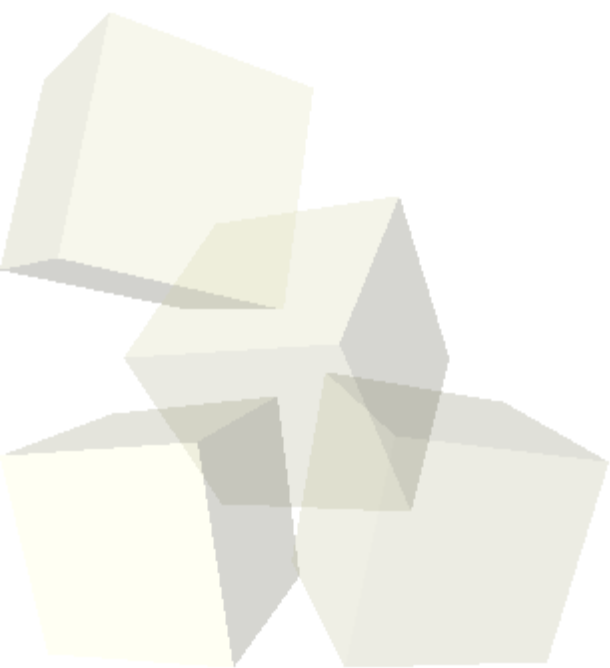




THROUGH THE FIREWALL

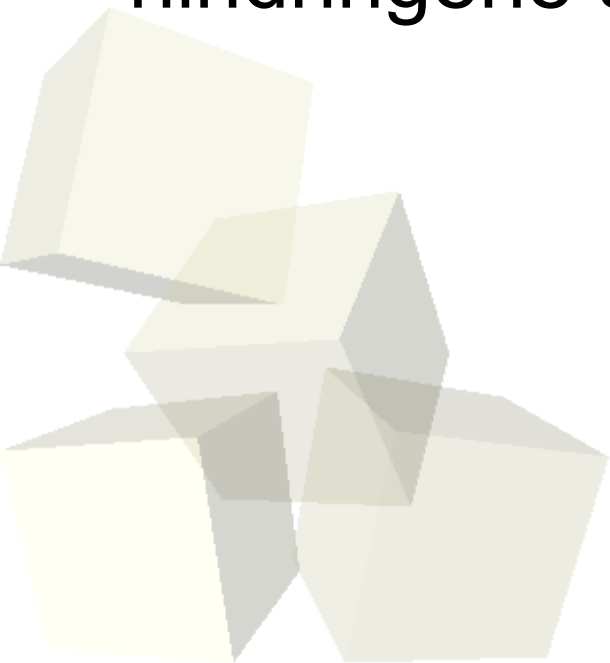
KAPITTEL 16
BUILDING SECURE SOFTWARE
INF329, HØST 2005
Isabel Maldonado
st10900@student.uib.no





Innledning

- Kort om firewall
- Hva er det som foresaker at en brannmur blokkerer en applikasjonen ?
- Hvordan designe applikasjoner slik at de går rundt hindringene som brannmuren setter?



Introduksjon til brannmur

- Plasseres mellom det lokale nettverket og det eksterne nettverket. Kun valgt datatrafikk kommer inn og ut av det interne nettverket.
- Skaper hodebry for utviklere av internett-aplikasjoner da det ikke fins en standard brannmur.
 - ♦ Applikasjoner som kjøres over nettverket og skal installeres må vurdere innvirkningen av de ulike brannmur-strategiene.
 - ♦ System administratorene ønsker å beskytte brukere fra å kjøre ondsinnet kode, mens applikasjonsutviklerne vil la brukeren kjøre enhver applikasjon de ønsker.



Litt mer om brannmurer

- Kontroll og filtrering av innkommende og utgående trafikk tvinger frem en access control policy.
 - ◆ Dette skal hindre angripere i å finne feil i ditt nettverk.
 - ◆ Stoppe Trojanske hester
 - ◆ Stoppe nedlasting av applikasjoner som kun er ondsinnet kode.
- To måter å kontrollere utgående trafikk:
 - ◆ Port-basert pakkefiltrering
 - Pakker som skal til bestemte porter, slipper gjennom
 - ◆ Applikasjons-proxy
 - Ingen trafikk slippes gjennom direkte. I stedet har proxyen et grensesnitt for utvalgte applikasjoner, dvs. den imiterer serveren for klienten og imiterer klienten for serveren.
 - Mer sikkerhet enn vanlig pakkefiltrering.
 - En proxy er skrevet for å være robust; enkel kode.



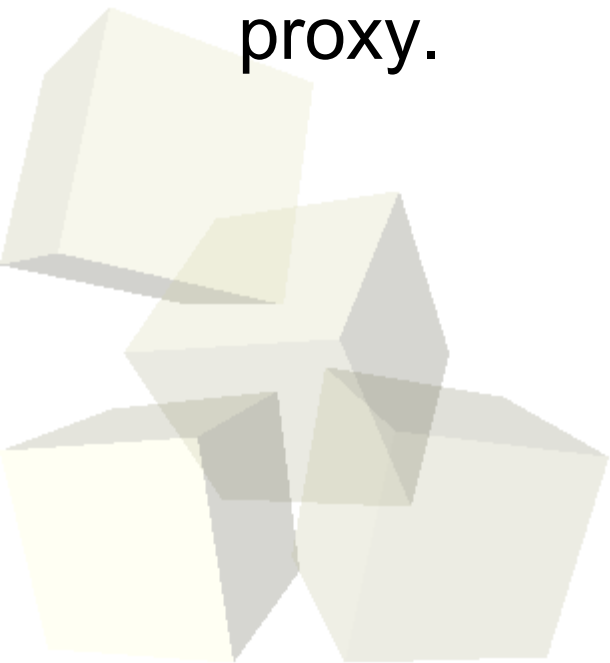
Gå rundt brannmuren

- Vanlig å la servere bruke HTTP-porten (port 80) for å få trafikken gjennom
 - ♦ Teorien går ut på at alle lar utgående HTTP-trafikk slippe gjennom.
 - ♦ Vil ikke fungere med en applikasjonsproxy, da HTTP-proxy vanligvis er på andre porter enn port 80.
 - ♦ Selv om det er en proxy på port 80, vil denne passe på at trafikken følger HTTP-protokollen, så du blir nødt til å pakke tilkoblingene inn i HTTP-forespørsler og svar. Dette kan være en stor utfordring for enkelte applikasjoner.
- Bruk port 443 (HTTPS-porten)
 - ♦ Dette er en mer universell løsning og er mindre utsatt for trafikkanalyse.



Gå rundt brannmuren

- Unngå server-inisialiserte tilkoblinger når du utvikler protokoller for en klient/server applikasjon.
 - ♦ Pakkefilteret vil stoppe innkommende pakker, siden den ikke kan vite at klienten har godtatt åpning av en gitt port.
 - ♦ Kan fungere med en proxy; du kan enten lage en selv eller bruke en generisk (slik som SOCKS). Men ofte vil ikke bedrifter med pakkefilter godta installasjon av proxy.





Klient proxy

- For å støtte alle potensielle brukere, må applikasjonen fungere bra sammen med proxyene.
- Skrive sin egen proxy server til egne applikasjoner.
 - ♦ Pass på å respektere ønskene til paranoide netverksadministratorer.
 - ♦ Gjør den så enkel som mulig. Det skal være lett for folk som gjennomgår koden å sette seg inn i den, og bestemme om den er god nok til å stole på. Den bør faktisk være så enkel at den ikke trenger kommentarer.
 - ♦ Design koden slik at den ikke trenger spesielle rettigheter, la porten være konfigurert og standard over 1024.
 - ♦ Bør støtte autentisering og access control.
 - ♦ Den skal aldri krasje, uansett grunn!



SOCKS

- Den største ulempen med proxy-baserte brannmurer er at det er vanskelig å få nye applikasjoner til å fungere med dem. Vi trenger en mer generell løsning, SOCKS.
- En SOCKS-server lytter på en spesifikk port, klienter kobler til denne porten og gir serveren tilkoblingsinformasjon. Serveren vil da åpne tilkobling til ønsket maskin.
- Det finnes biblioteker som gir en applikasjon SOCKS-støtte uten å forandre på koden; de bytter bare ut standard nettverkskall med tilsvarende SOCKS-kall.



SOCKS

- SOCKS støtter per idag ikke kryptering, bare autentisering (noe som er nytteløst uten kryptering).
- SOCKS er på mange måter utdatert, og har stort sett blitt byttet ut med NAT (Network Address Translation). NAT tilbyr samme funksjonalitet som SOCKS, men fremstår som en router for maskiner i det interne nettverket. Klient-applikasjoner trenger derfor ikke vite (og da heller ikke ta hensyn til) at de er bak en brannmur.



Konklusjon

- Vi må alltid foreta en avveining mellom sikkerhet og brukervennlighet. En brannmur øker sikkerheten, men er ofte grunnen til at applikasjoner ikke fungerer som ønsket.
- Å få en applikasjon til å fungere uansett hvilken brannmur som benyttes er en stor utfordring.
- For en klient/server-applikasjon kan man ofte bruke port 443 for å få trafikken gjennom. Der det ikke går kan man legge ved egen applikasjonsproxy; men husk: La den være enkel!