Tik-79.159 Tiedon salaus ja suojaus	
Tentin 19.11.1996 ratkaisut	Tuomas Aura

1. (a) Heikoilla avaimilla salatut sanomat ovat helpommin avattavissa kuin muilla avaimilla salatut. Pahimmassa tapauksessa salakielinen teksti on sama kuin selväkielinen. Hyvässä salausalgoritmissa heikkoja avaimia on vähän tai ei lainkaan.

   (b) Salauksessaa diffuusion tehtävänä on levittää kunkin selväkielisen bitin vaikutus laajalle alueelle salakieliseen tekstiin. Lohkosalauksessa diffuusio saadaan aikaan permutoimalla lohkon bittejä.

   (c) Message Authentication Code MAC on symmetriseen menetelmään perustuva allekirjoitus. Sanoma allekirjoitetaan ja sen alkuperä tarkistetaaan saman salaisen avaimen avulla. MAC-algoritmi voidaan konstruoida esim. yksisuuntaisesta hajautusfunktiosta.

   (d) Informaatioteoreettisen salausmenetelmän turvallisuus perustuu siihen, ettei pelkässä salakielisessä sanomassa ole riittävästi tietoa avaamiseen. Sanomaa ei voi avata ilman avainta, vaikka hyökkääjällä olisi käytössä rajaton laskentakapasiteetti. Esim. *one-time-pad*.

   (e) Bittivirtasalauksessa salattava bittijono xorrataan vahvalla pseudosatunnaislukugeneraattorilla tuotetun bittijonon kanssa. Salauksen avaaminen tapahtuu täsmälleen samalla tavalla. Lähettäjän ja vastaanottajan on alustettava satunnaislukugeneraattorinsa samalla avaimella. Tarvittaessa voidaan salaus suorittaa vaikka yksi bitti kerrallaan.

   (f) Subliminaalinen kanava tarkoittaa viestintätapaa, jota käytettäessä ulkopuolinen taho ei voi edes tietää, että viestejä lähetetään. Sanomat siis piilotetaan.

2. (a) Tehtävään jäi vahingossa vääriä lukuarvoja. $31 \cdot 43 = 1333 \neq 1189$. Jos $N = 1333$, $e = 33$ ei ole kelvollinen avain, koska $\gcd(\varphi(N), e) \neq 1$. Toisaalta $1234 > 1189$, joten allekirjoitus pitäisi tehdä useammassa osassa.

   Arvostelu: Olen pahoillani tästä mokasta. Pisteet sai joko kertomalla, mikä virhe tehtävässä on, tai laskemalla tehtävän oikein muutetuilla lukuarvoilla.

   (b) Ks. (a)-kohta.

   (c) Jukinen avain d (17) vai olla hyvinkin lyhyt, esim. juuri 17. Enintään se voi olla moduulin suuruinen. Moduulin N (1189) pitää olla vähintään 512 bittiä, mielellään enemmän. Hajautusfunktion arvo (321) on luultavasti 128 tai 160 bittinen.

3. (a) Lohkosalakirjoituksen blokin pituus (64 bittiä) pitää olla niin pitkä, ettei tunnetuista salakieli-selväkielipareista ole mahdollista koota sanakirjaa. Toisaalta lohkon tulisi olla lyhyt, jotta tiedon diffuusio lohkossa saadaan mahdollisimman täydelliseksi vähillä algoritmin kierroksilla. 64 bittiä on kahden potenssina sopiva valinta tietokonetoteutusta ajatellen.

   Kryptografiselta hajautusfunktiolta vaaditaan yksisuuntaisuus ja (laskennallinen) konfliktittomuus. Yksisuuntaisuuden takia funktion arvon pitää olla niin pitkä, ettei käänteisfunktion arvoa ole mahdollista löytää kokeilemalla. Tähän riittäisi esim. 80-bittinen hajautusarvo. Syntymäpäivähyökkäyksen takia konfliktittomuuteen vaaditaan kuitenkin enemmän bittejä. Syntymäpäivähyökkäys n-bittisiä avaimia vastaan onnistuu, jos pystytään tallettamaan noin $2^{n/2}$ argumentti-hajautusarvoparia. Muistivaatimus on samaa luokkaa kuin lohkosalauksen sanakirjahyökkäys, kun hajautusarvon pituus on suunnilleen kaksi kertaa salauslohkon pituudelta vaadittu 64 bittiä eli 128 bittiä.

(b) Jos hyökkääjällä on käytössään rajaton muistikapasiteetti, on mahdollista kerätä tunnetuista salakieli-selväkielipareista sanakirjaa. Tämä on kohtalokasta ainakin ECB-moodissa tapahtuvalle salaukselle. Jos valitun selväkielen hyökkäys on mahdollinen, sanakirjan kokoaminen tehostuu.

Lisäksi jos salausavaimen pituus on pieni (ehkä $<$ 72 bittiä), on hyökkääjän mahdollista salata sama selväkielinen lohko kaikilla avaimilla ja verrata kuulemiaan salakielisiä tekstejä näihin lohkoihin. Jos valitun selväkielen hyökkäys on mahdollinen, avain selviää välittömästi. Muussa tapauksessa hyökkääjän on valittava jokin usein käytetty samoma tai sanoman osa ja odotettava sen seuraavaa esiintymistä.

Kaksin- ja kolminkertaista salausta (ja muitakin algoritmejä) vastaan suunnattu *meet-in-the-middle*-hyökkäys vaatii yleensä suuren määrän muistia, joten osa näistä hyökkäyksistä tulee mahdolliseksi.

Näissä hyökkäyksissä ja monissa muissakin rajaton muistikapasiteetti mahdollistaa laskennan tekemisen etu- tai jälkikäteen ja siten vähentää laskennan kustannuksia.

4. Alice generoi kaksi satunnaislukua R_1 ja R_2. Hän laskee ja lähettää Bobille hajautusarvot $SHA(\text{"kruuna"}, R1)$ ja $SHA(\text{"klaava"}, R2)$ *satunnaisessa järjestyksessä.* Bob valitsee arvoista *satunnaisesti* toisen ja lähettää sen takaisin Alicelle. Tämän jälkeen Alice pajastaa Bobille *molemmat* satunnaisluvut R_1 ja R_2. Bob laskee hajautusarvot ja tarkistaa, että ne ovat samoja kuin Alicen lähettämät.

   Bob ei pysty erottamaan sanomia toisistaan, joten hän joutuu valitsemaan sarunnaisesti joko kruunan tai klaaavan. Alice ei voi huijata, koska Bob tarkistaa lopuksi sanomien sisällön. (Protokolla ei ole reilu siinä mielessä, että Alice voi jättää viimeisen sanoman lähettämättä, jos huomaa häviävänsä.)

   Monia muitakin vaihtoehtoja on. Kaikissa ensin lähetetään dataa vastapuolelle muodossa, jota tämä ei ymmärrä. Kun molemman osapuolet ovat tehneet satunnaisen valintansa, paljastetaan datan sisältö niin, että toinen voi tarkistaa toisen alkuperäisen valinnan.

5. Kaksinkertaista salausta vastaan on olemassa *meet-in-the-middle*-hyökkäys, joka vaatii kaksi tunnettua salakieli-selväkieliparia, 2^n (avaimen pituus n bittiä) lohkoa muistia ja 2^{n+1} salaus- tai avausoperaatiota. Vastaava hyökkäys kahdella avaimella EDE-moodissa tehtävää kolminkertaista salausta vastaan vaatii 2^{2n} salaus- tai avausoperaatiota, joten se ei ole merkittävä uhka. EDE-salausta vastaan kuitenkin on samantaipainen *meet-in-the-middle*-hyökkäys, joka vaatii suurin piirtein saman ajan ja muistin kuin hyökkäys kaksinkertaista salausta vastaan, mutta luokkaa 2^n *valittua* selväkielistä tekstiä. Niinpä kaksinkertainen salaus ei ole vaivaan nähden merkittävästi parempi kuin yksinkertainen, eikä sitä ole syytä käyttää. Kolminkertainen salaus sen sijaan on selvästi parempi. Jos massiivinen valitun selväkielisen testin hyökkäys on mahdoton, kolminkertainen salaus kahdella avaimella todella pidentää avainpituutta luvattujen bittimäärien mukaisesti. Hyökkäysten yksityiskohdat on selitetty Schneierissa sivuilla 357–359.

   Arvostelu: Tehtävä oli osattu todella huonosti. *Meet-in-the-middle*-hyökkäyksen mainitsemisesta ja mekanismeista sai pisteitä. Samoin vertailevista vahvuusarvioista.

6. Diffie–Hellman-avaintenvaihdosta voi kertoa toimintaperiaatteen lisäksi monenlaisia asioita, joita löytyy oppikirjasta, luennoilta ja laskuharjoituksista sekä kirjallisuudesta.

   Arvostelu: Tarkoituksesta ja toimintaperiaatteesta sai 2 pistettä. Diskreetti logaritmi ja *man-in-the-middle* toivat 2 lisää. Mistä tahansa muista tarkennuksista, huomioista tai laajennuksista sai myös pisteitä. Diffie-Hellmanin tulosta ei muuten käytetä suoraan salausavaimena, vaan se on ns. avainmateriaalia, josta avain voidaan tuottaa esim. yksisuuntaisella hajautusfunktiolla.