

Bienvenidos a la clase 65 de HDC :),

¿Están listos? Proactividad.

"¡Eso! ;Procrastividad!"

Eh... Bueno, está bien. Hoy veremos FTP. Hasta ahora lo que vimos fue...

"Robar cosas con malware."

... ¿De verdad? ¿Éso has aprendido? El tema es más extenso que "robar". **FTP** o **File Transfer Protocol**, es decir una manera de **transferir archivos**.

"¿De la misma manera que lo hago por Google Drive?"

Buena pregunta. Google Drive es otro tipo de servicio que nada tiene que ver con el FTP. Veamos ciertas cuestiones.

La idea principal es el **servidor**. Un servidor FTP instalado en algun equipo que **alojará archivos** y los **clientes** pueden conectarse a él y descargarse archivos o subirlos (todo depende de los **permisos** que se le brinden al usuario). Obviamente trabaja mediante **TCP** porque no es necesario que sea rápido pero sí efectivo.

Para hacer una clase interesante, vamos a instalar en nuestra máquina virtual con W7, un server FTP.

"Quiero hackear."

Manolo, primero caminar y luego correr. Sé que el camino es duro y largo pero te aseguro que es satisfactorio :).

Vamos a descargar **FileZilla** desde la página oficial (es **gratuito**, quédense tranquilos): <u>https://filezilla-project.org/</u>



El **cliente** lo descargan para su máquina **host**, y el **Filezilla Server** lo descargan en la máquina **virtual**. Aunque si quieren, pueden instalarlo al **revés** o los dos en el mismo equipo -que es lo que haré yo-.

Bueno, ya descargado luego de unos segundos, le damos a instalar al **exe** del server.



Le vamos a dar "*next*" hasta que aparezca esta ventana a la que vamos a dejar que Filezilla se instale como un **servicio** pero que **no empiece automáticamente** sino que nosotros le vamos a dar la orden. El puerto puede ser cualquiera pero yo prefiero elegir algo que recuerde (aunque es indistinto, no eligan los puertos que son estándar para otros servicios) y voy a querer que inicie el server una vez terminada la instalación, sobre todo para saber si anda correctamente.

FileZilla Server beta 0.9.50 Setup	
Startup settings Select startup behaviour for FileZilla Server	Z
Please choose how FileZilla Server should be started:	
	•
Please choose the port, the admin interface of FileZilla Server should listen	on (1-65535):
Start Server after setup completes	
< Back Next	Cancel

Luego de darle "*next*" nuevamente, aparecerá esta otra ventana.

🔁 FileZilla Server beta 0.9.50 Setup	
Startup settings Select startup behaviour for FileZilla Server	Z
Please choose how the server interface should be started:	
Start manually	•
Start Interface after setup completes	
Nullsoft Install System v2.46-Unicode Sack Install	Cancel

Una vez terminada la instalación del server, aparecerá esta ventanita la cual aparece porque iniciamos el Filezilla.

En donde dice "**Server Address**", apuntaremos a **nuestra IP** (recuerden que el servicio del server lo instalamos en nuestro equipo pero Filezilla puede actuar solo como interfaz gráfica de un

servidor FTP externo) que como yo estoy conectado a un **DHCP** (¿ésto lo explique anteriormente?), mi IP **no es fija** en la red interna. Lo mejor que puedo hacer es apuntar a la IP definida como un **sinónimo** de **LocalHost**. En la password dejen el cuadro en blanco.

Connect to Server	×
Server Address:	Port:
127.0.0.1	15000
Administration password:	
Always connect to this serve	
OK Ca	ancel

Entonces aparecerá la interfaz y si todo sale bien, como última linea el "*Logged on*" de su correcto logueo.



Ahora a la otra parte. Pero tengan **cuidado** al instalar el **cliente**, que intenta de instalarte unos adware desde el descargador de **SourceForge** (tuve que darle al Decline un par de veces). Va sin imágenes porque no hay opciones que tocar. Cuando termina nos aparece la interfaz del cliente.

FileZilla File Edit View Tran	sfer Server Bookmarks	i Help E 🕂 🚰 🎊						×
Host:	Username:	Password:		Port:	Quickconr	nect 💌		
Local site: \			Remote s	ite:				4
Filename	Filesize Filetype	Last modified	Filename	Filesize	Filetype	Last modified	Permissior	ns Ov
설 전 모				N	ot connected	to any server		
					III			۴
3 directories			Not conne	ected.				
Server/Local file Queued files Failed 1	Direction Rem	nsfers	Size	Priority Status				
						📾 Queue: emp	ty	•

En donde está el recuadro rojo vamos a poner los datos que sabemos. El host es la dirección IP de donde esté el server. En mi caso, es mi **localhost** por lo que la **IP** es **127.0.0.1**. El **puerto** es el **15000** y el usuario y password estarán en blanco. Una vez le dimos a Quickconnect que aparece al lado, se conectará. Pero vemos que pasa algo más luego de 20 segundos.

<u>H</u> ost:	127.0.0.1	<u>U</u> sername:	anonymous	Pass <u>w</u> ord:	Port:	15000	Quickconnect 💌
Status:	Connecting to	127.0.0.1:1500	0				
Status:	Connection est	tablished, wait	ing for welcome me	essage			
Respon	ise: FZS						
Respon	ise: J						
Respon	ise: P						
Respon	ise: J						
Respon	ise: +						
Respon	ise:						
Error:	Connection tin	ned out after 2	0 seconds of inactiv	ity			
Error:	Could not con	nect to server					

Las primeras dos líneas son el estado de conexión establecido. Pero luego aparecen dos errores. Dice que **si por 20 segundos no tuviste actividad, la conexión caerá**.

"¿Veinte segundos? Eso es muy poco, con razón los hackers hacen todo rápido."

Veinte segundos es **mucho tiempo** porque cada vez que hagas algo esos 20 segundos se renuevan. Y no tiene nada que ver con lo que estás pensando. Hagamos una práctica para subir y descargar archivos. Para ésto, primero vamos a **añadir un usuario** en el servidor FTP. Vamos a *Edit* \rightarrow *Users*. Y allí dentro, en *Add*, podemos crearlo.

	FileZilla Server (127.0.0.1) File Server Edit ? (000002)5/7/20 (000003)5/7/20 (000003)5/7/20 Groups	
Users Page: General Shared folders Speed Limits IP Filter	Account settings Enable account Password: Group membership: Bypass userlimit of server Maximum connection count: Connection limit per IP:	Users
	Force SSL for user login Description	Rename Copy
OK Cancel	You can enter some comments about the user	

Aquí tenemos una ventana donde podemos **agregar el nombre del usuario** -en mi caso *Roadd*- y luego **tildamos** la opción de **password** con una que queramos nosotros. Vean que también aparece la opcion de agregar ese usuario a un **grupo** determinado, para que podamos administrar permisos de a muchos pero **en este caso no es necesario**. Lo que aparece sobre *Maximum connection count* es la **cantidad máxima de conexiones** con este mismo usuario; y *Connection limit per IP* es la cantidad de conexiones limite por IP (redundante, si lo leen es obvio). Cuando los valores están en cero corresponden a un "sin límite".

Users		—
Page: General Shared folders Speed Limits IP Filter	Account settings Image: Content of the second of the se	Roadd Add Remove Rename Copy
OK Cancel	Description You can enter some comments about the user	

No podemos irnos sin agregarle un directorio por el cual va a operar el usuario, en la izquierda tenemos la opcion de *Shared folders*. Allí, en la parte izquierda, podemos ver que manejamos los directorios. Añadan uno que quieran, yo creé una carpeta en el escritorio y agregaré ésa.

Users				X
Page: General Shared folders Speed Limits IP Filter	Shared folders Directories Add Remo A directory alias will also ap path. Separate multiple alia If using aliases, please avo	Aliases	Files Read Write Delete Append Directories Create Delete Uist + Subdirs Set as home dir Ilocation. Aliases must with the pipe character uctures, it will only control	Users Roadd Add Remove Rename Copy t contain the full virtual arr (1) fuse FTP clients.
OK Cancel				

Cuando la agregamos, vemos que los botones que estaban en gris, se habilitan. Voy a habilitar las primeras tres opciones del *checkbox* "Files". Ya sabrán a qué se refieren :). Y luego ya aceptamos todo.

Users				—
Page: General Shared folders Speed Limits IP Filter	Shared folders Directories H C:\Users\w7\Des Add Ren A directory alias will also path. Separate multiple a If using aliases, please a	Aliases s hove Rename appear at the specifie sliases for one directory st	Files Read Vite Vite Delete Append Directories Create Delete Vitst V + Subdirs Set as home dir d location. Aliases muy with the pipe charace ructures, it will only con	Image: Users Roadd Add Remove Rename Copy st contain the full virtual ter (1) Infuse FTP clients.
OK Cancel				

Abramos una **línea de comandos**. Sé que les hice instalar un cliente del Filezilla y quisiera que lo **investiguen** por su cuenta, y mientras tanto yo les enseño a usar las herramientas que son más **flexibles** aunque un poco feas.

En fin, para abrir la conexión tenemos el comando "**ftp** <**ip**>". Recuerden que mi IP es mi propio localhost, por lo que será "**ftp 127.0.0.1**"

🔁 FileZilla Se	rver (127.0.0.1)			
File Server	Edit ?			
🦻 🔒 🛯) 🖸 🔮 🎖 🔤 /C/ C:\ 🚦			
Done sending a	ccount settings.			~
(000005)5/7/20 (000005)5/7/20 (000005)5/7/20	15 12:06:44 PM - (not logged in 15 12:06:44 PM - (not logged in 15 12:06:44 PM - (not logged in 15 12:06:44 PM - (not logged in) (127.0.0.1)> Conne) (127.0.0.1)> 220-Fil) (127.0.0.1)> 220-Wi	cted on port 21, sending welcome message eZilla Server version 0.9.50 beta itten by Tim Kosse @filezilla-project.org)	
	Assessment	, (127.0.0.1)7 22011.		
	Account	14	Transfer	
+ 000005	(not logged in)	127.0.0.1		
∢ Ready	C:\Users\w7>ftp Connected to 12 220-FileZilla S 220-written by	nmand Prompt - ft 127.0.0.1 7.0.0.1. erver versior Tim Kosse (t:	p 127.0.0.1 0.9.50 beta im.kosse@filezilla-project.org	>
	User (127.0.0.1	:(none)):	rezilla project.org/	

"¡Muchos mensajes! Ésto ya es muy complicado. Me voy."

¡Manolo! ¡Volvé! Vas a ver que todo es **informativo** y que no solamente es **necesario**, sino que es muy **interesante**. Lo primero que vemos de interesante es dentro del Filezilla Server, donde aparecen **4 mensajes**. Los mensajes tienen **separaciones**, veamos de qué se tratan.

- **(000005)**: Esto corresponde al ID de la conexión, porque en un server aparecen varios usuarios, varias conexiones y así tenemos una referencia de quién hablamos. Fíjense que también tenemos información de esa ID en la separación de abajo.
- **5/7/2015**: Fecha en formato mm/dd/aaaa.
- **12:06:44 PM**: Hora en formato hh:mm:ss.
- (not logged in): Indica que no está logueado con ningún usuario desde esa IP.
- (127.0.0.1): Cuál es la IP de donde se está haciendo la conexión.
- >: Luego de este símbolo aparece el mensaje o los estados del server.

La primer línea que aparece en negro es información sobre el puerto donde se conecta; lo que aparece en **verde** es el mensaje que se **envía y recibimos** desde el **cliente**. Allí, en nuestro **cmd**, tenemos la **IP** a donde nos conectamos (la mía es la del localhost), y el mensaje que me envío el

server. Para terminar me indica que me pide un **usuario** para poder conectarme y luego una **contraseña**.

(000006)5///2015 13:21:19 PM - (not logged in) (12/.0.0.1)> 220 Please visit https://filezilla-project.org/	
(000006)5/7/2015 13:21:24 PM - (not logged in) (127.0.0.1)> USER Roadd	
(000006)5/7/2015 13:21:24 PM - (not logged in) (127.0.0.1)> 331 Password required for roadd	
(000006)5/7/2015 13:21:31 PM - (not logged in) (127.0.0.1)> PASS	
(000006)5/7/2015 13:21:31 PM - roadd (127.0.0.1)> 230 Logged on	

		Administrator: Command Prompt - ftp 127.0.0.1
ID 🛆	Account	C:\Users\w7>ftp 127.0.0.1 Connected to 127.0.0.1.
- ©- 000006	roadd	220-FileZilla Server version 0.9.50 beta 220-written by Tim Kosse (tim.kosse@filezilla-project.org) 220 Please visit https://filezilla-project.org/ User (127.0.0.1:(none)): Roadd 331 Password required for roadd Password: 230 Logged on
Ready		

Allí tenemos la manera que me conecté. **Cuando ponen su password,** mientras escriben no van a ver asteriscos ni nada, simplemente **nada se mueve**. Ésto aparece para que nadie sepa **cuántas** letras -o mejor dicho caracteres- escribiste. Una vez terminado nos da el **prompt** para que demos comandos.

Hay un par de cosas a las cuales prestar atención:

- **El ID de la conexión cambió**: esto sucedió porque se me cerro la conexión anterior por lo que tuve que abrir otra (acostúmbrense porque va a pasarme seguido).
- Hay unos numeritos antes de los mensajes del server: esto corresponde a los ID de los mensajes.

Con *help* o ? listamos los comandos que tenemos.

ftp> help Commands may be	abbreviated.	Commands are:		
? ascii bell binary bye cd close ftp> _	delete debug dir disconnect get glob hash help lcd	literal ls mdelete mdir mget mkdir mls mput open	prompt put pwd quit quote recv remotehelp rename rmdir	send status trace type user verbose

Casi 45 comandos. No voy a enseñarles todos, pero sí algunos. Veamos una abreviación de lo que podemos entender rápidamente:

- 1. cd: Para ir navegando por los directorios. Tal cual el comando de Windows.
- 2. Close o disconnect: Desconectarse del server al cual está conectado sin salir del servicio.
- **3.** Bye o quit: Cierra la conexión y el servicio.

- **4. Rename <archivo> <nombre nuevo>:** renombra un archivo. Cuando se renombra es importante no olvidar la extensión del archivo.
- **5. Is:** lista los archivos y directorios que hay en el directorio.
- 6. Dir: lista los archivos y directorios con sus atributos.
- 7. **Open <IP/DNS>:** Intenta abrir una conexión con un servidor.
- **8. Pwd:** Pedirle al servidor que nos diga en qué directorio estamos. Cuando el directorio actual es "/" quiere decir que nos encontramos en la raíz, el primer directorio madre de todos los directorios (perdón la expresión, pero así se entiende xD).
- 9. Status: El estado de la conexión y sus parámetros.
- **10. Verbose:** Cambia a On o a Off. La idea del verbose es que nos diga si queremos tener más información del server (On) o que las cosas pasen por segundo plano y no nos interese(Off).
- **11. Delete <archivo>:** Borra un archivo.
- **12. Rmdir <directorio>:** Borra un directorio.
- **13. Get** <archivo> [nombre nuevo]: Transfiere un archivo desde el server ftp hacia donde estamos conectados nosotros. El segundo parámetro es opcional y es para ponerle un nuevo nombre cuando se termine de transferir.
- **14. Mget <archivo> [archivo] ... [archivo]:** Permite la transferencia de varios archivos simultáneamente
- **15. Put <archivo>, Mput <archivo> [archivo] ... [archivo]:** Igual que los anteriores pero desde la máquina host hacia el servidor FTP.
- **16. !**<**comando**>**:** El signo de admiración más el comando es para ejecutar ese mismo comando en la máquina local.
- **17. Binary y ascii:** Son modos de transferencia. El primero sirve para ejecutables, imágenes, mp3 -en fin, archivos binarios-. El segundo es para archivos de texto, html, y ese tipo de cosas.
- **18. Mkdir <directorio>:** Crea un directorio.
- 19. User: Cambia el usuario que se conecta con el FTP.

Sé que son comandos de oficina y administración, pasa que son necesarios. Vamos a ver la práctica:

Name	Date modified	Туре
archivoFTP.txt	5/8/2015 11:37 AM	Text Docume
Administrator: Command Prompt - ftp 12	7.0.0.1 - ftp 127.0.0.1	
C:\Users\w7>ftp 127.0.0.1 220-FileZilla Server version 0. 220-written by Tim Kosse (tim.k 220 Please visit https://filezi User (127.0.0.1:(none)): roadd 331 Password required for roadd Password: 230 Logged on ftp> 1s 200 Port command successful 150 Opening data channel for di archivoFTP.txt 226 Successfully transferred "/ ftp: 16 bytes received in 0.00S ftp>	9.50 beta cosse@filezilla-pr illa-project.org/ l rectory listing o c econds 16000.00Kb	oject.org) f "/" ytes/sec.

Lo que aparece arriba es la carpeta "**madre**" -de ahora en más "**raíz**"-. Hicimos un *ls* que correspondía al **listado** de directorios y archivos. Nos dice también dónde listó ("/"). Voy a intentar crear un nuevo directorio.



Pero **no tenemos permisos**. ¡Maldita sea la autoridad! ¡Anarquía! ¡Anarquía! Es broma, es broma. No se me duerman :P. Para que recuerden les vuelvo a mostrar la imagen donde no le pusimos **permisos de escritura**.

Users	
Page: General Shared folders Speed Limits IP Filter	Shared folders Files Directories Aliases H C:\Users\w7\Des Directories Append Directories Create Delete Users Add Remove Rename Set as home dir Add Remove Add Rename Set as home dir Copy
OK Cancel	

Voy a intentar **recuperar** el archivo de dentro del server.

Y sí, todo salió bien :). Veamos si aparece en nuestras carpetas. Recuerden que era *!* para ejecutar un comando dentro de **nuestra máquina local**.

ftp> !dir Volume in drive C has no label. Volume Serial Number is 3403-DB3F				
Directory	of C:\Users	\w7		
05/08/2015	12:57 PM	<dir></dir>		
05/08/2015	12:57 PM	<dir></dir>		
10/03/2014	02:33 PM	<dir></dir>		
05/08/2015	12:57 PM	(0 archivoFTP.txt	
01/28/2015	02:22 PM	<pre><dir> </dir></pre>	Contacts	
08/20/2014	12:20 PM	<dir></dir>	Documents	

Aparece perfectamente. Y ahora voy a **borrar** el archivo dentro del servidor.



Perfecto. Todo bien. Veamos el tema de pasar un archivo hasta allí. Esto lo vimos en la segunda clase de malware pero nunca está de más volver a ver los temas. El comando *put* nos deja hacer la tarea.

ftp> put archivoFTP.txt
200 Port command successful
150 Opening data channel for file upload to server of "/archivoFTP.txt"
226 Successfully transferred "/archivoFTP.txt"
ftp> dir
200 Port command successful
150 Opening data channel for directory listing of "/"
$\beta = \beta = \beta = 1$ ftn ftn $\beta = 0$ Mau 11 10:24 archiuoFTP tyt
226 Successfully twopsfeward "/"
220 Successfully transferred /
rtp. 05 hytes received in 0.005econds 05000.00khytes/sec.
rtp/_

En rojo están marcados las líneas que corresponden a comandos.

Allí vemos los mensajes 226 correspondientes a las transferencias exitosas. Todo muy lindo.

"¿Y ésto es todo por el puerto 21? Es un super-puerto."

En realidad no todo pasa por el mismo puerto. **El puerto 21 se encarga del control** pero (siempre con algún pero, eh) la realidad es otra. Existen **dos formas de conexión: modo activo y modo pasivo**. Imagina que sino no podría haber más de un cliente conectado.

En el **modo activo** la historia es ésta: el puerto 21 del servidor está abierto esperando un mensaje. El cliente, con otro **puerto de control y otro de datos** de algún número mayor a 1024 -aleatorios- le dice al servidor FTP desde su puerto de control (digamos 1030) cuál es ese puerto de datos.



La cursilería siempre hace todo más fácil de entender :P. Volvamos. Luego de ésto el server le responde con un ACK (recuerden el **three way handshake** de la conexión **TCP** que vimos casi al final de la **clase 17**) y envíando cuál es su puerto de datos.



Si está todo bien, el cliente vuelve a responder al server con un ACK.



Aquí tenemos un grave **fallo de seguridad** porque el cliente le manda el puerto y el servidor se conecta a él. No vaya a ser cosa que alguien quiera anteponerse y hacerse pasar por el server. Por este problema -el **firewall** bloquea las conexiones de este estilo- se usa el **modo pasivo**.

El modo pasivo cambia porque primero le dice al servidor que quiere abrir una conexión mediante el comando **PASV** y el **servidor** le responde con el **canal de datos que usa**, que a diferencia del modo anterior debe ser mayor a 1024. Luego el cliente se conecta a él.



Aquí resolvemos este problema de seguridad, pero seguimos teniendo **otro**: todos los datos pasan en **texto plano** de un lugar a otro. **¿Qué pasa con el texto plano?** Que cualquiera que pueda interceptar los datos, al **no estar cifrados, se pueden leer sin inconvenientes**. Ya veremos como y de qué manera se puede arreglar. Por ahora tenemos que conformamos con hacerle **fuerza bruta**. Para ésto vamos a usar una herramienta que es muy buena en flexibilidad y en velocidad: **Hydra**. Les dejo el comprimido en los archivos complementarios, pero pueden googlearlo (ojo que también esta para Linux, así que los que usan Windows fíjense).

En fin, para usarlo tenemos que hacerlo desde la consola. Vayan hasta la carpeta donde está el **hydra.exe** y ejecútenlo.

C:\Users\w7\Desktop\hydra-7.5\hydra>hydra Hydra v7.5 (c)2013 by van Hauser/THC & David Maciejak – for legal purposes only
Syntax: hydra [[[-1 LOGIN!-L FILE] [-p PASS!-P FILE]] ¦ [-C FILE]] [-e nsr] [-o FILE] [-t TASKS] [-M FILE [-T TASKS]] [-w TIME] [-W TIME] [-f] [-s PORT] [-x MI :MAX:CHARSET] [-SuvV46] [service://server[:PORT][/OPT]]
Options: -1 LOGIN or -L FILE login with LOGIN name, or load several logins from FILE -p PASS or -P FILE try password PASS, or load several passwords from FILE -C FILE colon separated "login:pass" format, instead of -L/-P options -M FILE list of servers to be attacked in parallel, one entry per line -t TASKS run TASKS number of connects in parallel (per host, default: 16) -U service module usage details -h more command line options (complete help) server the target server (use either this OR the -M option) service the service to crack (see below for supported protocols) OPT some service modules support additional input (-U for module help)
Supported services: asterisk cisco cisco-enable cvs ftp ftps http[s]-{head!get} http[s]-{get!post}-form http-proxy http-proxy-urlenum icq imap[s] irc ldap2[s] dap3[-{cram!digest}md5][s] mssql mysql nntp oracle-listener oracle-sid pcanywhe e pcnfs pop3[s] rdp rexec rlogin rsh sip smb smtp[s] smtp-enum snmp socks5 team peak telnet[s] vmauthd vnc xmpp
Hydra is a tool to guess/crack valid login/password pairs — usage only allowed for legal purposes. This tool is licensed under AGPL v3.0. The newest version is always available at http://www.thc.org/thc-hydra
Example: hydra -l user -P passlist.txt ftp://192.168.0.1

No sólo tenemos un buen modo de uso sino que el ejemplo es lo que queremos hacer. Voy a hacer un txt con contraseñas y **una será la correcta.** Ya se habrán dado cuenta que allí tenemos "**ftp:**//" que es algo muy parecido a "**http:**//" de cuando abrimos una página web. Bueno, ese antecesor a la dirección **especifica el protocolo**, que en este caso es ftp.

passList.txt - Notepad				
File	Edit	Format	View	Help
1234 admi admi soyu asci acie hack	156 in inist imacm ofhna ociaw cingd	rador sword if p1212 esdecer	~ 0	

C:\Users\w7\Desktop\hydra-7.5\hydra>hydra -1 roadd -P passList.txt ftp://127.0	0
.ı Hydra v7.5 (c)2013 by van Hauser/THC & David Maciejak — for legal purposes only	ļ
Hydra (http://www.thc.org/thc-hydra) starting at 2015-05-11 13:15:49 [DATA] 8 tasks, 1 server, 8 login tries (1:1/p:8), ~1 try per task [DATA] attacking service ftp on port 21 [21][ftp] host: 127.0.0.1 login: roadd password: hackingdesdecero 1 of 1 target successfully completed, 1 valid nassword found Hydra (http://www.thc.org/thc-hydra)	

Aquí vemos la información que nos tira y como encontró la **password correcta**. No hay mucho que decir aquí más que lo que ya sabemos :). **Investiguen la herramienta** que es excelente. **¿Y cómo hacemos para bloquear este tipo de ataques?**

Desde el Filezilla Server *Edit* → *Settings* y allí en la pestaña de *Autoban*.



FileZilla Server Options		— ———————————————————————————————————
FileZilla Server Options	Autoban By default, FileZilla Server throttles incoming connect attempts. This makes attempts to break into the serv However, a legitimate user would still be able to logor If you're feeling paranoid, you can enable an automa attempts. Be advised that automatic banning can be attacks, so use with care. Enable automatic bans Ban IP address after 10 failed attempts w Ban for 1 hours (1-999).	FileZilla Server s after some failed login ther impractical. In at any time. Attic ban upon multiple failed login abused for denial of service
OK Cancel		

Si habilitamos la opción allí podemos configurar para que en el primer recuadro limitemos la cantidad de intentos **simultáneos**. **Si yerra 10 veces** (por como está configurado por default) tiene **1 hora de prohibición** para intentar poner 10 nuevas contraseñas. **Así nos ahorramos problemas -o nos lo generamos si nos olvidamos la clave xD-.**

Para la próxima clase veremos un servicio conocido como Telnet :). Espero les haya gustado.

Pueden seguirme en Twitter: @RoaddHDC

Cualquier cosa pueden mandarme mail a: <u>r0add@hotmail.com</u>

Para donaciones, pueden hacerlo en bitcoin en la dirección siguiente: 1HqpPJbbWJ9H2hAZTmpXnVuoLKkP7RFSvw

Roadd.

Este tutorial puede ser copiado y/o compartido en cualquier lado siempre poniendo que es de mi autoría y de mis propios conocimientos.