



Bisoños Usuarios de GNU/Linux de Mallorca y Alrededores | Bergantells Usuaris de GNU/Linux de Mallorca i Afegitons

## De Mozilla ThunderBird a OpenLdap automáticamente (10728 lectures)

Per Elídir Moya R, [lillo](http://emoya.freeshell.org) (<http://emoya.freeshell.org>)

Creado el 05/07/2005 16:50 modificado el 05/07/2005 16:50

*Ya un artículo anterior, publicado en bulma aquí [[Q<sup>1</sup>](#)], había comentado como realizar una migración de un conjunto de direcciones, originalmente en Outlook Express a un servidor OpenLdap (con ayuda de Mozilla ThunderBird). Sin embargo cuando los nombres continene acentos o caracteres especiales es necesario realizar algún trabajo manual. Para superar esta limitación, en búsqueda de una solución encontré que Perl era lo que necesitaba. Sin ser un experto en Perl he creado tres pequeños scripts que ayudan a realizar el trabajo automáticamente.*

### Paso1. Decodificar las líneas dn:: que contienen acentos y caracteres especiales.

Cuado se realiza la exportación en el Mozilla ThunderBird, este codifica en base 64 todas aquellas líneas que contengan un acento o caracter especial. Por lo tanto, el primer paso es decodificar estas líneas para luego proceder a modificarlas. Esto lo logramos con el siguiente script:

```
#!/usr/bin/perl
#-----
# Este script recibe como argumento el nombre del archivo .LDIF
# que deseamos importar en el Servidor OpenLDAP y decodifica aquellos nombres
# que están en base 64 por contener acentos y caracteres especiales
#
# Elídir Moya R. 2005
#-----

# incluye las librerías que usaremos para decodificar
# los nombres de los usuarios que poseen acentos y caracteres especiales
use MIME::Base64 ();

$file = $ARGV[0];
# abre el archivo que se pasó como argumento
open (ARCHIVO, $file) || die('Error');
while ($linea = ) # Recorre el archivo línea por línea
{

    if ($linea =~ /^dn::/) # selecciona las líneas que inician con dn::
    {

        # Asigna a la variable hilera solamente los caracteres que están codificados en base 64
        $hilera = substr($linea, 4 ,(length($linea)-4) );
        # instrucción para que el final de la línea nueva sea idéntica al de las otras
        $fin = substr($linea, (length($linea)-2),2);
        $decodificado = MIME::Base64::decode($hilera); # decodifico
        print('dn:: ');
        print($decodificado);
        print ($fin); #imprime la línea decodificada, con el final apropiado

    }
    else

```



```

    {
        print ($linea); #imprime la línea normalmente
    }
}

```

Podemos almacenarlo con el nombre convierte.pl y ejecutarlo de la siguiente forma, asumiendo que está almacenado en la carpeta /home/emoya y que el archivo que exportamos en el mozilla ThuderBird se llama exportado.ldif

```
/home/emoya# ./convierte.pl exportado.ldif > decodificado
```

La anterior instrucción creará un nuevo archivo, llamado decodificado el cual contiene todas las líneas presentes en exportado.ldif de manera que aquellas líneas que contenían acentos y caracteres especiales ahora están decodificadas para que las podamos modificar en el siguiente paso.

## Paso 2. Modificar las líneas dn para que se ajusten a nuestras necesidades

El siguiente paso es modificar todas las líneas dn, que en este momento lucen así

```
dn: nombre apellidos, mail = correo@dominio.com
```

para que se muestren así

```
dn: nombre apellidos, dc=institucion,dc=com
```

Esto lo lograremos con el siguiente script

```

#!/usr/bin/perl
#-----
# Este script recibe los datos
# que deseamos importar en el Servidor OpenLDAP (después de haber sido decodificados por el primer script)
# lo que hace es modificar todas las líneas dn: nombre, mail por dn: nombre, dc=institucion,dc=com ...
#
# Elídir Moya R. 2005
#-----
$hilera_reemplazo = "o=institucion,dc=com"; #Debe modificar esta línea

while ($linea = ) # lee de la entrada estándar
{
    if ($linea =~ /^dn/) # verifica cuales líneas inician en dn
    {
        $fin = substr($linea, (length($linea)-2),2);
        #Reemplaza todo despues de la palabra mail, incluyendo el fin de línea
        $linea =~ s/mail=.*\n/$hilera_reemplazo/;
        print($linea); # imprime la línea
        print($fin); # imprime el fin de línea
    }
    else
    {
        print ($linea); #imprime todas las otras líneas
    }
}

```



}

Asumiendo que el script está almacenado en la carpeta /home/emoya y que fue salvado con el nombre modifica.pl, podríamos utilizarlo de la siguiente forma:

```
cat decodificado | ./modifica.pl > modificado
```

El resultado sería un archivo denominado modificado que contiene nuestro documento casi listo para ser importado en el OpenLDAP. El único paso pendiente sería volver a codificar las líneas dn que contienen acentos y caracteres especiales.

**Nota:** No olvide modificar la línea donde se inicializa \$hilera\_reemplazo antes de ejecutar el script

### Paso 3. Codificando nuevamente

Este es el script que podemos utilizar para codificar en base 64 todas las líneas dn:: que decodificamos con el primer script

```
#!/usr/bin/perl

#-----
# Este script recibe los datos
# que deseamos importar en el Servidor OpenLDAP
# después de haber sido:
# 1. Decodificados por el primer script
# 2. Modificados por el segundo
# lo que hace es codificar en base 64 todos aquellos nombres que contienen acentos y
# caracteres especiales
#
# Elídir Moya R. 2005
#-----

# incluye las librerías necesarias para codificar en base 64
use MIME::Base64 ();

while ($linea = ) # lee el archivo de la entrada estándar
{
    if ($linea =~ /^dn::/) #importante: solo codifica las líneas que comienzan con dn::
    {
        $hilera = substr($linea, 5, (length($linea)-7) );
        $fin = substr($linea, (length($linea)-2),2);
        $hilera = MIME::Base64::encode($hilera,"");
        print('dn:: ');
        print($hilera);
        print($fin);
    }
    else
    {
        if ($linea !~ /^modifytimestamp/)
        { # elimina esta línea, ya que no es necesaria.
            print ($linea);
        }
    }
}
}
```

Este lo podemos ejecutar así

```
cat modificado | ./codifica.pl > codificado.ldif
```



## Resumen

En resumen, los pasos para darle el formato apropiado a nuestro archivo sería:

1. Crear los tres scripts
2. Darle privilegios de ejecución a los scripts (Por ejemplo: chmod 700)
3. Modificar la línea donde se inicializar \$hlera\_reemplazo en el segundo script, de manera que se ajuste a nuestro servidor
4. Ejecutar los Script.

## En una sola línea

Puede ejecutar las instrucciones en una sola línea así:

```
./convierte.pl archivo.ldif | ./modifica.pl | ./codifica.pl > Archivo_listo.ldif
```

---

### Lista de enlaces de este artículo:

1. <http://bulma.net/body.phtml?nIdNoticia=2188>
- 

E-mail del autor: emoya\_ARROBA\_costarricense.cr

Podrás encontrar este artículo e información adicional en: <http://bulma.net/body.phtml?nIdNoticia=2209>