

REDES DE COMUNICACIONES II

Práctica 2 - Seguridad y criptografía

[Volver a: Prácticas ➡](#)

2 Funcionalidad

2.1 Gestión de identidades y usuarios

Esta funcionalidad hace referencia a la capacidad del cliente para gestionar la identidad digital del usuario, de forma que quede registradas en SecureBox. Esta identidad deberá ser almacenada localmente por el cliente, típicamente en forma de fichero o ficheros en un directorio.

Concretamente, el cliente deberá poder:

- **Crear una nueva identidad**, con la opción `--create_id [alias]`. El cliente deberá generar un nuevo par de claves pública y privada, con el alias adecuado (si se ha especificado alguno), y registrarla en SecureBox con la llamada adecuada, lo que generará un ID único para el usuario. **En cada momento, cada usuario sólo puede tener una identidad activa asociada a un token de autenticación.** Si se crea y registra una nueva identidad, los datos asociados a la antigua (nombre, email y clave pública) se perderán.

```
# python securebox_client.py --create_id "Mi primera identidad"
```

```
Generando par de claves RSA de 2048 bits...OK
```

```
Identidad con ID#01020304 creada correctamente
```

- **Buscar una identidad** existente, con la opción `--search_id`. Esto permite encontrar un usuario del que sólo sabemos su nombre o correo, por ejemplo, y buscar su ID para poder enviarle un archivo de forma segura.

```
# python securebox_client.py --search_id "Antonio"
```

```
Searching for identity...OK
```

```
Identity with ID#01020304 found
```

```
Name: Antonio Chicharro
```

```
Email: antonio.chicharro@estudiante.uam.es
```

- **Borrar una identidad** existente, con la opción `--delete_id`. Obviamente, sólo es posible borrar identidades creadas por el propio usuario.

```
# python securebox_client.py --delete_id 01020304
```

```
Requesting removal of identity #01020304...OK
```

```
Identity with ID#01020304 successfully deleted
```

[Volver a: Prácticas ➡](#)