UNIVERSITAT POLITÈCNICA DE VALÈNCIA

etsinf

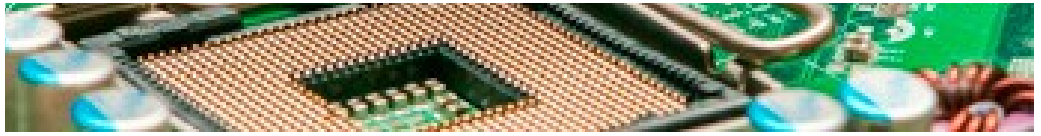
Máster Universitario en Ingeniería Informática

CyberSeguridad
Tema 5: Cifrado de ficheros

José Ismael Ripoll Ripoll

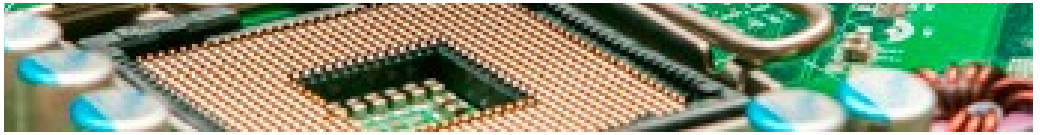
Máster Oficial Universitario en Ingeniería Informática
muiinf.webs.upv.es

MUIInf



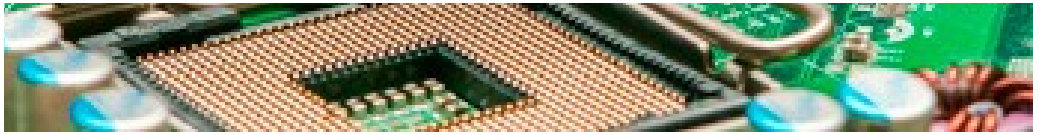
Contenidos

- Cifrado de ficheros con GPG
  - Cifrado simétrico.
  - Cifrado asimétrico.
  - Firmado de ficheros.
- Cifrado de dispositivos.
  - LUKS (Linux Unified Key Setup)



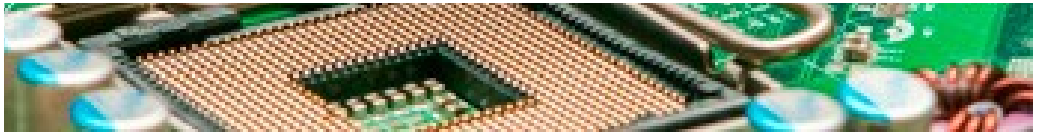
GnuPG: Gnu Pretty Good Privacy

- gpg2 is the OpenPGP part of the GNU Privacy Guard (GnuPG). It is a tool to provide digital encryption and signing services using the OpenPGP standard.
- Es la versión de GNU del conocido PGP.
- Permite:
  - Des/Cifrado simétrico.
  - Des/Cifrado asimétrico.
  - Firmado.
  - Comprobación de firma.



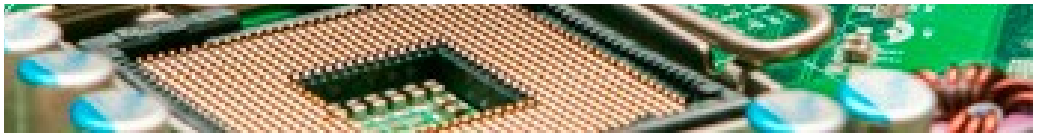
GPG: soporte

```
$ gpg --version
gpg (GnuPG) 1.4.16
Copyright (C) 2013 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later
<http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

Home: ~/.gnupg
Algoritmos disponibles:
Clave pública: RSA, RSA-E, RSA-S, ELG-E, DSA
Cifrado: IDEA, 3DES, CAST5, BLOWFISH, AES, AES192, AES256, TWOFISH,
         CAMELLIA128, CAMELLIA192, CAMELLIA256
Resumen: MD5, SHA1, RIPEMD160, SHA256, SHA384, SHA512, SHA224
Compresión: Sin comprimir, ZIP, ZLIB, BZIP2
```

Cifrado simétrico

- Cifrar un fichero con un password (clave simétrica)
  - gpg -c fichero
  - Produce: fichero.gpg
- Cifrar un fichero con password y el algoritmo AES256
  - gpg -c --cipher-algo AES256 fichero
  - Produce: fichero.gpg
- Cifrar generando un fichero en base 64
  - gpg -c -a --cipher-algo AES256 fichero
  - Produce: fichero.asc
- Descifrado:
  - gpg -d fichero.asc



GPG: generación de clave asimétrica

§ Generar una pareja de claves pública/privada

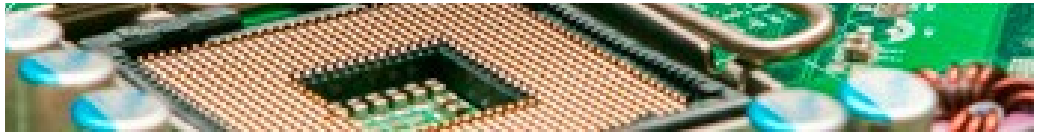
♦ gpg2 --gen-key

```
$ gpg2 -gen-key
gpg (GnuPG) 1.4.16; Copyright (C) 2013 Free Software
Foundation, Inc.
This is free software: you are free to change and
redistribute it.
There is NO WARRANTY, to the extent permitted by law.
Please select what kind of key you want:
   (1) RSA and RSA (default)
   (2) DSA and Elgamal
   (3) DSA (sign only)
   (4) RSA (sign only)
Your selection? 2
DSA keys may be between 1024 and 3072 bits long.
What keysize do you want? (2048) 3072
Requested keysize is 3072 bits
Please specify how long the key should be valid.
         0 = key does not expire
      <n>  = key expires in n days
      <n>w = key expires in n weeks
      <n>m = key expires in n months
      <n>y = key expires in n years
Key is valid for? (0) 10
Key expires at dom 22 nov 2015 17:23:30 CET
Is this correct? (y/N) y
You need a user ID to identify your key; the software
constructs the user ID
from the Real Name, Comment and Email Address in this form:
    "Heinrich Heine (Der Dichter) <heinrichh@duesseldorf.de>"
```

```
Real name:  Perico de Los palotes
Email address: pericos@nosite.no
Comment:
You selected this USER-ID:
    "Perico de Los palotes <pericos@nosite.no>"


Change (N)ame, (C)omment, (E)mail or (O)kay/(Q)uit? O
You need a Passphrase to protect your secret key.
```

La clave privada está protegida con una passphrase

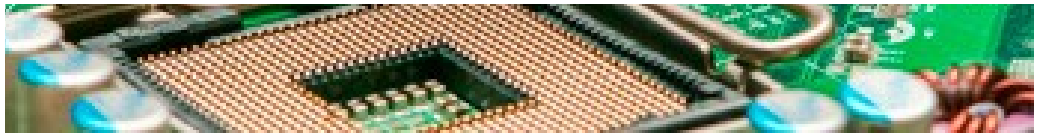
```
We need to generate a lot of random bytes. It is a good idea to pe
some other action (type on the keyboard, move the mouse, utilize t
disks) during the prime generation; this gives the random number
generator a better chance to gain enough entropy.
gpg: WARNING: some OpenPGP programs can't handle a DSA key with th
..++++++++++++++++++++.+++++..++++++++++...++++++++++.+++++.
+++++.++++++++++++++...++++++++++++++++++++.++++++++++++++++
++++++++++.++++++++++++++++++++..+++++..++++++++++.+++++>.++
++++++++++++++++.++++++++++.+++++.++++++++++>+++++>+++++<.
+++++>++++
+.............................................................


Not enough random bytes available.  Please do some other work to g
the OS a chance to collect more entropy! (Need 185 more bytes)
```

- Listar el "anillo de claves"
  - `gpg -k`

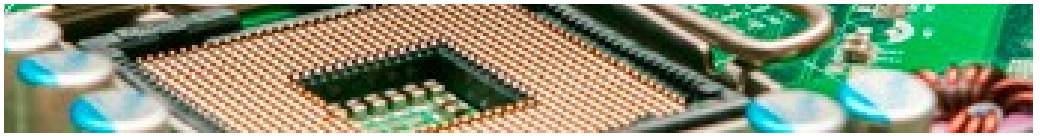
```
home/iripoll/.gnupg/pubring.gpg
-------------------------------
pub   3072D/1FE90B56 2015-11-12 [expires: 2015-11-22]
uid                  Perico de Los palotes <pericos@nosite.no>
sub   3072g/F768E14B 2015-11-12 [expires: 2015-11-22]
```

- Las claves públicas y privadas se guardan en un fichero llamado "anillo de claves"
  - No tiene nada que ver con el señor de los anillos.
  - Sino con la anilla que se usa para llevar las llaves agrupadas.
- Podemos "subir" nuestra clave pública a un servidor de claves centralizado:
  - `gpg --send-keys --keyserver pgp.mit.edu [ID-CLAVE]`
- Y podemos consultar las claves de mucha gente allí.



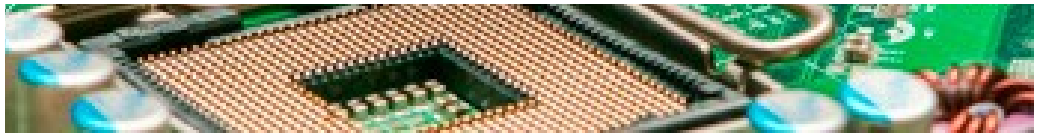
GnuPG: Cifrar con una clave pública

- Se obtiene el identificador de la clave pública y:
  - `gpg --encrypt --recipient F768E14B fichero.txt`
- El resultado es:
  - Fichero.txt.gpg
- Para enviarlo por mail es mejor que el resultado sea ascii plano: base64:
  - `gpg --encrypt -a --recipient F768E14B fichero.txt`
- La salida será:
  - Fichero.txt.asc
- Siempre podemos extraer el contenido de un fichero en base64, con la utilidad base64 :-)



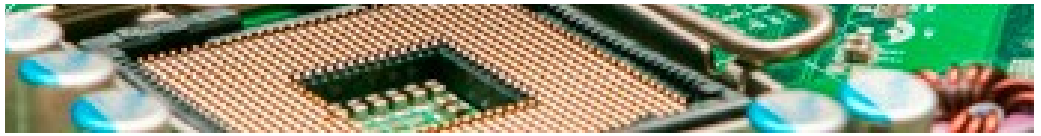
GnuPG descifrado

- Se necesita la clave privada para descifrar un fichero cifrado con la pública:
  - `gpg -d -o fichero.txt fichero.txt.gpg`
- Puede que esté instalado el gpg-agent
  - Es un demonio que maneja las claves privadas.
  - Guarda durante la sesión las passphases.
  - Solo es necesario introducirla una sola vez.



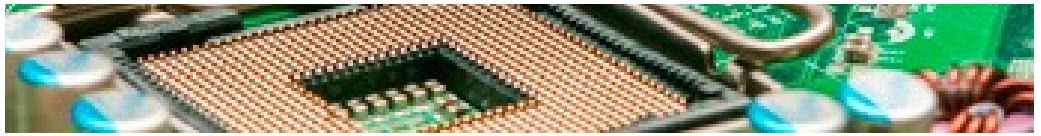
GnuPG: Firmar un documento

- Firmar en claro:
  - `gpg --clearsign -a -u F768E14B fichero.txt`
  - El fichero resultante "fichero.txt.asc" contiene el texto original y la firma al final.
  - Se suele utilizar para enviar mails. El receptor puede ver el mail sin tener la clave pública del firmante.
- Firmar:
  - `gpg --sign -a -u F768E14B fichero.txt`
- Verificar la firma:
  - `gpg --verify fichero.txt.asc`
  - `gpg -d fichero.txt.asc`

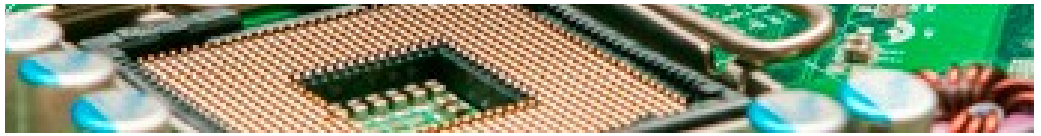


Cifrado de una partición: LUKS

- LUKS: Linux Unified Key Setup-on-disk-format
  - `# apt-cache search cryptsetup`
- Se puede utilizar sobre una partición o sobre sistemas en loop.
  - `$ dd if=/dev/zero of=fs.raw bs=1025 count=10000`
- Crear el cifrado:
  - `$ cryptsetup --verify-passphrase luksFormat fs.raw`
- Acceder al dispositivo cifrado:
  - `# cryptsetup luksOpen fs.raw my_fs`
- Formatear el dispositivo:
  - `# mkfs.ext4 /dev/mapper/my_fs`



- Montar el sistema cifrado:
  - mount /dev/mapper/my_fs /mnt
- Todo esto se puede automatizar utilizando el fichero
  - cat /etc/crypttab
  - my_fs fs.raw none luks
- Cuando Linux arranque tratará de abrir el dispositivo cifrado (LuksOpen) y luego montarlo:
  - cat /etc/fstab
  - /dev/mapper/my_fs /mnt ext4 errors=remount-ro 0 0



- Para desmontar un sistema cifrado:
  - `umount /mnt`
  - `cryptosetup close my_fs`