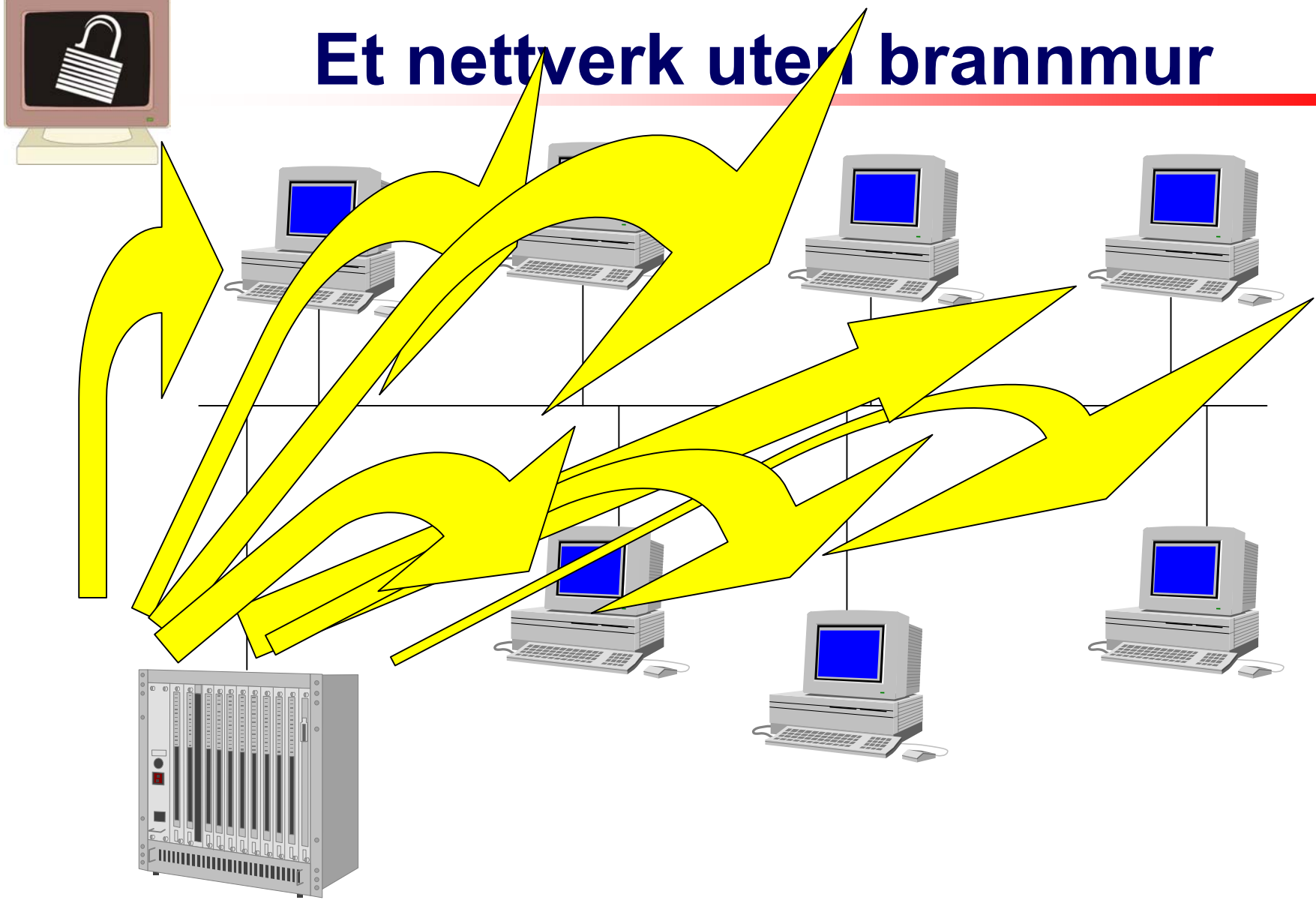
Forelesning 12

Brannmurer

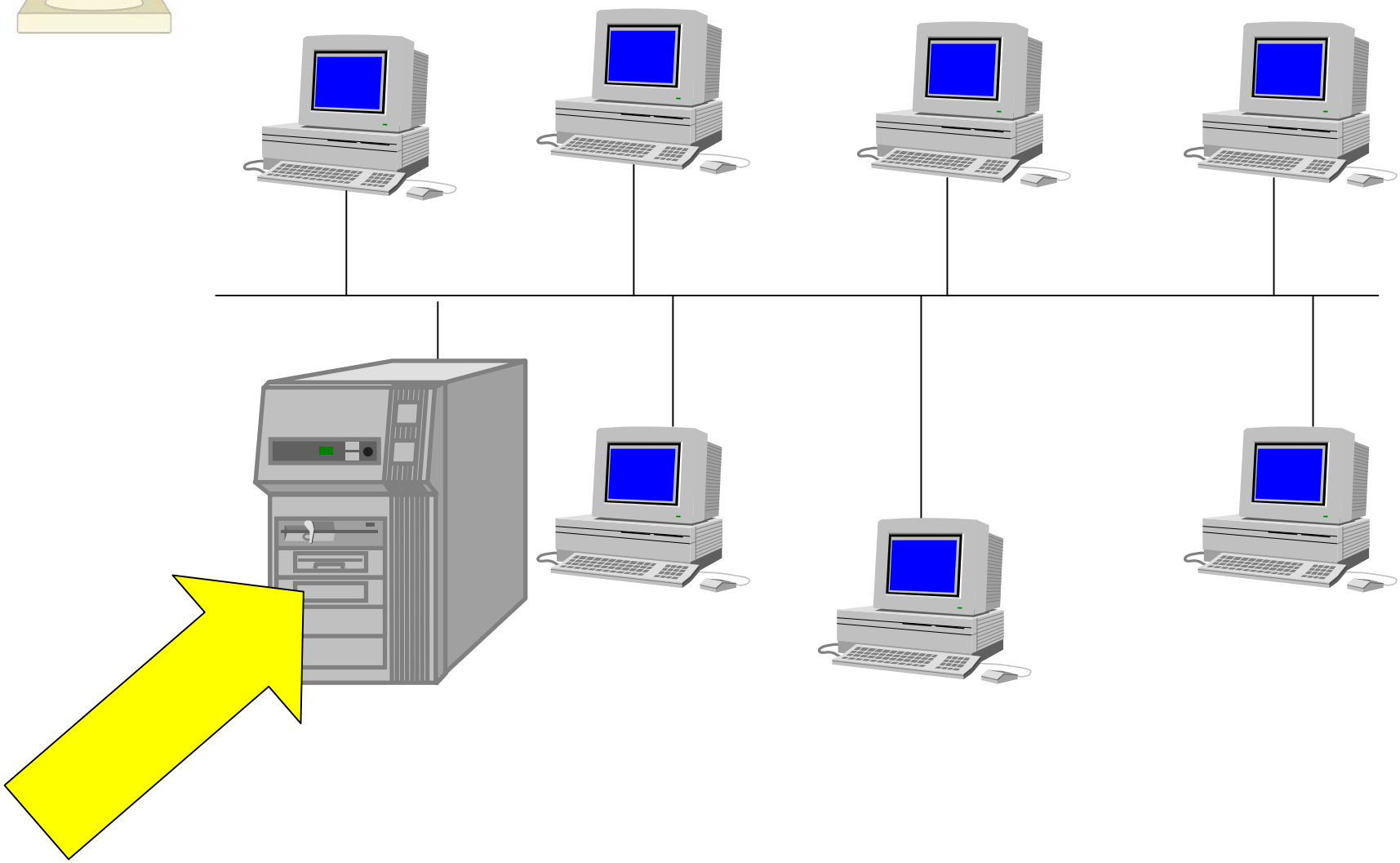
Brannmurer

- En brannmur blir vanligvis plassert mellom et intranett og Internett
- Brannmurer kan (og bør) også brukes mellom intranett

Et nettverk uten brannmur



Et nettverk med brannmur



Hvorfor brannmur?

- Internett er et farlig sted
- Å knytte et stort lokalnett med hundre-vis av maskiner til Internett medfører en betydelig risiko
- Kanskje umulig å beskytte hver enkelt maskin?
- Bedre å ha ett sted hvor man kan kontrollere trafikk inn og ut

Mål med Brannmurer

- Hindre at illegal trafikk slipper inn på intranettet
  - Maskiner kjører ofte programvare de ikke er klar over
  - Programvare som brukerne laster ned kan inneholde feil som utgjør en fare
- Ett sentralt punkt for administrasjon av sikkerhetspolicy

Andre betraktninger

- En brannmur bedrer sikkerheten, men nettverket *bak* brannmuren blir ikke nødvendigvis sikrere
- Det er viktig å patche og vedlikeholde arbeidsstasjoner og servere, men dette er da naturlig nok mange ganger så viktig for brannmurer!
- Brannmurer må oppdateres regelmessig

Politikk

- Alt som ikke er eksplisitt tillatt er forbudt

eller

- Alt som ikke er forbudt, er tillatt

Hvilken ville du valgt?

Fysiske løsninger

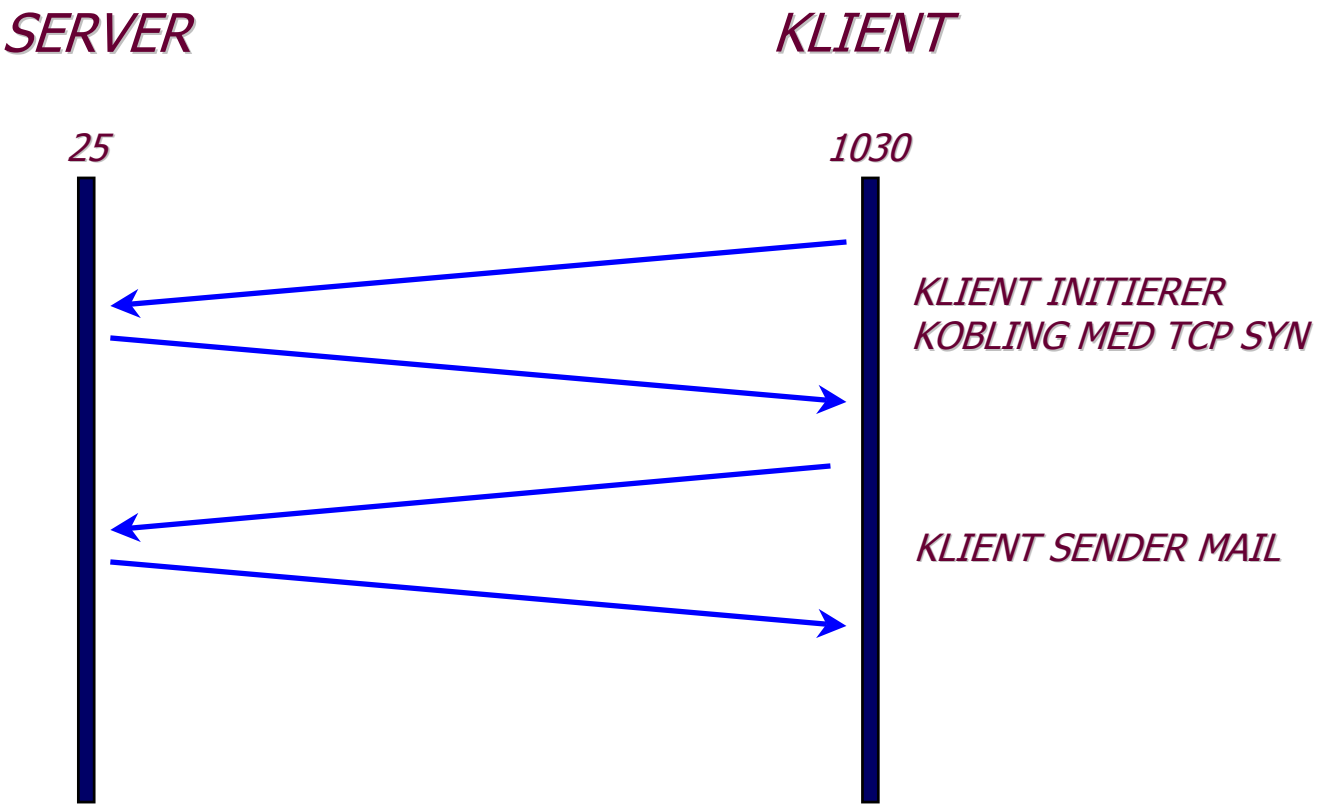
- Ruter (router)
- PC
- PC og ruter
- etc...

- Også: Personal Firewall
  - ZoneAlarm

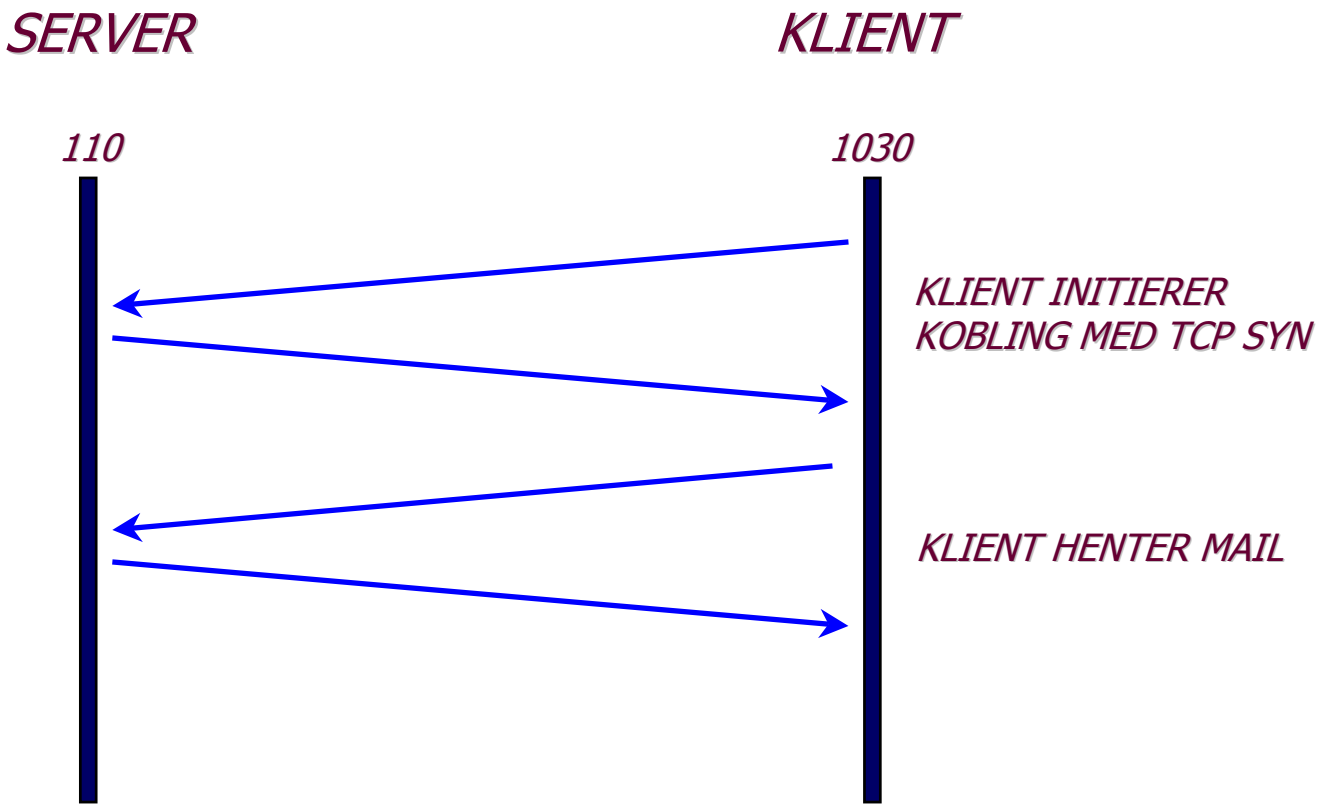
Protokoller gjennom brannmuren

- En brannmur konfigureres vanligvis til å slippe enkelte protokoller gjennom
  - Epost
  - Filoverføring
  - Web-aksess
  - ...

Epost - SMTP


SERVER
KLIENT
25
1030
KLIENT INITIERER
KOBLING MED TCP SYN
KLIENT SENDER MAIL

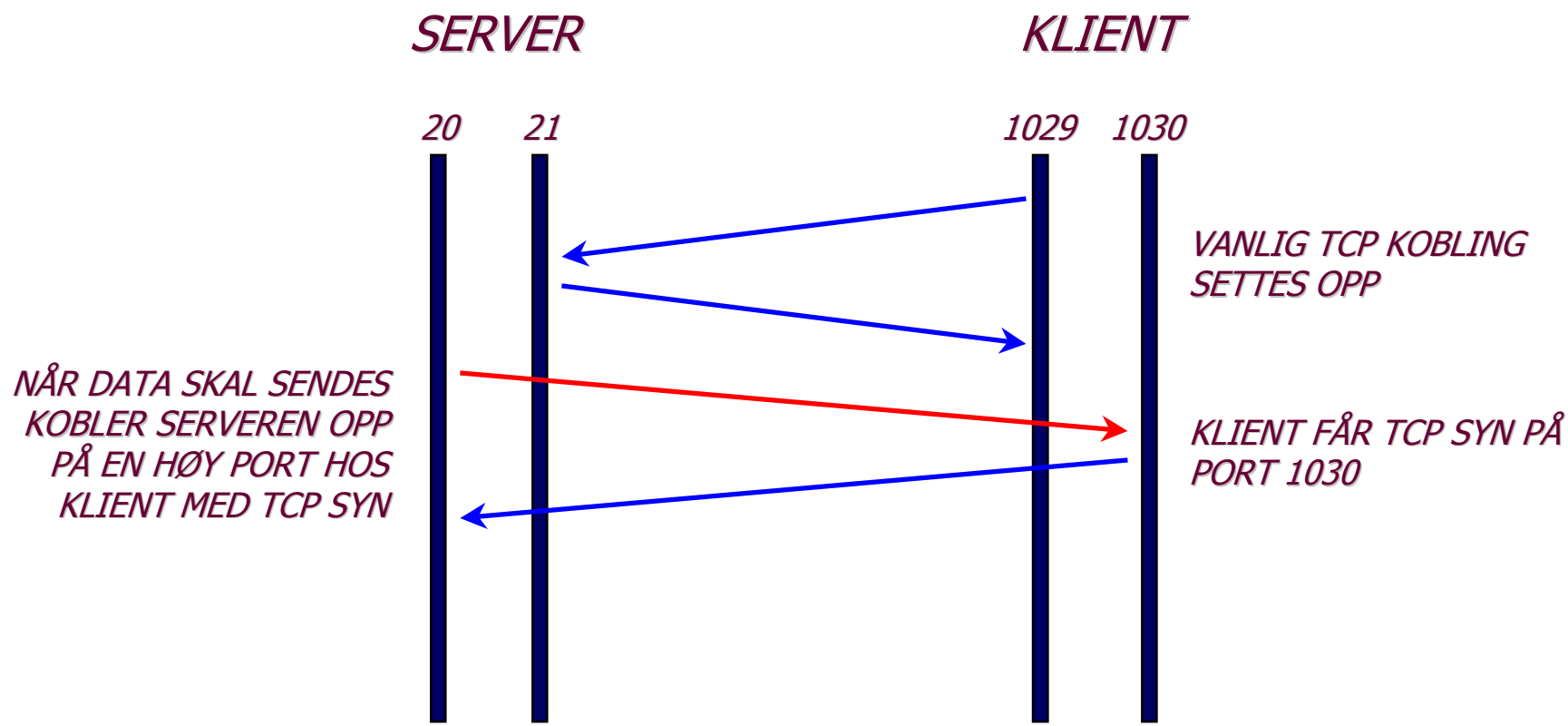
Epost - POP


SERVER
KLIENT
110
1030
KLIENT INITIERER
KOBLING MED TCP SYN
KLIENT HENTER MAIL

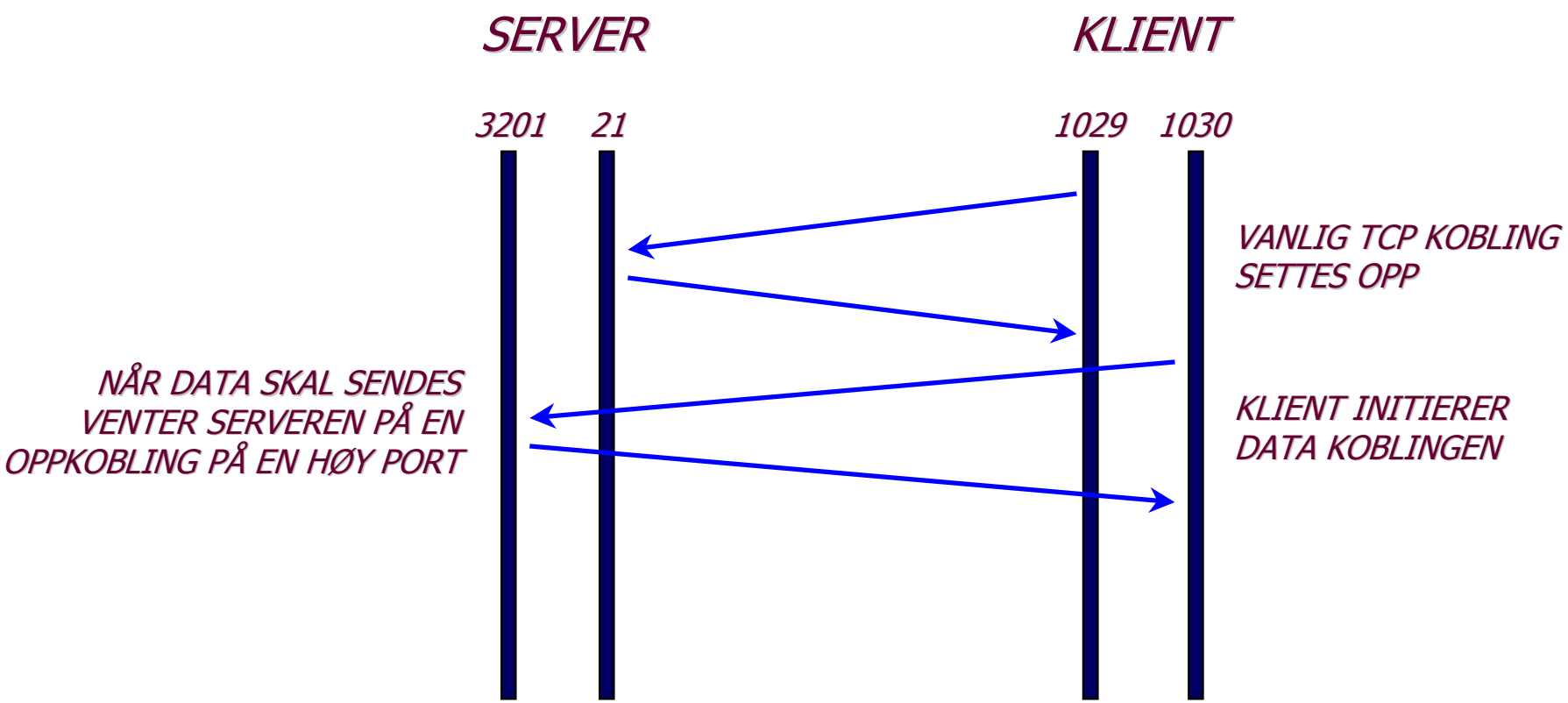
Andre Protokoller

- De fleste tjenestene bruker tilsvarende kommunikasjon som SMTP og POP
- For eksempel
  - DNS - port 53 (når TCP brukes)
  - Telnet - port 23
  - HTTP - port 80

Aktiv FTP


SERVER
KLIENT
20
21
1029
1030
VANLIG TCP KOBLING SETTES OPP
NÅR DATA SKAL SENDES KOBLER SERVEREN OPP PÅ EN HØY PORT HOS KLIENT MED TCP SYN
KLIENT FÅR TCP SYN PÅ PORT 1030

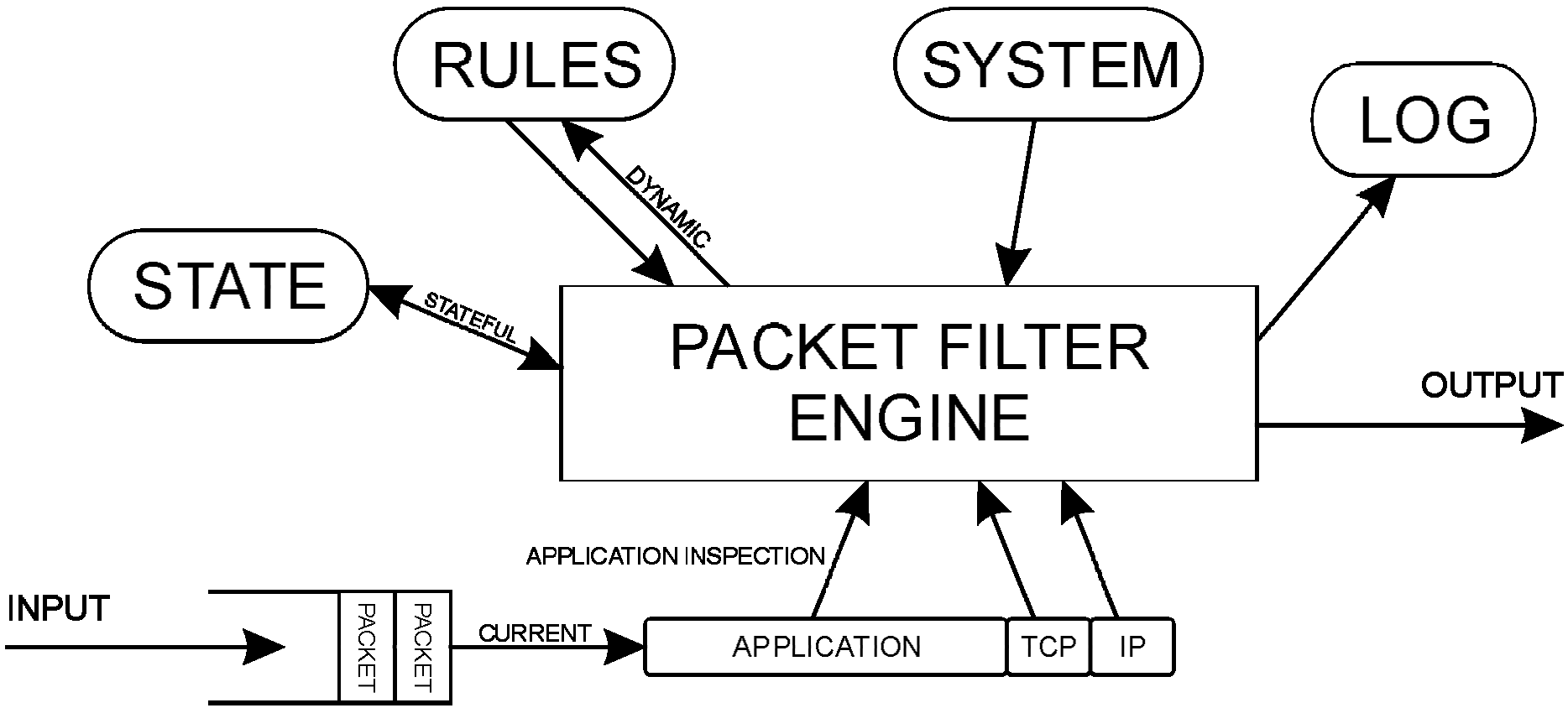
Passiv FTP


SERVER
KLIENT
3201
21
1029
1030
VANLIG TCP KOBLING SETTES OPP
NÅR DATA SKAL SENDES VENTER SERVEREN PÅ EN OPPKOBLING PÅ EN HØY PORT
KLIENT INITIERER DATA KOBLINGEN

Brannmurteknologier

- Pakkefilter
  - Header / Application inspeksjon
  - Tilstandsbasert / dynamisk filtrering
  - Bufferstrategi
- Network Address Translation
- Proxy
  - Generisk/Dedikert
  - Transparent

Pakkefilter



Pakkefilter - Header inspeksjon

- IP-header - til/fra adresse
- TCP-header - port (tjeneste)
- Flagg (SYN)

Application Inspection

- Se på innholdet i pakkene
- Eks: FTP
  - Datakanalen opprettes av server mot klient, portnummer på klient kommuniseres på applikasjonsnivået
  - Dette eksempelet krever også tilstandsbasert og dynamisk filtrering!

Dynamisk filtrering

- Tidlige pakkefiltre benyttet statiske filtreringsregler
- Moderne pakkefiltre er i stand til å endre regler basert på input til filteret

Tilstandsbasert filtrering

- Hvis man bare ser på hver enkelt pakke, blir det vanskelig å håndtere fragmenterte TCP-pakker eller UDP-trafikk
- Med en tilstandstabell kan filteret holde styr på at det f.eks. er opprettet en FTP-forbindelse

Banal bufferstrategi

- Motta pakke
- Bufre internt
- Kjør filtrering
- Ta avgjørelse
- Utfør handling

Avansert bufring

- Buffer and release
  - Motta og bufre opp pakker pr. forbindelse
  - Fragmenterte pakker vurderes samlet, og datafeltet i flere pakker kombineres
  - Når innholdet er godkjent, slippes alle pakkene gjennom
- Reflection
  - Alle pakker kopieres til en tredjepart som kontrollerer innholdet
  - Uhumskheter kan medføre terminering

Hva gjør vi med pakkene?

- Stopp
- Videresend
- Avvis (med beskjed)
- Reflektering
- Logging
- Modifikasjon

Pakkefilter-eksempler

- IP spoofing
  - Kast alle pakker som kommer utenfra som har en av våre interne adresser som avsender
- Source routing
  - Kast pakker / skru av flagg
- Tiny fragment
  - Kast fragmenterte pakker

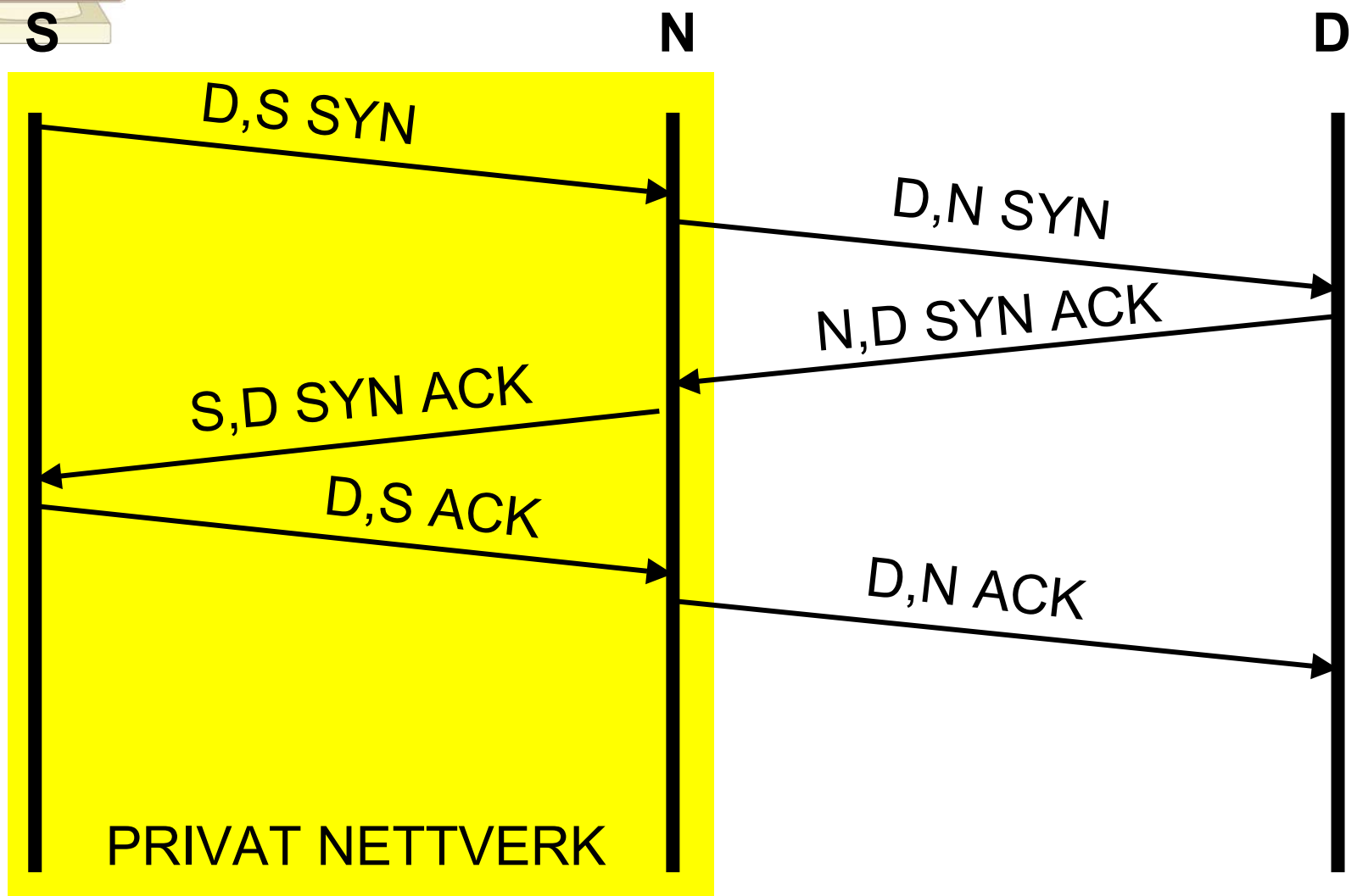
”Egress”-filtrering

- Kan i mange sammenhenger være ønskelig å begrense hva som kommer fra det interne nettet
  - Pakker som ikke har lovlige (dvs. våre) avsenderadresser
- Mange organisasjoner blokkerer tilgang til enkelte servere
  - BigBrother
  - Playboy.com

Network Address Translation

- Gir mulighet for å skjule interne adresser i trafikk mot Internett
- Omverdenen ser bare én IP-adresse
- Source NAT: Trafikk initiert innenfra
- Destination NAT: Trafikk initiert utenfra (eks: WWW-trafikk)

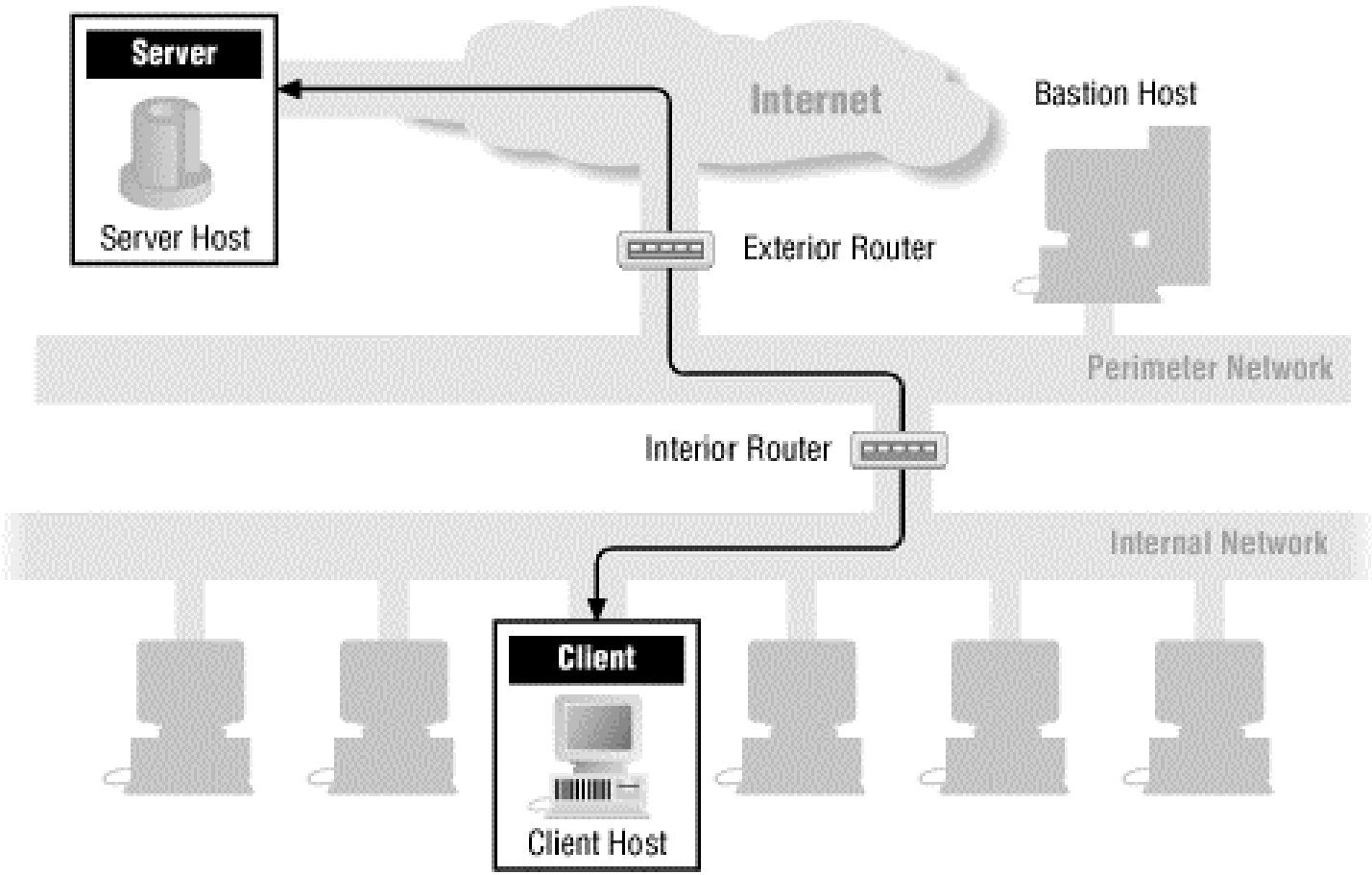
Source NAT


S
N
D
D,S SYN
D,N SYN
N,D SYN ACK
S,D SYN ACK
D,S ACK
D,N ACK
PRIVAT NETTVERK

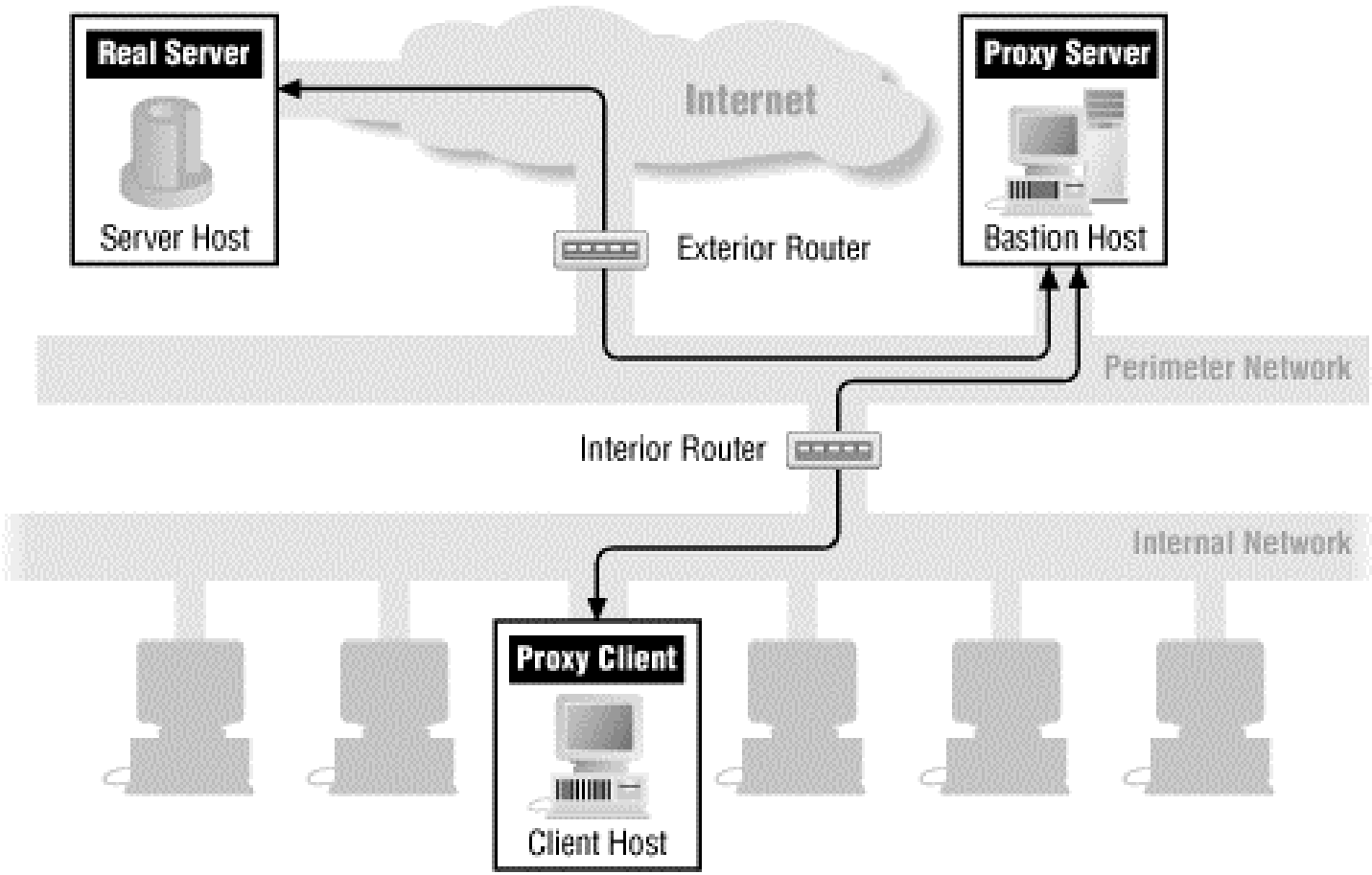
Proxy

- ”På vegne av”
- En proxy vil stå mellom to kommuni-serende parter, og ha en TCP-forbindelse med hver av dem
- Klient-programvare på ”innsiden” må endres slik at den tar kontakt med proxyen i stedet for den egentlige motparten

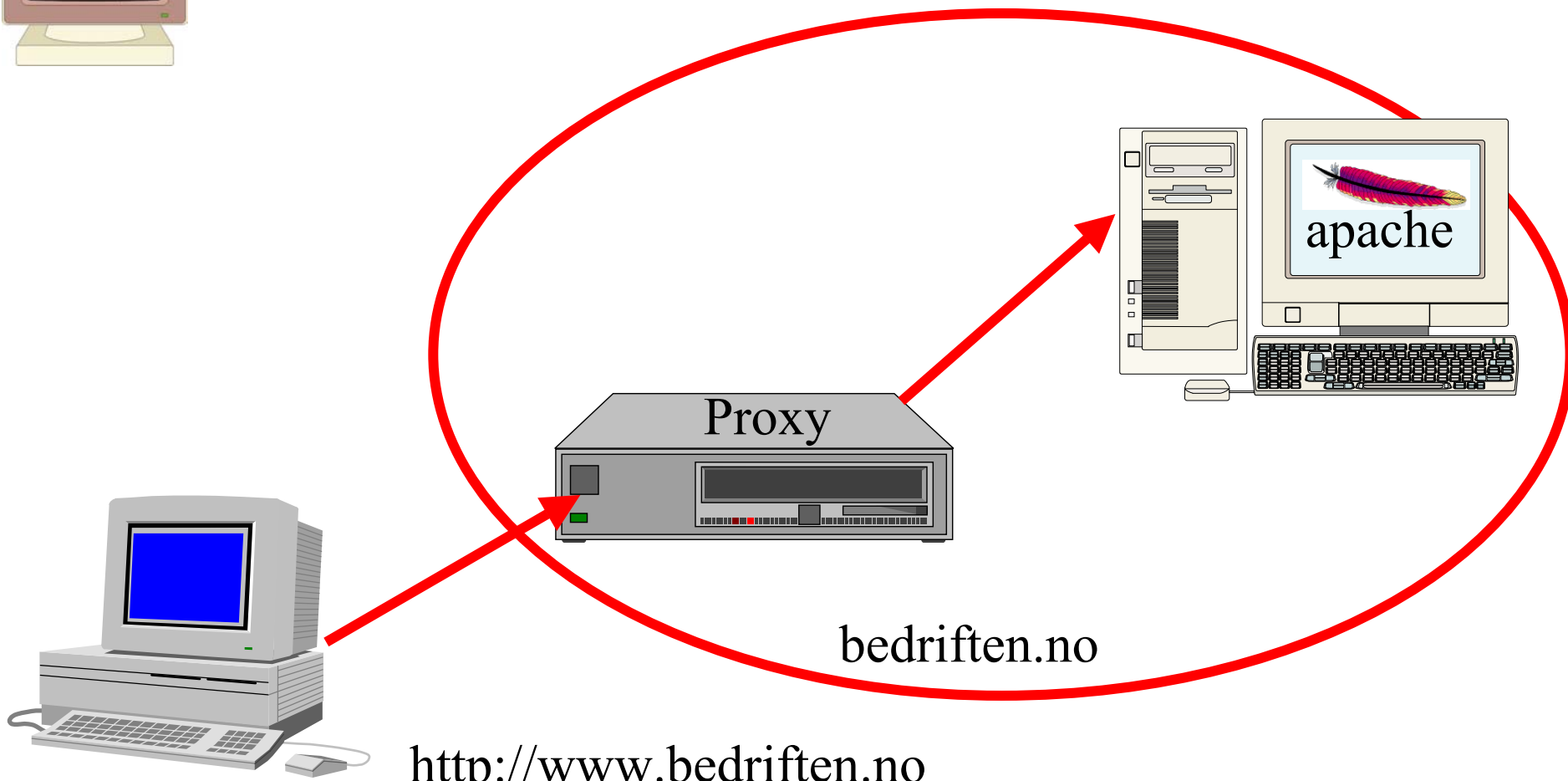
Arkitektur uten FW



Arkitektur med proxy



Proxy illustrert



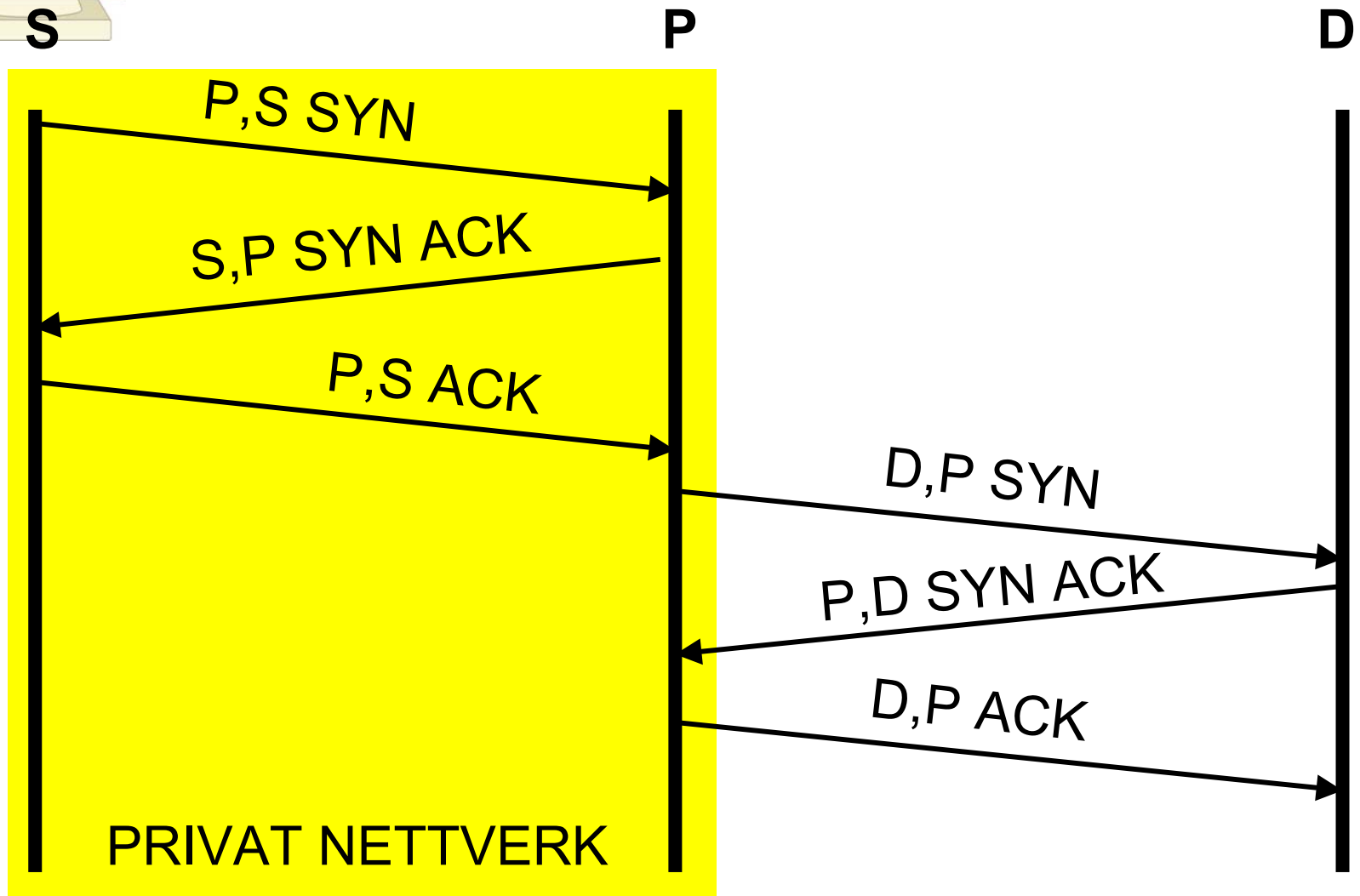
Generisk proxy

- Kan i sin enkleste form se ut som en NAT-variant
- Har mulighet til å utføre tilleggsprosessering i oppstartsfasen (autentisering, etc.)
- Samarbeidende proxyer kan f.eks. sette opp kryptert forbindelse mellom to lokalnett
- Eksempel: SOCKS

Dedikert proxy

- Forstår (i større eller mindre grad) protokollen som formidles
- Applikasjonsspesifikk – må ha en egen proxy per protokoll
- Brukes gjerne for komplekse applikasjoner med en forhistorie full av hull
- Eksempel: SMTP proxy
  - Tolker et subsett av SMTP-protokollen
  - Kaster alt som faller utenom

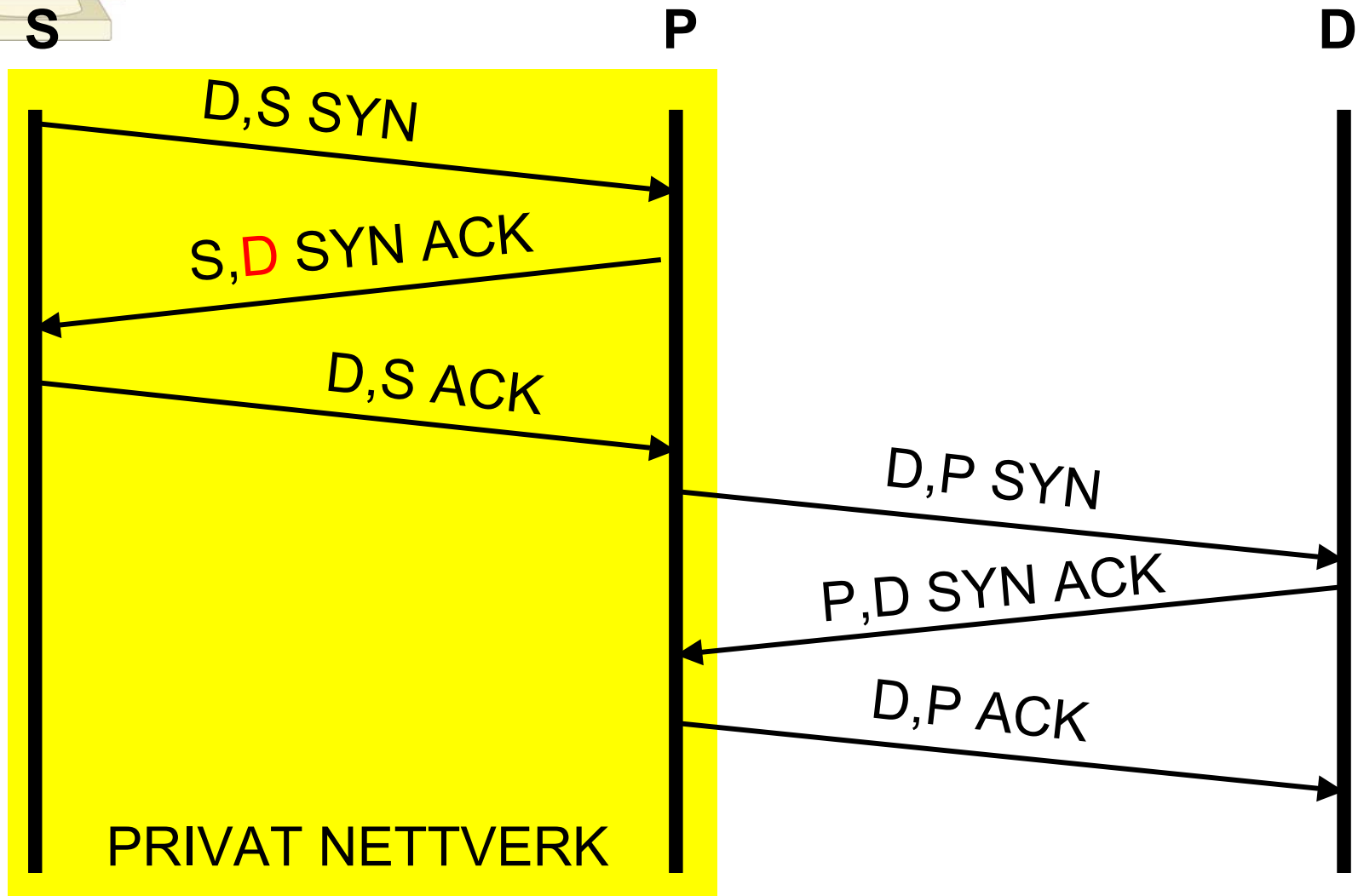
Oppsett av forbindelse m/proxy



Transparent proxy

- Brukerprogrammer må *vite om* vanlige proxyer – må modifiseres til å kontakte proxy i stedet for destinasjonsadressen
- En transparent proxy snapper opp pakker som skal til den eksterne adressen, og svarer på vegne av denne
- Oppretter ny forbindelse til ekstern adresse

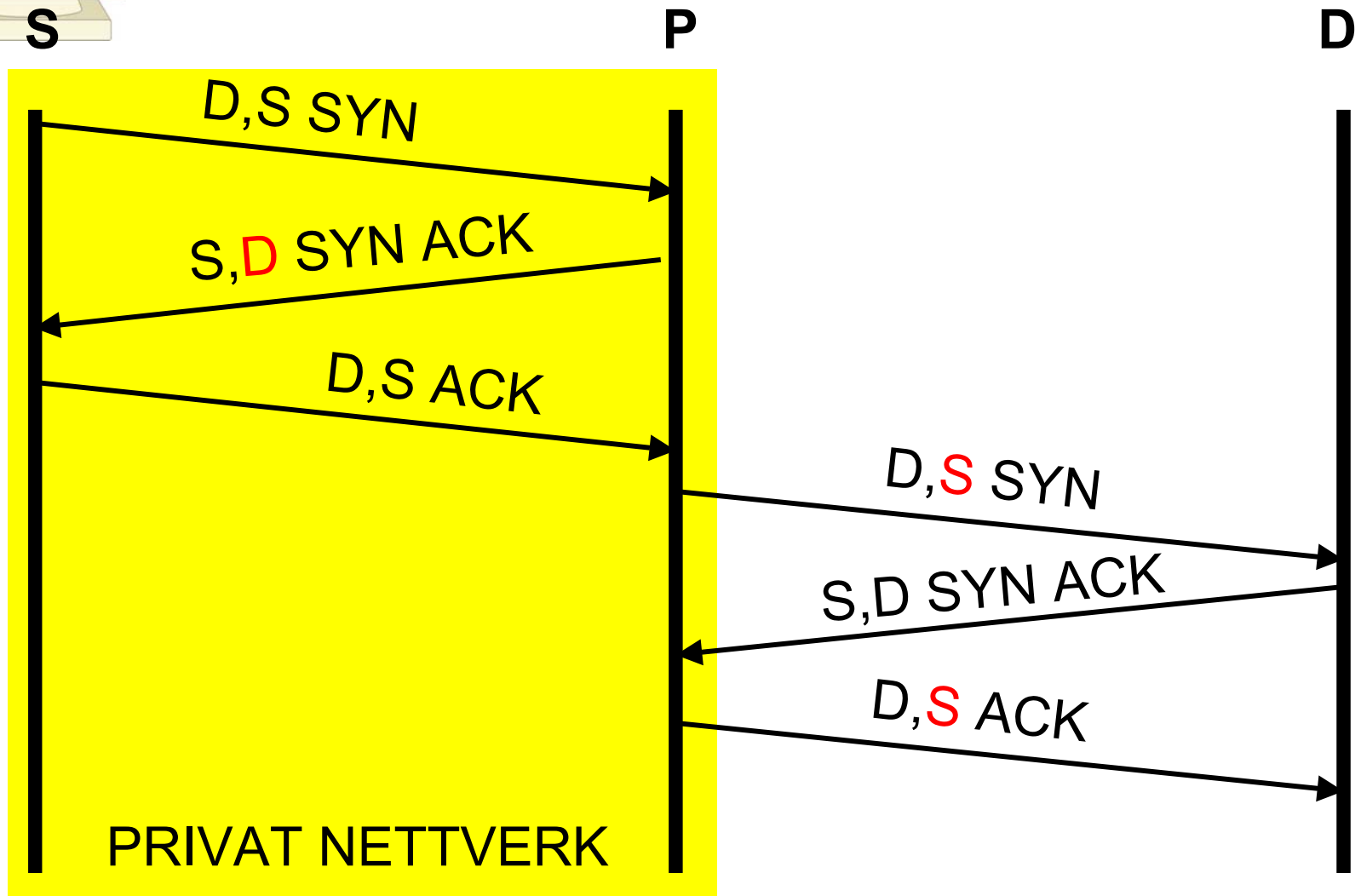
Transparent proxy m/skjult adr.



Transparent proxy forts.

- En transparent proxy kan også la være å skjule avsenderadressen
- Proxyen må da ”snappe” opp trafikk i begge retninger, og gi seg ut for å være begge parter etter tur
- En transparent proxy bryter en del fundamentale nettverksregler (det er ikke ”lov” å gi seg ut for å være andre)

T.P. uten skjult adresse



Forskjellen mellom NAT og T.P.

- Det essensielle er fortsatt at en proxy har **to** TCP-forbindelser
- Mye av det samme kan oppnås med NAT og payload scanning, men avanserte funksjoner som introduserer større forsinkelser får problemer med tidsbegrensninger i TCP (timeout, etc.)
- NAT redirection + proxy en mulighet!

Andre tjenester

- Mange leverandører pakker inn mye annen funksjonalitet i brannmuren, f.eks.:
  - VPN
  - Viruskontroll
  - IDS
- En brannmur kan ofte være et greit sted å gjøre slike ting, men vi regner ikke nødvendigvis slike funksjoner til de fundamentale arbeidsoppgavene til en brannmur

Dagens website

- http://www.wilyhacker.com/
- Fulltekst (!) av :

  Firewalls and Internet Security:

  Repelling the Wily Hacker

  William R. Cheswick and Steven M. Bellovin