Tjenester I internett og datateknologi og sikkerhet

Time 1:

Litt repetisjon av transportlaget... Transportlaget består bl.a av **TCP** og **UDP**, som er transportprotokoller. Førstnevnte er **tilkoblingsorientert**, som betyr at den oppretter en forbindelse med den/de den skal kommunisere med før den begynner å sende data. Dette gjøres via en three-way-handshake, som betyr at maskinene sender litt info frem og tilbake for å stadfeste at de er tilkoblet. I tillegg har TCP garantier for at all data leveres og i riktig rekkefølge. UDP er ikke tilkoblingsorientert og har ingen garantier for at data leveres riktig. Dette gjør at UDP er godt egnet for strømming og direkte broadcasting, siden disse krever rask flyt av data og effektiv bruk av bredbåndet.

Forbindelsesstrategier er ulike omstendigheter der data blir utvekslet. Vi har **"pull"** (dra), som er den vanligste strategien, der klienten ber om en tjeneste fra tjener. For eksempel da du trykket på linken til dette dokumentet. Det motsatte av dette er **"push"** (dytte), som er når tjener sender klienten data. En variant av push er **"publish-subscribe"**, som er ganske likt, men som innebærer at klientene er abonnenter av tjenesten de får dyttet på seg.

Litt repetisjon av aksessmodeller... aksessmodellene vi har snakket om er **klient-tjener** og **peer-to-peer**. Førstnevnte er den mest vanlige modellen. Det er når man har et klart skille mellom klienter (brukere) og tjenere, der klienter ofte ber om tjenester fra tjenere, som sender dem ut fra sine servere. Ulempen med dette er at nettet rundt tjeneren (kjernenettet) blir fort belastet når mange ber om tjenester derfra. En løsning på dette er et **CDN** (content delivery network), der man har flere lokale servere fordelt geografisk, slik at vi i Norge for eksempel slipper å be om tjenester helt fra USA. Disse lokale serverne får sine tjenester fra en source-server (kilde-tjener), slik at alle har samme data.

Proxy-cache er lagring som brukes av proxy-servere i et CDN. En **proxy-server** er et mellomledd mellom en klient og en source-server. Når man laster inn nettsider lastes disse ofte fra proxy-servere. Når proxy-serveren får en forespørsel om data sjekker den om den har dataen inne og hvor ny denne dataen er. Hvis den mangler data eller dataen er gammel laster den data fra

source-serveren og lagrer en kopi i proxy-cachen sin. En **forward proxy** er en "vanlig" proxy, som ligger nærmere klientene enn tjeneren. En **reverse proxy** er motsatt og ligger nærmere tjeneren.

Peer-to-peer har vi snakket litt om før. Du husker kanskje at det betyr at maskiner i denne aksessmodellen kan være både tjenere og klienter og at P2P ofte brukes til ulovlig fildeling. P2P er fint å bruke til fildeling og vil virke selv om det skjer endringer i nettverket. Det er derimot vanskelig å ha kontroll på alt som skjer og det kan være komplisert å finne en spesifikk ressurs. Særlig siden peers kan komme og gå. Vi kan nevne to typer av P2P. Vi har "ren" Peer-to-peer, som er det jeg snakket om her og **hybrid peer-to-peer**. Hybrid innebærer at den har noen likheter med klient-tjener. I stede for at *alle* noder er likeverdige har man noen **superpeers** som har litt mer funksjonalitet enn de andre. Disse kan for eksempel ha mer harddiskplass eller kraftigere CPU. Det kan også være et poeng at disse ikke står bak en brannmur og/eller har NAT. Disse funksjonene gjør at forsinkelsene som er forbundet med vanlig P2P blir redusert.

HTTP eller hypertext trasfer-protocol er en protokoll i applikasjonslaget. Den baserer seg på klient-tjener-modellen ved at klienter som spør om web-objekter får disse av tjenere. Http bruker TCP som transport og er **stateless** (sparer ikke på informasjon om tilstand).

HTTP-streaming (DASH) er en måte å streame på som sikrer kvaliteten bedre enn med UDP. I stede for å laste innhold som en kontinuerlig strøm lastes data som pakker på noen par sekunder av gangen. Hver av disse segmentene blir gjort tilgjengelig i ulik kvalitet (bit-rate), slik at klienter med ulik kapasitet kan streame innholdet mest mulig problemfritt. Klientene som mottar innholdet bruker en algoritme for å velge høyest mulig bit-rate som er mulig akkurat da.

Persistente/ikke-persistente forbindelse er ulike ved at persistente forbindelser holder oppe den opprettede forbindelsen etter at man har sendt over data (i tilfelle mer skal sendes), mens ikke-persistente lukker forbindelsen så snart data er ferdig sendt over. Hvis mye data skal sendes kan dette bli slitsomt, siden nye forbindelser må opprettes hver eneste gang.

Cookies tar vare på tilstander, slik som brukerautentisering, brukerpreferanser og annen aktivitet bruker gjør, slik som hvilke produkter bruker ser på og hvilke nettsider de kommer fra

og går til. Man kan samle veldig mye info om en person via cookies, som er potensielt problematisk med tanke på personvern.

Betinget GET er når det sjekkes om bruker har data i cache før det sendes eller sendes på nytt.

DNS prefetching er når man er på en nettside og klikker på en link. Da er ofte ip-adressen til denne linken allerede hentet via DNS.

E-post består av mail-klienter og mail-tjenere. Klientene bruker noe som kalles en MUA (message user agent), som er applikasjonen er man kan lese, skrive og sende e-poster. Mailtjenerne har klientens innkommende (uleste) eposter og utgående eposter lagret, så det når man skal lese en mail sender man egentlig en forespørsel til tjeneren om å få lese mailen sin. Epost-tjener bruker noe som kalles **SMTP (simple mail transfer protocol)** for behandre utgående og inngående mail. SMTP bruker så klart TCP som transport-protokoll. Mailer består av en header og en body. Headeren kan f.eks. ha avsender, mottaker og emne, mens body-en er selve mailen, som gjerne er ASCII-tegn.

MIME eller multipurpose internet mail extension brukes i multimedia-mailer. Altså mailer som inneholder mer enn bare tekst.

IMAP eller internet mail access protocol har en rekke med funksjoner, slik som å:

- Opprette, slette, bytte navn på epostmapper (på tjeneren)
- Se etter nye meldinger, fjerne meldinger sette og fjerne flagg
- Lese, søke og hente basert på søkeresultat
- Søke i innholdet i meldinger
- STORE and conditional STORE
- CATENATE

POP (post office protocol) er et mindre komplekst alternativ til IMAP, som byr på autorisering og nedlasting. Pop brukes gjerne i en egen applikasjon, mens IMAP brukes via browser. Hovedforskjellen mellom de to er at når POP laster ned mailer fra webserveren blir disse slettet fra serveren, mens IMAP synkroniserer med en webserver, som betyr at du kan bruke ulike maskiner til å sjekke mailen din. POP er nyttig hvis du har dårlig internett og ønsker å lese

mailer offline (som allerede er lastet inn), siden mailene ligger på maskinen din og ikke oppe i skyen.

Internet of things er samlebetegnelsen for devices fra dagliglivet som leverer ekstra tjenester via internett. Smartklokker, automatiske målere og enheter i smarthus er noen eksempler. Noen ganger er dette nyttig, mens andre ganger er det mer til bry eller kan by på sikkerhetstrusler ved at de mangler sikkerhetsoppdateringer eller ble solgt uten innebygget sikkerhet.

Time 2:

Når man tenker på sikkerhet i forbindelse med internett tenker man kanskje at farene for det meste befinner seg "ute" på nettet og ikke på din personlige datamaskin eller ditt lokale nettverk. Dette er ikke helt sant. I denne forelesningen ble det gjennomgått litt om hvordan brukere blir lurt eller angrepet via applikasjonsprogramvare og systemprogramvare.

SQL-injisering er en måte å hacke via en applikasjon. Hvis en applikasjon ber om input kan man legge inn en kode-setning, som systemet til applikasjonen vil kjøre. Dette kan brukes til å for eksempel slette data fra systemet, slik som i eksemplet fra den svenske valgkampen.

XSS eller cross site scripting er når en angriper plasserer kode, som brukerinput på en side, som lagres hos web-tjeneren. En annen bruker kan bli rammet av denne koden, som betyr at angriperen nå har kontroll over brukerens maskin eller data. Dette kan for eksempel skje via en gjestebok eller et kommentarfelt på en nettside.

Phishing er når svindlere fisker etter info ved å prøve å lure deg til å gi den bort. Du kan få en epost om at du må logge deg inn hos en tjeneste for å rette opp i en feil eller for å få en gevinst. Som regel med en link til innlogging. Hvis du følger denne linken får du ofte opp en nettside som ser legit ut, men hvis du skriver inn informasjonen din her blir den sendt til svindlerne. Noen ganger ser man at det er svindel lett (google translate, mistenkelig URL... e.l.), men ofte ser alt helt vanlig ut. Derfor er det ofte lurt å ikke trykke på linken i eposten, men heller gå inn manuelt via web-browseren din, til nettstedet det er snakk om og sjekke om det faktisk var en autentisk mail.

DNS-sikkerhet er også noe å tenke på. En DNS-server kan bli utnyttet ved at en uvedkommende legger inn feil IP knyttet til et domenenavn. I tillegg har vi **man in the middle-angrep** som er når en hacker plasserer seg imellom to endepunkter i en dataoverføring og tyvlytter på all data som sendes fra ende til ende. For de to som kommuniserer virker alt helt normalt, som om de kommuniserer direkte med hverandre, men i virkeligheten strømmer all data gjennom hackeren først. Hackeren kan også sende data, som for de to partene vil virke som legit, ufarlig data, men som egentlig er skadevare.

Maskinvare kan også hackes. Mange komponenter i datamaskiner i dag er så små at du og jeg ikke har sjanse for å oppdage at noe er galt hvis vi åpner enhetene våre. Uvedkommende enheter kan brukes til å gjøre maskinen din mer sårbar for angrep.

En digital sårbarhet er en feil i et system. Det kan være en svakhet i systemet eller en feil som er lett å utnytte. Digitale sårbarheter kan være kjente av leverandørene, eller ukjente, som gjør at utnyttelse av disse sårbarhetene kommer som en overaskelse. En **Nulldags-sårbarhet** er en sårbarhet som ennå ikke er kjent for leverandøren av systemet. Den kan altså være kjent av angripere, som gjør at man helst burde rette opp i den fortest mulig. En **angrepsvektor** er den konkrete metoden en angriper velger for å utføre et angrep på en datamaskin, et nettverk eller et system, mens en **exploit** er en konkret utnyttelse av en sårbarhet.

Skadevare er programvare som har som mål å ødelegge for eller utnytte personen som kjører den. Skadevare kan være et eget program eller en bestanddel av et program. Skadevare vil ofte prøve å spre seg til andre maskiner. Skadevare består som oftest av et **stridshode**, som er måten den infiserer et datasystem, en **spredning**, som er måten den sprer seg på og **nyttelast**, som er det konkrete den gjør.

Fra forelesning:

 Virus	Kjennetegnes ved at det sprer seg hver gang en fil med virus blir åpnet av bruker eller systemet.
 Dataorm	Skadevare som på egenhånd er i stand til å spre seg og finne nye ofre.
 Trojaner	Skadevare som utgir seg for å være et nyttig program.
Bakdør	En ubeskyttet og ukjent åpning inn i et datasystem, laget av en angriper.
 Spionvare	Programvare for å spionere på en bruker i håp om å få tak i f.eks. informasjon, passord, kontoinformasjon.
 Tastelogger (Keylogger)	En enhet eller programvare som er koblet til tastatur eller maskin og som lagrer eller videresender alle tastetrykk. En variant av spionvare.
Rootkit	Skadevare som endrer eller erstatter funksjoner i OS eller applikasjoner, med mål å ta kontroll over en maskin eller et system.
Exploit	Et program, en sekvens med data, eller en oppskrift på hvordan en sårbarhet kan utnyttes.

En **buffer-overskridelse** skjer når mer data enn det det er plass til blir lagt inn i en buffer. Hvis dette gjøres ubevisst kan det føre til at programmer kræsjer eller at data blir korupte. Bevisst utnyttelse av buffer-overskridelse kan også føre til dette, men kan også brukes til å overskrive data og til og med legge inn egen programkode med skadevare.

En **TPM** (trusted platform module) sjekker om ting kjører som forventet når du starter PC-en din. TPM bruker kryptografi til å sjekke integritet og autentisering av programvare som starter.

Å vite om alt dette bidrar til **sikkerhetskultur**. Det kan riktignok sies at mer bevisste brukere fører til mer kompliserte angrep, siden hackere ofte ikke lever under en stein, men er klare over folks holdninger til datasikkerhet.