1. Clases

1.1. Cliente

La clase cliente tiene la responsabilidad de verificar que los parámetros de la terminal sean correctos (cantidad de parámetros y modo de ejecución) y además es quien se comunica con el server para crear un certificado o para revocarlo.

1.2. InfoCertificado

Lee el archivo que tiene la información necesaria para crear el certificado, y permite obtener los distintos campos por separado. Eso es, subject, fecha inicial y fecha final. En caso de no haber una fecha final, esta se calcula a partir de la fecha inicial.

1.3. Certificado

Permite armar el certificado. Se pueden setear cada uno de los campos y luego obtener una cadena con el certificado completo. También es posible armar el certificado leyendo el archivo del certificado. Además permite guardar el certificado en un archivo.

1.4. Clave

Esta clase representa una clave, ya sea privada o publica. La misma se crea con un exponente (1 byte sin signo) y un modulo (2 bytes sin signo).

1.5. Encriptador

El encriptador usa el algoritmo RSA para encriptar y desencriptar. Se usa luego para encriptar con una clave publica o privada.

1.6. Hash

Función de hashing que aplica la sumatoria del valor ASCII de todos los caracteres que componen una cadena en un entero sin signo de 2 bytes.

1.7. InfoClave

Lee el archivo que tiene la información de la/las clave/claves del server/cliente. Luego es posible obtener dichas claves por separado.

1.8. Mensajero

Esta clase sabe como enviar cada tipo de dato. Se utiliza y sobrecarga el operador >> y el operador <<.

1.9. Skt

Se hicieron algunos cambios con respecto a lo que habia hecho para el tp1. La clase socket es la que sabe como comunicarse, envia y lee mensajes. Permite ademas cerrar canal de escritura/lectura.

1.10. SktAceptador

El SktAceptador lo usa el server primero para escuchar a los clientes (clientes ya pueden quedar en lista de espera) y luego ya puede aceptar clientes. Al aceptar un cliente se obtiene un Skt con el cual nos vamos a poder comunicar con el cliente que corresponda. Para esto ultimo (devolver un Skt al aceptar un cliente) utilice move semantics.

1.11. Indice

Al crearse el objeto lee el archivo índice y guarda en memoria la información de todos los subjects y claves. Permite tener registro de los certificados, así como también agregar y borrar los mismos. Luego podemos pedirle que guarde el índice actualizado.

1.12. Server

Esta clase se encarga de validar los parámetros (cant. de parámetros) y de la comunicación con el cliente.

2. Aclaraciones

Debido a los cambios que tuve que hacer para el TP2, no pude llegar a terminar el TP3. Me falta la parte de threads, de la cual ya tengo una idea de como se podría llegar a hacer y pienso terminar en los próximos días.