

# Laboratorio 2 – Cifrado II

## Objetivos:

1. Aprender el uso de herramientas tipo PGP (OpenPGP, GnuPG) para asegurar las comunicaciones.
2. Aprender sobre los anillos de confianza.
3. Aprender sobre otras funciones que ofrece GPG.

## Recursos necesarios:

- Ubuntu (Usuario/password: lsi/lsi).

## Índice:

1. Asegurando las comunicaciones mediante GPG.
2. Confianza sobre las claves.
3. Firmas GPG.
4. Otras funcionalidades GPG.

## Asegurando las comunicaciones mediante GPG

GnuPG<sup>1</sup> (GPG) es un programa libre que nos permite cifrar, descifrar y firmar información cumpliendo el estándar OpenPGP<sup>2</sup> y así asegurar nuestras comunicaciones. La mayoría de distribuciones de Linux traen por defecto el paquete GPG, pero si fuese necesario se puede instalar en Ubuntu mediante el siguiente comando:

```
$ sudo apt install gpg
```

GPG ofrece muchas posibilidades. Es conveniente familiarizarse con ellas:

```
$ gpg --help
```

Para trabajar con GPG, lo primero es generar un par de claves (pública y privada):

```
$ gpg --generate-key
```

Es muy importante proveer una dirección de email válida. La frase clave es opcional y sirve para proteger el acceso al llavero de claves privadas. En PGP, el llavero es el almacén donde se almacenan las claves con las que se va a trabajar. Existe un llavero de

---

1 <https://gnupg.org/>

2 <https://www.openpgp.org/>

#### Sistemas de Gestión de Seguridad de la Información 2023/2024

claves privadas, y otro llavero de claves públicas. Es aconsejable proteger el llavero con una frase clave.

Usando el comando **gpg --full-generate-key** se puede especificar qué longitud de clave deseáis usar, y qué algoritmo queréis usar para su creación. GnuPG soporta RSA, DSA y ElGamal. Para la creación del par de claves se usa una medida denominada entropía, que simboliza la cantidad de aleatoriedad o desorden que tiene la clave. A mayor entropía, mayor aleatoriedad y por lo tanto más complicado de realizar un criptoanálisis. En la generación de claves la entropía se obtiene en base a datos de la máquina como el estado de la CPU, la fecha, el número de ventanas abiertas, etc. Así que mientras se genera la clave es aconsejable navegar, abrir ventanas, teclear cosas, etc. para generar una entropía lo mayor posible.

Una vez terminada la generación de las claves se da la posibilidad de crear un certificado de revocación de las claves. El certificado de revocación sirve para indicar que tu clave ya no es válida porque la has perdido, te la han robado, etc. Cread el certificado de revocación y guardadlo.

Una vez creadas las claves, para verlas:

```
$ gpg --list-keys
```

¿Qué quiere decir **[ultimate]**?

Es importante que la clave pública esté accesible. Se puede publicar en una página web personal<sup>3</sup>, se puede enviar adjunta en un email, o se puede publicar en servidores específicos como **keys.openpgp.org**.

GPG puede ser usado integrado en clientes de correo como Thunderbird, o directamente en la línea de comandos. Para enviar archivos que han sido cifrados en la línea de comandos mediante GPG simplemente basta con adjuntarlos en el email.

- Cifrad este archivo PDF y enviáoslo entre vosotros de forma que consigáis los principios de seguridad **Confidencialidad, Integridad, Autenticidad y No Repudio**.

Razonad qué habéis tenido que hacer para conseguir cada uno de ellos.

## Confianza sobre las claves

Como habéis podido comprobar, es muy fácil crear un par de claves y poner cualquier nombre. No se realiza ningún tipo de comprobación. Por lo que si recibimos un archivo firmado y/o cifrado por X, no podemos estar seguros de que realmente sea X a no ser que tengamos alguna manera de preguntarle a X a ver si esa es realmente su clave. Sin

<sup>3</sup> Por ejemplo: <https://mikel-egana-aranguren.github.io/about/>

#### Sistemas de Gestión de Seguridad de la Información 2023/2024

embargo, existen mecanismos para que podamos confiar en las claves de una persona aun sin necesidad de conocerla o haber hablado previamente con ella para comprobar si es su clave.

- En cada grupo se designará a uno de los estudiantes como “de confianza”. Ese estudiante enviará su clave pública al profesor. El grupo tendrá que conseguir que al enviar las claves de los otros estudiantes al profesor aparezcan como de confianza en el anillo de claves del profesor ([full]).

Razonad qué habéis tenido que hacer para conseguirlo.

## Firmas GPG

En la página web de los desarrolladores de Enigmail<sup>4</sup> se pueden descargar dos ficheros, la extensión para Thunderbird (.xpi) y otro fichero llamado “GPG Signature”.

¿Para qué sirve ese segundo fichero? ¿Cómo se usa?

En GitHub existe la opción de firmar commits mediante GPG, para aumentar la seguridad y trazabilidad de dichos commits. Por ejemplo, en este commit aparece la palabra “Verified” y si pinchamos en la misma nos da detalles de quién la firmó y con qué clave:

<https://github.com/mikel-egana-aranguren/mikel-egana-aranguren.github.io/commit/03c5037c8f56196aaa4ed4fd4b303e7a70fdb422>

Usa tus claves GPG para firmar un commit del repositorio GitHub en el que estás desarrollando tu sistema web para las entregas, de modo que aparezca como “Verified” en GitHub.

## Otras funcionalidades GPG

Es importante que seáis capaces de usar vuestras claves en otros equipos.

¿Cómo se exporta una clave GPG para poder usarla en otro equipo?

Puede pasar que una clave quede comprometida.

¿Cómo revocarías tu clave?

Aunque su función principal es el cifrado asimétrico, GPG también se puede usar para cifrado simétrico.

¿Como cifrarías este documento de manera simétrica, y qué pasos seguirías para que el receptor lo descifre?

<sup>4</sup> <http://www.enigmail.net/download>