Втора домашна задача по информациска безбедност

Марија Трпчевска

196002

Документација за имплементацијата на упростена верзија на Kerberos

1. Иницијални подготовки

Во рамките на проектот дефинирани се четири класи: Main, User, KDC и AES. Main игра улога на интерфејс за тестирање на функционалностите имплементирани во останатите класи, во User и KDC се одредени однесувањата на корисниците и центарот за дистрибуција на клучеви соодветно додека AES е во голем дел ист како и приложената класа со тоа што се додава иницијализирачки вектор од класата IvParameterSpec во функциите encrypt и decrypt за да можат да работат во CBC мод со PKCS5PADDING.

1. Интеракција испраќач-KDC

Во Main се инстанцира KDC и се креираат две инстанци од User - Alice и Bob. Секој корисник има случајно одберен 128-битен клуч Key кој е генериран во конструкторот преку објект од SecureRandom (што значи секој run на програмата ќе даде различна вредност за клучевите) и низа од бајти session\_key која ќе биде иницијализирана понатаму. Приватна членка на KDC е HashMap KEK која преку add\_user\_to\_KDC() 16 бајтниот клуч на секој корисник го хешира со името на корисникот кое понатаму ќе служи како нивна идентификација IDa и IDb. Аlice ја повикува методата generate\_nonce() која на ист начин како и при генерирањето на клучевите креира 128 битен nonce и налага повикување на методата RQST() од KDC која како аргументи го прима името на корисникот што сака да воспостави комуникација (Alice), корисникот со кој се комуницира (Bob) и nonce-от.

Во RQST(), со get() од хeш-мапата се пребаруваат клучевите на корисниците и се проверува дали низите sender\_key и receiver\_key имаат null вредности (нема мапирање за тоа име/ID, не е валиден корисник). Ако извршената идентификација е во ред, се генерира сесиски клуч преку generate\_session\_key() која пак генерира случајна 128 битна низа и прави XOR меѓу неа, sender\_key и receiver\_key. Со generate\_lifetime() се враќа инстанца на ZonedDateTime која го зема моменталното време нагодено за временските зони и додава 30 min (што би било период на важење на сесискиот клуч).

Следи енкрипција на сесискиот клуч, nonce-от, lifetime и името на примачот со sender\_key и сесискиот клуч, името на испраќачот и lifetime со receiver\_key. Бидејќи шифрираните низи имаат различна големина согласно колку се долги имињата на корисниците и бројот на карактери во lifetime (и согласно добиваат различен padding од AES), не се конкатенираат и шифрираат заедно бидејќи практичното разделување на низите по дешифрирање е комплицирано. Затоа сите шифрирани низи се ставаат во листа за праќачот и примачот, а бидејќи функција може да врати само една вредност, истите се ставаат како членки на друга листа и се праќаат како одговор од KDC.

Методата KDC\_RQST\_response\_parse повикана преку Alice ја прифаќа листата, ја добива листата наменета за испраќачот (онаа за примачот само ќе ја препрати), го декриптира сесискиот клуч, nonce-от, lifetime и името на примачот и проверува дали nonce-от добиен од KDC е ист со оној генериран од објектот, а истото се прави и со името на примачот. Ако се совпаѓаат, се проверува дали добиениот lifetime е свеж, односно од нова инстанца на ZonedDateTime со моменталното време се одзема lifetime преку Duration.between(). Ако времето е позитивна вредност и помала од 30 min тогаш верификацијата е успешна и може да се изгенерира timestamp на ист начин како и lifetime со дополнителни 30 min важност.

Сега користејќи го сесискиот клуч, се енкриптира timestamp и името на испраќачот, се ставаат во листа низите и заедно со листата од KDC наменета за примачот се праќа до Bob, а се иницијализира приватната членка session\_key на добиениот сесиски клуч..

1. Интеракција испраќач-примач

Преку Bob се повикува методата receiver\_parse\_and\_verify() која ги отпакува двете листи, го добива сесискиот клуч, името на испраќачот според KDC и lifetime со декрипција со неговиот клуч, а со сесискиот дополнително ги добива timestamp и lifetime на испраќачот според испраќачот. Се верифицира еднаквоста на името на испраќачот, на ист начин како и верификацијата на lifetime кај испраќачот се верифицираат lifetime и timestamp. Ако верификацијата е успешна, се иницијализира session\_key. Она што останува е да се започне разговор за чии цели пораката Hello со повикување на send\_message() Alice ја шифрира со сесискиот клуч. Добиената енкриптирана порака со методата receive\_message() повикана преку Bob се дешифрира.

1. Пример излез од програмата



