Документација за домашна работа

Интегритет на пораки

Имплементацијата на CCMP протоколот користен во IEEE 802.11i се состои од 4 класи напишани во програмскиот јазик Java. Во класата "CCMPImplementation" се наоѓа main методот во кој се инстанцираат објекти од класата Sender и Receiver и исто така се генерира и 128 битен симетричен клуч (клуч кој ќе се користи и за енкрипција и за декрипција на рамките).

Класата CCMPFrameNE претставува рамка која не е енкриптирана и во неа се чуваат податоци за бројот на пакетот (PN) кој го составува CCMP хедерот. Содржи и податоци кој го составуваат хедерот на рамката (Frame header), односно изворна MAC адреса, дестинациска MAC адреса, qos. Потоа ги содржи и податоците кој ќе треба да се енкриптираат и MIC кој ќе се генерира во оваа класа и служи за одржување на интегритетот на пораката, односно да се утврди дека врз пакетот не е извршена никаква неавторизирана промена.

Со методот generateNonce се генерира nonce така што се земаат 48 бита од pn (Packet Number), 48 бита од sourceAddress (изворна MAC адреса) и 8 бита од qos (Quality of service), сите овие податоци се спојуваат заедно и се добива 104 бита nonce кој е неопходен за генерирање на MIC како и енкрипција на податоците. Потоа во зависност од тоа во кој режим на работа се користи AES можно е останатите 24 бита да се надополнат со нули или да се надополнат со 24 битен бројач.

Со методот generateMIC се генерира вредноста за MIC така што, прво се спојуваат дестинациската адреса, изворната адреса, бројот на пакетот и data во една низа од бајти input и со помош на режимот Cipher-Block Chaining на алгоритамот AES од оваа низа се генерира MIC. Процесот започнува така што:

1. Се земаат 128 бита од низата и се стават во друга низа од бајти block
2. Потоа врз низите block и nonce се применува XOR операцијата и излезот се користи како влез во алгоритамот AES (се енкриптира блокот со приватниот клуч генерирани во main методот)
3. Врз излезот од AES и врз нови 128 бита од Input низата се применува XOR чиј што резултат повторно се става како влез во алгоритамот AES.
4. Чекорот 3 се повторува се додека да се измине цела input низа

Кога ќе се измине цела низа, од последниот блок кој ќе биде енкриптиран со AES се земаат првите 64 бита и тие ќе служат како вредност за MIC.

Самата рамка се енкриптира во втората класа (CCMPFrameE) каде се чуваат истите податоци (непроменети, source/destination address, QoS, PN-incremented). Конструкторот на оваа класа прима CCMPFrameNE објект од кој ги зема непроменетите податоци. Ја зема пораката и ја енкриптира. Процесот на енкрипција се одвива во методот encrypt кој го користи Counter mode на AES. Работи така што на 104 битен nonce му додава 24 битен бројач кој започнува од 1, nonceCounter (бројач + nonce) го енкриптира со помош на клучот и на резултатот му прави XOR со блок од пораката (128 bits), така прави се' до крај на пораката, притоа зголемувајќи го бројачот за 1 со секој нареден блок. Накрајот и самиот MIC се енкрипртира на истиот начин во методот encryptM, само што бројачот е 0.

Со класите Sender и Receiver се симулира праќањето и примањето на пакетите така што во main методот прво се инстанцира објект од Sender класата на кој му се испраќа клучот за енкрипција како параметар и другите податоци неопходни за создавање на една рамка преку конструкторот, потоа се креира и објект од класата Receiver кој како аргумент во конструкторот го прима само клучот. Потоа се повикува методот send од Sender класата кој започнува со испраќање на рамката, а во процесот на испраќање прво се енкриптира рамката па потоа се испраќа до примачот кој е испратен како аргумент во методот send и во примачот се извршува декрипцијата на рамката и споредба на енкриптираниот MIC и пресметаниот MIC на добиената рамка, доколку овие се совпаѓаат се прифаќа рамката, а доколку не се совпаѓаат тогаш рамката се отфрла бидејќи се смета дека нејзиниот интегритет е нарушен. Во примачот има метод за декрипција decrypt во кој се одвива декрипцијата така што се одвива во обратна насока од енкрипцијата. Прво се зема nonce и бројач иницијализиран на 1 и се соединуваат во една заедничка низа од бајти наречена nonceCounter оваа низа се енкриптира со AES со помош на клучот и на резултатот му се прави XOR со 128 битен блок на податоци од енкриптираните податоци и резултатот се става во низа од бајти која ќе биде резултатот од декрипцијата. Ова се повторува со тоа nonceCounter со секој нареден блок се инкрементира за 1 се додека има податоци за декрипција. На исти начин се декриптира и MIC единствена разлика е тоа што се започнува со бројач сетиран на 0.