Løsningsforslag til øvingsoppgaver uke 45

Oppgave 1 Praktisk oppgave uten løsningsforslag.

Oppgave 2 Praktisk oppgave uten løsningsforslag.

Oppgave 3

a) og b) Praktiske oppgaver uten løsningsforslag, men:

- Kommandoen `ls -l <mappenavn>` viser deg innholdet i mappen, inkludert hvilke rettighetsbit som er satt på en fil eller undermappe, hvilken bruker som eier de ulike filer og mapper, osv.
- Kommandoen `ls -ld <mappenavn>` viser deg informasjon om mappen selv, dvs. hvilke rettighetsbit som er satt og hvilken bruker som eier mappen.

Noen observasjoner:

- Du kan ikke se (lese) innholdet i en mappe uten å lesetilgang (r-bit satt).
- Hvis eksekveringsrettigheter (x-bit) ikke er satt på en mappe kan du ikke eksekvere innholdet i den, dvs. heller ikke se innholdet eller lese en fil som ligger i mappen (uavhengig av hvilke retigheter som er satt på fila).
- Du må ha skrivetilgang (w-bit satt) for å kunne skrive i en mappe.

c)

Konfidensialitet: Man kan bidra til å sikre konfidensialitet ved at kun de brukerne som skal ha tilgang gis tilgang. Dette kan styres ved å sette lesetilgang (r-bitet) riktig.
Ved å kun sette eksekvering-tilgang (x-bit) på en mappe kan heller ikke innholdet i mappen listes, mens en fil som ligger i mappen fint kan både leses og eksekveres (bruker må dog kjenne full sti (path) til fila). Dvs. at kun de som har fått kjennskap til full sti har tilgang (skjule informasjon, Security by obscurity). Dette kan være nyttig hvis man ønsker å gi en bruker tilgang til enkeltelementer (en bestemt fil/et program), samtidig som man fortsatt vil skjule alt annet som ligger under samme mappe.

Dataintegritet: Ved å styre og begrense hvem som kan skrive til mapper og filer i filsystemet bidrar man til å sikre at uvedkommende ikke kan endre innholdet i filer.

Oppgave 4

a) Nei, sjekksummen blir en annen når fila endres. Dette er en grunnleggende egenskap ved sjekksumalgoritmer. Men observer gjerne at sjekksummen blir den samme når du bruker programmet sha256sum på samme fil flere ganger.

b) Man kan sammenligne sjekksum av en datafil opp mot en kjent sjekksum av denne fila. Er den lik? Isåfall er fila ikke endret mellom første og andregangskjøring av sjekksum.

c) Sjekksumalgoritmer brukes ofte ved deling av programvare (f.eks. nedlasting fra nett). Tilbyderen av programvare genererer en sjekksum av orignalen (ofte et filarkiv (f.eks. .zip-fil), som publiseres sammen med programvaren. Brukere som laster ned programvare kan enkelt selv kjøre en sjekksumalgoritme etter nedlasting, og kan med det finne ut om programvaren det den utgir seg for å være, eller er den kanskje endret etter at sjekksummen ble generert? Vil også kunne avdekke hvorvidt en nedlasting har gått greit eller ikke (pakketap?). Helt konkret eksempel: `https://www.postgresql.org/ftp/source/v9.6.6/`, nedlasting av PostgreSQL source-kode. Her finner du både programvaren og tilhørende sjekksummer.

Oppgave 5

a) Ved digital signatur benytter avsender sin private nøkkel til kryptering, mens mottager benytter avsenders offentlig nøkkel til dekryptering.

b) Ved kryptering av innhold i en melding benytter avsender mottages offentlige nøkkel til kryptering av meldingen, hvorpå meldingen så kun kan dekrypteres ved å benytte mottagers private nøkkel.

Oppgave 6

a) Konfidensialitet

Trussel: Lekkasje eller tyveri av informasjon som skal hemmeligholdes. Bedrifshemmeligheter, personinformasjon. Hva med anonymitet? Noen finner ut hvem som er involvert/engasjert i hva.
Tiltak: Kryptering (av data (disker?), kommunikasjon, osv). Tilgangskontroll. Perimeterforsvar/skallsikring. Unngå mennskelige feil. Men krever også gode policy for hvordan data skal håndteres og hvordan kommunikasjon skal foregå. Det hjelper lite med sikker lagring hvis en ansatt likevel legger en kopi av en sensitiv fil på Google Drive.

b) Integritet

Trussel: Hovedtrusselen er korrupsjon av data og systemer. Har noen endret innholdet i en datafil? Har noen endret konfigurasjonen for et system? Hvordan kan vi med sikkerhet vite at noen ikke har gjort endringer?
Tiltak: Sjekksumalgoritmer (kryptografisk integritetssjekk), kryptering, endring- og tilgangskontroll, perimeterforsvar (skallsikring), digital signering også av programvare (code signing).

c) Tilgjengelighet

Trussel: (D)DoS-angrep, som fører til at tjenester utilgjengeliggjøres eller forsinkes for de som rettmessig har tilgang. Systemfeil er også en trussel, og forekommer nok oftere enn DDoS-angrep.
Tiltak: Redundans for kritiske tjenester/ressurser. I tillegg ha failover management", dvs. gode prosedyrer for å enkelt kunne bytte til redundante eller standby tjenester/systemer ved feil. Sikkerhetskopi av filer, systemer og konfigurasjoner er selvsagt. Ha også rutiner for hvordan hendelser skal håndteres.

Oppgave 7 Ingen fasit, men se hvilken kriterier som er satt, samt test ulike passord på `https://passord.uio.no/changePassword`. Hva skal til for å oppnå en passordstyrke på 10?

Oppgave 8 Punkter som må vurderes:

Alle sider av informasjonssikkerhet (konfidensialitet, integritet og tilgjengelighet) må vurderes.

Skal alle alltid ha tilgang til all informasjon selv om man jobber på samme prosjekt?

Hvis alle kan bruke en ulåst pc vet man ikke hvem som har endret/lagt til/slettet informasjon. Hvilke konsekvenser vil dette få, og for hvem? (Arbeidsplassen, deg selv, andre brukere)

Kan du få problemer dersom en kollega bruker din pc til å utføre handlinger som ikke er tillatt? F.eks. laste ned filer ulovlig eller utført andre handlinger i strid med intern sikkerhetsinstruks.

Oppgave 9 Praktisk oppgave uten løsningsforslag.