_body)
    ripe=tmp['objects']['object']
    final_result=[]
    for i in ripe:
        if 'inetnum' in i['type']:
            total=i
            line_result=[]
            for line in total['attributes']['attribute']:

                if "inetnum" in line['name'] or "netname" in line['name'] or "descr" in line['name']:

                    file={str(line['name']):str(line['value'])}
                    line_result.append(file)

            final_result.append(line_result)

    copy_csv(final_result)

if __name__ == '__main__':
    search_ripe()
```

Se automatiza el proceso y se genera un CSV con los datos, el script consiste en la obtención de los rangos, nombre y descripciones por la que está registrado una organización buscada dentro de RIPE.



	A	B	C	D	E
1	inetnum	netname	descr	descr	descr
2	194.165.144.48 - 194.165.144.55	EY	E&Y	LL customer for Orange	
3	194.165.144.224 - 194.165.144.231	EY	Ernst & Young	Amman- Jordan	LL CUSTOMER FOR ORANGE
4	213.178.229.128 - 213.178.229.135	EY	Ernst & Young branch in Damascus Syria.		

```
#!/usr/bin/python
# -*- coding: utf-8 -*-
import pythonwhois
import argparse
import os.path
def copy_csv(final_result, csv, domain):
    heads_csv=['city', 'fax', 'name', 'state', 'phone', 'street', 'country', 'postalcode', 'organization', 'email']
    head={}
    head['domain']=domain
    head['city']=''
    head['fax']=''
    head['name']=''
    head['state']=''
    head['phone']=''
    head['street']=''
    head['country']=''
    head['postalcode']=''
    head['organization']=''
    head['email']=''

    for key in final_result:
        for key2,value in key.items():
            for head_csv in heads_csv:
                if key2 in head_csv:
                    head[head_csv]=value

    csv.write(head['domain']+','+head['city']+','+head['fax']+','+head['name']+','+head['state']+','+head['phone']+','+head['street']+','+head['country']+','+head['postalcode']+','+head['organization']+','+head['email']+','+'\n')

def find_csv(final_result, domain):
    if os.path.exists('whois_csv.csv') is False:
        csv = open('whois_csv.csv', 'a')
        csv.write('Domain,City,Fax,name,State,Phone,Street,Country,Postal Code,Organization,Email\n')
        copy_csv(final_result, csv, domain)
        csv.close()
    else:
        csv = open('whois_csv.csv', 'a')
        copy_csv(final_result, csv, domain)
        csv.close()

def search_whois(domain):
    print domain
    final_result=[]
    domain_whois = pythonwhois.get_whois(domain)
    result_domain=domain_whois['contacts']

    for key,value in result_domain.items():
        if 'registrant' in key:
            print value
            final_result.append(value)

    find_csv(final_result, domain)

if __name__ == '__main__':
    parser = argparse.ArgumentParser()
    parser.add_argument("-d","--domain", help="insert domain")
    args = parser.parse_args()
    if not args.domain:
        print "Introduce domain with -d or --domain"
    else:
        search_whois(args.domain)
```

script_whois.py

Paquetes necesarios

pip install python-whois

El script consiste en la obtención de la información relacionada con el whois de un objetivo concreto.

Se automatiza el proceso y se genera un CSV con los datos.



	A	B	C	D	E	F	G	H	I	J	K
1	Domain	City	Fax	name	State	Phone	Street	Country	Postal Code	Organization	Email
2	marca.com	Madrid		UNIDAD EDITORIAL INFORMACION DEPORTIVA S L U		34.914.435.907	Avda San Luis, 25	ES	28033	UNIDAD EDITORIAL INFORMACION DEPORTIVA, S.L.U	dominios@herrero.es

```
#!/usr/bin/python
# -*- coding: utf-8 -*-
import socket
import argparse
import os
import json
from geoiP import geolite2

def copy_csv(geoiP_result, csv, domain):

    domain_result=domain
    country=str(geoiP_result.country)
    continent=str(geoiP_result.continent)
    timezone=str(geoiP_result.timezone)
    location=str(geoiP_result.location)

    csv.write(domain_result+';'+country+';'+continent+';'+timezone+';'+location+';\n')

def find_csv(geoiP_result, domain):

    if os.path.exists('geoiP_csv.csv') is False:
        csv = open('geoiP_csv.csv', 'a+')
        csv.write('Domain;Country;Continent;Timezone;Location\n')
        copy_csv(geoiP_result, csv, domain)
        csv.close()
    else:
        csv = open('geoiP_csv.csv', 'a+')
        copy_csv(geoiP_result, csv, domain)
        csv.close()

def search_geoiP(domain):
    print domain

    ip = socket.gethostbyname(domain)
    geoiP_result = geolite2.lookup(ip)
    find_csv(geoiP_result, domain)

if __name__ == '__main__':
    parser = argparse.ArgumentParser()
    parser.add_argument("-d", "--domain", help="insert domain")
    args = parser.parse_args()
    if not args.domain:
        print "Introduce domain with -d or --domain"
    else:
        search_geoiP(args.domain)
```

script_geoiP.py

	A	B	C	D	E
1	Domain	Country	Continent	Timezone	Location
2	google.com	US	NA	America/Los_Angeles	(37.419200000000004, -122.0574)
3	marca.com	ES	EU	None	(40.4, -3.6833)



Paquetes necesarios pip install python-geoiP-geolite2

El script consiste en la obtención de información relacionada con la geolocalización de un dominio concreto. Se automatiza el proceso y se genera un CSV con los datos.

script_mx_ns.py

```
#!/usr/bin/python
# -*- coding: utf-8 -*-
import argparse
import os
import json
import dns.resolver
import csv

def copy_csv(domain,a,ns,mx,csv):
    list_ns={}
    list_mx={}
    list_ns['ns1']=1
    list_ns['ns2']=1
    list_ns['ns3']=1
    list_ns['ns4']=1
    list_mx['mx1']=1
    list_mx['mx2']=1
    list_mx['mx3']=1
    list_mx['mx4']=1
    cont_ns=0
    cont_mx=0

    for i in ns:
        cont_ns=cont_ns+1
        list_ns['ns'+str(cont_ns)]=i

    for i in mx:
        cont_mx=cont_mx+1
        list_mx['mx'+str(cont_mx)]=i

    csv.write(domain+';'+a[0]+';'+list_ns['ns1']+';'+list_ns['ns2']+';'+list_ns['ns3']+';'+list_ns['ns4']+';'+list_mx['mx1']+';'+list_mx['mx2']+';'+list_mx['mx3']+';'+list_mx['mx4']+';\n')

def find_csv(domain,a,ns,mx):
    if os.path.exists('domains_csv.csv') is False:
        csv = open('domains_csv.csv', 'a+')
        csv.write('Domain;A;NS1;NS2;NS3;NS4;MX1;MX2;MX3;MX4\n')
        copy_csv(domain,a,ns,mx,csv)
        csv.close()
    else:
        csv = open('domains_csv.csv', 'a+')
        copy_csv(domain,a,ns,mx,csv)
        csv.close()

def search_ns(domain):
    list_ns=[]
    num_ns=0
    try:
        answers_ns = dns.resolver.query(domain, 'NS')
        for i in answers_ns:
            num_ns=num_ns+1
            if num_ns==1:
                ns1=i.to_text()
                list_ns.append(ns1)
            elif num_ns==2:
                ns2=i.to_text()
                list_ns.append(ns2)
            elif num_ns==3:
                ns3=i.to_text()
                list_ns.append(ns3)
            elif num_ns==4:
                ns4=i.to_text()
                list_ns.append(ns4)
    except:
        print "error"
        return list_ns
```

```
def search_mx(domain):
    list_mx=[]
    num_mx=0
    try:
        answers_mx = dns.resolver.query(domain, 'MX')
        for i in answers_mx:
            num_mx=num_mx+1
            if num_mx==1:
                mx1=i.exchange
                list_mx.append(mx1)
            elif num_mx==2:
                mx2=i.exchange
                list_mx.append(mx2)
            elif num_mx==3:
                mx3=i.exchange
                list_mx.append(mx3)
            elif num_mx==4:
                mx4=i.exchange
                list_mx.append(mx4)
    except:
        mx1=""
        mx2=""
        mx3=""
        mx4=""
        return list_mx

def search_domains(domain):
    print domain
    a=[]
    try:
        answer = dns.resolver.query(domain, 'A')
        for i in answer:
            a.append(str(i))
    except:
        a=[]
        print a
        ns=search_ns(domain)
        mx=search_mx(domain)
        print a, domain, ns, mx
        find_csv(domain,a,ns,mx)

if __name__ == '__main__':
    parser = argparse.ArgumentParser()
    parser.add_argument("-d", "--domain", help="insert domain")
    args = parser.parse_args()
    if not args.domain:
        print "Introduce domain with -d or --domain"
    else:
        search_domains(args.domain)
```

El script consiste en la obtención de la información relacionada con la IP, nombres de dominio y servidores de correo de un dominio concreto.

Se automatiza el proceso y se genera un CSV con los datos.



	A	B	C	D	E	F	G	H	I	J
1	Domain	A	NS1	NS2	NS3	NS4	MX1	MX2	MX3	MX4
2	google.com	216.58.214.174	ns2.google.com.	ns1.google.com.	ns4.google.com.	ns3.google.com.	alt4.aspmx.l.google.com.	alt1.aspmx.l.google.com.	aspmx.l.google.com.	alt2.aspmx.l.google.com.

El script consiste en la generación de un listado de un dominio, combinándolo con diferentes TLDs para obtener de manera automatizada todos los registros A, NS y MX de cada uno de ellos. Permitirá comprobar si cada unos de los dominios tiene relación entre si.

```
#!/usr/bin/python
# -*- coding: utf-8 -*-
import pythonwhois
import argparse
import os.path
from script_tlds_mx_ns import search_domains

def search_tld(domain,file):
    final_result=[]
    domain_all=[]
    domain_sintld=domain.split('.')

    tlds=open(file,'r')
    for tld in tlds.read().split('\r\n'):
        domain_all.append(domain_sintld[0]+tld)

    for domain in domain_all:
        print domain
        search_domains(domain)

if __name__ == '__main__':
    parser = argparse.ArgumentParser()
    parser.add_argument("-d","--domain", help="insert domain")
    parser.add_argument("-f","--file", help="insert file")
    args = parser.parse_args()
    if not args.domain:
        print "Introduce domain with -d or --domain"
    elif not args.file:
        print "Introduce file with -f or --file"
    else:
        search_tld(args.domain,args.file)
```



script_tlds_mx_ns.py

#	A	B	C	D	E	F	G	H	I	J
1	Domain	A	NS1	NS2	NS3	NS4	MX1			
2	uah.it	[80.92.84.139]	ns4.eurodns.com.	ns1.eurodns.com.	ns2.eurodns.com.	ns3.eurodns.com.	mail.b-io.co.			
3	uah.biz	[176.31.179.191], [37.187.83.72]	dns2.ru-tld.ru.	dns1.ru-tld.ru.						
4	uah.eu	[109.106.167.8]	dns4.blavem.nl.	dns5.blavem.nl.			postdul.natuurpark.nl.			
5	uah.net	[185.53.179.8]	ns2.parkingcrew.net.	ns1.parkingcrew.net.			mail.h-email.net.			
6	uah.org	[75.119.201.101]	ns.xmission.com.	ns1.xmission.com.	ns2.xmission.com.		mx.xmission.com.			
7	uah.com	[208.73.210.212], [208.73.210.214], [208.73.210.217], [208.73.210.219]	ns1.dsredirection.com.	ns2.dsredirection.com.						
8	uah.info	[37.187.83.72], [176.31.179.191]	dns2.ru-tld.ru.	dns1.ru-tld.ru.						
9	uah.gov	[]								
10	uah.ch	[72.52.4.119]	ns1.sedoparking.com.	ns2.sedoparking.com.			localhost.			
11	uah.ru	[193.124.57.181]	ns.sutel.ru.	ns.66p.su.			emx.mail.ru.			
12	uah.us	[63.172.201.153]	ns1.uniregistrymarket.link.	ns2.uniregistrymarket.link.						
13	uah.post	[]								
14	uah.es	[159.146.56.125]	dns.uah.es.	dns3.uah.es.	chico.rediris.es.	sun.rediris.es.	uah-es.mail.protection.outlook.com.			
15	uah.fr	[52.58.78.16]	ns1.undeveloped.com.	ns2.undeveloped.com.			mx.uah.fr.			
16	uah.int	[]								
17	uah.pro	[]								
18	uah.tel	[]								
19	uah.co	[184.168.221.96]	ns01.cashparking.com.				smtp.secureserver.net.	mailstore1.secureserver.net.		
20	uah.uk	[]								
21	uah.su	[82.146.61.215]	ns110.inhostedns.com.	ns310.inhostedns.org.	ns210.inhostedns.net.		mx15.ukraine.com.ua.	mx20.ukraine.com.ua.		
22	uah.ar	[]								
23	uah.at	[109.235.63.103]	ns1.golem.eu.	ns2.golem.eu.	ns3.golem.eu.					
24	uah.ae	[]								
25	uah.be	[213.132.196.245]	ns.nlhosting.net.	ns1.nlhosting.net.			mailfilter1.webguru.nl.			
26	uah.br	[]								
27	uah.oa	[208.73.210.212], [208.73.211.163], [208.73.211.236], [208.73.210.219]	ns1.dsredirection.com.	ns2.dsredirection.com.						
28	uah.cl	[190.96.232.125]	secundario.nic.cl.	ns.uahurtado.cl.			proxmx.uahurtado.cl.	mx2.uahurtado.cl.		
29	uah.oy	[]								
30	uah.oz	[54.72.9.51]	ns2.parkingcrew.net.	ns1.parkingcrew.net.			mail.h-email.net.			
31	uah.de	[81.169.145.72]	docks05.rzone.de.	shades15.rzone.de.			smtpin.rzone.de.			
32	uah.dk	[46.30.213.154]	ns02.one.com.	ns01.one.com.			mxcluster2.one.com.	mxcluster1.one.com.		
33	uah.ee	[]								
34	uah.gr	[]								
35	uah.fi	[54.72.9.51]	ns1.parkingcrew.net.	ns2.parkingcrew.net.			mail.h-email.net.			
36	uah.hk	[]								
37	uah.hr	[88.99.30.30]	ns3.totohost.com.	ns2.totohost.com.	ns1.totohost.com.	ns4.totohost.com.	uah.hr.			
38	uah.lu	[]								
39	uah.mc	[]								
40	uah.me	[122.10.94.116]	fig1ns2.dnspod.net.	fig1ns1.dnspod.net.						
41	uah.nl	[109.106.167.8]	dns1.blavem.nl.	dns2.blavem.nl.	dns3.blavem.nl.		postdul.natuurpark.nl.			
42	uah.no	[194.63.248.52]	ns1.hyp.net.	ns2.hyp.net.	ns3.hyp.net.					
43	uah.pt	[]								
44	uah.pl	[80.92.84.139]	ns2.eurodns.com.	ns3.eurodns.com.	ns4.eurodns.com.	ns1.eurodns.com.	mail.b-io.co.			
45	uah.py	[]								
46	uah.se	[72.52.4.119]	ns2.sedoparking.com.	ns1.sedoparking.com.			localhost.			
47	uah.si	[]								
48	uah.ua	[]								

script_combinatoria_tlds.py

```
#!/usr/bin/python
# -*- coding: utf-8 -*-
import argparse
import os
import json
import shodan
from conf import API_SHODAN
def copy_csv(activo, rango, csv):
    for activo in activos:
        csv.write(rango+';'+str(activo['ip'])+';'+str(activo['asn'])+';'+str(activo['isp'])+';'+str(activo['puertos'])+';'+str(activo['modulo'])+';'+str(activo['org'])+';'+
            +str(activo['domains'])+';'+str(activo['location'])+';'+str(activo['hostnames'])+';'+str(activo['os'])+';'+str(activo['cpe'])+';'+str(activo['product'])+';'+
            +str(activo['vuln'])+';'+str(activo['ssh'])+';\n')
def find_csv(activo, rango):
    if os.path.exists('shodan_csv.csv') is False:
        csv = open('shodan_csv.csv', 'a+')
        csv.write(rango+';'+str(activo['ip'])+';'+str(activo['asn'])+';'+str(activo['isp'])+';'+str(activo['puertos'])+';'+str(activo['modulo'])+';'+str(activo['org'])+';'+
            +str(activo['domains'])+';'+str(activo['location'])+';'+str(activo['hostnames'])+';'+str(activo['os'])+';'+str(activo['cpe'])+';'+str(activo['product'])+';'+
            +str(activo['vuln'])+';'+str(activo['ssh'])+';\n')
        csv.close()
    else:
        csv = open('shodan_csv.csv', 'a+')
        copy_csv(activo, rango, csv)
        csv.close()
```

```
def search_shodan(rango):
    api = shodan.Shodan(API_SHODAN)
    dict_shodan['rango']=rango
    activo=[]
    print dict_shodan['rango']
    try:
        # búsqueda de rangos IP en Shodan
        host = api.search('net:'+'%s'%rango)
        # recorrido de cada resultado de shodan
        for i in host.matches:
            dict_shodan['ips']=[]
            dict_shodan['asn']=i.asn
            dict_shodan['isp']=i.isp
            dict_shodan['puertos']=i.puertos
            dict_shodan['org']=i.org
            dict_shodan['domains']=i.domains
            dict_shodan['hostnames']=i.hostnames
            dict_shodan['location']=i.location
            dict_shodan['ip']=i.ip
            dict_shodan['os']=i.os
            dict_shodan['cpe']=i.cpe
            dict_shodan['product']=i.product
            dict_shodan['modulo']=i.modulo
            dict_shodan['vuln']=i.vuln
            dict_shodan['ssh']=i.ssh
            host_buscar=api.host(i.ip_str)
            if 'ssh' in i:
                dict_shodan['ips'].append(i.ssh)
            if 'cpe' in i:
                dict_shodan['ips'].append(i.cpe)
            dict_shodan['ips'].append(i.os)
            dict_shodan['ips'].append(i.modulo)
            dict_shodan['ips'].append(i.domains)
            for it in host_buscar['data']:
                if 'org' in it:
                    dict_shodan['ips'].append(i.org).decode('utf-8')
                if 'asn' in it:
                    dict_shodan['ips'].append(i.asn).decode('utf-8')
            if 'vuln' in host_buscar:
                dict_shodan['ips'].append(i.vuln)
            dict_shodan['ips'].append(i.vuln)
            # Imprimamos los banners
            dict_shodan['ips'].append(i.ip)
            print(host_buscar['ip_str'])
            if 'product' in i:
                dict_shodan['ips'].append(i.product)
            dict_shodan['ips'].append(i.puertos)
            dict_shodan['ips'].append(i.ip)
            dict_shodan['ips'].append(i.location)
            dict_shodan['ips'].append(i.country_name)
            activo.append(dict_shodan['ips'])
            print activo
            find_csv(activo, rango)
            print '\n-----\n'
    except Exception as e:
        print 'Ha ocurrido un error: %s' % e
```

```
if __name__ == '__main__':
    parser = argparse.ArgumentParser()
    parser.add_argument('-r', '--rango', help="inserte rango")
    args = parser.parse_args()
    if not args.rango:
        args.rango='194.165.144.224-194.165.144.231'
    search_shodan(args.rango)
```

script_shodan.py



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	rango	ip	asn	isp	puertos	module	org	domains	location	hostnames	os	cpe	product	vuln	ssh
2	194.165.144.224-194.165.144.231	194.165.144.226	AS8376	Jordan Data Communications Company LLC	[500, 4500]	ike	Jordan Data Communications Company LLC	[]	Jordan	None	[]			[]	[]
3	194.165.144.224-194.165.144.231	194.165.144.232	AS8376	Jordan Data Communications Company LLC	[2123, 2152]	gtp-v1	Jordan Data Communications Company LLC	[]	Jordan	None	[]		GPRS Tunneling Protocol	[]	[]
4	194.165.144.224-194.165.144.231	194.165.144.226	AS8376	Jordan Data Communications Company LLC	[500, 4500]	ike-nat-t	Jordan Data Communications Company LLC	[]	Jordan	None	[]			[]	[]
5	194.165.144.224-194.165.144.231	194.165.144.232	AS8376	Jordan Data Communications Company LLC	[2123, 2152]	gtp-v1	Jordan Data Communications Company LLC	[]	Jordan	None	[]		GPRS Tunneling Protocol	[]	[]
6	194.165.144.224-194.165.144.231	194.165.144.229	AS8376	Jordan Data Communications Company LLC	[123]	ntp	Jordan Data Communications Company LLC	[]	Jordan	None	[]			[]	[]

Paquetes necesarios: pip install shodan

El script consiste en la obtención de la información relacionada con los datos que shodan posee sobre el objetivo a partir de un rango dado. Se automatiza el proceso y se genera un CSV con los datos.

```
#!/usr/bin/python
# -*- coding: utf-8 -*-

import argparse
import os.path
import nmap

def copy_csv(final_result, csv):
    for key, value in final_result.items():
        ip=key
        ports=""
        product=""
        state=""
        version=""
        extra_info=""
        cpe=""

        if value.has_key('status'):
            listaProtocolos = ['udp', 'tcp', 'icmp']
            for protocolo in listaProtocolos:
                if value.has_key(protocolo):
                    list_ports = value[protocolo].keys()
                    for port in list_ports:
                        ports=port
                        product = value[protocolo][port]['product']
                        state = value[protocolo][port]['state']
                        version = value[protocolo][port]['version']
                        extra_info = value[protocolo][port]['extrainfo']
                        cpe = value[protocolo][port]['cpe']
                        print str(key)+";"+str(ports)+";"+str(product)+";"+str(state)+";"+str(version)+";"+str(extra_info.replace(';',';'))+";"+str(cpe)+"\n"
                        csv.write(str(key)+";"+str(ports)+";"+str(product)+";"+str(state)+";"+str(version)+";"+str(extra_info.replace(';',';'))+";"+str(cpe)+"\n")

def find_csv(final_result):
    if os.path.exists('nmap_csv.csv') is False:
        csv = open('nmap_csv.csv', 'a')
        csv.write(ip;Ports;Product;State;Version;Extra Info;cpe\n')
        copy_csv(final_result, csv)
        csv.close()
    else:
        csv = open('nmap_csv.csv', 'a')
        copy_csv(final_result, csv)
        csv.close()

def ejecutar_nmap(host, argumentos, puertos=""): # Ejecutar comandos de nmap
    try:
        escaneo = nmap.PortScanner() # Abre el portscanner de nmap
        escaneo.scan(host, arguments=argumentos) # Ejecuta el comando con los argumentos pasados
        return escaneo
    except Exception:
        RuntimeWarning

def search_nmap(ip):
    print ip
    final_result=[]
    nmap_result=ejecutar_nmap(ip, "-sS -O -sV")

    for clave, valor in list(nmap_result._dict__items()): # recorre el resultado
        if isinstance(valor, dict):
            print valor['scan']
            find_csv(valor['scan'])

    #find_csv(final_result, domain)

if __name__ == '__main__':
    parser = argparse.ArgumentParser()
    parser.add_argument('-ip', '--ip', help="insert IP")
    args = parser.parse_args()
    if not args.ip:
        print "Introduce IP with -ip or --ip"
    else:
        search_nmap(args.ip)
```

script_nmap.py

Paquetes necesarios: pip install python-nmap

El script consiste obtener información de los servicios activos de una dirección IP como objetivo. Se automatiza el proceso y se genera un CSV con los datos.



	A	B	C	D	E	F	G
1	ip	Ports	Product	State	Version	Extra Info	cpe
2	127.0.0.1	22	OpenSSH	open	7.2p2 Ubuntu 4ubuntu2.1	Ubuntu Linux, protocol 2.0	cpe:/o:linux:linux_kernel

The Harvester es una herramienta desarrollada en Python, que permite recopilar información pública a través de los buscadores. Se pueden obtener emails, subdominios, IPs, nombre de perfiles de usuario en redes sociales.



<https://github.com/laramies/theHarvester>

```
python theHarvester.py -d uah.es -l 300 -b all -f result.html
```



Listado de
Emails

Listado de
Subdominios e IPs

Listado de nombres de
usuarios de Redes Sociales

```
*****
*
* TheHarvester
*
* TheHarvester Ver. 2.7
* Coded by Christian Martorella
* Edge-Security Research
* cmartorella@edge-security.com
*****

Usage: theharvester options

-d: Domain to search or company name
-b: data source: baidu, bing, bingapi, dogpile.google, googleCSE,
    googleplus, google-profiles, linkedin, pgp, twitter, vhost,
    yahoo, all

-s: Start in result number X (default: 0)
-v: Verify host name via dns resolution and search for virtual hosts
-f: Save the results into an HTML and XML file (both)
-n: Perform a DNS reverse query on all ranges discovered
-c: Perform a DNS brute force for the domain name
-t: Perform a DNS TLD expansion discovery
-e: Use this DNS server
-l: Limit the number of results to work with(bing goes from 50 to 50 results,
    google 100 to 100, and pgp doesn't use this option)
-h: use SHODAN database to query discovered hosts

Examples:
theHarvester.py -d microsoft.com -l 500 -b google -h myresults.html
theHarvester.py -d microsoft.com -b pgp
theHarvester.py -d microsoft -l 200 -b linkedin
theHarvester.py -d apple.com -b googleCSE -l 500 -s 300
```

HaveIbeenPwned es una herramienta web que permite comprobar si un email se encuentra en alguno de las base de datos de leaks. Se pretende desarrollar una herramienta que permita consultar mediante su API, que emails se encuentran filtrados en algún leak, utilizando el listado obtenido de la herramienta anterior.

API <https://haveibeenpwned.com/api/v2/breachedaccount/carlos.iglesias@uah.es>



```
[{"Title":"LinkedIn","Name":"LinkedIn","Domain":"linkedin.com","BreachDate":"2012-05-05","AddedDate":"2016-05-21T21:35:40Z","PwnCount":164611595,"Description":"In May 2016, <a href='\"https://www.troyhunt.com/observations-and-thoughts-on-the-linkedin-data-breach\" target='\"_blank\" rel='\"noopener\">LinkedIn had 164 million email addresses and passwords exposed</a>. Originally hacked in 2012, the data remained out of sight until being offered for sale on a dark market site 4 years later. The passwords in the breach were stored as SHA1 hashes without salt, the vast majority of which were quickly cracked in the days following the release of the data.\"",DataClasses":{"Email addresses","Passwords"},"IsVerified":true,"IsSensitive":false,"IsActive":true,"IsRetired":false,"IsSpamList":false,"LogoType":"svg"}]
```



```
▼ array [1]
  ▼ 0 {14}
    Title : LinkedIn
    Name : LinkedIn
    Domain : linkedin.com
    BreachDate : 2012-05-05
    AddedDate : 2016-05-21T21:35:40Z
    PwnCount : 164611595
    Description : In May 2016, <a href='\"https://www.troyhunt.com/observations-and-thoughts-on-the-linkedin-data-breach\" target='\"_blank\" rel='\"noopener\">LinkedIn had 164 million email addresses and passwords exposed</a>. Originally hacked in 2012, the data remained out of sight until being offered for sale on a dark market site 4 years later. The passwords in the breach were stored as SHA1 hashes without salt, the vast majority of which were quickly cracked in the days following the release of the data.
    DataClasses [2]
      0 : Email addresses
      1 : Passwords
    IsVerified :  true
    IsSensitive :  false
    IsActive :  true
    IsRetired :  false
    IsSpamList :  false
    LogoType : svg
```

<https://haveibeenpwned.com/>

The screenshot shows the HaveIBeenPwned website interface. At the top, there is a navigation bar with links: Home, Notify me, Domain search, Who's been pwned, Pastes, API, About, and Donate. The main heading asks "';--have i been pwned?'". Below this, it says "Check if you have an account that has been compromised in a data breach". There is a search input field labeled "email address or username" and a button labeled "pwned?".

Statistics shown:

- 183 pwned websites
- 2,051,020,243 pwned accounts
- 43,869 pastes
- 40,402,537 paste accounts

Top 10 breaches:

Website	Accounts	Description
myspace	359,420,698	MySpace accounts
NetEase	234,842,089	NetEase accounts
LinkedIn	164,611,595	LinkedIn accounts
Adobe	152,445,165	Adobe accounts
Badoo	112,005,531	Badoo accounts
VK	93,338,602	VK accounts
Рамблер	91,436,280	Rambler accounts
Dropbox	68,648,009	Dropbox accounts

script_leak.py



```
#!/usr/bin/python
# -*- coding: utf-8 -*-
import argparse
import json
import requests
import os
import time

def copy_csv(email, list_leak, csv):

    csv.write(email+';'+str(list_leak)+'\n')

def find_csv(email, list_leak):

    if os.path.exists('leak_csv.csv') is False:
        csv = open('leak_csv.csv', 'a')
        csv.write(email+'\n')
        copy_csv(email, list_leak, csv)
        csv.close()

    else:
        csv = open('leak_csv.csv', 'a')
        copy_csv(email, list_leak, csv)
        csv.close()

def search_all_leaks(leak_user):

    leaks=[]
    url = "https://haveibeenpwned.com/api/v2/breachedaccount/"+leak_user
    response = requests.get(url)
    if response.status_code == 200:
        results = response.json()
        for result in results:
            leaks.append(result['Name'])

    else:
        print "Error code %s" % response.status_code

    return leaks

def search_leak(file):

    emails_leak=[]

    emails=open(file, 'r')
    for line in emails.read().split('\r\n'):
        emails_leak.append(line.encode('utf8'))

    for line_email in emails_leak:
        leak_dict={}
        leak_dict['user']=''
        leak_dict['leaks']=[]
        leak_dict['user']=line_email

        leak_dict['leaks']=search_all_leaks(leak_dict['user'])
        print leak_dict['user'] , leak_dict['leaks']
        find_csv(leak_dict['user'],leak_dict['leaks'])
        time.sleep(2)

if __name__ == '__main__':
    parser = argparse.ArgumentParser()
    parser.add_argument("-f", "--file", help="insert file")
    args = parser.parse_args()
    if not args.file:
        print "Insert a File with emails"
    else:
        search_leak(args.file)
```

DNSTwist es una librería open-source creada para el análisis de amenazas de Typosquatting y suplantación de dominios.



Comando ejecución

```
python dnstwist.py -b -w -g -m google.es
```

```
dnstwist (1.04b)
Disabling multithreaded job distribution in order to query WHOIS servers
Processing 198 domain variants .....20%.....41%.....62%.....
..... 85 hits (42%)

Original*  google.es  216.58.201.131/United States 2a00:1450:4003:804::2003 NS:ns1.google.com MX:alt1.aspmx.l.google.com HTTP:"gws" SMTP:"mx.google.com ESMTP l131si26608711fb.6 - gsmtpp
Addition  googlea.es  176.28.103.205/Spain NS:ns1.srv-hostalia.com MX:mx.googlea.es HTTP:"Apache" SMTP:"relayin06.dominioabsoluto.net ESMTP"
Addition  googleb.es  -
Addition  googlec.es  -
Addition  googled.es  185.53.177.20/Germany NS:ns1.eurodns.com MX:mail.googled.es HTTP:"nginx" SMTP:"corellia.eurodns.com EuroDNS Mail Server"
Addition  googlee.es  185.53.178.9/Germany NS:ns1.parkingcrew.net MX:mail.h-email.net HTTP:"nginx"
Addition  googlef.es  -
Addition  googleg.es  217.76.128.34/Spain NS:dns23.servidoresdns.net HTTP:"Microsoft-IIS/7.5"
Addition  googleh.es  -
Addition  googlei.es  212.227.247.183/Germany NS:ns63.land1.es MX:mx00.land1.es HTTP:"Apache" SMTP:"554-kundenserver.de (mxeue003) Namesis E"
Addition  googlej.es  -
Addition  googlek.es  -
Addition  googl.el.es  -
Addition  googlem.es  -
Addition  googlen.es  -
Addition  googleo.es  -
Addition  googlep.es  -
Addition  googleq.es  -
Addition  googler.es  82.194.64.60/Spain NS:dns1.canaldominios.com MX:mx01.canaldominios.com HTTP:"Apache" SMTP:"mx01.canaldominios.com ESMTP"
Addition  googles.es  185.53.178.7/Germany NS:ns1.parkingcrew.net MX:mail.h-email.net HTTP:"nginx"
Addition  googlet.es  -
Addition  googleu.es  -
Addition  googlev.es  -
Addition  googlew.es  188.165.119.163/Spain NS:ns1.cdmon.net MX:yahoo.es HTTP:"Apache"
Addition  googlex.es  -
Addition  googley.es  37.152.88.54/Spain NS:ns2.dondominio.com HTTP:"Apache"
Addition  googlez.es  -
Bitsquatting  google.es  185.53.178.7/Germany NS:ns1.parkingcrew.net MX:mail.h-email.net HTTP:"nginx"
Bitsquatting  eoogle.es  -
Bitsquatting  ooogle.es  37.152.88.54/Spain NS:ns1.dondominio.com HTTP:"Apache"
Bitsquatting  ooogle.es  50.63.202.10/United States NS:ns61.domaincontrol.com MX:mailstore1.secureserver.net HTTP:"HTTP 302" SMTP:"554 p3plibsmtp02-12.prod.phx3.secureserv"
Bitsquatting  woogles.es  37.152.88.54/Spain NS:ns2.dondominio.com HTTP:"Apache"
Bitsquatting  gnogle.es  -
Bitsquatting  gkogle.es  -
Bitsquatting  gkogle.es  -
Bitsquatting  gkogle.es  0.0.0.0 NS:ns1.ggoogle.es
Bitsquatting  gongle.es  -
Bitsquatting  gomgle.es  -
Bitsquatting  gokgle.es  -
Bitsquatting  goggle.es  -
Bitsquatting  goofle.es  213.186.33.5/France NS:dns10.ovh.net MX:redirect.ovh.net HTTP:"nginx"
Bitsquatting  goonee.es  -
Bitsquatting  goocle.es  37.152.88.54/Spain NS:ns1.dondominio.com HTTP:"Apache"
Bitsquatting  goonle.es  50.63.202.14/United States NS:ns61.domaincontrol.com MX:mailstore1.secureserver.net HTTP:"HTTP 302" SMTP:"554 p3plibsmtp01-09.prod.phx3.secureserv"
Bitsquatting  goowle.es  -
Bitsquatting  goowme.es  -
Bitsquatting  googme.es  -
Bitsquatting  googhe.es  -
Bitsquatting  qoogde.es  -
Bitsquatting  qoogld.es  -
```



Arquitectura de una plataforma de Inteligencia

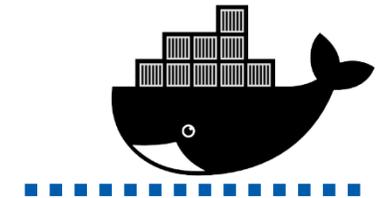
- ▶ ¿Quiénes somos?
- ▶ Definiciones
- ▶ Búsqueda de información en OSINT
- ▶ Automatización de scripts de Inteligencia
- ▶ Arquitectura de una plataforma de Inteligencia

Instalación de Docker

```
echo "deb https://apt.dockerproject.org/repo ubuntu-xenial main"  
>> /etc/apt/sources.list; apt-get update;  
sudo apt install docker-engine
```

Comandos Básicos

- `docker images` Ver imágenes instaladas.
- `docker ps` Ver contenedores arrancados.
- `docker build -t nombre_imagen:versión` Crear una imagen de docker.
- `docker run -i -t nombre_imagen:versión --name nombre_contenedor /bin/bash` Arrancar un contenedor asociado a una imagen de docker.
- `docker rmi -f image_id / docker rm -f container id` Eliminar una imagen / contenedor de docker.
- `ctrl q + p` Salir de un contenedor de docker sin perder cambios.
- `docker attach nombre_contenedor` Acceder a contenedor.
- `docker commit nombre_contenedor nombre_imagen:versión` Guardar cambios en el contenedor y creación de una nueva imagen.
- `docker save nombre_imagen:version > imagen.tar` Exportar imagen en un tar.
- `docker load < imagen.tar` Importa la imagen desde un tar.
- `docker cp fichero_a_copiar hash_contenedor:/root/fichero_a_copiar` Copia un archivo desde máquina anfitriona a contenedor docker.



```
FROM ubuntu:latest

# Instalamos las actualizaciones
RUN apt-get update
RUN apt-get upgrade -y

# Instalamos cliente y servidor de MySQL
RUN DEBIAN_FRONTEND=noninteractive apt-get -y install mysql-client mysql-server curl
RUN apt-get install -y inetutils-tools inetutils-ping nano wget
RUN apt-get install -y net-tools
# Se habilita el acceso remoto desde MySQL
RUN sed -i -e"s/^bind-address\s+=\s+127.0.0.1/bind-address = 0.0.0.0/" /etc/mysql/my.cnf

# Se crea el volumen
VOLUME ["/opt/mysql"]

# Añadimos el archivo con la base de datos y las tablas.
ADD ./bbdd.sql /var/db/bbdd.sql

# Añadimos la configuración estándar
ENV user admin
ENV password admin
ENV url file:/var/db/bbdd.sql
ENV right WRITE

# Ejecutamos el script
ADD ./start.sh /usr/local/bin/start.sh
RUN chmod +x /usr/local/bin/start.sh

# Habilitamos el puerto 3306 de MySQL
EXPOSE 3306

# Iniciamos el motor de MySQL
CMD ["/usr/local/bin/start.sh"]
```

Archivo Dockerfile

```
#!/bin/bash

# This script starts the database server.
echo "Creando el usuario $user para la base de datos desde la url $url"

# Import database if provided via 'docker run --env url="http://ex.org/db.sql"'
echo "Añadiendo datos a MySQL"
service mysql start
# /usr/sbin/mysqld
sleep 5
curl $url -o import.sql

mysql --default-character-set=utf8 < import.sql
rm import.sql
mysqladmin shutdown
echo "finished"

# Now the provided user credentials are added
# /usr/sbin/mysqld &
service mysql start
sleep 5
echo "Creando usuario"
echo "CREATE USER '$user' IDENTIFIED BY '$password' | mysql --default-character-set=utf8"
echo "REVOKE ALL PRIVILEGES ON *.* FROM '$user'@'%'; FLUSH PRIVILEGES" | mysql --default-character-set=utf8
echo "GRANT SELECT ON *.* TO '$user'@'%'; FLUSH PRIVILEGES" | mysql --default-character-set=utf8
echo "Finalizado"

if [ "$right" = "WRITE" ]; then
echo "añadiendo permisos al usuario"
echo "GRANT ALL PRIVILEGES ON *.* TO '$user'@'%'; WITH GRANT OPTION; FLUSH PRIVILEGES" | mysql --default-character-set=utf8
fi

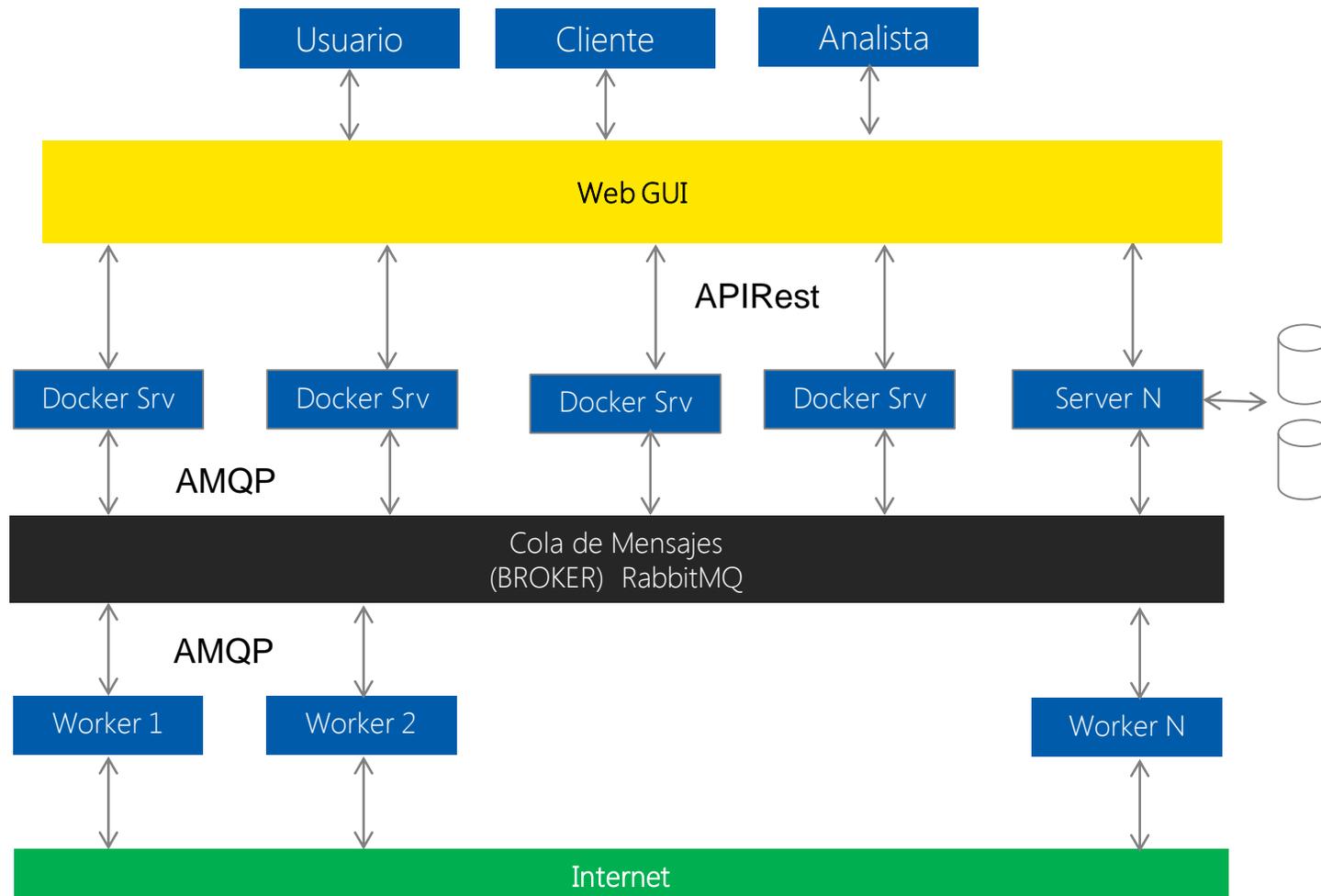
# And we restart the server to go operational
mysqladmin shutdown
echo "Iniciando MySQL Server"
service mysql start
# /usr/sbin/mysqld
```

Archivo start.sh

Creación de una imagen con MySQL `docker build -t taller_mysql:latest .`

Creación del contenedor `docker run -it -p 3306:3306 taller_mysql bash`

Script a ejecutar dentro del contenedor `sh /usr/local/bin/start.sh`



<https://education.github.com/pack>

 Education

  educate

 DigitalOcean

 UDACITY

Microsoft Imagine 

 axosoft GitKraken

 CrowdFlower

 FLATIRON
SCHOOL

 namecheap

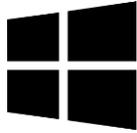
 SendGrid

 DATADOG

 bitnami

 ATOM

dnsimple



<https://imagine.microsoft.com>

The screenshot shows the Microsoft Imagine website. At the top, there is a navigation bar with the Microsoft logo and links for Technologies, Documentation, and Resources. Below this is a secondary navigation bar with the Microsoft Imagine logo and links for Account, Downloads, Code, Compete, About, and Blog. A cookie consent banner is visible below the navigation. The main content area features a large heading "Software and services catalog" followed by a sub-heading "Microsoft Imagine has the tools you need to build a game, design an app or launch a project." Below this is a call to action "Sign in or create an account to download these software & development tools at no cost." and a "Sign in /create profile >" button. The "Featured" section contains three items: "Services" (Microsoft Azure for Students), "Tools" (Parallels Desktop for Mac Pro Edition), and "Tools" (Visual Studio Community 2015). Each item includes a brief description and a link to get or download the tool.

Microsoft Technologies Documentation Resources Sign in

Microsoft Imagine Account Downloads Code Compete About Blog

By using this site you agree to the use of cookies for analytics, personalized content and ads. [Learn More.](#)

Software and services catalog

Microsoft Imagine has the tools you need to build a game, design an app or launch a project.

Sign in or create an account to download these software & development tools at no cost.

[Sign in /create profile >](#)

Featured

Services	Tools	Tools
Microsoft Azure for Students Free cloud services Develop in the cloud at no cost with Azure App Services, Notification Hubs, SQL database, and more. Get Azure	Parallels Desktop for Mac Pro Edition Free 3-month trial Powerful virtualization solution for running Windows and Mac apps side-by-side, without rebooting. Download Parallels	Visual Studio Community 2015 Free, fully-featured, extensible IDE Create modern apps for Android, iOS, and Windows as well as web apps and cloud services. Download Visual Studio Community



<https://shodan.io>

- ▶ Crear cuenta de Shodan.
- ▶ Solicitar upgrade por mail a jmath@shodan.io

The screenshot shows the Shodan Account Overview page. The top navigation bar includes 'Shodan', 'Scanhub', and 'Developers'. The user is signed in as 'wiktory.nykiel'. The left sidebar has 'ACCOUNT' with options for 'Overview', 'Settings', and 'Change Password'. The main content area is titled 'Account Overview' and displays the following information:

API Key	SfWfHnMM0QsFVjkdNyeaAGj2VUucKpuB
Display Name	wiktory.nykiel
Email	wiktory.nykiel@edu.uah.es
Export credits	100
Query credits	200000
Scan credits	65536
Member since	2014-08-25 09:09:41.084000

Profile card for John Matherly, founder of Shodan. The card features a background image of a world map with a heatmap overlay and a portrait of John Matherly. The text on the card reads:

John Matherly
@achilleian
Founder of Shodan, Internet Cartographer
Austin, Texas
shodan.io
Se unió en octubre de 2007



<https://github.com/ey-ciberseg-wiktor-nykiel-ivan-portillo/scripts-taller-creacion-tools-de-inteligencia>

WiktorNykiel committed on GitHub Update README.md		Latest commit 4b4de77 2 days ago
📁 docker_mysql	Add files via upload	2 days ago
📄 .gitattributes	Added .gitattributes	2 days ago
📄 README.md	Update README.md	2 days ago
📄 script_combinatoria_tlds.py	Add files via upload	2 days ago
📄 script_geoiip.py	Add files via upload	2 days ago
📄 script_leak.py	Add files via upload	2 days ago
📄 script_mx_ns.py	Add files via upload	2 days ago
📄 script_nmap.py	Add files via upload	2 days ago
📄 script_ripe_api.py	Add files via upload	2 days ago
📄 script_shodan.py	Add files via upload	2 days ago
📄 script_tlds_mx_ns.py	Add files via upload	2 days ago
📄 script_whois.py	Add files via upload	2 days ago
📄 tld.txt	Add files via upload	2 days ago

Talent Management

Manda tu CV a: eytalent@es.ey.com

<http://www.ey.com/es/es/careers/students>



Universidad
de Alcalá

JORNADAS DE
SEGURIDAD Y
CIBERDEFENSA



Wiktor Nykiel



wiktor.nykiel@es.ey.com



[wiktornykiel](https://www.linkedin.com/in/wiktornykiel)



[@wiktornykiel](https://twitter.com/wiktornykiel)

Iván Portillo



ivan.portillomorales@es.ey.com



[ivanportillomorales](https://www.linkedin.com/in/ivanportillomorales)



[@ivanPorMor](https://twitter.com/@ivanPorMor)



Universidad
de Alcalá

JORNADAS DE
SEGURIDAD Y
CIBERDEFENSA



Building a better
working world