IN1020 - Kryptografi og operativsystemer: Øvingsoppgaver gruppetime 4 (Uke 38)

1. Praktisk anvendelse av kryptografi

   (a) I hvilke situasjoner og til hvilke formål kan kryptografi benyttes til å ivareta konfidensialitet til informasjon/data?

   (b) Hvilke andre sikkerhetsmål kan understøttes ved bruk av kryptografi? Forklar kort.

   (c) Bruker du kryptografi i hverdagen? Identifiser og diskuter.

2. Nøkkel og nøkkeldistribusjon

   (a) Hva er en kryptografisk nøkkel?

   (b) Nevn de tre hovedtypene av kryptografiske nøkler, og for hver av dem forklar kort hvilke *sikkerhetsmål* som er viktig å ivareta for å beskytte de ulike typene nøkler.

   (c) Beskriv kort hovedformålet med *offentlig-nøkkel sertifikater* og *offentlig-nøkkel infrastruktur (PKI)*, og hvilke utfordringer disse løser.

3. Symmetrisk kryptering

   Som vi vet fungerer symmetrisk kryptering litt som en dagbok med en lås. Du bruker en nøkkel til å låse ned innholdet i dagboken og den samme nøkkelen brukes til å få tilgang til innholdet igjen.

   Det finnes en myriade av forskjellige måter for hvordan utveksle data i dag. Noen er «fysiske» mens andre foregår trådløst. Tenk deg at du skal sende data som er kryptert symmetrisk via disse ulike kanalene. Diskuter hver av dem. Hvilke utfordringer de byr på og hvordan dere vil sørget for at ikke uvedkommende får tilgang til informasjonen?

   (a) Radio

   (b) CD/DVD

   (c) USB

   (d) E-post

4. Asymmetrisk kryptering

   I asymmetrisk kryptering har man et nøkkelpar, der én nøkkel brukes til å kryptere og én brukes til å dekryptere. Som regel er en av disse hemmelige, mens den andre er offentlig. Hvem har den offentlige nøkkelen og hvem har den hemmelige nøkkelen i disse scenarioene?

   (a) Du vil sende et hemmelig dokument til en kollega.

   (b) Du benytter digital signatur i et dokument du sender til en kollega.

   (c) Hvilke sikkerhetsmål blir ivaretatt i oppgave a og b?

5. Hash-algoritmer (sjekksumalgoritmer) Hash-algoritmer (også omtalt som hash-funksjoner og sjekksumalgoritmer) gjør kryptering uten nøkler. Hva er de tre vikiggste egenskapene til en god hash-algoritme?

6. Eksamensoppgave 3.5 og 3.6 fra Eksamen i IN1020 H2023

7. Case-oppgave (Hash-kryptering/sjekksumalgoritme)

   (a) Du jobber som utvikler og teamet ditt er i ferd med å bygge en nettside. Besøkende på nettsiden kan registrere seg med e-post og lage et passord for å opprette en brukerprofil på siden. Teamet ditt har forstått at det er en dårlig idé å lagre passordene til brukerne i klartekst, så dere vil i stedet benytte en hash-algoritme for å kryptere passordene. Er dette en god idé? Hvilke svakheter kan dere identifisere med denne strategien?

   (b) Hvilke sikkerhetsmål er hash-algoritmer med på å ivareta når passord hashes som omtalt her?

   (c) For den interesserte: I forbindelse med hash-kryptering nevnes ofte «message-digest» eller «hash-digest». Hvorfor brukes dette som uttrykk?

8. Operativsystemer:

   (a) Hvorfor har vi operativsystemer?

   (b) Hvor finner vi operativsystemer?

   (c) Hvilke to perspektiver bruker vi for hva et operativsystem er?

9. Hvilke hovedkomponenter finner man i et operativsystem?

10. Diskuter hvilke mekanismer et operativsystem bruker for å ivareta sikkerhet, og sørge for at et program ikke får tilgang til data fra andre programmer?

Praktiske oppgaver

11. Sjekksumalgoritme (hash-funksjon) – praktisk oppgave

    Gjennomføres på Ifis linux-maskiner.

    (a) Programmet `sha256sum` genererer en sjekksum av en fil basert på algoritmen SHA256. Lag en liten tekstfil med innhold, kall den f.eks. `abc.txt`. Kjør så programmet sha256sum:

    ```
    [user@vestur]>sha256sum abc.txt
    [6ee0c32c675ce6d3bd3f6e326c81e45b3d6675c29c0c9ced1398684a667804e9 abc.txt
    ```

    Gjør endringer i fila `abc.txt`, og kjør programmet en gang til. Er nøkkelen den samme?

    (b) Hvordan kan sjekksumalgoritmer bidra til å sikre dataintegritet?

    (c) Finn og diskuter situasjoner/eksempler hvor bruk av sjekksumalgoritme kan være nyttig for deg.

12. SSL – praktisk oppgave

    SSL Labs er et forskningssamarbeid som jobber for økt bevissthet rundt bruk av SSL. De tilbyr også en online-tjeneste for å teste nettlesere og web-tjenere med tanke på hvor godt (sikkert) SSL er implementert og konfigurert i den enkelte nettleser/tjener.

    (a) Bruk https://www.ssllabs.com/for å sjekke nettleseren din. Nettleser på Ifi, bærbar PC, mobil, nettbrett, etc. Er den sårbar? Hvilke protokoller støtter den?

    (b) Bruk https://www.ssllabs.com/ for å sjekke kjente web-tjeneres sertifikat og konfigurasjon. Finner du sårbarheter? Kan du si noe om sertifikat, nøkkel samt protokoll(er) den enkelte tjener støtter?

    (c) Ta også en kikk på https://www.ssllabs.com/ssl-pulse/