



Click-to-Dial: Aspectos Prácticos de Implementación

Whitepaper

Documento versión: 1.0
Creado: 8 de febrero de 2008
Última actualización: 25 de abril de 2008

Barcelona, España



Reconocimiento - Compartir bajo la misma licencia 2.5 España
<http://creativecommons.org/licenses/by-sa/2.5/es/>

Todas marcas y símbolos gráficos usados aquí son propiedad de sus propietarios.

Se pueden encontrar más whitepapers en http://www.lexatel.com/whitepapers_es

Contenido

Introducción	3
Ventajas.....	4
Diseño funcional.....	5
Integración	6
Seguridad.....	8
Limitar las capacidades de Asterisk y AMI	8
Limitar el interficie dónde hace bind (nivel Asterisk)	8
Limitar las IP's/redes origenes del script.....	8
Sobre Lexatel	9
Licencia.....	10

Introducción

Asterisk es un sistema de telecomunicación de licencia libre desarrollado y distribuido por Digium (<http://www.digium.com>). Esto garantiza un fácil acceso al código fuente y ha repercutido que Asterisk tiene un gran número de características pensadas por y para los usuarios y muchas facilidades de integración con otros sistemas que aprovecharemos en este whitepaper.

Se explicará la experiencia de Lexatel Technologies integrando Asterisk con sistemas ya funcionando para facilitar una marcación rápida: normalmente con un sólo click de aquí al nombre: "Click to dial".

Con este whitepaper tendremos la base teórica y el código y configuración necesario para conseguir una fácil integración de otros sistemas con Asterisk. El protocolo para la comunicación del sistema existente y Asterisk será mediante HTTP ya que es un protocolo muy estándar y simple. Conseguiremos tener una URL para hacer la petición y Asterisk conectará los dos teléfonos.

Esta capacidad para realizar las llamadas nos repercutirá con una productividad mucho mayor en el centro de llamadas.

Las tecnologías principales de interconexión de Asterisk son AMI (el usado en este caso), AGI y EAGI:

- AMI (Asterisk Manager Interface). Permite conectarse desde otros procesos externos al servidor Asterisk mediante el protocolo TCP (por defecto el puerto 5038) y controlar remotamente Asterisk.
- AGI permite que Asterisk ejecute comandos externos y recoja el resultado. Es Asterisk que inicia la comunicación
- EAGI es como AGI pero permite tratar flujos de audio

Los requisitos para el aprovechamiento del artículo:

- Conocimientos básicos de administración de sistemas (instalar paquetes, configurar Apache, PHP, etc.)
- Conocimientos de Asterisk

Ventajas

Cada día hay más sistemas heterogéneos. En las empresas hay sistemas que ya están consolidados y no han llegado al final de su ciclo de vida, mientras que otros sistemas necesitan una renovación y adaptación a las nuevas tecnologías como suele ser VoIP.

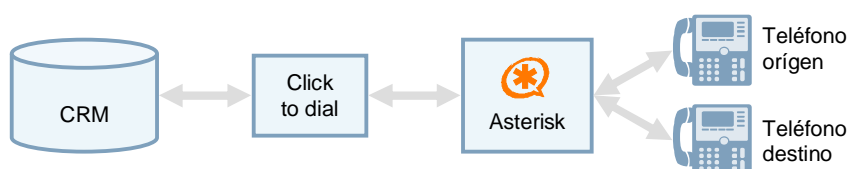
Mediante el click to dial tendremos varios ahorros:

- Ahorro en tiempo. En un call center de llamadas salientes el tiempo de marcación es un tiempo considerable. Con la solución “click to dial” ahorramos un tiempo considerable
- Errores: con el “click to dial” se evitan muchos errores de marcación. Esto ahorra costes en llamadas a números equivocados así como tiempo perdida de tiempo hablando con gente que no son los potenciales clientes.

En nuestra experiencia, la implementación de “Click to Dial” ofrece entre un 10% y 12% más de productividad en cada estación.

Diseño funcional

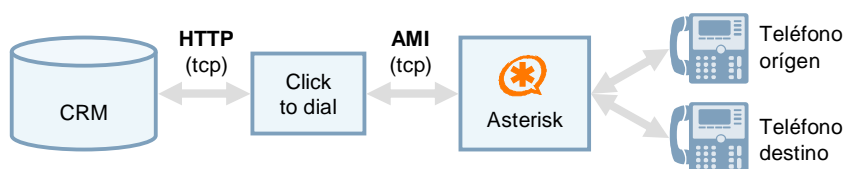
Hay tres sistemas diferenciados:



Los objetivos del sistema son:

- Minimizar el impacto en la integración del CRM
- Conseguir que la integración sea independiente de la versión de Asterisk o otras posibles PBX futuras
- Hacer una integración con costes y impacto mínimo

Una de las tecnologías más estándares y consolidadas para la comunicación entre diferentes equipos de red y con un acoplamiento bajo es el protocolo HTTP. Por esto Lexatel Technologies plantea este escenario:



La mayoría de lenguajes de programación disponen de librerías para hacer peticiones HTTP GET/POST. En caso que no se disponga de ninguna librería, el protocolo HTTP es un protocolo simple y se podría implementar la petición GET fácilmente desde el entorno del CRM existente.

El sistema "Integración" recibirá la petición con sus parámetros (número origen y destino) y comunicará con Asterisk usando la tecnología AMI para que este llame al número origen y al destino.

El sistema CRM podría comunicar directamente con Asterisk. Pero complica mucho la gestión del sistema: a veces es necesario añadir prefijos en el origen o destino (muy fácil hacerlo en un sólo sitio, en "Integración"), si el Asterisk cambiase sus métodos de acceso sólo habría un sitio dónde hacer cambios y no en cada instalación local en el CRM.

La parte del sistema llamada "Integración" y "Asterisk" pueden estar en el mismo servidor o en servidores diferentes. Si es posible por la arquitectura de la red recomendamos configurarlo en el mismo servidor por simplicidad también por facilidad en minimizar la exposición de Asterisk a agentes innecesarios.

Integración

En la solución de Lexatel Technologies, "ClickToDial" es un script en PHP5 que se ejecuta en Apache2. Esta es la solución que expondremos en este whitepaper. En muchos casos ya se usa Apache en el mismo servidor de Asterisk para otras tareas.

Recomendamos instalar el sistema "ClickToDial" en el directorio `/var/www/remote/`. El sistema consta de 2 ficheros: `config.php` y `call.php`. Aquí mostraremos los ficheros básicos que se pueden usar. En la implementación final se puede tener que configurar para cada caso prefijos especiales, control de origen de llamadas, usuario y contraseñas HTTP, etc.

`config.php`:

```
<?php
$asterisk_login="TheLogin";
$asterisk_passwd="ThePassword";
$asterisk_ip="127.0.0.1";
$asterisk_port="5038";
?>
```

`call.php`:

```
<?php
include "config.php";
error_reporting(E_ALL);
$source=$_REQUEST["src"];
$destination=$_REQUEST["dst"];
$socket =
fsockopen($asterisk_ip,$asterisk_port,$errnum,$errdesc) or
die ("Connection to Asterisk failed");
$temp=fread($socket,1024);
fputs($socket,"Action: login\r\n");
fputs($socket,"Events: off\r\n");
fputs($socket,"Username: $asterisk_login\r\n");
fputs($socket,"Secret: $asterisk_passwd\r\n");
fputs($socket,"\r\n");

$temp=fread($socket,1024);

fputs($socket,"Action: originate\r\n");
fputs($socket,"Channel: SIP/$source\r\n");
fputs($socket,"WaitTime: 30\r\n");
fputs($socket,"Exten: $desti\r\n");
fputs($socket,"Context: extensiones-locales\r\n");
fputs($socket,"Callerid: <$source>\r\n");
fputs($socket,"Priority: 1\r\n");
fputs($socket,"\r\n");

$temp=fread($socket,1024);
```

```
fputs($socket,"Action: Logoff\r\n");  
fputs($socket,"\r\n");  
?>
```

El login y contraseña de config.php es el mismo que configuraremos en el fichero de Asterisk `/etc/asterisk/manager.conf`.

La IP en este caso es 127.0.0.1. Hay casos que nos puede interesar tener en dos servidores el Asterisk y el servidor Apache2. No hay ningún problema tenerlo en dos servidores siempre que tengan conectividad TCP entre ellos.

En el fichero `/etc/asterisk/manager.conf` deberemos configurar el acceso. La configuración básica es:

```
[general]  
enabled = yes  
port = 5038  
bindaddr = 0.0.0.0  
displayconnects = yes  
  
[user_01]  
secret=1234567  
deny=0.0.0.0/0.0.0.0  
permit=192.168.1.0/255.255.255.0  
permit=127.0.0.1/255.255.255.255  
read = system,call,log,verbose,command,agent,user  
write = system,call,log,verbose,command,agent,user
```

Una vez configurado ya podremos usar un enlace como:

<http://IP/remote/call.php?src=10&dst=93200900>

Y nos conectará la extensión 10 con el teléfono 93200900.

Seguridad

Es muy importante mantener el sistema tan seguro como sea posible. Aunque el sistema es seguro en sí, siempre es necesario aplicar las buenas prácticas en entornos informáticos y de redes para reducir al máximo el riesgo.

Actuaremos en varios puntos: en la exposición de Asterisk a posibles agentes maliciosos así como como en la exposición del script de PHP hacia fuera. La segunda parte es de administración de sistemas pero haremos un comentario aquí.

Limitar las capacidades de Asterisk y AMI

Conseguir que el script de click to dial sólo pueda hacer dial. Para esto modificaremos en el fichero `/etc/asterisk/manager.conf` para dejar el read y write en:

```
read = call,log,verbose
write = call,log,verbose
```

Limitar el interfície dónde hace bind (nivel Asterisk)

Si desde el script PHP accedemos a Asterisk desde localhost: configuramos el `manager.conf` para hacer bind sólo en el interfície de red de localhost. Sinó podemos hacer algo como:

```
deny=0.0.0.0/0.0.0.0
permit=192.168.1.100/255.255.255.255
```

si quiséramos que sólo se pueda acceder desde la IP 192.168.1.100.

Limitar las IP's/redes orígenes del script

Es importante proteger el sistema desde todos los niveles, por tanto también desde nivel aplicación. P. ej. podemos usar los ficheros `.htaccess` de Apache para limitar desde qué IP's se conectan al script de PHP.

También es recomendable poner un usuario y contraseña para acceder a este script (o bien añadir dos parámetros más en la URL que sean usuario y contraseña).

Si es posible también podemos reducir la exposición de Apache2 usando iptables en la misma máquina o el firewall corporativo.

Se recomienda, si los clientes lo soportan, usar conexiones seguras HTTPS.

Sobre Lexatel

Lexatel es proveedor de soluciones integradas de telecomunicaciones para empresas. Damos la gama completa de servicios – voz, datos, integración, servicios de valor añadido. Además desarrollamos aplicaciones integradas con sistemas de informática ya existentes.

Información de contacto:

Dirección: Lexatel Technologies, S.L.
Rambla de Catalunya 124, 4-1
08008 Barcelona
España

Central: 902 023 958 (desde el territorio nacional)
+34 933 688 239 (desde extranjero)

Fax: +34 933 686 656

Email: info@lexatel.com
whitepapers@lexatel.com

Página web: www.lexatel.com

Licencia



Reconocimiento - Compartir bajo la misma licencia 2.5 España

<http://creativecommons.org/licenses/by-sa/2.5/es/>

Usted es libre de:

- copiar, distribuir y comunicar públicamente la obra
- hacer obras derivadas

Bajo las condiciones siguientes:

Reconocimiento. Debe reconocer los créditos de la obra de la manera especificada por el autor o el licenciador (pero no de una manera que sugiera que tiene su apoyo o apoyan el uso que hace de su obra).

Compartir bajo la misma licencia. Si altera o transforma esta obra, o genera una obra derivada, sólo puede distribuir la obra generada bajo una licencia idéntica a ésta.

Al reutilizar o distribuir la obra, tiene que dejar bien claro los términos de la licencia de esta obra.

Alguna de estas condiciones puede no aplicarse si se obtiene el permiso del titular de los derechos de autor

Nada en esta licencia menoscaba o restringe los derechos morales del autor.