

Tema del trabajo

Diseño e implementación de un esquema de intercambio de clave para n-usuarios

Objetivos

El objetivo del trabajo es realizar una implementación del esquema desarrollado en la publicación "***Authenticated Key Establishment: From 2-Party To Group***" [1] habiendo seleccionado como algoritmo 2-parte Cramer Shoup [2] y se valorará sustituirlo por un algoritmo post-cuántico

Metodología seguida

La metodología seguida en el desarrollo del trabajo es la siguiente:

Una primera parte (Iteración 0) donde a partir de la documentación proporcionada por la directora del trabajo, se adquirió conocimiento sobre las herramientas criptográficas necesarias para la implementación del esquema a implementar.

En la fase de desarrollo se realizó siguiendo la metodología de desarrollo iterativo de dos semanas por iteración. Cada iteración presentaba a mi directora de trabajo los avances y planificábamos las tareas hasta la siguiente iteración.

Herramientas empleadas

Para desarrollar el trabajo, se ha utilizado Python como herramienta de programación, y las librerías (*cryptography* [3], *sympy* [4], *dill*) que se han requerido para poder implementarlo. Estas últimas están incluidas en el fichero requirements.txt con el fin de facilitar la instalación de las dependencias necesarias utilizando PIP (herramienta de instalación de librerías Python). Adicionalmente, se ha utilizado GitHub como herramienta de gestión de cambios.

Grado de avance

Actualmente la herramienta es funcional en dos formas de ejecución. La primera, menos realista, las instancias de todos los participantes están en una misma función de tal manera que los participantes se encuentran en el mismo proceso. Una segunda iteración, se implementó una ejecución más realista, donde los participantes se conectan a un proceso que simula ser un broadcast, y cada participante se ejecuta en un proceso independiente.

Dificultades encontradas

La mayor dificultad encontrada ha sido la implementación de la comunicación entre el broadcast y los clientes de los participantes. Esta dificultad se debe a la serialización de los mensajes y objetos que se comparten por el canal.

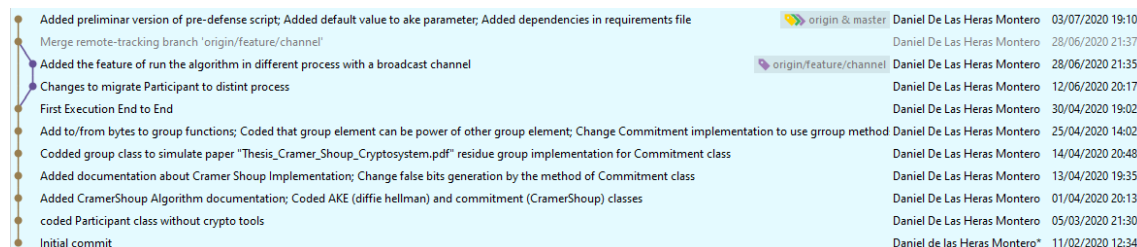
Planificación para terminar

Las tareas planificadas para terminar se componen de:

- Medición tiempo de cómputo vs # participantes
- Compleción de documentación en el repositorio
- Evaluación de posibles algoritmos post-cuánticos que replacen el algoritmo Cramer Shoup y el algoritmo de intercambio de claves (Diffie-Hellman) por las correspondientes herramientas de Kyber-Cristal
- Redacción final de la memoria
- Incorporación de sugerencias y recomendaciones resultantes de la sesión de pre-defensa

Considerando el ritmo actual, se estima que será necesario un plazo entre 4-6 semanas para la realización de las tareas anteriores.

Commits



| | | | |
|--|------------------------|------------------------------|------------------|
| Added preliminar version of pre-defense script; Added default value to ake parameter; Added dependencies in requirements file | origin & master | Daniel De Las Heras Montero | 03/07/2020 19:10 |
| Merge remote-tracking branch 'origin/feature/channel' | | Daniel De Las Heras Montero | 28/06/2020 21:37 |
| Added the feature of run the algorithm in different process with a broadcast channel | origin/feature/channel | Daniel De Las Heras Montero | 28/06/2020 21:35 |
| Changes to migrate Participant to distinct process | | Daniel De Las Heras Montero | 12/06/2020 20:17 |
| First Execution End to End | | Daniel De Las Heras Montero | 30/04/2020 19:02 |
| Add to/from bytes to group functions; Coded that group element can be power of other group element; Change Commitment implementation to use group method | | Daniel De Las Heras Montero | 25/04/2020 14:02 |
| Coded group class to simulate paper "Thesis_Cramer_Shoup_Cryptosystem.pdf" residue group implementation for Commitment class | | Daniel De Las Heras Montero | 14/04/2020 20:48 |
| Added documentation about Cramer Shoup Implementation; Change false bits generation by the method of Commitment class | | Daniel De Las Heras Montero | 13/04/2020 19:35 |
| Added CramerShoup Algorithm documentation; Coded AKE (diffie hellman) and commitment (CramerShoup) classes | | Daniel De Las Heras Montero | 01/04/2020 20:13 |
| coded Participant class without crypto tools | | Daniel De Las Heras Montero | 05/03/2020 21:30 |
| Initial commit | | Daniel de las Heras Montero* | 11/02/2020 12:34 |

Referencias

- [1] M. Abdalla, J.-M. Bohli, M. I. González Vasco and R. Steinwandt, "(Password) Authenticated Key Establishment: From 2-Party To Group," 2007.
- [2] S. Ulrick, «Implementation of Cramer-Shoup Cryptosystem,» 2017.
- [3] «pyca/cryptography,» [En línea]. Available: <https://cryptography.io/en/latest/>.
- [4] «SymPy's documentation,» [En línea]. Available: <https://docs.sympy.org/latest/index.html>.
- [5] A. Hänninen, «The Cramer-Shoup Public-Key,» 2006.