Bien, pues pasamos aquí ahora a la parte de GPG Tool, de la herramienta GPG.

Vamos a ver algunas opciones y alguna descripción sobre esas opciones.

Gpg es una suite que les permite poder cifrar, descifrar archivos, mensajes, firmar, verificar, es decir, podemos utilizar tanto claves simétricas como clave asimétrica para realizar diferentes tipos de tareas.

Además podemos conectar con servidores de cle que están en Internet para poder tanto enviar las claves como para poder descargarnos las claves públicas, en este caso de estos servidores.

Por ejemplo, si nosotros quisiéramos enviar nuestra clave pública a otro usuario que se encuentra lejos, podemos utilizar el correo electrónico o podríamos subir nuestra clave pública a este servidor de claves, bueno, a estos servidores de claves que existen alrededor del mundo.

Esa clave, nuestra clave pública se replica y cualquier usuario en Internet podría buscarnos por el email, buscar nuestra clave pública y poder descargarla.

Para eso están estos servidores de Clef.

Podemos gestionar la revocación de las primeras claves, podemos generarnos las claves, podemos hacer búsquedas en estos key servers, en nuestras servidores de claves a través de ciertos parámetros como estamos viendo aquí, por lo cual estos parámetros son importantes.

La ayuda de la herramienta también es importante para entender mejor cómo funciona GPG y en muchos casos los parámetros son bastante auto explicativos.

Vamos a empezar a ver algunos ejemplos.

Por ejemplo, cuando queremos realizar la generación de una clave o de un par de claves, en este caso clave pública, clave privada, podemos utilizar el parámetro generate key.

Si queremos estar en el proceso donde indicar todos los pasos que queremos llevar a cabo desde los bits que queremos nuestra clave, el tipo de de clave simétrica que

queremos utilizar, podríamos utilizar el parámetro full generate key como indica ahí.
Fijaros en la nota que nos están poniendo en la cuarta línea de la ejecución del comando, ahí estamos utilizando el nombre, estamos utilizando la dirección de email que queremos asociar, queremos utilizar para asociarlas a clave pública, a ser clave privada.
Eso es muy importante porque luego utilizaremos el email para poder utilizar las claves.
Después del proceso de registro, que ya os digo que en el caso del Generate Key es mucho más corto que en el caso del Full Generate Key porque hay menos cosas que seleccionar, pues nos va a preguntar si está todo ok.
Si decimos que está todo Ok empieza a generar un conjunto de bytes aleatorios para generar entropía.
También deciros que nos va a pedir, aunque aquí no se ve, nos va a pedir si queremos proteger la clave privada con un PIN, una contraseña, una pequeña contraseña para que en el caso de que usemos la clave privada pues automáticamente nos pida ese segundo factor de autenticación a través de una clave, como puede ocurrir por ejemplo en el caso del certificado digital, cuando utilizamos un certificado digital de identidad para firmar un documento, nos va a pedir la contraseña para usar la clave privada.
Bueno, fijaros que la primera vez que se generan las claves.
Cada vez que se generamos claves se almacenan en una carpeta, en el home del usuario hay una carpeta gnupg, que ahí es donde vamos a tener los datos, las claves creadas.
Fijaos, también nos dan ahí el fingerprint, que se puede ver la clave pública, vemos que hay una clave que es pub, ahí veis el valor de la clave pública, tenemos el fingerprint de esa clave, ahí lo tenemos.
Luego también, si queremos listar las claves que tenemos almacenadas en nuestro sistema desde el usuario en el que me encuentro, puedo utilizar el comando gpg listkeys, me lista las diferentes claves que tenemos, ya sea porque las hemos importado de otros usuarios o ya sea porque las hemos generado en el sistema.

Para mí, para tener diferentes claves, fijaos que hay un pugre, hay un fichero pugr, que es donde se encuentra el anillo de claves y en este caso la que hemos creado es la única que tenemos.

Ahí se puede ver donde nos dan la información y nos están diciendo, oye, tenemos una clave pública y también tenemos una clave privada.

Importante generar un certificado de revocación.

¿Esto qué es?

Bueno, veis el comando de arriba, el parámetro output es importante porque nos permite volcar ficheros de salida.

En este caso estamos generando un certificado mediante el parámetro gen revoke para la clave pública asociada a la cuenta de correo, en este caso del usuario Pablo.

Este certificado nos pide una razón para generar esa, un certificado de revocación para esta clave.

Bueno, fijaos que ahí nos están diciendo que la clave, por ejemplo, ha sido comprometida o la clave ha sido, bueno, no es demasiado larga, por ejemplo, cualquier razón, podemos elegir una razón y el certificado se acaba generando.

Bien, más acciones.

Tenemos la posibilidad de importar, exportar y buscar claves.

Importar y exportar es importante.

El parámetro export nos permite exportar con el parámetro output, el parámetro output, exportar la clave a un fichero.

El parámetro armor lo que nos permite es generar un feature asociado de ASCII.

Si queremos importar claves de otro usuario, por ejemplo, otro usuario nos envía su clave pública, lo queremos importar a nuestro niño de claves para luego poder, por ejemplo, cifrar un fichero para ese usuario, que solamente ese usuario lo pueda descifrar.

O si quisiéramos a posteriori validar la firma de un usuario sobre un documento con su

clave pública, podríamos validar.

Luego tenemos el comando search, que es el parámetro search keys, que es importante porque tenemos la posibilidad de decirle, oye, con gpg, toma, esto es un dominio, un keyserver, un dominio de un servidor de claves, busca nuestras claves y le podemos dar una dirección de correo y ver si en los servidores de claves que hay en Internet existe alguna clave pública asociada a esa dirección servicio.

Bien, aquí tenéis un ejemplo de exportación de una clave, ahí veis con el gpg output estamos guardando un fichero de clave exported key y ahí lo tenemos.

Fijaros que si decimos con el comando file a un fichero key qué tipo de fichero es, nos dice es una pgp, entonces lo está identificando claramente que es un fichero pgp, bueno, que es una clave pgp que estamos utilizando o que hemos conseguido.

Fijaos también un parámetro, un parámetro interesante es el del sendkeys asociado al parámetro de keyserver, le estamos diciendo oye queremos subir nuestra clave pública a un servidor de claves y le tenemos que pasar el fingerprint, el fingerprint de la clave, recordad que la podemos recuperar con el parámetro fingerprint asociado a un email para obtener esa clave y obtener esa información de la que queremos sacar la huella.

Más cosas, tenemos también por ejemplo parámetros importantes como es el encrypt y el decrypt, encrypt cifrar, entonces podemos cifrar con el parámetro armor, lo que conseguimos es que nos devuelve un documento cifrado pero en formato ascii, un texto, el texto contenido del texto del fichero en formato ascii pero está cifrado.

Podremos verlo más antes en un ejemplo.

Si no ponemos el armor lo que estamos obteniendo como salida es un fichero binario gpg.

Es bastante diferente porque el asc nos va a devolver un mensaje cifrado pero que el contenido lo vamos a poder entender que es un texto y en el caso del Benof solamente sin el armor nos va a devolver un fichero binario, un nuevo fichero gpg, un fichero ilegible.

Y luego tenemos el parámetro de write que con una redirección sobre el fichero pues

vamos a poder descifrar el fichero que queremos descifrar y vamos a obtener lógicamente un fichero en texto plano o binario que no esté cifrado.

Aquí tenéis algunos ejemplos, aquí tenéis una operación de firma, arriba tenemos una operación de firma con parámetros sign, firmamos el fichero secrets txt, no estamos cifrándolo, estamos solamente firmándolo.

La salida decimos que lo queremos en un fichero secret sig y luego con el parámetro verify vemos que podemos verificar con la clave pública lo que está ocurriendo.

Arriba es la clave privada, estamos utilizando una clave privada del Usuario, en este caso solamente tenemos un usuario, el usuario Pablo, estamos usando esa clave privada para firmar el fichero secrets, esa firma se está almacenando en secret sig y lo que estamos haciendo luego es si nosotros enviamos ese fichero firmado o ese fichero secreterssecrets.

Txt y el secret sig asociado.

Pueden verificar el secret sig sabiendo que no se ha modificado el mensaje original con el parámetro menos verify.

Y luego justo debajo tenemos un ejemplo de cifrado y descifrado.

Tenemos el parámetro output donde vamos a guardar en este caso un binario gpg, tenemos el parámetro encrypt para cifrar, tenemos el parámetro recipient que éste asociamos a un email, la clave pública que queremos utilizar y el fichero que queremos cifrar.

Esto nos genera un binario que es secret gpg y con el parámetro de crypt podemos hacer el descifrado y obtener un nuevo fichero con el texto plano de ya disponible.

Utilizando en el primer caso descifrado utilizamos la Clave Pública del Usuario Pablo y para descifrar bueno utilizamos la Clave Privada del Usuario Pablo.

Bueno como conclusiones llegamos al final, hemos estado viendo el tipo de criptografía simétrica, hemos visto un poco más en detalle en qué consiste, hemos jugado con la parte

de de la criptografía asimétrica, hemos visto algunos puntos clave acerca de las claves y los tipos de ésta y hemos visto un ejemplo teórico, hemos visto un ejemplo práctico pero en slide de GPG.

Más adelante veremos un ejemplo práctico ya utilizando una terminal e iremos ejecutando y viendo un poco los resultados de esta aplicación.

Bien, con esto ahora sí llegamos al final de la sesión y nos vemos en la próxima sesión.