(12) **PATENT**

(19) NO (11) **324332** (13) **B1**

NORGE

(51) Int Cl.
H04L 9/00 (2006.01)
H04L 12/56 (2006.01)
H04L 29/06 (2006.01)

Patentstyret

(21)	Søknadsnr	19970611	(86)	Int.inng.dag og søknadsnr	1996.06.16 PCT/IL96/00017
(22)	Inng.dag	1997.02.10	(85)	Videreføringsdag	1997.02.10
(24)	Løpedag	1996.06.16	(30)	Prioritet	1995.06.15, IL, 114182
(41)	Alm.tilgj	1997.04.15			
(45)	Meddelt	2007.09.24			

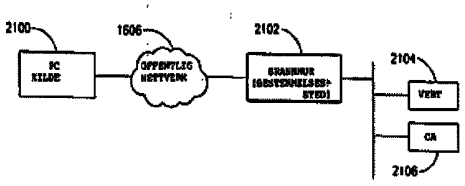
(73) Innehaver Checkpoint Software Technologies Ltd, Hateomim Building 2, 35 Jabotinsky Street, Ramat Gan 52511, IL
(72) Oppfinner Gil Shwed, Tel Aviv, IL
Shlomo Kramer, Tel Aviv, IL
Nir Zuk, Ramat Gan, IL
Gil Dogon, Hertzlia, IL
Ehud Ben-Reuven, Tel Aviv, IL
(74) Fullmektig Onsagers AS, Postboks 6963 St Olavs Plass, 0130 OSLO

(54) Benevnelse **System for å sikre flyt av og for å selektivt modifisere pakker i et datamaskinnettverk**

(56) Anførte publikasjoner
Artikkel med tittel "Intrusion Protection for Nettworks fra "Byte"
Artikkel med tittel "Network Firewalls" fra "Iecc Communication Magazine"

(57) Sammendrag

Den foreliggende oppfinnelse beskriver et nytt system for å kontrollere flyt av inngående og utgående datapakker i et datamaskinnettverk. Ved å kontrollere flyt av pakkene i et datamaskinnettverk, kan private nettverk sikres fra utenforliggende angrep i tillegg til å kontrollere flyt av pakker fra innenfor det private nettverk til den ytre verden. En bruker genererer en regelbase som konverteres til et sett instruksjoner i filterspråk. Hver regel i regelbasen omfatter en kilde, et bestemmelsessted, en tjeneste, enten å godta eller å avvise pakken og muligheten for å logge hendelsen. Settet av instruksjoner av filterspråk installeres og eksekveres på inspeksjonsmaskiner som er anordnet i datamaskiner som virker som brannmur. Brannmurene er anordnet i datamaskinnettverket slik at all trafikk til og fra nettverket som skal beskyttes tvinges til å passere gjennom brannmuren. På denne måten filtreres pakkene mens de flyter inn og ut av nettverket ifølge reglene omfattet i regelbasen. Inspeksjonsmaskinen virker som en virtuell pakkefiltreringsmaskin som fastslår per pakkebasis enten å avvise eller å godta pakken. Hvis en pakke er avvist, droppes den. Hvis den er godtatt, kan den modifiseres. Modifiseringen kan omfatte kryptering, dekryptering, signaturgenerering, signaturverifisering eller adresseoversettelse. Alle modifikasjoner utføres i følge innholdet i regelbasen. Den foreliggende oppfinnelse tilveiebringer ytterligere sikkerhet for et datamaskinnettverk ved å kryptere kommunikasjoner mellom to brannmurer, mellom en kunde og en brannmur. Dette tillater anvendelse av usikre offentlige nettverk i bygning av et WAN som omfatter både private og offentlige nettverkssegmenter, og derved dannelse av et virtuelt privat nettverk.



Søknaden vedrører generelt en fremgangsmåte for å kontrollere sikkerheten ved et datamaskinnettverk. Nærmere bestemt vedrører den en lett modifiserbar og utvidbar metode for datamaskinnettverksikkerhet som kontrollerer flyt av informasjon i nettverket fra/til eksterne og interne bestemmelsessteder.

Oppkoblingsmuligheter og sikkerhet er to mål som er i konflikt i datamaskinmiljøet i de fleste organisasjoner. Det typiske datamaskinsystem med modem er bygget rundt nettverkkommunikasjoner, og frembringer transparent tilgang til en mengde tjenester. Den globale tilgjengelighet av disse tjenester er muligens det ene viktigste trekk med modembaserte datamaskinløsninger. Krav for oppkoblingsmulighet kommer både fra innenfor organisasjonene og utenfra.

Beskyttelse av nettverktjenester mot uautorisert bruk er ytterst viktig i en hvilken som helst organisasjon. For eksempel vil UNIX arbeidsstasjoner som er tilkoblet internett, tilby de samme tjenester til hele verden som de tilbyr maskinen ved siden av. Ved dagens teknologi må en organisasjon ofre en stor del av oppkoblingsmulighetene med den hensikt å hindre sårbarhet, i noen tilfelle i den grad at alle forbindelser til den ytre verden eller andre beliggenheter fjernes.

Etter hvert som kravet til sikkerhet øker blir midlene for å kontrollere tilgang til nettverkressurser et administrativt prioritetsområde. Med den hensikt å spare kostnader og beholde produktiviteten, må tilgangskontroll være enkel å konfigurere og transparent både for brukerne og for anvendelsene. Minimalisering av oppsettkostnader og unyttbar tid er også viktige faktorer.

Pakkefiltrering er en fremgangsmåte som tillater oppkoblingsmuligheter og likevel frembringer sikkerhet ved å kontrollere trafikken som går igjennom, og som på denne måte hindrer ulovlige forbindelsesforsøk, både innenfor enkelte nettverk og mellom tilkoplede nettverk.

Dagens implementering av pakkefiltrering tillater spesifisering av tilgangslistetabeller ifølge et fast format. Denne metoden er begrenset i fleksibiliteten til å uttrykke sikkerhetspolitikken for en gitt organisasjon. Den er også begrenset til protokollsettet og tjenestene som er definert i nevnte tabell. Denne metoden tillater ikke innføring av forskjellige protokoller eller tjenester som ikke er spesifisert i den opprinnelige tabell.

En annen fremgangsmåte for implementering av pakkefiltrering er å tilpasse datamaskinens operative systemkode manuelt i hvert strategiske punkt i organisasjonen. Denne metoden er begrenset i fleksibiliteten til fremtidige forandringer i nettverkstopologi, nye protokoller, forbedrede tjenester og til fremtidige sikkerhetstrusler. Den krever en utstrakt mengde arbeid utført av spesialister som skal modifisere rettighetsbeskyttede datamaskinprogrammer. Dette gjør den utilstrekkelig og kostbar for oppsett og vedlikehold.

Et eksempel på en fremgangsmåte for inspisering av datapakker av den ovennevnte type er beskrevet i dokumentet: Dawson J.B., "Intrusion Protection for Networks", Byte, McGraw-Hill Inc. St. Petersborough, US, Bind 20, nummer 4, sider 171-172 XP000562745 ISSN:0360-5380.

I dokumentet Bellovin S.M. o.a. "Network firewalls", IEEE Communications Magazine, IEEE Service Center, Piscataway, N.J. DU, Bind 32, nummer 9, sider 50-57 XP000476555 ISSN:0163-6804, beskrives en fremgangsmåte for å inspisere inngående og utgående datapakker i et datamaskinnettverk ifølge innledningen i det vedlagte krav 1. Et sikkerhetssystem ifølge innledningen i krav 19 er også kjent fra dette dokumentet.

I tillegg er behovet for sikre fjernforbindelser mellom firmaer, avdelingskontorer, og samarbeidspartnere blitt et hovedkrav i moderne forretningsvirksomhet. Reserverte punkt-til-punkt-forbindelser mellom nettverk har vært anvendt for fullstendig privat handel og lang-avstandtransaksjoner mellom firmaer. Imidlertid, har mangel på fleksibilitet og høye kostnader ved nevnte forbindelser hindret deres utstrakte anvendelse. Offentlige nettverk slik som internett tilveiebringer en fleksibel og rimelig løsning for lang avstand kommunikasjon mellom nettverk. Istedenfor å etablere dedikerte linjer, kan firmaene kommunisere ved anvendelse av internett som forbindelse. Med en gang de er tilkoblet en lokal internettleverandør, kan private nettverk raskt tilkobles et hvilket som helst bestemmelsessted i verden.

Et privat nettverk som anvender noen offentlige segmenter er kalt et virtuelt privat nettverk (VPN). Et VPN er betydelig mindre kostbart og mer fleksibelt enn et dedikert privat nettverk. Hvert private nettverk trenger bare å være tilkoblet en lokal internettleverandør. Addering av nye forbindelser er enkelt og kostnadsfritt. Imidlertid, er en stor ulempe ved et VPN at det er usikkert på grunn av de usikre segmentene. Internettilkoblingen utsetter firmaet for de

følgende to farer: (1) uautorisert internettilgang i interne firmanettverk (break-in) og (2) tjuvlytting og utførelse av ulovlige endringer med firmaets kommunikasjon ved passering gjennom Internett.

Sikkerhetsrisikoene forbundet med kommunikasjon over Internett har holdt firmaer tilbake fra fullt ut å utnytte fordelene ved VPN. Forretningsføring over internett (for eksempel å overføre beløp, å oppnå og verifisere kredittinformasjon, å selge og å levere produkter) krever en pålitelig og effektiv sikkerhetsløsning.

Følgelig er hensikten med den forliggende oppfinnelse å tilveiebringe en forbedret fleksibel, lett modifiserbar sikkerhetsfremgangsmåte som kontrollerer flyt av informasjon i et datamaskinnettverk tilsvarende den beskrevet i US Patentsøknad 08/168041.

En annen hensikt med den forliggende oppfinnelse er å kontrollere informasjonsflyt i nettverket fra/til interne og eksterne bestemmelsessteder hvor kontrollen omfatter i det minste ett av trinnene å kryptere informasjon og å modifisere kildens og/eller bestemmelsesstedets adresse.

En ytterligere hensikt med oppfinnelsen er å kontrollere informasjonsflyt ved hjelp av et pakkefilter som er i stand til å undersøke hver informasjonspakke som flyter gjennom en node i systemet, idet pakken er kryptert.

En ytterligere hensikt med oppfinnelsen er å kontrollere informasjonsflyt ved hjelp av pakkefilteret hvor pakkefilteret er i stand til å la pakken passere bare hvis den er autorisert på forhånd, fortrinnsvis etter en ikke-destruktiv gyldighetstest for forbindelsen.

En annen hensikt med oppfinnelsen er å tilveiebringe en generisk pakkefiltermodul som er kontrollert ved et sett instruksjoner for å implementere en gitt sikkerhetspolitikk ved en node for å godta (la passere) eller avvise (droppe) pakken, hvor pakken passerer bare hvis passeringen er autorisert på forhånd.

En annen hensikt med oppfinnelsen er å tilveiebringe en sikkerhetsmetode for et datamaskinettverk som er lett modifiserbar ved systemets administrator uten behov for å forandre egenskapene for selve pakkefilter eller for å skrive omfattende kode.

En annen hensikt med oppfinnelsen er å tilveiebringe en forbedret gyldighetstest for en forbindelse.

En ytterligere hensikt med oppfinnelsen er å tilveiebringe muligheten til å modifisere pakken ved enten å kryptere den, å modifisere adressen for et bestemmelsessted, eller ved å godta eksterne input som kriterier for å godta, avvise eller modifisere nettverkskommunikasjon.

En annen hensikt med den foreliggende oppfinnelse er å tilveiebringe et krypteringsmetode for å sikre flyt av data over usikre offentlige nettverk slik som internett, for derved å danne et VPN.

For å oppnå disse hensiktene er fremgangsmåten for å inspisere og selektivt å modifisere inngående og utgående datapakker i et datamaskinsystem karakterisert ved de trekkene angitt i krav 1-18. Oppfinnelsen tilveiebringer også et system som angitt i kravene 19-20.

Ifølge et aspekt av den foreliggende oppfinnelse, tilveiebringes et datamaskinsystem for å sikre transaksjoner over nettverk ved å kryptere dem, ved å forbinde flere nettverk sammen med forskjellige addresseringsmetoder og å tilveiebringe måter å la informasjonspakker passere bare når kommunikasjonsskilden er autorisert og å detektere gyldighet av trafikken gjennom nettverket mens informasjon som kreves for å utføre dette minimaliseres, fortrinnsvis i en feilsikker arkitektur.

Ifølge en første utførelse av den foreliggende oppfinnelsen tilveiebringes en fremgangsmåte for å inspisere og selektivt modifisere inngående og utgående datapakker i et datamaskinnettverk, idet inspeksjonen og den selektive modifikasjonen av datapakkene foregår ifølge en sikkerhetsregel, idet fremgangsmåten omfatter trinnene å generere en definisjon av hvert aspekt ved datamaskinnettverket som inspiseres ved sikkerhetsregelen, å generere sikkerhetsregelen ved definisjonen av aspektene, idet sikkerhetsregelen kontrollerer i det minste ett av aspektene, å konvertere sikkerhetsregelen i en pakke av filterspråkinstruksjoner for å kontrollere operasjonen av en pakkefiltreringsmodul som inspiserer og selektivt modifiserer datapakkene ifølge sikkerhetsregelen, å koble pakkefiltermodulen til datamaskinnettverket for å inspisere og selektivt modifisere datapakkene ifølge sikkerhetsregelen, idet pakkefiltermodulen implementerer en virtuell pakkefiltreringsmaskin, og idet pakkefiltermodulen utfører pakkefilterspråkinstruksjonene for å drive

den virtuelle pakkefiltreringsmaskinen for enten å godta eller å avvise passering av datapakkene inn og ut av dadamaskinnettverket og selektivt å modifisere datapakkene som er godtatt.

Aspektene kan ytterligere omfatte nettverksobjekter eller -tjenester. I tillegg omfatter objektdefinisjonene objektets adresse og instruksjonene i filterspråk for konversjonstrinnet er i form av skript og omfatter ytterligere en kompilator for å kompilere skriptet til instruksjoner som utføres i utførelsestrinnet.

Ytterligere er både trinnene å generere nettverksaspekter og aspekter av sikkerhetsregelen definert grafisk og den selektive modifikasjon velges fra gruppen omfattende kryptering, dekryptering, signaturgenerering og signaturverifisering.

Det er også frembrakt ifølge en foretrukket utførelse av den foreliggende oppfinnelse i et sikkerhetssystem for å inspisere og selektivt å modifisere inngående og utgående datapakker i et datamaskinnettverk, idet sikkerhetssystemet inspiserer og selektivt modifiserer datapakkene i datamaskinnettverket ifølge en sikkerhetsregel, hvor hvert aspekt av datamaskinnettverket inspisert ved sikkerhetsregelen er definert på forhånd, idet sikkerhetsregelen er definert på forhånd ved aspektene og konvertert til instruksjoner i pakkefilterspråk, en fremgangsmåte for å drive sikkerhetssystemet omfattende trinnene å tilveiebringe en pakkefiltermodul tilkoblet datamaskinnettverket i det minste i en enhet i datamaskinnettverket som skal inspiseres ved sikkerhetsregelen, idet pakkefiltermodulen implementerer en virtuell pakkefiltreringsmaskin som inspiserer og selektivt modifiserer datapakkene som passerer inn og ut av datamaskinnettverket, og pakkefiltermodulen utfører instruksjonene i pakkefilterspråk for å drive den virtuelle pakkefiltreringsmaskin til enten å godta eller å avvise passering av datapakkene inn og ut av datamaskinnettverket og til selektivt å modifisere datapakkene som er godtatt.

Også ifølge en foretrukket utførelse av den foreliggende oppfinnelsen frembringes i et sikkerhetssystem for å inspisere og selektivt å modifisere inngående og utgående datapakker i et datamaskinnettverk, idet sikkerhetssystemet inspiserer og selektivt modifiserer datapakkene i datamaskinnettverket ifølge en sikkerhetsregel, hvor hvert aspekt av datamaskinnettverket inspisert ved sikkerhetsregelen er definert på forhånd,

idet sikkerhetsregelen er definert på forhånd ved hjelp av aspektene, og konvertert til instruksjoner i pakkefilterspråk, en fremgangsmåte for å drive sikkerhetssystemet omfattende trinnene å tilveiebringe en pakkefiltermodul tilkoblet datamaskinnettverket i det minste i en enhet i datamaskinnettverket som skal kontrolleres ved sikkerhetsregelen, idet pakkefiltermodulen simulerer en virtuell pakkefiltreringsmaskin som inspiserer og selektivt modifiserer datapakkene som passerer inn og ut av datamaskinnettverket, idet pakkefiltermodulen leser og utfører instruksjonene i pakkefilterspråket for å utføre pakkefiltreringsoperasjoner, å lagre resultatene oppnådd i trinnene å lese og å utføre instruksjonene i pakkefilterspråket i en lagringsanordning, og idet pakkefiltermodulen benytter de lagrede resultater fra tidligere inspeksjoner, for å drive pakkefiltermodulen til å godta eller avvise passering av datapakkene inn og ut av datamaskinnettverket og for selektivt å modifisere datapakkene som blir godtatt.

I tillegg tilveiebringes også ifølge en foretrukket utførelse av den foreliggende oppfinnelse, et sikkerhetssystem for å inspisere og selektivt å modifisere inngående og utgående datapakker i et datamaskinsnettverk, idet sikkerhetssystemet inspiserer og selektivt modifiserer datapakkene som passerer gjennom datamaskinnettverket ifølge en sikkerhetsregel, hvor hvert aspekt av datamaskinnettverket kontrollert ved sikkerhetsregelen er definert på forhånd, hvor sikkerhetsregelen er definert på forhånd ved hjelp av aspektene og konvertert til instruksjoner i pakkefilterspråk, idet sikkerhetssystemet omfatter en pakkefiltermodul koblet til datamaskinnettverket idet pakkefiltermodulen opererer ifølge sikkerhetsregelen, og pakkefiltermodulen implementerer en virtuell pakkefiltreringsmaskin som inspiserer og selektivt modifiserer datapakkene som passerer inn og ut av datamaskinnettverket og prosesseringsmidler for å lese og å utføre instruksjonene i pakkefilterspråket integrert med pakkefiltermodulen, idet prosesseringsmidlene driver pakkefiltermodulen for enten å godta eller å avvise passering av pakkene inn og ut av datamaskinnettverket og selektivt modifisere de godtatte datapakkene.

Oppfinnelsen skal i det følgende beskrives nærmere under henvisning til det som er illustrert ved hjelp av de vedlagte tegninger hvor:

Fig. 1 er et eksempel på nettverktopologi;

Fig. 2 viser et sikkerhetssystem ifølge den foreliggende oppfinnelsen anvendt ved nettverktopologien i figur 1;

Fig. 3 viser datamaskinskjermen til nettverkadministratoren i figur 2 i større detalj;

Fig. 4 er et flytskjema for delsystemet for å konvertere grafisk informasjon til filterskript;

Fig. 5 er et flyskjema for informasjonsflyt i et datamaskinnettverk som anvender den forliggende oppfinnelse;

Fig. 6 er et flytskjema for drift av pakkefilteret vist i figur 5;

Fig. 7 er et flyskjema som viser operasjonene til den virtuelle maskinen vist i figur 6;

Fig. 8 er et flytskjema av datautvinningsmetoden i figur 7;

Fig. 9 er et flytskjema for den logiske operasjonsmetode i figur 7;

Fig. 10 er et flytskjema for sammenligningsoperasjonsmetoden i figur 7;

Fig. 11 er et flytskjema for metoden for å innføre en alfanumerisk verdi i minnet;

Fig. 12 er et flytskjema for en betinget grenoperasjon;

Fig. 13 er et flytskjema for en aritmetisk og bitvis operasjon;

Fig. 14 er et flytskjema for en oppslagsoperasjon;

Fig. 15 er et flytskjema for en lagringsoperasion;

Fig. 16 er et høynivå-blokkdiagram som illustrerer en eksempelkonfigurasjon med brannmur bygget ifølge den foreliggende oppfinnelse;

Fig. 17 er et høynivå-blokkdiagram som illustrerer data som overføres mellom to brannmurer under en nøkkelutvekslingssesjon;

Fig. 18 er et høynivå-logiskflytdiagram som illustrerer prosessen utført av en brannmur ved overføring av en pakke ved hjelp av kryptering til en annen brannmur under en datautvekslingssesjon;

Fig. 19 er et høynivå-logiskflytskjema som illustrerer prosessen utført av en brannmur ved mottagelse av en kryptert pakke fra en annen brannmur under en datautvekslingssesjon;

Fig. 20 er et høynivå blokkdiagram som illustrerer data som øverføres mellom to brannmurer under en basisk utvekslingsnøkkel;

Fig. 21 er en høynivå-blokkdiagram som illustrerer en eksempelkonfigurasjon som benytter en kundes private datamaskin og en brannmur bygget ifølge den foreliggende oppfinnelse;

Fig. 22 er et høynivå blokkdiagram som illustrerer data overført mellom en privat kundes datamaskin og en brannmur under en utvekslingsnøkkelsesjon; og

Fig. 23 er et høytnivå blokkdiagram som illiustrerer dataene overført mellom en kundes privatmaskin og en brannmur under utvekslingssesjon for basisnøkler.

Sikring av inngående og utgående datapakkeflyt.

Med henvisning til figur 1, vises det et eksempel på nettverkstopologi. Hovedbeliggenheten 100 omfatter en systematadministrator-funksjon innarbeidet i en arbeidsstasjon 102. Denne arbeidsstasjon er koblet til nettverket som omfatter arbeidsstasjonene 104, ruter 110 og gateway 106. Ruter 110 er koblet via satellitt til en fjerntliggende beliggenhet via gatewayen 122. Gatewayen 106 er koblet via ruter 108 til internett. Den fjerntliggende beliggenhet 120 omfatter arbeidsstasjoner 124 som er koblet til nettverket og via gatewayen 122 til internett. Den spesielle konfigurasjon som vises her er valgt som et eksempel bare og er ikke ment å begrense nettverkstyper hvor den foreliggende oppfinnelsen kan operere. Antall konfigurasjoner som et nettverk kan ha er virtuelt begrenset og metoder for å sette opp disse konfigurasjoner er kjent i teknikken. Den foreliggende oppfinnelsen kan operere i en hvilken som helst av disse mulige konfigurasjoner.

Figur 2 viser nettverket i figur 1 i hvilket den foreliggende oppfinnelse er installert. I figur 2 har elementene som også vises i figur 1 samme henvisningstall. Som vist omfatter systemadministratoren 102 en kontrollmodul 210, en pakkefiltergenerator 208, en fremvisningsenhet 206 og et lagringsmedium 212. Pakkefilteret 204 er installert på systemadministratoren, arbeidsstasjonene 104, og gatewayen 106. Gatewayen 106 har to slike filtere, ett i sin forbindelse til nettverket og ett i forbindelsen til ruteren 108. Ruterne 108 og 110 har hver en programmeringsskripttabell som genereres ved sikkerhetssystemet, men som ikke er en del av den foreliggende oppfinnelsen, og som ikke vil beskrives i detalj. Disse tabeller tilsvarer tabellene som anvendes vanligvis ved programrutere, noe som er kjent på området.

Pakkefiltrene 204 er også installert i gatewayen 122 i den fjerntliggende beliggenhet 120. Et pakkefilter er installert i koblingen mellom satellitten og gatewayen 122, et andre pakkefilter er installert i koblingen mellom internett og gatewayen 122 og et tredje pakkefilter er installert i koblingen mellom gatewayen og nettverket.

Som kjent for fagmannen, flyter informasjon i nettverket i form av pakker. Plassering av pakkefilterne i figur 2 er valgt slik at dataflyt til eller fra et bestemt objekt i nettverket, slik som en arbeidsstasjon, en ruter og/eller en gateway kan kontrolleres. Således har arbeidsstasjonene 104 hver et pakkefilter slik at informasjonsflyt til/fra disse arbeidsstasjoner kontrolleres adskilt. Ved den fjernliggende beliggenhet 120, er imidlertid pakkefilteret plassert i koblingen mellom gatewayen 122 og nettverket, således er det ingen individuell kontroll over dataflyt til/fra stasjonene 124. Hvis en slik individualisert kontroll er krevd, kan pakkefiltrene plasseres i hver av arbeidsstasjonene 124 også. Hvert pakkefilter er installert ved tidspunktet for oppsettet av nettverket og ved installasjon av sikkerhetssystemer, selv om ytterligere pakkefiltere kan installeres ved senere tidspunkter. Pakkefiltrene er innstallert på hver anordning slik som arbeidsstasjonen eller gatewayen ved hvilken beskyttelse er ønsket.

Hvert pakkefilter opererer med et sett av instruksjoner som er generert ved pakkefiltergeneratoren 208 i systemadministratoren 102. Disse instruksjoner tillater utførelse av komplekse operasjoner ved pakken, heller enn bare sjekking av innholdet i pakken mot en tabell omfattende parametrene for å godta eller å avvise pakken. Således kan hvert pakkefilter håndtere forandringer i sikkerhetsreglene med stor fleksibilitet og også håndtere multiple sikkerhetsregler uten å forandre strukturen til selve pakkefilteret.

Systemadministratoren innfører sikkerhetsregelen via et grafisk brukergrensesnitt (GUI) som vises i monitoren 206 og som forklares i større detalj med henvisning til figur 3. Denne informasjon prosesseres ved pakkefiltergeneratoren 208 og den resulterende kode overføres til det riktige pakkefilter eller filtre i nettverket som skal utføre den ønskede funksjon. Kontrollmodulen 210 tillater at systemadministratoren holder rede på operasjonene i nettverket, og lager 212 kan benyttes for å føre en logg over operasjoner på nettverket og forsøk på ulovlig inngang i nettverket. Systemoperatoren kan derved få full rapport på nettverksoperasjonen og på suksess eller feil ved sikkerhetsreglene. Dette setter sikkerhetsadministratoren i stand til å foreta forandringer som er hensiktsmessige for å opprettholde nettverkssikkerheten uten å begrense nettverkets oppkoblingsmuligheter. Figur 3 viser dataskjermen 206 i figur 2 i større detalj. Dataskjermen er delt i 4 vinduer, to mindre vinduer på venstre side og 2 større vinduer på høyre side. Nettverkobjekter og tjenester er to

aspekter av nettverket som skal defineres i sikkerhetsmetoden ifølge den foreliggende oppfinnelse. Vinduet 304 anvendes for å definere nettverksobjekter slik som arbeidsstasjoner, gatewayer og andre datamaskinutstyr tilkoblet systemet. Det er også mulig å gruppere flere anordninger sammen slik som for eksempel finansavdelingen, forskning og utvilingsavdelingen, selskapets direktører. Det er således mulig å kontrollere dataflyt ikke bare til individuelle datamaskiner i nettverket, men også til grupper av datamaskiner i nettverket ved riktig plassering av pakkefiltrene. Dette gir systemoperatøren stor fleksibilitet i håndtering av kommunikasjoner i nettverket. Det er mulig for eksempel å gi sjefen på finansavdelingen og andre hierarki ansatte i firmaet slik som CEO og direktørene mulighet til å kommunisere direkte med finansgruppen, og å filtrere ut kommunikasjoner fra andre grupper. Det er også mulig å tillate elektronisk post fra alle grupper men å begrense andre informasjonskrav til et spesifisert sett av datamaskiner. Dette setter systemoperatøren i stand til å frembringe intern og ekstern sikkerhet i nettverket. Objektdefinisjonen vil omfatte objektets adresse i nettverket såvel som navn og gruppe, hvorvidt objektet er internt eller eksternt i forhold til nettverket, hvorvidt et pakkefilter er installert på objektet, og et grafisk symbol. Det grafiske symbol anvendes i forbindelse med den regelbaserte administratoren 302.

På samme måte defineres nettverktjenester i blokk 306 på skjermen. Disse nettverktjenester kan omfatte innlogging, ruting, syslog og telnet for eksempel. Hver tjeneste defineres ved generiske og spesifikke egenskaper. De generiske egenskaper omfatter kodestrengen som identifiserer tjenesten for eksempel "dgateway" (bestemmelsessted gateway) som er lik 23 for telnet. Kodestrengen som identifiserer de inngående og utgående pakkene er identifisert. De spesifikke egenskaper omfatter tjenestens navn, gatewayen som anvendes for å tilveiebringe tjenesten, tidsavbrudd i sekunder for hvor lenge en sesjon uten tilkobling kan være inaktiv, dvs., uten at noen pakke overføres i noen av retningene før man antar at sesjonen er fullført. Andre elementer i tjenestedefinisjonen kan omfatte programnummer for RPC tjenester og de utgående koblinger for mottatte tjenester som bruker koblingsløse protokoller slik som UDP. Det grafiske symbol og farven er spesifisert.

Blokk 302 er regelbasestyreren eller administrator som tillater at en ny sikkerhetsregel legges inn i systemet på grafisk måte, og på denne måte

frigjøres systemadministratoren fra å behøve å skrive kode for å implementere en bestemt sikkerhetsregel eller for å forandre en sikkerhetsregel. Bare fire elementer kreves for å innføre den nye sikkerhetsregel i systemet. Det første element er kilden for datapakken og det tredje element er bestemmelsesstedet for pakken. Det andre element er type tjeneste som regelen gjelder for og det fjerde element er handlingen som skal foretas. Handlingen som kan foretas omfatter å godta pakken i hvilket tilfelle pakken passerer fra kilden til bestemmelsesstedet eller å avvise pakken i hvilket tilfelle kilden passerer ikke fra kilden til bestemmelsesstedet. Hvis pakken er avvist, kan ingen handling foretas eller et negativt kvitteringssignal kan sendes som angir at pakken ikke har passert til bestemmelsesstedet. Et ytteligere element som kan spesifiseres, er installasjonsplassering for regelen som spesifiserer på hvilke objekter regelen skal tre i kraft (se fig. 2). Hvis en installasjonsplassering ikke er spesifisert, plasserer systemet pakkefiltermodulen på kommunikasjonsbestemmelsesstedet. Disse objekter er ikke nødvendigvis bestemmelsesstedet. For eksempel må en kommunikasjon fra internett med bestemmelsessted en lokalvert, nødvendigvis passere gjennom en gateway. Derfor er det mulig å håndheve regelen på gatewayen, selv om gatewayen hverken er kilden eller bestemmelsesstedet. Ved å innføre data med akronymer eller grafiske symboler, kan hver regel raskt innføres og verifiseres uten behov for å skrive, å kompilere, og å sjekke ny kode for dette formål. Således behøver ikke systemadministratoren å være ekspert i programmering av en datamaskin for sikkerhetsformål. Så lenge tjenesten er en av tjenestene som allerede er innført i systemet vil datamaskinen som tjener som vert for systemadministratoren prosessere informasjonen til et sett instruksjoner for det riktige pakkefilter, som beskrevet i større detalj nedenfor.

Blokk 308 er et øyeblikksbilde av systemet som viser oppsett og operasjonen av sikkerhetssystemet. Den er ikke nødvendig for å utføre den foreliggende oppfinnelse. Øyeblikksbildet fremviser et sammendrag av systemet ved hjelp av grafiske symboler. Sammendraget kan omfatte, for eksempel, vert-ikonet, vert-navn, regelbasenavn, som er navnet til filen som omfatter regelbasen, og dato for installasjon av regelbasen i verten. Den kan også vise tilstanden til verten og angi om det har vært kommunikasjoner med verten eller ikke og også antall pakker inspisert ved, droppet og logget av verten.

Kontroll passerer da til blokk 410 i hvilken kode genereres til å samsvare med regeltjenestene som var valgt. Regeltjenestene har vært definert på forhånd og er lagret innenfor systemet eller i de tilfeller de ikke er definerte vil defineres ved tidspunktet der sikkerhetsregelen som styrer tjenesten innføres i systemet. Kontroll passerer da til 412 i hvilken kode genereres for å godta eller avvise pakken hvis datablokkene 406, 408 og 410 var i samsvar, dvs. at resultatene av testen var sanne. Handlingen å godta eller å avvise baseres på handlingen valgt i sikkerhetsregelen. Kontroll passerer da til bestemmelsesblokk 414 hvor det fastslås om flere regler skal innføres i systemet eller ikke. Hvis ingen flere regler skal innføres i systemet, slutter delsystemet ved blokk 418. Hvis flere regler skal innføres i systemet, passerer kontroll til blokk 416 som fastslår neste regel og passerer kontrollen tilbake til 406 på hvilket tidspunkt prosessen gjentas og neste sikkerhetsregel, funnet i neste tekstlinje i GUI prosesseres.

Kommunikasjonsprotokoller er lagdelte, noe som også omtales som protokollstabel. ISO (International Standardization Organisation) har definert en generell modell som tilveiebringer en ramme for utforming av kommunikasjonsprotokollag. Denne modell tjener som grunnleggende referanse for å forstå funksjonaliteten av ekstisterende kommunikasjonsprotokoller.

ISO MODELL

Lag	Funksjon	Eksempel
7	Anvendelse	Telnet, NFS, Novell NCP
6	Presentasjon	XDR
5	Sesjon	RPC
4	Overføring	TCP, Novell SPX
3	Nettverk	IP, Novell IPX
2	Datalink (maskinvare grensesnitt)	Lenke
1	Fysisk (maskinvare tilkobling)	

Forskjellige kommunikasjonsprotokoller anvender forskjellige nivåer i ISO modellen. En protokoll i et bestemt lag kan være uvitende om protokoller anvendt ved andre lag. Dette er en viktig faktor ved sikkerhetshandlinger. For eksempel, en anvendelse (nivå 7) kan eventuelt ikke være i stand til å identifisere datamaskinkilden for et kommunikasjonsforsøk (nivå 2-3), og kan derfor evenutelt ikke være i stand til å tilveiebringe tilstrekkelig sikkerhet.

Figur 5 viser hvordan en filterpakkemodul ifølge den foreliggende oppfinnelsen anvendes innenfor ISO modellen.

Kommunikasjonslagene ved ISO modellen vises ved 502 på venstre siden i figur 5. Nivå 1, blokk 504, er maskinvaretilkobling på nettverket som kan være ledningen anvendt til å koble de forskjellige objekter i nettverket. Det andre nivå, blokk 506 i figur 5 er nettverkets grensesnitt maskinvare som er anordnet ved hver datamaskin på nettverket. Pakkefiltermodulen ifølge den foreliggende oppfinnelsen arbeider mellom dette nivå og nivå 3 som er nettverkets maskinvare. De andre nivåer på ISO modellen er nivå 4, blokk 510 som vedrører utsendelsesdata fra et segment til neste segment, nivå 5, blokk 512 som synkroniserer åpning og lukking av en sesjon på nettverket. Nivå 6, blokk 514 som vedrører en datautveksling mellom flere datamaskiner på nettverket, og nivå 7, blokk 516 er anvendelsesprogrammet.

En pakke som går inn i datamaskinen på hvilken pakkefiltermodulen er anordnet passerer gjennom lag 1 og 2 og avledes deretter til pakkefilteret 520, vist på høyre side i figur 5. Pakken mottas i blokk 522. I blokk 524, sammenlignes pakken med sikkerhetsregelen og en avgjørelse tas på om pakken oppfyller regelen eller ikke. Hvis pakken oppfyller regelen, kan den tas inn på systemadministratoren's log og hvis et ulovlig forsøk på å komme inn i systemet har vært utført, kan en alarm utløses. Kontroll passerer da til blokk 534 i hvilken en avgjørelse tas på om pakken passerer eller ikke, basert på kravene i sikkerhetsregelen. Hvis avgjørelsen er å passere pakken, passeres pakken til nivå 3, blokk 508. Hvis avgjørelsen er å ikke la pakken passere, sendes en nagativ kvittering (NACK) ved blokk 528, hvis denne mulighet har vært valgt, og kontroll overføres til blokk 530 hvor pakken droppes, det betyr at den ikke passerer til bestemmelsesstedet. På samme måte, hvis en anvendelse genererer en pakke som skal sendes til et annet bestemmelsessted, forlater pakken ISO modellen ved nivå 3, blokk 508 og

kommer inn i blokk 522 og samme prosess utføres med untak av hvis pakken skal passere, sendes den til nivå 2, blokk 506 og ikke nivå 3, blokk 508. I nivå 2 sendes pakken til nettverket ved blokk 504, nivå 1. Hvis pakken ikke oppfyller regelen, vil neste regel bli funnet og pakken undersøkt for å fastslå om den oppfyller denne regel. En "default" regel tilveiebringes som en hvilken som helst pakke vil oppfylle uansett kilden, bestemmelsessted eller tjeneste som er spesifisert. Denne "tomme regel" har bare en hendelse, som er å droppe pakken. Hvis ingen andre regler er godtatt, vil denne regel bli funnet og vil utføres til å droppe pakken. Dropping av pakken er det sikreste trinn å utføre under disse omstendigheter. Den "tomme regel" kan selvfølgelig settes til å la pakken passere.

Det henvises nå til figur 6, hvor 600 er en detaljert beskrivelse av blokk 520 i figur 5. Den generelle beskrivelse i figur 6 og de mer detaljerte beskrivelser vist i figurene 7-10 omfatter en definisjon av begrepet "pakkefiltermodul" som anvendes ved dem. Mulighetene vist i disse figurer er de minimale muligheter for pakkefiltermodulens drift. Figurene 11-15 viser ytterligere trekk som også kan innbefattes av pakkefiltermodulen, men som ikke er krevet i den mindre definisjonen av begrepet.

Pakkefiltermodulen er anordnet i en virtuell maskin, som, i forbindelse med denne søknad, kan defineres som en etterligning av maskinen vist i figurene 6-10 liggende i hver datamaskin, som er en datamaskin på nettverket.

Den virtuelle maskin starter ved blokk 602 i hvilken pakken mottas, tilsvarende i blokk 522 i figur 5. Kontroll overføres til blokk 604 i hvilken filteroperasjonene mottas fra instruksjonen i et minne (ikke vist). Disse filteroperasjoner er filteroperasjonene som ble generert ved pakkefiltergeneratoren 208 vist i figur 2. Kontroll overføres da til blokk 604 i hvilken filteroperasjonene finnes og deretter til blokk 606 i hvilken minnet 618 initialiseres. I blokk 608, oppnås den første operasjon av den virtuelle maskin og den utføres i blokk 610. Den virtuelle maskin omfatter en minnemekanisme slik som en stabel eller et register 618 som kan anvendes til å lagre mellomliggende verdier. Anvendelse av denne stabel eller dette register vises i større detalj i forbindelse med tabellen vist nedenfor. Kontroll passerer da til avgjørelseseblokk 614 i hvilken det fastslås om stopptilstanden har vært nådd eller ikke. Hvis stopptilstanden er nådd, har avgjørelsen vært tatt om å godta eller avvise pakken, denne avgjørelsen utføres ved blokk 616.

Hvis pakken er passert, vil pakken forflytte seg som vist på figur 5. Hvis pakken er avvist, vil den droppes og en negativ kvittering kan sendes som vist i blokker 528 og 530. Hvis stopptilstanden ikke er nådd i blokk 614, er neste operasjon tatt inn i blokk 616 og prosessen gjentas med startpunkt i blokk 610.

De typer operasjoner som kan utføres i trinn 5, blokk 610 vises klarere i figur 7. I figur 7 er blokk 610 og blokk 614 identiske med blokkene vist i figur 6. Koblingen 613 avbrytes ved 3 operasjoner som vises i parallell. For operasjonen som skal utføres i blokk 610, vil kontroll passere til den riktige blokk 702, 704 eller 706 i hvilken oppgaven skal utføres. I blokk 702 utføres datautlesning, i blokk 704 utføres logiske operasjoner og i blokk 706 utføres en sammenligningsoperasjon. Som vist på høyre side i figur 7, kan andre blokker adderes i parallell til operasjonene som den virtuelle maskinen er i stand til å utføre. Blokkene 702, 704 og 706 er hovedelementene i den virtuelle maskin ved den foreliggende oppfinnelse. Disse elementer vises i større detalj i figurene 8, 9, henholdsvis 10. Ytterliggere elementer som kan valgfritt inkluderes i operasjonene som den virtuelle maskin er i stand til å utføre, vises i respektive figurene 11- 15.

Datautvinningsblokk 702 vises i større detalj i figur 8. Prosessen starter ved blokk 802 og kontroll går til 804 i hvilke data hentes fra en spesifikk adresse innenfor pakken 806. Denne adresse tas fra stabelminne 618 eller fra instruksjonskoden. Datamengden som hentes er også fastslått ved stabelminnet eller instruksjonskoden. De hentede data settes inn i minnestabelen 810 ved blokk 808. Prosessen avsluttes ved blokk 812. I disse figurer vises kontrollflyt ved piler med en enkel linje mens dataflyt vises med piler med doble linjer.

Figur 9 viser den logiske operasjon 704 i større detalj. Den logiske operasjon starter ved blokk 902 og kontroll overføres til blokk 904 i hvilken en første verdi tas inn fra minne 906. I blokk 908 mottas en andre verdi fra minnet og den logiske operasjonen utføres i blokk 910. Hvis den logiske operasjonen er sann, plasseres en én i minne 906 ved blokk 912 og hvis den logiske operasjonen er usann, plasseres en null i minne 906 ved blokk 914. Prosessen avsluttes ved blokk 916.

Den tredje og siste nødvendige operasjon for den virtuelle maskin vises i større detalj i figur 10. Sammenligningsoperasjonen, blokk 706, starter ved

1002 og kontroll overføres til 1004 i hvilken den første verdi mottas fra minnet 1006. Kontroll overføres til blokk 1008 i hvilken en andre verdi mottas fra minnet 1006. En sammenligningsoperasjon mellom den første og den andre verdi foregår ved blokk 1010. Hvis sammenligningsopersjonen er sann, plasseres en èn i minnet 1006 ved blokk 1012, og hvis sammenligningsoperasjonen er usann plasseres en null i minnet 1006 ved blokk 1014. Prosessen avslutter ved blokk 1016.

De følgende operasjoner vises ikke i figur 7 men de kan adderes på høyreside av figuren ved de stiplede linjer og er tilkoblet på samme måte som blokk 702, 704 og 706, dvs. i parallell. Figur 11 viser inngangen av en alfanumerisk verdi i minnet. Prosessen starter ved blokk 1102 og kontroll overføres til blokk 1106 i hvilken den alfanumeriske verdi mottas fra instruksjonskoden. Verdien plasseres i minne ved blokk 1108 og prosessen slutter ved blokk 1110.

En betinget grenoperasjon vises i figur 12. Prosessen starter ved blokk 1202 og kontroll passerer ved 1204 i hvilken gren operasjonen, mottatt fra instruksjonskoden, testes. Hvis grenoperasjonen er sann, mottas verdien fra minnestabelen 1206 ved blokk 1208 og sjekkes ved blokk 1210. Hvis resultatene av sammenligningen i blokk 1210 er sanne, er neste trinn satt til N og prosessen avslutter ved blokk 1216. Hvis sammenligningen ved blokk 1210 er usann, avslutter prosessen ved blokk 1216. Hvis grenbetingelsen er usann, overføres kontroll direkte til blokk 1214, ved blokk 1204.

En aritmetisk eller bitvis operasjon vises i figur 13. Prosessen starter ved blokk 1302 og kontroll overføres til blokk 1304 i hvilken den første verdi mottas fra minnet 1306. Den andre verdi mottas fra minnet 1306 ved blokk 1308 og en aritmetisk eller bitvis operasjon utføres på de to verdier mottatt fra minnet 1306 i blokk 1310. Resultatet av den aritmetiske eller bitvise operasjon plasseres i minne i blokk 1312 og prosessen avsluttes i blokk 1314.

Figur 14 viser en oppslagsoperasjon som kan anvendes hvis data skal passeres fra et første instruksjonssett som implementerer en sikkerhetsregel til et andre instruksjonssett for en annen sikkerhetsregel. Som vist i blokk 606 i figur 6, initialiseres minnet hver gang en ny sikkerhetsregel prosesseres. Derfor vil ikke informasjonen plassert i minnet ved en første sikkerhetsregel være tilgjengelig for anvendelse ved en annen sikkerhetsregel. Med den hensikt å løse dette problem, tilveiebringes et

separat minne 1410 som omfatter tabellene 1-3 som kan anvendes for dette formål. Innføring av data i tabellene vises i figur 15 og beskrives nedenfor. Oppslagsoperasjonen starter ved 1402 og kontroll overføres til 1404 hvor verdiene mottas fra minne 1406. Kontroll overføres til blokk 1408 i hvilken data mottas fra tabellene 1-3 ved blokk 1410 ved å slå opp verdiene i nevnte tabeller. Kontroll overføres til blokk 1412 i hvilken en avgjørelse tas om hvorvidt blokken er i tabellen eller ikke. Hvis avgjørelsen er positiv, plasseres en èn i minnet 1406 ved blokk 1416. Hvis avgjørelsen er nei, plasseres en null i minnet 1406 ved blokk 1414. Prosessen avsluttes ved blokk 1418.

Med henvisning til figur 15, starter prosessen ved blokk 1502 og kontroll overføres til blokk 1504 i hvilken verdier mottas fra minne 1506. Kontroll overføres da til blokk 1508 i hvilken verdiene mottatt fra minnet 1506 plasseres i de riktige plasseringer i tabellene 1-3 ved blokk 1510. Kontroll overføres til blokk 1512 i hvilken en avgjørelse tas om lagring av verdiene i tabellen har vært vellykket. Hvis lagring er vellykket plasseres en èn i minnet 1506 ved blokk 1516. Hvis lagring ikke er vellykket, plasseres det en null i minnet 1506 ved blokk 1514. Prosessen avsluttes ved blokk 1518.

Et eksempel på en sikkerhetsregel som implementeres ved hjelp av pakkefiltreringsfremgangsmåten ifølge den foreliggende oppfinnelse vil nå beskrives og det skal anvendes sikkerhetsregel for å nekte Telnet-tjenester i systemet som eksempel. Telnet er definert som en TCP tjeneste med en spesifikk TCP bestemmelsessted-gateway. Den vil identifiseres ved at den har en TCP protokollverdi på 6 i byte-plassering 9 i pakken, og ved at den har et protokollnummer 23 for bestemmelsessted i Telnet i byte-plassering 22 i pakken, idet verdien er en to-byte-verdi. Dette finner man på hver Telnet anmodningspakke.

Den første operasjonen i tabellen vist nedenfor er å utvinne IP protokollen fra pakkeplasseringen 9 og plassere den i minnet. Som vist i "minneverdier" kolonnen på høyre side av tabellen, plasseres denne verdi, 6, på toppen av stabelen.

Den andre operasjonen, TCP protokoll (gateway) nummer, som fastslås å være 6 ovenfor, plasseres ved den andre plassering i minnet. I trinn 3 sammenlignes verdiene på de første to lag i stabelen, og et positivt resultat oppnås.

Droppe Telnet Prosedyre

#	Pakke-filterkode	Virtual maskin operasjon	Minneverdier (Stabelrekkefølge)		
1	pushbyte [9]	Utvinningsoperasjon: Utvinn IP protokollnummer fra pakkeplassering 9 til minne.	6		
2	push 6	Innfør alfanumerisk verdi i minnet: Putte TCP protokollnummer i minnet.	6	6	
3	eq	Sammenligningsoperasjon: Sammenlign IP protokoll og TCP, oppnåelse av positivt resultat.	1		
4	push [22]	Henteoperasjon: Hent TCP protokoll-nummer fra pakkeplassering 22 til minne.	1	23	
5	push 23	Innfør alfanumerisk verdi i minne: Putt TELNET protokollnummer i minne.	1	23	23
6	eq	Sammenligningsoperasjon: Sammenlign TCP protokoll og TELNET, oppnåelse av positivt resultat.	1	1	
7	og	Logisk operasjon: Sjekk om både protokoll TCP og TELNET er oppfylt.	1		
8	btrue drop	Betingelsesgrenoperasjon: Hvis minneverdi er sann, gå til dropptilstand.			

Verdiene på 6 ved de to topp-lagene i stabelen slettes og en 1, som angir det positive resultat, plasseres på toppen av stabelen. I trinn 4 utvinnes TCP protokollnummer for pakkplasseringen 23 og plasseres i minneplassering ved det andre lag i stabelen. I trinn 5 plasseres den alfanumeriske verdi som er Telnet protokollnummeret i minnet ved stabelens tredje lag. I trinn 6, sammenlignes minnelag 2 og 3 omfattende TCP protokollen for Telnet med den forventede verdi, og et positivt resultat oppnås. Verdiene av det andre og tredje lag i stabelen slettes og erstattes med en 1, som angir det positive resultat. I trinn 7 utføres en logisk operasjon for å undersøke om både TCP og Telnet har vært oppfylt. Dette fastslås ved en annen operasjon. I dette tilfelle er resultatet positivt og enerne i de første to lag i stabelen slettes og erstattes ved en 1, som angir det positive resultat. I trinn 8 utføres en betinget grenoperasjon, i hvilken, hvis minneverdien er sann, programmet vil gå til dropptilstanden. I dette tilfelle, er resultatet sant og programmet går til droppetilstanden i hvilken Telnet sin anmodning ikke aksepteres. På denne måten er regelen å droppe Telnet implementert.

Kryptering av dataflyt. Innføring.

Som nevnt ovenfor, er fjernkommunikasjoner mellom firmaer, avdelingskontorer og forretningspartnere en betydelig del av moderne forretningsvirksomhet. Ved hjelp av den foreliggende oppfinnelse, kan virtuelle private nettverk (VPN) bygges over usikre offentlige nettverk slik som internett for å tilveiebringe sikre og fleksible kommunikasjoner.

Pakkemodifikasjonen ved for eksempel kryptering av utgående pakker, dekryptering av inngående pakker, signering av pakker eller adresseoversettelse utføres i pakkefiltermodulen. Avgjørelsen om å modifisere en pakke eller ikke, fastslås fra regelbasen. Alle forandringer dvs. kryptering, dekryptering, signering og adresseoversettelse utføres på en selektiv basis ifølge innholdet i regelbasen. For at kryptering for eksempel skal finne sted, må en regel i regelbasen uttrykkelig angi at kryptering skal foregå ved pakker som har en bestemt kilde, bestemmelsessted, og tjenestetype. Krypteringsinstruksjonene oversettes til pakkefilterspråk som er installert og utføres på de virtuelle pakkefiltermaskinene i nettverket.

Som nevnt ovenfor, avgjør pakkefiltermodulen om en pakke skal avvises eller godtas. Hvis den avvises, droppes pakken. Hvis den aksepteres, kan pakken modifiseres på et antall måter. Eksempler på mulige modifikasjoner

omfatter, men er ikke begrenset til, kryptering, dekryptering og adresseoversettelse. Nedenfor beskrives i detalj kryptering og dekryptering av pakker som blir selektivt utført ved pakkefiltermodulen.

Betegnelsesbeskrivelse.

De følgende betegnelser anvendes i dokumentet:

Symbol	Beskrivelse
g	felles rot anvendt for alle Diffie-Hellman nøkler
p	felles modulus anvendt for alle Diffie-Hellman nøkler
S_{pvt}	kildens private nøkkel
S_{pub}	kildens offentlige nøkkel
D_{pvt}	bestemmelsesstedets private nøkkel
D_{pub}	bestemmelsesstedets offentlige nøkkel
B	basisnøkkel
TB	avkuttet basisnøkkel
A	hjelpenøkkel
R	sesjonsnøkkel
E	sesjonens datakrypteringsnøkkel
I	sesjonens dataintegritetsnøkkel
M	datadel av en pakke
P	ikkekryptert passord
$ENC_x(Y)$	krypter Y med X som nøkkel
$DCR_x(Y)$	dekrypter Y med X som nøkkel
SIG(Y)	signatur for Y

Definisjoner av anvendte begreper.

De følgende definisjoner kan være til hjelp for å forstå operasjonen av den foreliggende oppfinnelse.

Begrep	Definisjon
Ren tekst	tekst som ikke er kryptert.
Klar tekst	et annet begrep for ikke-kryptert tekst.
Siffertekst	kryptert tekst.
Nøkkel	et stykke informasjon som bare en kjent for senderen og for den ønskede mottager.
Kryptering	å konvertere renteksten i en beskjed til siffertekst for å gjøre beskjeden uleselig for de som ikke har nøkkelen.
Dekryptering	å konvertere siffertekst til rentekst ved hjelp av samme nøkkel anvendt for å kryptere beskjeden.
Sertifisering	en pålitelig tredje part, kjent som sertifiseringsmyndighet (CA), fra hvilken man med pålitelighet kan motta en offentlig nøkkel, til og med over en usikker kommunikasjonskanal, og som genererer et sertifikat for den offentlige nøkkel som kan verifiseres av mottageren.
Digital signatur	informasjon generert ut fra innholdet til selve beskjeden og anvendt av mottageren for å verifisere dataintegritet ved beskjeden og/eller dens opprinnelse.
Nettverksobjekt	en stykke datamaskinvare som er koblet til nettverket og som har et bestemt samvirke med nettverket.
Gateway	et nettverksobjekt som er koblet til i det minste to nettverk og som passerer informasjon mellom dem.
Brannmur eller Brannsikret nettverksobjekt	et nettverksobjekt, vanligvis en gateway eller en endevert, som sikrer flyt av inngående og utgående datapakker i et datamaskinnettverk og som også selektivt modifiserer datapakker ifølge en sikkerhetsregelbase.

Et høynivåblokkdiagram som illustrerer en eksempelkonfigurasjon som anvender brannmur bygget ifølge den foreliggende oppfinnelse vises i figur 16. Eksempelnettverket vist i denne figur vil anvendes til å forklare krypteringsmulighetene ved den foreliggende oppfinnelse.

Nettverkkonfigurasjonen vist er bare for illustrasjonsformål. En fagmann på området kan tilpasse den foreliggende oppfinnelse til andre nettverkkonfigurasjoner. Både vert1 og vert2 er tilkoblet deres respektive private LANs. I tillegg er brannmur1 1604 koblet til vert1 gjennom dens LAN og brannmur2 er koblet til vert2 gjennom dens LAN. Begge brannmurer er koblet til et offentlig nettverk 1606 slik som internett. Det er også antatt at det offentlige nettverk er usikkert og ikke kan stoles på. Sertifiseringsmyndighet 1 (CA1) 1602 fungerer som sertifiseringsmyndighet for vert1 og brannmur1. CA2 1612 fungerer som sertifiseringsmyndighet for vert2 og brannmur2. I andre utførelser, kan det anvendes bare en enkel CA for begge brannmurer. Uansett utførelse er funksjonene til CA de samme. Den eneste forskjell er hvilken CA som anvendes ved brannmuren for å motta offentlige nøkler.

Det er ønskelig at kommunikasjonen mellom vert1 og vert2 skal sikres. Kommunikasjonene fra vert1 rutes til internett (dvs. det offentlige nettverk) via brannmur1 som fungerer som et brannmur-nettverksobjekt. På samme måte rutes kommunikasjonene fra vert2 til internett via brannmur2 som også fungerer som et brannmur-nettverksobjekt. Ved kommunikasjonene til vert2, vil brannmur1 sperre av og kryptere pakkene som den mottar fra vert1 i rute til vert2. Brannmur2 mottar de krypterte pakkene med bestemmelsessted vert2 og dekrypterer slike pakker. I den motsatte retning, krypterer brannmur2 pakkene fra vert2 ved bestemmelsessted vert1. Brannmur1 mottar de krypterte pakker, dekrypterer dem og lar dem passere til vert1. Kryptering- og dekrypteringsoperasjonene utført ved brannmur1 og brannmur2 er transparent for vert1 og vert2.

Gitt at vert1 initierer sesjonen med vert2, sender den en internett-protokollpakke til vert2. Brannmur1 vil sperre av pakken og fastslå at kommunikasjonene mellom vert1 og vert2 skal modifiseres på en bestemt måte, for eksempel kryptering, dekryptering, adresseoversettelse, osv. Avgjørelsen tas separat for hver kobling basert på informasjon fra alle ISO-lagene og basert på informasjon beholdt fra tidligere pakker. Denne avgjørelsesprosess kalles for tilstandsflerlags-inspeksjon (SMLI). Hver

brannmur vedlikeholder en regelbase som instruerer brannmuren om hvordan inngående og utgående kommunikasjon mellom nettverksobjekter skal håndteres, som beskrevet i detalj ovenfor. Etter å fastslå at kommunikasjonene mellom vert1 og vert2 skal krypteres eller signeres digitalt, parkerer brannmur1 pakken midlertidig og starter en utveksling av sesjonsnøkkel, som beskrives i større detalj nedenfor. Før kryptert kommunikasjon eller signering kan utføres, må begge sider være enige om en delt nøkkel. Denne nøkkel kalles for sesjonsnøkkel, R, og genereres på nytt ved start av hver sesjon. Det er viktig å merke seg at bare kommunikasjonen mellom brannmur1 og brannmur2 er kryptert på grunn av anvendelse av det usikre internett eller offentlige nettverk. Kommunikasjonene mellom vert1 og brannmur1 og mellom vert2 og brannmur2 er ikke kryptert fordi de foregår over private LANs som kan antas å være private og sikre.

Utveksling av sesjonsnøkkel - brannmur/brannmur

Et høynivå blokkdiagram som illustrerer dataene overført mellom to brannmurer under en utveksling av sesjonsnøkkel vises i figur 17. Det følgende skjema er bare et eksempel på implementering av kryptering med SMLI og er ikke ment å begrense omfanget av den foreliggende oppfinnelse til andre krypteringsteknikker. Det er innlysende for en fagmann å tilpasse andre krypteringsteknikker til SMLI prosessen for å utføre læren ved den foreliggende oppfinnelsen. For eksempel i en alternativ utførelse kan SKIP standard anvendes. For å starte datakrypteringen sender brannmur1 først en anmodningspakke til vert2. Anmodningspakken sendes til vert2 og ikke til brannmur2 fordi brannmur1 ikke nødvendigvis kjenner IP adressen til brannmuren som har ansvar for vert2. Brannmur2 sperrer av denne anmodningspakke og sender tilbake en svarpakke. Anmodnings- og svarpakkene tillater at begge sider blir enige om en delt sesjonsnøkkel R som skal anvendes for all kommunikasjon som skal krypteres mellom vert1 og vert2. Som nevnt ovenfor er bare kommunikasjonene mellom brannmur1 og brannmur2 faktisk kryptert.

Generelt genereres sesjonsnøkkel R av den ikke-initierende part (dvs. brannmur2 1608) også kalt bestemmelsesstedet og den sendes kryptert til den initierende part (dvs. brannmur1 1604) også kalt kilden. Denne utveksling av to pakker må foreligge før kryptert kommunikasjon kan utføres. Etter at den krypterte sesjon etableres, opprettholdes tilstandsinformasjon i begge

brannmurer og den originale pakke som var parkert overføres nå kryptert gjennom brannmurene. Den samme sesjonsnøkkel R anvendes ved brannmur2 for å kryptere pakker som skal sendes fra vert2 til vert1.

Utveksling av sesjonsnøkkel vil nå beskrives i større detalj. For å bli enige om en felles hemmelig seksjonsnøkkel R, anvender den foreliggende oppfinnelse et statisk Diffie-Hellman skjema. Hver Diffie-Hellman nøkkel omfatter en privat del og en offentlig del. Hver side har en egen privat og offentlig del. Den private nøkkel for kilden (dvs. brannmur1) og bestemmelsessted (dvs. brannmur2) er S_{PVT} og D_{PVT}, respektivt. De offentlige deler for kilde og bestemmelsessted er da definert som følger:

$$S_{pub} = g^{Spvt} \text{ (mod) } p$$

$$D_{pub} = g^{Dpvt} \text{ (mod) } p$$

Både kilder og bestemmelsessted må kjenne til hverandres offentlige nøkkel for at utveksling av sesjonsnøkkel skal virke. Hvis en side ikke kjenner den andres offentlige nøkkel eller den nøkkelen den har er utdatert, utløses det en utveksling av basisnøkler, noe som forklares i større detalj nedenfor. Begge sider anvender hverandres offentlige nøkkel til å basisnøkkelen B. Kilden utfører det følgende:

$$B = \{g^{Dpvt} \text{ (mod) } p\}^{Spvt} \text{ (mod) } p = g^{SpvtDpvt} \text{ (mod) } p$$

På samme måte utfører bestemmelsesstedet det følgende:

$$B = \{g^{Spvt} \text{ (mod) } p\}^{Dpvt} \text{ (mod) } p = g^{SpvtDpvt} \text{ (mod) } p$$

Således deler begge sider basisnøkkelen B. For anvendelse for kryptering av sesjonsnøkkel R, genereres en avkuttet versjon av den basiske nøkkel B, som kalles TB.

Generelt vedlikeholder hver brannmur en tabell av forbindelser mellom Diffie-Hellman nøkler og brannmurbeskyttede nettverksobjekter. I tillegg må en brannmur ha en forbindelse mellom IP adresser og et slikt objekt. I konfigurasjonen vist i figur 17, må en database innenfor brannmur1 konfigureres slik at den vet om tilstedeværelsen av brannmur2. Brannmur1 må også være klar over at vert2's krypteringsbrannmur er brannmur2. Brannmur1 kan ha en liste av potensielle brannmurer som kan anvendes som krypteringsbrannmurer for brannmur2. Forbindelsene og nettverksobjektets

database for hver brannmur håndteres på statisk måte ved en separat håndteringsenhet.

For å kryptere kommunikasjoner mellom brannmurer, må en brannmur ha kjennskap til sin egen basis private nøkkel og de offentlige basisnøkler for hvert brannmurbeskyttet nettverksobjekt som den behøver å kommunisere med. De offentlige basisnøkler som tilhører eksterne brannmurbeskyttede nettverksobjekter slik som en brannmur som tilhører en forretningspartner må også være kjent for at de krypterte sesjonene skal utføres. Denne statiske forbindelse mellom basiske nøkler og brannmurbeskyttede nettverkobjekter kan allerede etableres i en database internt i brannmuren eller den kan oppnås på stedet ved hjelp av basisnøkkelutveksling beskrevet nedenfor.

Når begge brannmurene er enige om en felles og hemmelig basisnøkkel B, anvendes denne for å kryptere den aktuelle nøkkel som anvendes for sesjonen, Dvs. sesjonsnøkkel. Den samme sesjonsnøkkel R anvendes både av kilden og av bestemmelsesstedet for å kryptere data fra vert1 til vert2 og fra vert2 til vert1.

Elementene i anmodningen fra kilden til bestemmelsesstedet vises over den høyre pil i figur 17. Siffermetoden omfatter en eller flere krypteringsmetoder for å kryptere sesjonsdataene som kilden er i stand til å utføre (dvs., DES, FWZ1, RC4, RC5, IDEA, Tripple-DES, etc). Nøkkelmetoden omfatter en eller flere krypteringsmetoder for å kryptere sesjonsnøkkel R som kilden er i stand til å utføre (for eks., DES, FWZ1, RC4, RC5, IDEA, Tripple-DES, etc.). d-metoden (dvs. meldingssystematiseringsmetode) eller meldingsintegritetsmetode omfatter en eller flere metoder eller algorytmer for å utføre dataintegritet som kilden er i stand til å utføre (dvs., MD5, SHA, etc.). Dataintegriteten omfatter typisk å beregne en kryptografisk spredefunksjon på en del eller hele meldingen.

Den foreslåtte kildens offentlige nøkkel ID identifiserer, via et ID nummer, den offentlige basisnøkkel som kilden angir at bestemmelsesstedet vil anvende. På samme måte identifiserer det foreslåtte bestemmelsessteds basisk offentlige nøkkel ID den offentlige basisnøkkel som kilden angir at bestemmelsesstedet vil anvende. Hvis det er mer enn ett mulig brannmur-beskyttet nettverksobjekt som server vert2, vil kilden omfatte flere foreslåtte offentlige basisnøkler i en anordningspakke siden den ikke vet hvilken av de brannmursbeskyttet nettverksobjekter aktuelt serverer vert2. Hver av de

foreslåtte basisk offentlige nøkler tilsvarer et spesielt brannmursbeskyttede nettverksobjekt.

Anmodningen omfatter også en oppfordringsnøkkel C som er et "random bit" felt valgt av kilden (dvs. brannmur1) og som anvendes til å motarbeide angrep mot sesjonens nøkkelutveksling eller selve sesjonsdata.

Bestemmelsesstedet (dvs. brannmur2) mottar anmodningspakken og basert på dens innhold genererer en svarpakke som skal sendes tilbake til kilden. Elementene i svarpakken vises over den venstre pil i figur 17. Svarpakken har tilsvarende format som anmodningspakken med unntak av oppfordringsnøkkel C feltet som erstattes ved et felt som har den krypterte sesjonsnøkkel R. Hver siffermetode, nøkkelmetode og md metode har nå heller bare ett element enn en liste av valg som i anmodningen. Elementene som er i listen er elementene valgt ved bestemmelsesstedet fra valgmulighetene ført i anmodningen. På samme måte vil den valgte offentlige basisnøkkelide for kilden og den basisk offentlige nøkkel ID for bestemmelsesstedet begge omfatter en enkel nøkkel ID som representerer nøkkel ID'en valgt ved bestemmelsesstedet fra valglisten sendt i anmodningen.

Sesjonsnøkkelen R som sendes i svaret omfatter faktisk to nøkler: en krypteringsnøkkel E for sesjonsdata og en integritetsnøkkel I for sesjonsdata. Således er sesjonsnøkkel R definert som

$$R = E + I$$

Sesjonsnøkkelen er en vilkårlig strøm av bytes som genereres både for siffermetoden (dvs. krypteringsmetoden) og md metoden eller meldingssystematiserings metoden. Lengden er summen av nøkkellengdene som kreves ved siffermetoden og md metoden. Når den er generert, oppnås en signatur av sesjonsnøkkel ved hjelp av den valgte md-metode, for eksempel, MD5, og representert ved SIG(R). Kombinasjonen av sesjon R og SIG(R) er da kryptert ved hjelp av en nøkkel dannet ved kombinasjonen av den avkuttede basisnøkkel TB og oppfordringen C, som således danner

$$ENC_{(TC + C)}(R + SIG(R))$$

som er det som sendes i svaret til kilden.

Signaturen eller beregning av nøklenes kontrollsum tilveiebringer bekreftelse til kilden om at pakken den mottok virkelig er utformet av en enhet som kjenner basisnøkkelen B og derved tilveiebringes sterk pålitelighet for svarpakken. I tillegg, siden kilden valgte oppfordringsnøkkel C er det ingen mulighet for svar.

Utveksling av sesjonsdata

Et høynivå logisk flytdiagram som illustrerer prosessen utført ved en brannmur for overføring av en pakke ved hjelp av kryptering til en annen brannmur under en utveksling av sesjonsdata vises i figur 18. Selv om den ikke vises i figurene, anvender en alternativ utførelse IPSEC standard for utførelse av utveksling av sesjonsdata. Som nevnt ovenfor, når kilden og bestemmelsesstedet er enige om en sesjonsnøkkel R, kan kryptert kommunikasjon mellom begge brannmurer utføres. Avsperring av og forandringer på pakkene foregår mellom nivå 2 og nivå 3 i ISO modellen. Kommunikasjonen som foregår begge veier skal krypteres og dekrypteres ved hjelp av den samme sesjonsnøkkel R. Pakkene som sendes ut ligner veldig på normale TCP/IP pakker. Pakkene omfatter ingen opplysninger som angir om pakkene er krypterte eller ikke, eller hvilken nøkkel som skal anvendes. Denne informasjonen er tilstede bare i tilstanden opprettholdt av begge brannmurene. Krypteringen utføres på stedet uten å forandre pakkens lengde som anvendes for å øke effektiviteten og båndbredde på den krypterte trafikk. Generelt deles hver overførte pakke i to deler, en rentekst del som ikke er kryptert og en siffertekst del som er kryptert. Rentekst delen omfatter IP hodeopplysninger og TCP/UDP hodeopplysninger. Resten av pakken dvs. dataene M er kryptert ved hjelp av en kombinasjon av sesjonsnøkkel R og en hjelpenøkkel A som beregnes fra klartekstdelen. Prosessen skal nå beskrives i større detalj.

Det første trinn utført av en brannmur for overføring av en pakke er å generere en hjelpenøkkel A fra rentekstinnholdet i selve pakken (trinn 1800). Delene som anvendes er avhenglige av typepakken og av protokollen (for eks., RPC, UDP, TCP, ICMP, osv.) og kan omfatte de følgende felt, for eksempel, IP_ID (bare en del), RPC-XID, RPC-PROGNUM, TCP-SEQ, TCP-ACK, UDP-LEN, ICMP-TYPE, ICMP-CODE og IP-PROTO. Deretter plasseres hjelpenøkkel A, integritesnøkkel I for sesjonsdata og datadelen av

pakke M i en buffer (trinn 1802). En signatur genereres da på innholdet i bufferen ved hjelp av md metoden (trinn 1804) og uttrykket ved

$$SIG(A + I + M)$$

Bitene av signaturen som genereres plasseres da i pakkens topptekst (trinn 1806). Addering av signaturbitene til pakken er viktig for å sikre dataintegritet. Siden pakkens lengde ikke modifiseres må noen deler av pakken overskrives med signaturbitene. Merk at signaturbitene lagres i pakken før pakken krypteres. For TCP pakker lagres en 28 bit signatur som følger:

- de 8 LSBits i signaturen erstatter de 8 MSBits i IP-ID
- de neste 16 bits adderes til TCP-CSUM feltet ved hjelp av 1 komplement aritmetikk.
- de neste 4 bits lagres i de ubrukte TCP-X2 nibbel (dette er valgfritt)

For UDP pakker lagres en 32 bit signatur som følger:

- de første 16 bits i signaturen adderes til UDP-CSUM feltet ved hjelp av 1 komplement aritmetikk; hvis det opprinnelige UDP-CSUM felt er null, UDP-SGATEWAY og UDP-DGATEWAY feltene adderes til UDP-CSUM også ved hjelp av 1 komplement aritmetikk
- de neste 16 bits lagres i UDP-LEN feltet.

Når signaturbitene er lagret i pakken, krypteres datadelen av pakken M (trinn 1808), og kan uttrykkes ved

$$ENC_{(E + A)}(M)$$

Krypteringen utføres ved hjelp av siffermetoden med en kombinasjon av krypteringsnøkkel for sesjonsdata E og hjelpenøkkelen A. Til slutt overføres pakken over det offentlige nettverk (trinn 1810).

Et høytnivå logisk flytdiagram som illustrerer prosessen utført ved en brannmur ved mottagelse av en kryptert pakke fra en annen brannmur under en utveksling av sesjonsdata vises i figur 19. Først, med den hensikt å verifisere signaturen, må hjelpenøkkelen genereres fra innholdet av pakken

(trinn 1900). Deretter, dekrypteres pakken's datadel M ved hjelp av siffermetoden og en kombinasjon av krypteringsnøkkel for sesjonsdata E og hjelpenøkkel A (trinn 1902), som kan uttrykkes ved

$$DCR_{(E+A)}(ENC_{(E+A)}(M))$$

Deretter utvinnes signaturbitene fra pakkens topptekst (trinn 1904). En signatur på hjelpenøkkelen A, integritetsnøkkel for sesjonsdata I og pakkedataene M genereres da ved ghelp av md metoden (trinn 1906), og uttrykkes ved

$$SIG(A + I + M)$$

Deretter sammenlignes de to signaturene med hverandre (trinn 1908). Hvis de tilsvarer hverandre overføres pakken etter å erstatte data i pakken som var overskrevet med signaturdata (1910). Hvis signaturene ikke tilsvarer hverandre droppes pakken (trinn 1912).

Utveksling av basisnøkkel

Som forklart ovenfor, med den hensikt å kryptere kommunikasjoner mellom objekter beskyttet ved hjelp av brannmur, må en brannmur ha kjennskap til sin egen privat basisnøkkel og de offentlige basisnøkler til hver brannmur som den skal kommunisere med. De offentlige basisnøkler som tilhører eksterne brannmurer slik som en brannmur som tilhører en forretningspartner må også være kjent for å kunne foreta krypterte sesjoner. Denne statiske forbindelse mellom basisnøkler og brannmur kan allerede etableres i en database internt i brannmuren eller den kan oppnås på stedet ved hjelp av basisnøkkelutveksling. I tillegg kan basisnøklene oppdateres på en sjelden basis for å øke sikkerhet. Den foreliggende oppfinnelse tilveiebringer mulighet for at offentlige basisnøkler oppnås på stedet hvis de ikke allerede er i en database innenfor brannmuren. Generelt må en basisk offentlig nøkkel oppnås hvis kilden ikke har kunnskap til bestemmelsesstedets offentlige basisnøkkel eller bestemmelsesstedet fastslår at bestemmelsesstedets offentlige basisnøkkel som anvendes av kilden er utdatert.

I begge tilfeller bekreftes utveksling av den offentlige basisnøkkel med den hensikt å sikre ektheten av de Diffie-Hellman nøkkel som overføres. Bekreftelse av meldinger, generelt, har som hensikt å unngå angrep mot systemet.

Prosessen for utveksling av basisnøkler vil nå beskrives i større detalj. Et høynivå blokkdiagram som illustrerer data overført mellom to brannmurer under en utveksling av basisnøkler vises i figur 20. Når en av sidene blir oppmerksom på at den ikke har en gyldig nøkkel for den andre side eller at den har en foreldet nøkkel anmoder den den andre siden om å sende en bekreftet basisnøkkel. Utveksling av basisnøkkelen kan utløses på to måter avhengig av hvilken side som oppdager at den offentlige basisnøkkel må oppdateres eller utveksles. Typisk vil det være siden som oppdager at den ikke har den andre sides basisnøkkel. For eksempel, med henvisning til figur 16, vil en utveksling av basisnøkler utløses hvis den startende side dvs. brannmur1 oppdager at den ikke har den offentlige basisnøkkel for brannmur2. I en annen situasjon ser brannmur2, ved mottakelse av en anmodning fra brannmur1, at den har en foreldet versjon av den offentlige basisnøkkel for brannmur1 (ved å sammenligne innholdet i dens database til den foreslåtte offentlige basisnøkkel sendt i anmodningen). Denne siste situasjonen er den som vises i figur 20.

Elementene i anmodningen om en basisnøkkel vises over den venstre pil i figur 20. Basisanmodningen omfatter kildens offentlige basisnøkkel ID, bestemmelsesstedets offentlige basisnøkkel ID, siffermetoden, nøkkelmetoden og md.metoden. Disse elementer er identiske med de omtalt ovenfor i delen med tittel Utveksling av sesjonsnøkkel - Brannmur/Brannmur. Når en utveksling av basiske nøkler må foretas, vil siden som ønsker at den andre sender en bekreftet nøkkeloppdatering eller nøkkelsynkronisering addere en CA offentlig nøkkel ID felt til anmodningen. Dette nye felt angir hvilken nøkkel som krever oppdatering og er ID til den sertifiseringsmyndighetsnøkkel (for eks. RSA nøkkel) ved hvilken brannmur2 ønsker å motta svaret fra brannmur1. Ved mottakelse av denne melding, vil brannmur1 sende sin offentlige basisnøkkel S_{pub} til brannmur2 etter å bekrefte den med den CA-offentlige nøkkel mot en bekreftelse som var utført ved CA. Bekreftelse er prosessen å generere en digital signatur for den basiske offentlige nøkkel. For brannmur 1 genererer CA1 1602 de CA-offentlige nøkler for å verifisere brannmur1's offentlige nøkler (figur 16). For at brannmur 2 skal verifisere signaturen, må den oppnå den CA-offentlige nøkkel fra CA1, sertifiseringsmyndighet for brannmur1.

Elementene i svaret til brannmur 1 vises over den høyre pil i figur 20. Elementene omfatter CA offentlig nøkkel ID, kildens offentlige basisnøkkel

S_{pub} og IP adressen eller adressen til kilden. I tillegg sendes signaturen til kildens offentlige basisnøkkel, som kan representeres ved

$$SIG(S_{pub})$$

I en foretrukket utførelse, genereres signaturen ved først å generere en mellomliggende signatur fra den basiske offentlige nøkkel som skal sendes ved hjelp av md.metoden for å generere digitale signaturer. Deretter innføres denne mellomliggende signatur til RSA dekrypteringsfunksjonen for å generere signaturen som overføres til slutt. IP adressen til kilden (dvs. brannmur1) er inkludert for å verifisere forbindelse mellom brannmuren, (dvs. brannmur1), og en offentlig basisnøkkel (S_{pub}).

Ved mottagelse av bekreftelsen fra brannmur1, kan brannmur2 verifisere den ved hjelp av den CA offentlige nøkkelen. Hvis den verifiserer riktig, oppdaterer brannmur 2 sin database med den nye basiske offentlige nøkkel til brannmur1. Nå kan sesjonens nøkkelutveksling fullføres og sesjonsdata kan kommuniseres.

Merk at de offentlige basisnøkler kommuniseres mellom hver brannmur og dens CA over sikre kommunikasjonskanaler. Hvis det er mer enn en enkel CA sendes den offentlige nøkkel av en CA innenfor den andre CA. Denne melding er enten signert ved hjelp av en eldre verdi av den CA-offentlige nøkkel eller den nyere oppnådde CA-offentlige nøkkel kan verifiseres ved andre manuelle metoder slik som faksimile eller telefon.

<u>Utveksling av sesjonsnøkkel - klient/brannmur</u>

Som beskrevet ovenfor, er det et voksende behov for ekstern tilgang til selskapsnettverk. Flere og flere ansatte jobber fysisk utenfor selskapets LAN eller WAN miljø men trenger å kobles til det. Den foreliggende oppfinnelse tilveiebringer muligheten for å bekrefte eksterne brukere av et system og å tilveiebringe kryptert kommunikasjon mellom den eksterne bruker eller kunde og vertsystemet.

Et høynivå blokkdiagram som illustrerer et eksempel på en konfigurasjon som anvender en kundes personlige datamaskin og en brannmur bygget ifølge den foreliggende oppfinnelse vises i figur 21. En personlig datamaskin (PC) 2100, kalt kilden for forklaringen, anvendes av kunden eller av den eksterne bruker for å logge inn på verten 2104 som vises koblet til et LAN.

Datamaskinen er koblet til et offentlig nettverk 1606 og kommuniserer med verten via brannmur 2102, kalt bestemmelsesstedet eller server for denne forklaring. All kommunikasjon mellom datamaskinen og verten rutes gjennom brannmuren. Datamaskinen er hensiktsmessig programmert til å utføre funksjonene som kreves for å logge inn på verten og utføre kryptert kommunikasjon mellom datamaskinen og brannmuren. På samme måte som ved konfigurasjonen vist i figur 16, er kommunikasjonen kryptert bare mellom datamaskin og brannmur i den konfigurasjonen vist i figur 21. For verten er brannmuren transparent og den tror at data kommer direkte fra datamaskinen.

Prosessene for utveksling av sesjonsdata for kunden til brannmurkryptering ligner de ved brannmur til brannmur kryptering. Forskjellene ligger imidlertid i utvekslingen av sesjonsnøkkel og utvekslingen av basis nøkkelen. Ved utveksling av sesjonsnøkkel i brannmur-til-brannmur sesjoner mottok hver sesjon en forskjellig sesjonsnøkkel. En sesjon er ikke bare en kobling mellom to bestemte nettverksobjekter men den kan også omfatte forskjellige tjenester mellom et og samme nettverksobjekt. I motsetning til dette, starter kunden en sesjon med verten og all kommunikasjon mellom kunden og verten under sesjonen er kryptert ved hjelp av samme nøkkel, uansett aktivitetene eller tjenestene som kunden anmoder om. I tillegg har ved brannmur til brannmur kommunikasjon begge sider hverandres sertifiserte offentlige nøkkel. I kunde til brannmurkommunikasjon gjelder dette bare kunden, mens serveren identifiserer kunden ved hjelp av et navn/passordpar sendt den av kunden.

Et høynivå blokkdiagram som illustrerer data overført mellom en kundes personlige datamaskin og en brannmur under en utveksling av sesjonsnøkler vises i figur 22. Elementene sendt i anmodningen fra kunden vises over høyre pil. Elementene omfatter navn, siffermetode, nøkkelmetode, md.metode, passord-metode, kildens offentlige basisnøkkel S_{pub}, foreslått bestemmelsesstedets offentlige basisnøkkel ID, oppfordringsnøkkel C, kryptert passord og en signatur. Navnet anvendes for å identifisere brukeren som faktisk anvender kunden. Siffermetoden, nøkkelmetoden og md-metoden er som beskrevet ovenfor. Passord-metoden angir hvilken krypterings-metode som skal anvendes for å kryptere passordet. Det krypterte passord kan yttrykkes som

$$ENC_{(TB + C)}(P)$$

Kildens offentlige basisnøkkel S_{pub} sendes alltid siden brannmuren ikke har en liste av brukere og deres tilknyttede offentlige basisnøkler. Dataene som sendes er tilsvarende data sendt ved brannmur1 til brannmur2 (figur 20) som beskrevet i delen med tittel Basisnøkkelutveksling - Brannmur/Brannmur. Bestemmelsesstedets offentlige basisnøkkel ID er som beskrevet ovenfor i delen med tittel Utveksling av Sesjonsnøkkel - Brannmur/Brannmur.

Signaturen virker for å sikre bestemmelsesstedet, det mottagende stedet, at meldingen ikke er modifisert. Signaturen genereres ved å ta hele innholdet i anmodningen eller meldingen, representert ved T i figur 22, bortsett fra signaturfeltet, og ved å kombinere T med det ikke-krypterte passordet og den avkuttede offentlige basisnøkkel TB, uttrykket som følger

$$SIG(T + P + TB)$$

Signaturen adderes til anmodningen og anmodningen sendes til brannmuren.

Etter mottagelse av anmodningen, kjenner brannmuren til kundens offentlige basisnøkkel S_{pub}. Den kan nå generere basisnøkkelen B og den avkuttende basisnøkkelen TB. Den kan da dekryptere passordet P. Når P er kjent, kan brannmuren bekrefte signaturen i anmodningen. Brannmuren genererer deretter en vilkårlig sesjonsnøkkel R og krypterer R og R's signatur ved hjelp av den avkuttede basisnøkkel TB og oppfordringen C sendes i anmodningen fra kunden, og gitt ved

$$ENC_{(TB + C)}(R + SIG(R))$$

En signatur genereres da fra innholdet av anmodningen angitt ved U i figur 22 i kombinasjon med den avkuttede offentlige basisnøkkel TB, som gitt ved

$$SIG(U + TB)$$

Brannmuren genererer da et svar med elementene som vises over den venstre pil i figur 22. Svaret omfatter bestemmelsesstedets offentlige basisnøkkel ID, siffermetoden, nøkkelmetoden og md-metoden, nøkkelforkryptetsesjon og signaturen.

Når sesjonsnøkkelen er kjent av både kunden og brannmuren, kan kommunikasjonssesjonen utføres mellom datamaskinen og verten og via brannmuren, og den krypterte kommunikasjonen mellom datamaskinen og verten er transparent for verten. For å redusere antall nøkkelutvekslinger, anvendes sesjonsnøkkel R for alle krypterte koblinger som passerer gjennom den samme brannmur. Etter en bestemt tidsvarighet, dvs. for eksempel flere minutter, droppes sesjonsnøkkel R.

Utveksling av basisnøkler - kunde/brannmur

I motsetning til kommunikasjoner fra brannmur til brannmur, er det bare nødvendig med utveksling av sertifisert nøkkel for oppdatering av kunden med brannmurens offentlige basisnøkkel. Utveksling av en basisnøkkel kan utløses på en av to måter. Den første, hvis kunden ikke har brannmurens offentlige basisnøkkel eller den andre hvis brannmuren fastslår at den offentlige basisnøkkel anvendt av kunden i anmodningen er foreldet.

Prosessen er tilsvarende utvekslingen av basisnøkler som forklart overfor i delen med tittel Utveksling av basisnøkler - Brannmur/Brannmur. Imidlertid finnes det forskjeller som forklares nedenfor. Hvis kunden blir oppmerksom på at den ikke har brannmurens offentlige basisnøkkel, erstatter den et CA offentlig nøkkel ID felt for bestemmelsesstedets offentlige basisfelt ID felt i anmodningen. Dette vises over øverste høyre pil i figur 23 som er et høynivå blokk diagram som illustrerer dataene overført mellom en kundes personlige datamaskin og en brannmur under utveksling av basisnøkler. Denne nøkkel ID er ID'en til sertifiseringsmyndighetens nøkkel (for eks., RSA nøkkel) ved hvilken kunden vil motta svaret fra brannmuren.

Når brannmuren mottar anmodningen fra kunden, fastslår den fra anmodningen om kunden ber om brannmurens offentlige basisnøkkel eller nøkkel ID i anmodningen ikke tilsvarer brannmurens offentlige nøkkel. Elementene i brannmurens svar vises over den venstre pil. Svaret omfatter den opprinnelige foreslåtte offentlige nøkkel for bestemmelsesstedets ID, CA offentlig nøkkel ID, bestemmelsesstedets offentlige basisnøkkel D_{pub}, IP

adresse til bestemmelsesstedet og en signatur. Den opprinnelige offentlige basisnøkkel til bestemmelsesstedet tas fra anmodningen. Signaturen til den basiske offentlige nøkkel til bestemmelsesstedet sendes, representert ved

$$SIG(D_{pub})$$

I en foretrukket utførelse, genereres signaturen ved første å generere en mellomliggende signatur fra den offentlige basisnøkkel som skal sendes ved hjelp av md metoden for å generere digitale signaturer. Deretter innføres den mellomliggende signatur til RSA dekrypteringsfunksjon for å generere signaturen som overføres til slutt. IP adressen til bestemmelsesstedet (dvs. brannmuren) inkluderes med den hensikt å bekrefte forbindelse mellom brannmur, og en offentlig basisnøkkel (D_{pub}).

Ved mottagelse av bekreftelsen fra brannmuren, kan kunden verifisere den ved å anvende CA's offentlige nøkkel. Hvis bekreftelsen er korrekt, oppdaterer kunden sin database med en ny offentlig basisnøkkel til brannmuren.

Etter mottagelse av brannmurens svar, sender kunden tilbake en melding for å fullføre bekreftelsen. Elementene i meldingen vises over den nedre høyre pil i figur 23. Meldingen omfatter det krypterte passord og en signatur. Når svaret er mottatt kan kunden generere den basisnøkkel B og den avkuttede basisnøkkelen TB. Kunden krypterer da passordet P, utrykket som

$$ENC_{(TB + C)}(P)$$

Signaturen genereres ved hjelp av md metoden på kombinasjonen av innholdet i den opprinnelige anmodning sendt til brannmuren som vist over høyre pilen i figur 22, representert ved T, klartekst passordet P og den avkuttede offentlige basisnøkkel TB, uttrykt ved

$$SIG(T + P + TB)$$

Det kryperte passord og signaturen sendes da til brannmuren. Utvekslingen av sesjonsnøkkel er fullført og kommunikasjon av sesjonsdata kan begynne. Mens oppfinnelsen er beskrevet i forbindelse med et begrenset antall utførelser, skal det bemerkes at flere varianter, forandringer og andre anvendelser av oppfinnelsen kan utføres.

PATENTKRAV

1. Fremgangsmåte for å inspisere og selektivt å modifisere inngående og utgående datapakker i et datamaskinnettverk (100,120), idet inspeksjonen av nevnte datapakker foregår ifølge en sikkerhetsregel (302-308), hvor fremgangsmåten omfatter trinnene:
å generere en definisjon av hvert aspekt av datamaskinnettverket (100,120) inspisert ved sikkerhetsregelen (302-308);
å generere nevnte sikkerhetsregel (302-308) ved hjelp av nevnte aspektdefinisjoner, idet sikkerhetsregelen (302-308) kontrollerer i det minste et av aspektene;
å konvertere sikkerhetsregelen (302-308) til et sett (400) instruksjoner i pakkefilterspråk for å kontrollere operasjonen av en pakkefiltreringsmodul (204,520) som inspiserer datapakkene ifølge sikkerhetsregelen (302-308);
å koble pakkefiltermodulen (204-520) til datamaskinnettverket (100,120) for å inspisere nevnte datapakker ifølge nevnte sikkerhetsregel (302-308), idet pakkefiltermodulen (204,520) implementerer en virtuell pakkefiltreringsmaskin (600); og
hvor nevnte pakkefiltermodul (204,520) utfører instruksjonene i pakkefilterspråket for å operere den virtuelle pakkefiltreringsmaskinen (600) for enten å godta eller å avvise passering av datapakkene inn og ut av datamaskinnettverket (100,120),
k a r a k t e r i s e r t v e d at en selektiv modifisering av nevnte datapakker foregår ifølge nevnte sikkerhetsregel (302-308) , idet nevnte pakkefiltermodul (204,520) som kontrolleres av nevnte filterspråkinstruksjoner (400) selektivt modifiserer nevnte datapakker ifølge nevnte sikkerhetsregel (302-308), og
ved at den virtuelle pakkefiltreringsmaskinen (600) som enten godtar eller avviser passeringen av datapakkene er operert for selektivt å modifisere de godtatte datapakkene.

2. Fremgangsmåte ifølge krav 1,
k a r a k t e r i s e r t v e d at nevnte aspekter omfatter nettverksobjekter.

3. Fremgangsmåte ifølge krav 1,
k a r a k t e r i s e r t v e d at nevnte aspekter omfatter nettverktjenester.

4. Fremgangsmåte ifølge krav 2,
k a r a k t e r i s e r t v e d at nevnte aspekter omfatter nettverkstjenester.

5. Fremgangsmåte ifølge krav 4,
k a r a k t e r i s e r t v e d at objektdefinisjonene omfatter objektets adresse.

6. Fremgangsmåte ifølge krav 1,
k a r a k t e r i s e r t v e d at instruksjonene i filterspråk (400) i konverteringstrinnet er i form av skript og omfatter videre en kompilator for å kompilere nevnte skript til instruksjonene utført i utføringstrinnet.

7. Fremgangsmåte ifølge krav 1, nevnte
k a r a k t e r i s e r t v e d at begge trinnene generering av aspektene av nettverket og av sikkerhetsregelen (302-308) defineres grafisk.

8. Fremgangsmåte ifølge krav 1,
k a r a k t e r i s e r t v e d at nevnte selektive modifisering velges fra gruppen omfattende kryptering, dekryptering, signaturgenerering og signaturverifisering.

9. Fremgangsmåte ifølge krav 1,
k a r a k t e r i s e r t v e d at nevnte virtuelle maskinen (600) utfører en datautvinningsoperasjon (702).

10. Fremgangsmåte ifølge krav 9,
k a r a k t e r i s e r t v e d at nevnte virtuelle maskinen (600) utfører en logisk operasjon (704).

11. Fremgangsmåte ifølge krav 10,
k a r a k t e r i s e r t v e d at nevnte virtuelle maskinen (600) utfører en sammenlikningsoperasjon (706).

12. Fremgangsmåte ifølge krav 11,
k a r a k t e r i s e r t v e d at nevnte selektive modifisering er valgt blant gruppen omfattende kryptering, dekryptering, signaturgenerering, og signaturverifisering.

13. Fremgangsmåte ifølge krav 1,
k a r a k t e r i s e r t v e d lagring av resultatene fremskaffet i trinnet å lese og å utføre instruksjonene i pakkefilterspråk i en lagringsanordning, idet pakkefiltermodulen (204,520) anvender nevnte lagrede resultater, fra

tidligere inspeksjoner, for å operere nevnte pakkefiltermodul (204,520) for å godta eller avvise passering av nevnte datapakker inn og ut av nevnte datamaskinnettverket (100, 120) og for selektivt å modifisere de således godkjente datapakkene, og
idet trinnet å lese instruksjonene i pakkefilterspråket (400) er innledet av lesing av nevnte pakkefilterspråk.

14. Fremgangsmåte ifølge krav 13,
k a r a k t e r i s e r t v e d at nevnte aspekter omfatter nettverksobjekter.

15. Fremgangsmåte ifølge krav 13,
k a r a k t e r i s e r t v e d at nevnte aspekter omfatter nettverktjenester.

16. Fremgangsmåte ifølge krav 14,
k a r a k t e r i s e r t v e d at nevnte aspekter omfatter nettverkstjenester.

17. Fremgangsmåte ifølge krav 16,
k a r a k t e r i s e r t v e d at objektdefinisjonene omfatter objektets adresse.

18. Fremgangsmåte ifølge krav 13,
k a r a k t e r i s e r t v e d at nevnte selektive modifiseringen omfatter kryptering, dekriptering, signaturgenerering og signaturverifisering.

19. Sikkerhetssystem for å inspisere og selektivt å modifisere inngående og utgående datapakker i et datamaskinnettverk (100, 120), hvor nevnte sikkerhetssystem inspiserer nevnte datapakker som passerer gjennom datamaskinnettverket (100,120) ifølge en sikkerhetsregel (302-308), hvor hvert aspekt av datamaskinnettverket (100,120) kontrollert av sikkerhetsregelen (302-308) har vært definert på forhånd ved hjelp av nevnte aspekter og konvertert til instruksjoner i pakkefilterspråk (400),hvor sikkerhetssystemet omfatter:
en pakkefiltermodul (204,520) koblet til datamaskinnettverket (100,120) hvor pakkefiltermodulen opererer ifølge sikkerhetsregelen (302-308), hvor pakkefiltermodulen implementerer en virtuell pakkefiltreringsmaskin (600) som inspiserer nevnte datapakker som passerer inn og ut av datamaskinnettverket (100,120); og
prosesseringsanordninger for å lese og å utføre nevnte instruksjoner i pakkefilterspråk (400) integrert med pakkefiltermodulen (204,520) hvor prosesseringsanordningen opererer pakkefiltreringsmodulen (204,520) for enten å godta eller å avvise passering av datapakkene inn og ut av

datamaskinnettverket (100, 120),
k a r a k t e r i s e r t v e d at nevnte sikkerhetssystem selektivt modifiserer datapakkene som passerer gjennom datamaskinnettverket (100,120) ifølge nevnte sikkerhetsregel (302-308) med nevnte virtuell pakkefiltreringsmaskin (600) som inspiserer og selektivt modifiserer pakkene som passerer inn og ut av datamaskinnettverket (100,120) og nevnte pakkefiltreringsmodul (204,520) enten aksepterer eller avviser passeringen av nevnte pakker som opereres for selektivt å modifisere datapakkene som er akseptert på denne måten.

20. System ifølge krav 19,
k a r a k t e r i s e r t v e d at nevnte selektive modifiseringen er valgt fra gruppen kryptering, dekryptering, signaturgenerering, og signaturverifisering.

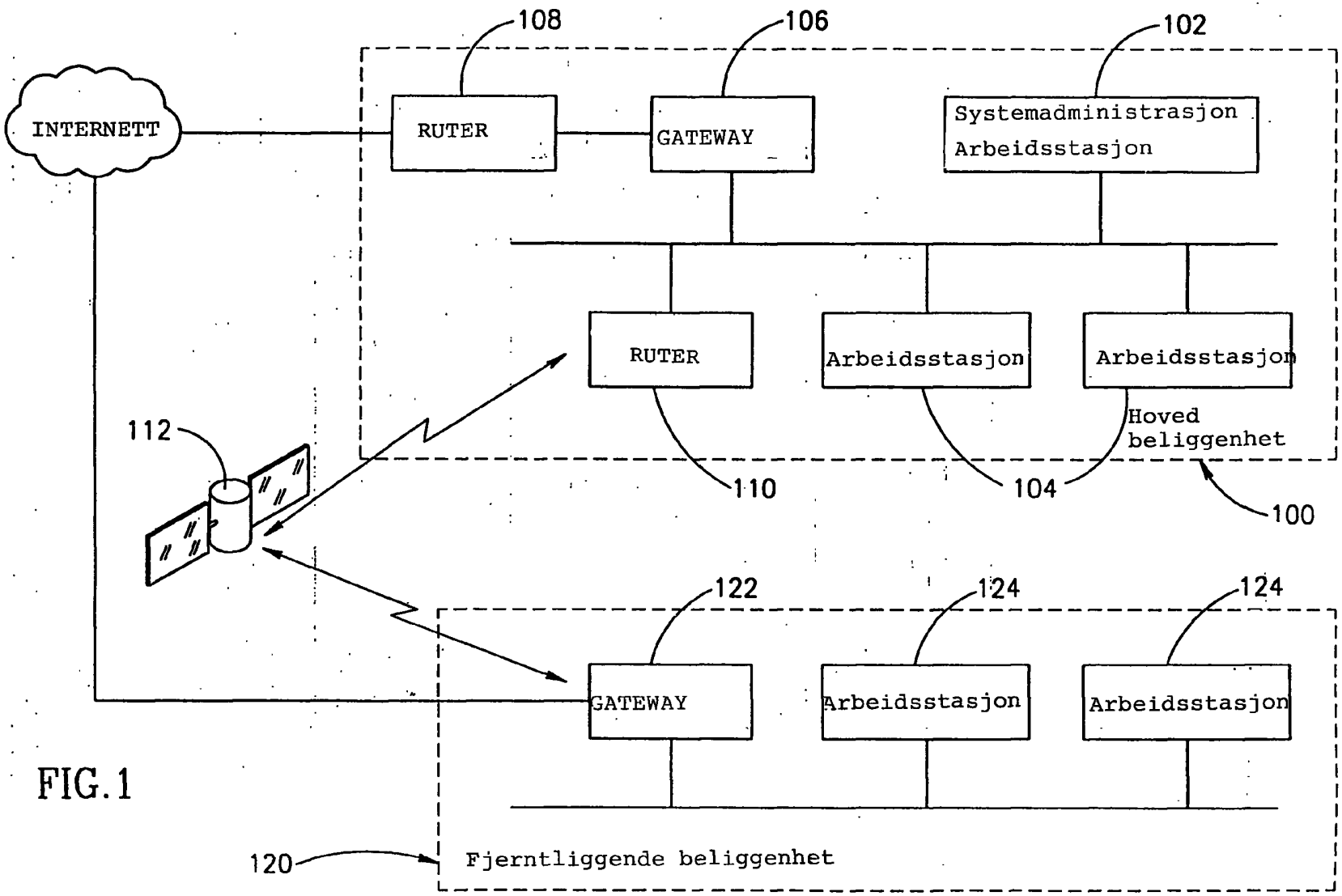

108
106
102
INTERNETT
RUTER
GATEWAY
Systemadministrasjon
Arbeidsstasjon
RUTER
Arbeidsstasjon
Arbeidsstasjon
Hoved
beliggenhet
112
110
104
100
122
124
124
GATEWAY
Arbeidsstasjon
Arbeidsstasjon
120
Fjerntliggende beliggenhet

FIG. 1

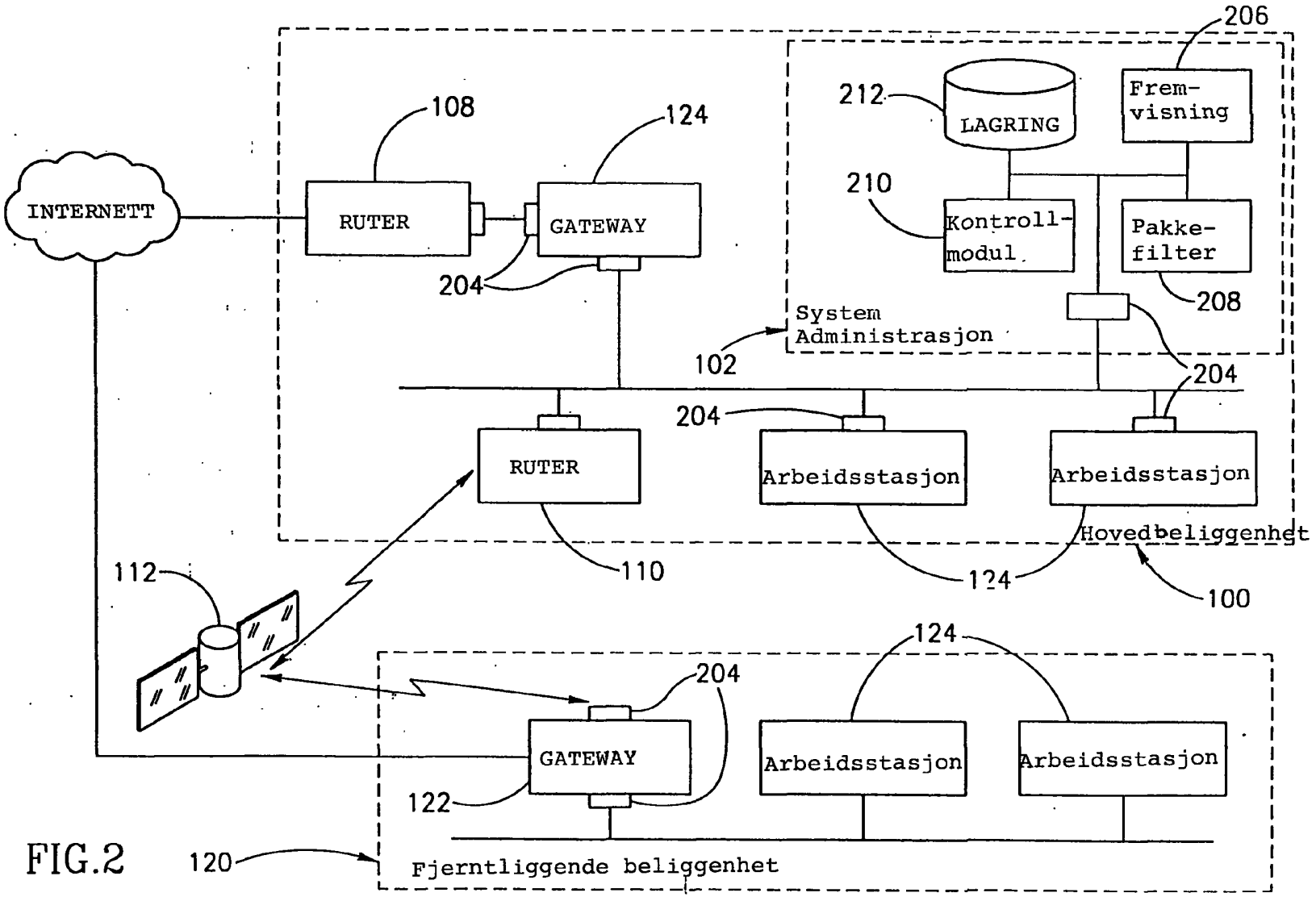

INTERNETT
RUTER
108
GATEWAY
124
204
206
212
LAGRING
Frem-
visning
210
Kontroll-
modul
Pakke-
filter
208
System
Administrasjon
102
204
204
RUTER
Arbeidsstasjon
Arbeidsstasjon
Hovedbeliggenhet
110
124
100
112
124
204
GATEWAY
Arbeidsstasjon
Arbeidsstasjon
122
120
Fjerntliggende beliggenhet

FIG.2

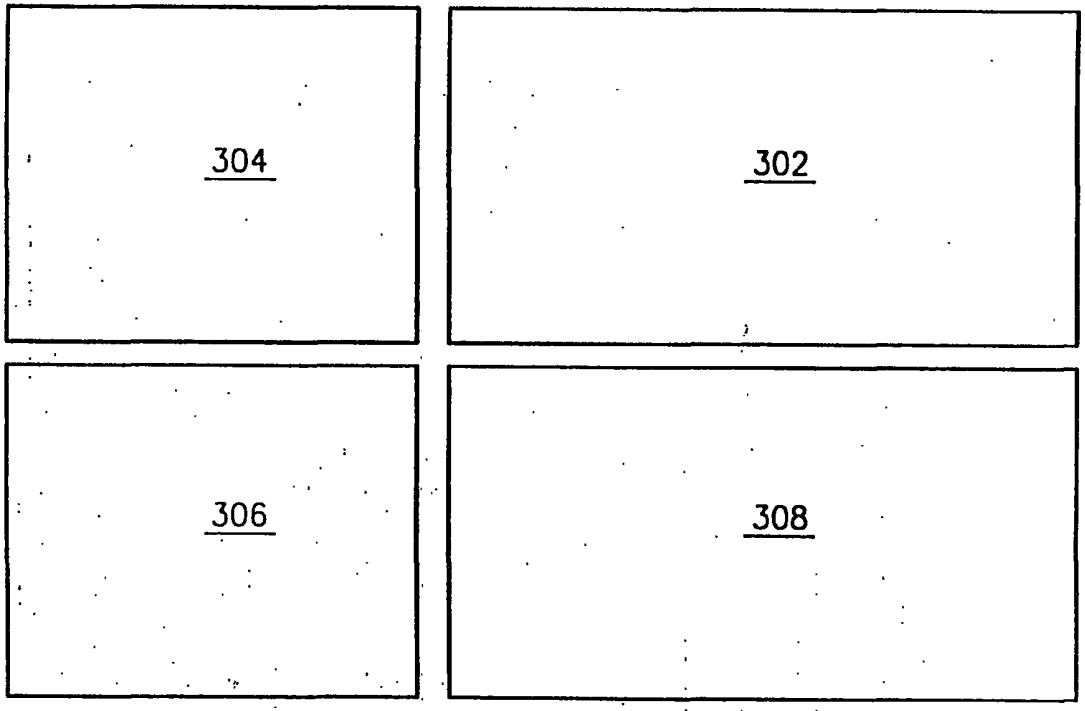


FIG.3/1

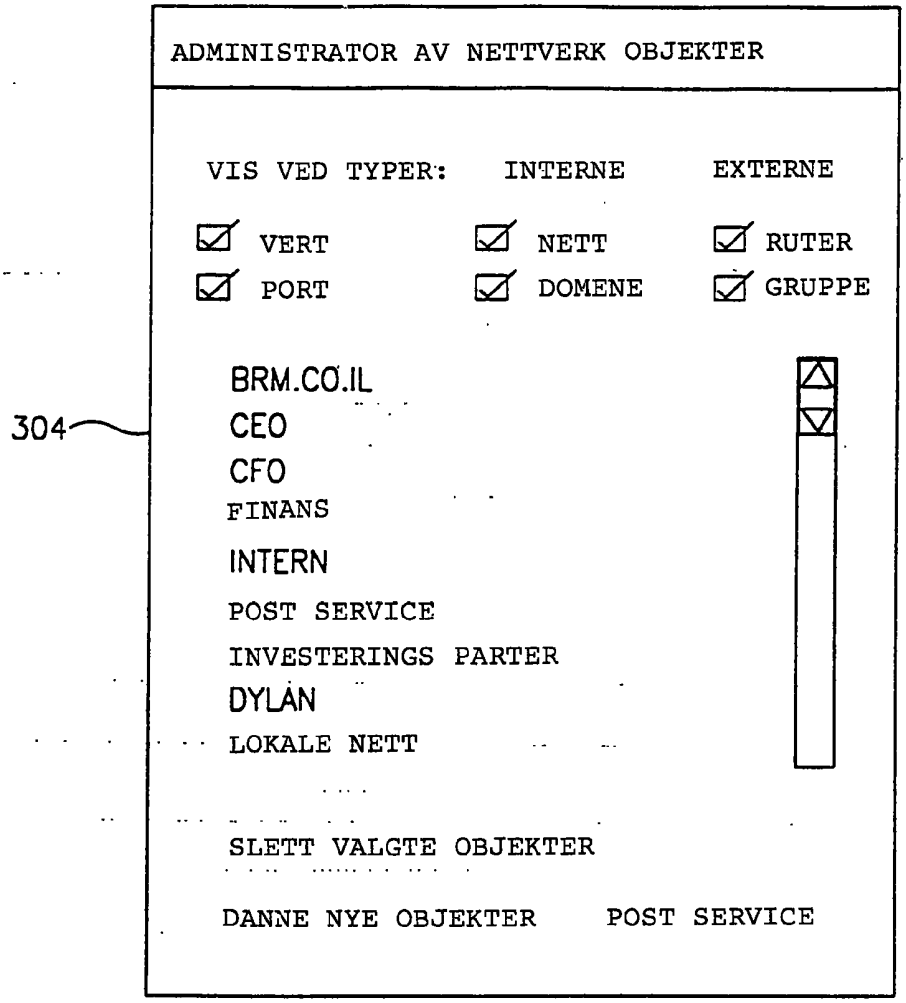

ADMINISTRATOR AV NETTVERK OBJEKTER
VIS VED TYPER:
INTERNE
EXTERNE
VERT
NETT
RUTER
PORT
DOMENE
GRUPPE
304
BRM.CO.IL
CEO
CFO
FINANS
INTERN
POST SERVICE
INVESTERINGS PARTER
DYLAN
LOKALE NETT
SLETT VALGTE OBJEKTER
DANNE NYE OBJEKTER
POST SERVICE

FIG.3/2

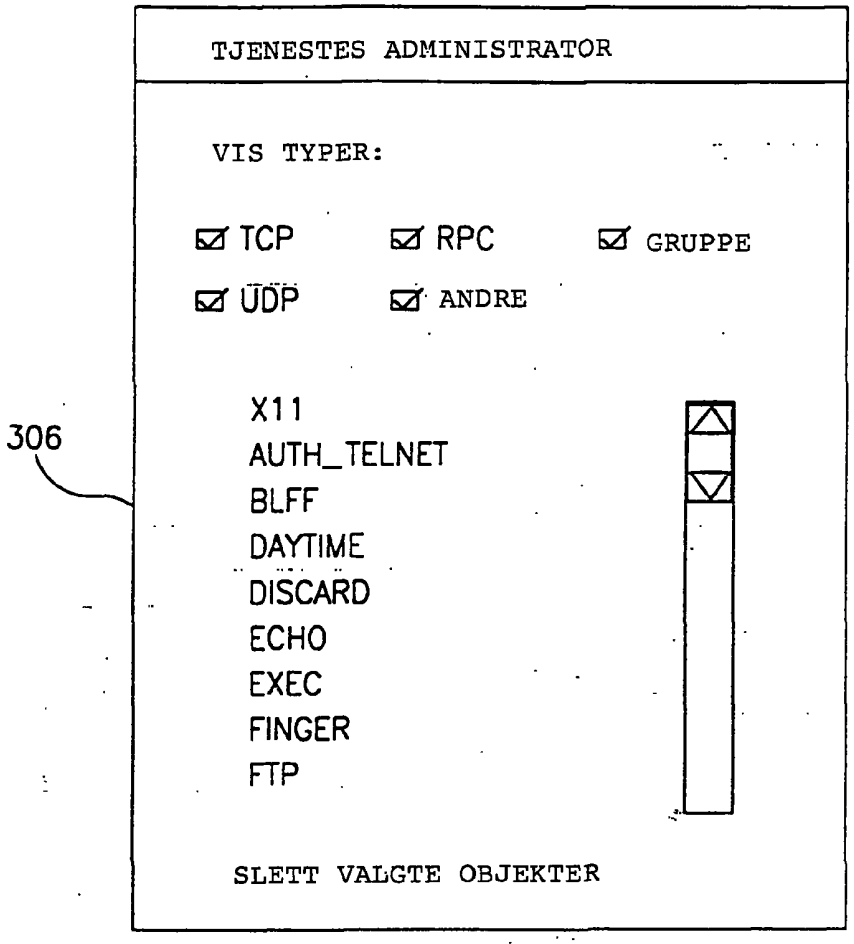

TJENESTES ADMINISTRATOR
VIS TYPER:
TCP
RPC
GRUPPE
UDP
ANDRE
306
X11
AUTH_TELNET
BLFF
DAYTIME
DISCARD
ECHO
EXEC
FINGER
FTP
SLETT VALGTE OBJEKTER

FIG.3/3

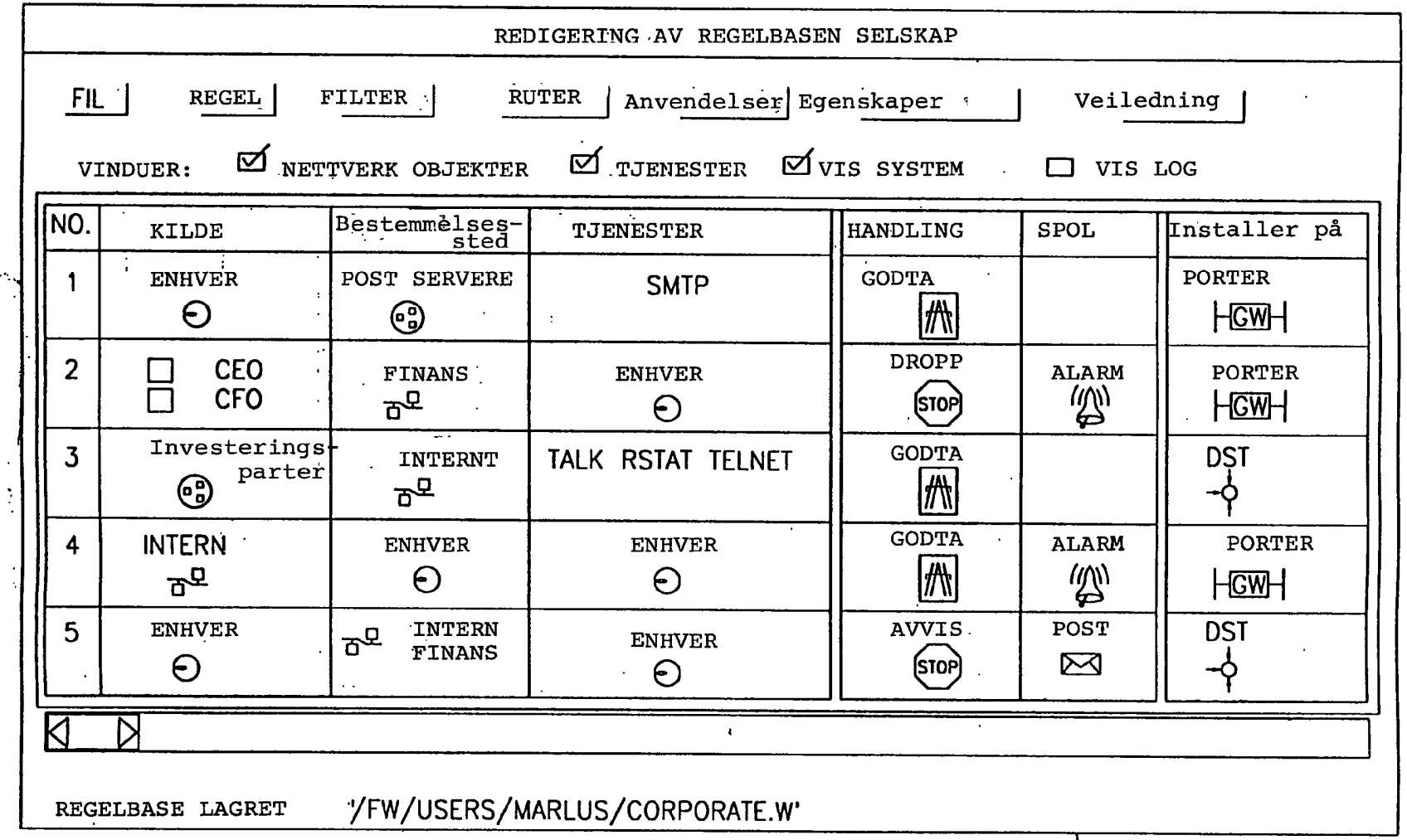

REDIGERING AV REGELBASEN SELSKAP
FIL
REGEL
FILTER
RUTER
Anvendelser
Egenskaper
Veiledning
VINDUER:
NETTVERK OBJEKTER
TJENESTER
VIS SYSTEM
VIS LOG
NO.
KILDE
Bestemmelses-sted
TJENESTER
HANDLING
SPOL
Installer på
1
ENHVER
POST SERVERE
SMTP
GODTA
PORTER
GW
2
CEO
CFO
FINANS
ENHVER
DROPP
STOP
ALARM
PORTER
GW
3
Investerings-parter
INTERNT
TALK RSTAT TELNET
GODTA
DST
4
INTERN
ENHVER
ENHVER
GODTA
ALARM
PORTER
GW
5
ENHVER
INTERN FINANS
ENHVER
AVVIS
STOP
POST
DST
REGELBASE LAGRET
'/FW/USERS/MARLUS/CORPORATE.W'

FIG.3/4

302

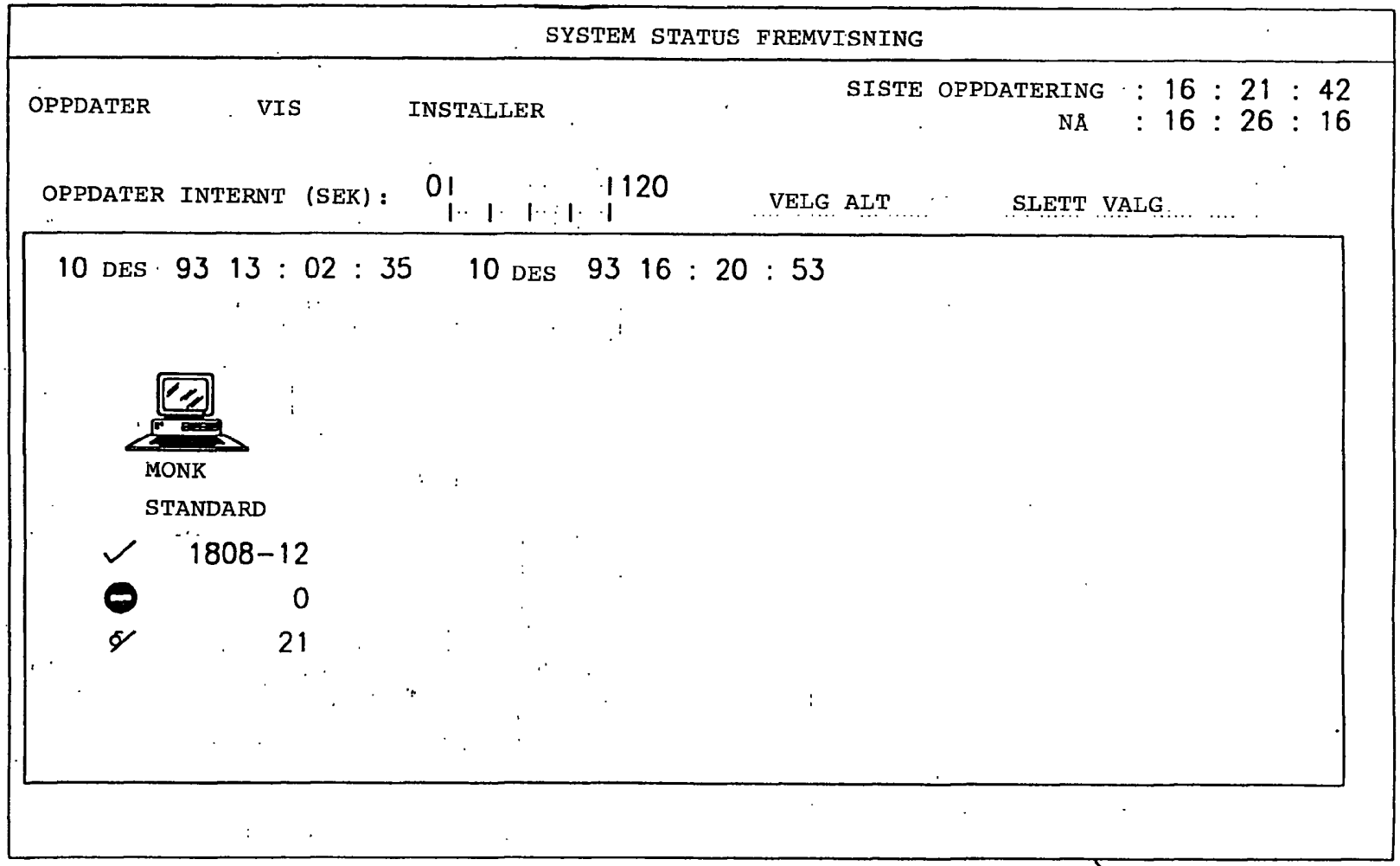

SYSTEM STATUS FREMVISNING
OPPDATER
VIS
INSTALLER
SISTE OPPDATERING : 16 : 21 : 42
NÅ : 16 : 26 : 16
OPPDATER INTERNT (SEK): 01 120
VELG ALT
SLETT VALG
10 DES 93 13 : 02 : 35
10 DES 93 16 : 20 : 53
MONK
STANDARD
1808-12
0
21
308

FIG.3/5

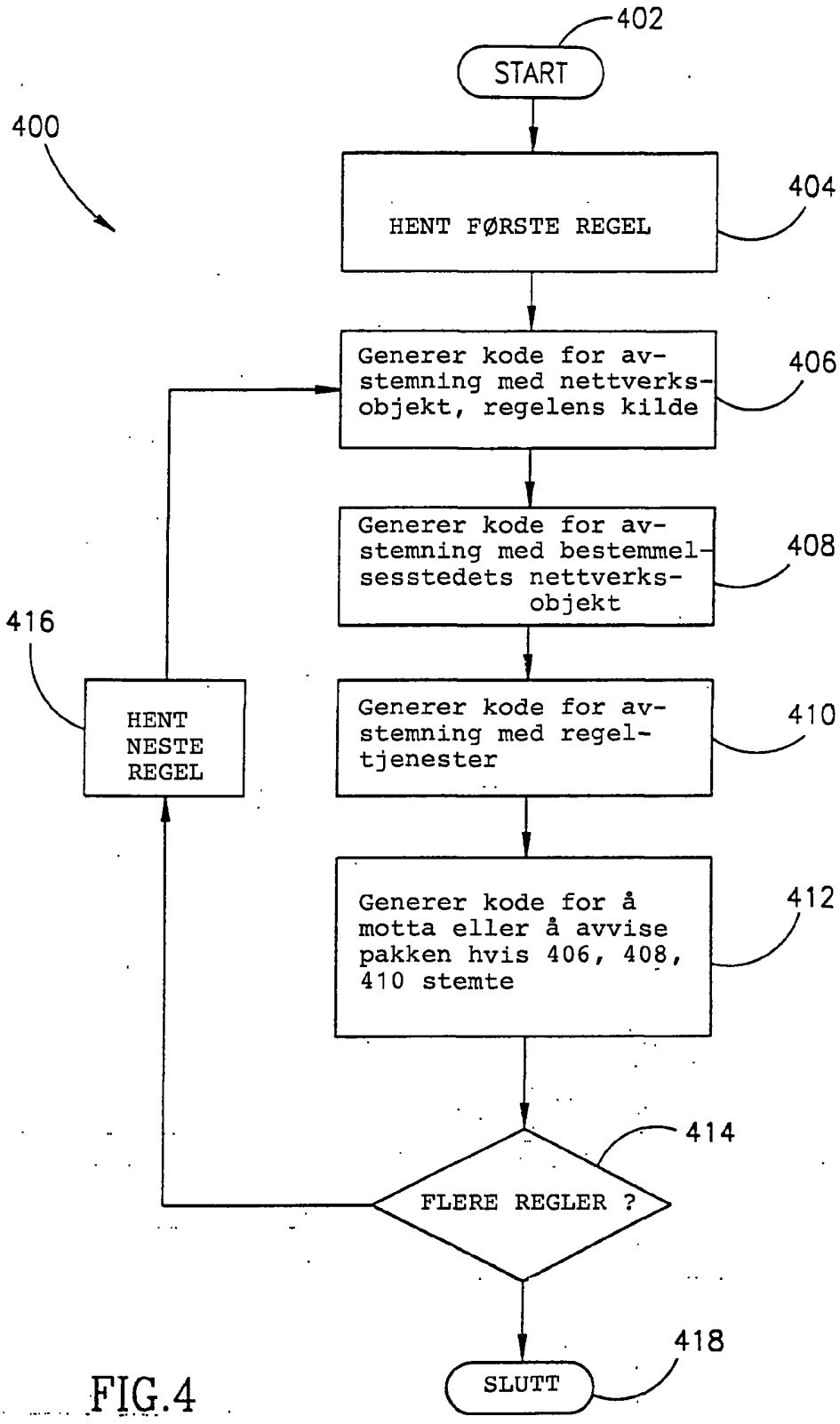

400
402
START
HENT FØRSTE REGEL
404
Generer kode for av-stemning med nettverks-objekt, regelens kilde
406
Generer kode for av-stemning med bestemmel-sesstedets nettverks-objekt
408
Generer kode for av-stemning med regel-tjenester
410
Generer kode for å motta eller å avvise pakken hvis 406, 408, 410 stemte
412
FLERE REGLER ?
414
HENT NESTE REGEL
416
SLUTT
418

FIG.4

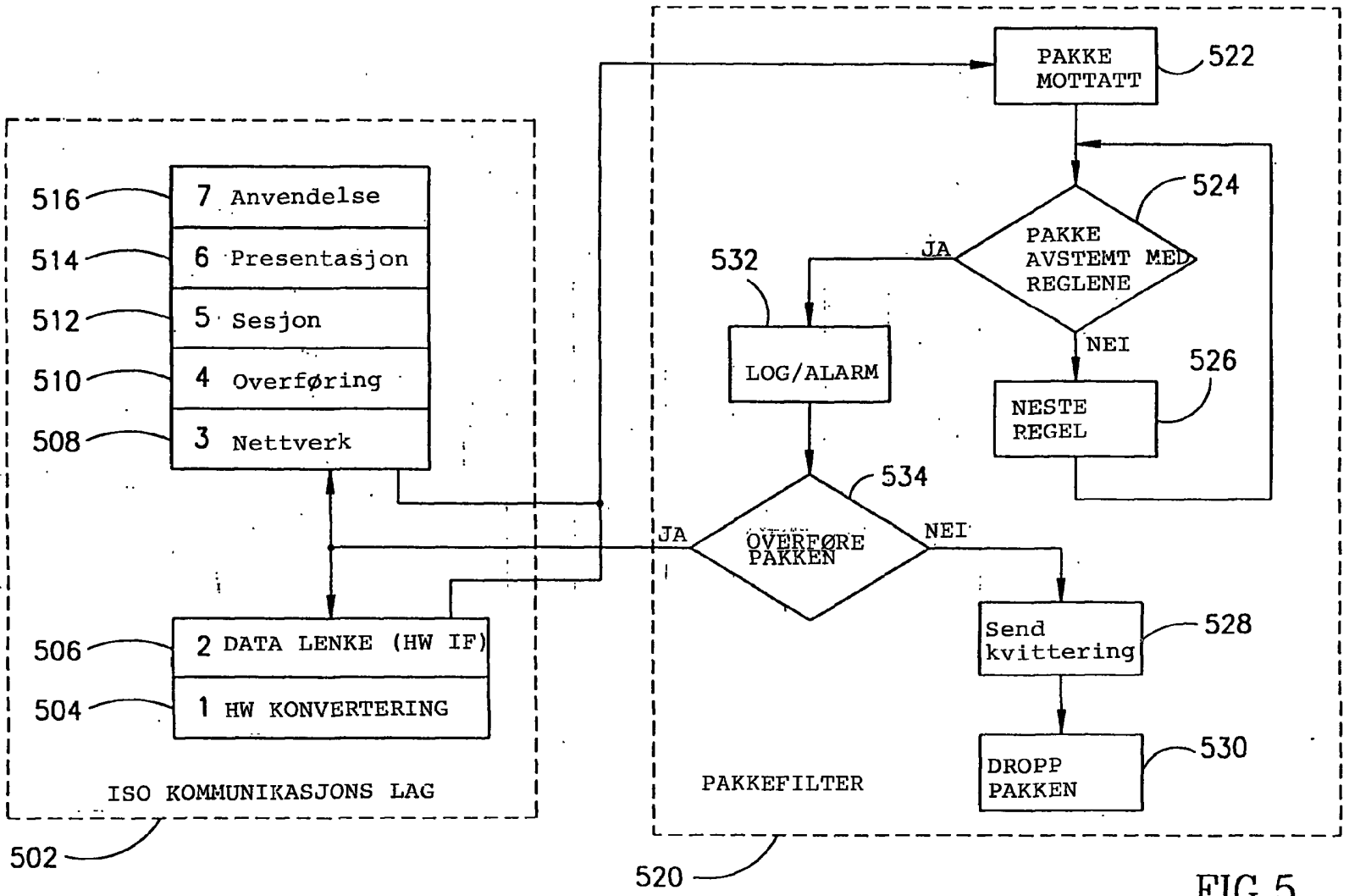

PAKKE MOTTATT
522
524
PAKKE AVSTEMT MED REGLENE
JA
NEI
532
LOG/ALARM
526
NESTE REGEL
534
JA
OVERFØRE PAKKEN
NEI
Send kvittering
528
DROPP PAKKEN
530
PAKKEFILTER
520
516
7 Anvendelse
514
6 Presentasjon
512
5 Sesjon
510
4 Overføring
508
3 Nettverk
506
2 DATA LENKE (HW IF)
504
1 HW KONVERTERING
ISO KOMMUNIKASJONS LAG
502

FIG.5

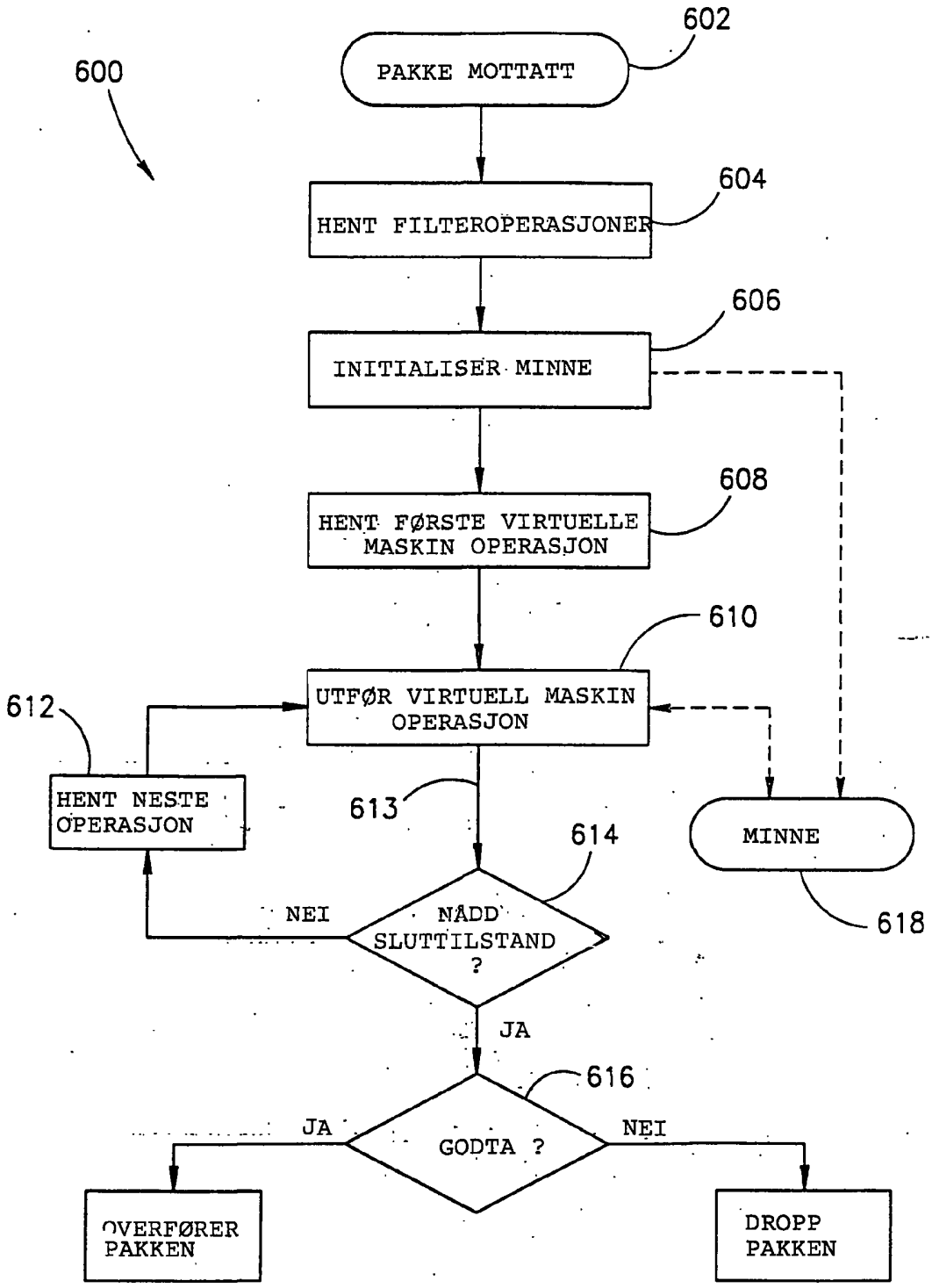

600
602
PAKKE MOTTATT
604
HENT FILTEROPERASJONER
606
INITIALISER MINNE
608
HENT FØRSTE VIRTUELLE MASKIN OPERASJON
610
UTFØR VIRTUELL MASKIN OPERASJON
612
HENT NESTE OPERASJON
613
614
NEI
NÅDD SLUTTILSTAND ?
JA
MINNE
618
616
JA
GODTA ?
NEI
OVERFØRER PAKKEN
DROPP PAKKEN

FIG.6

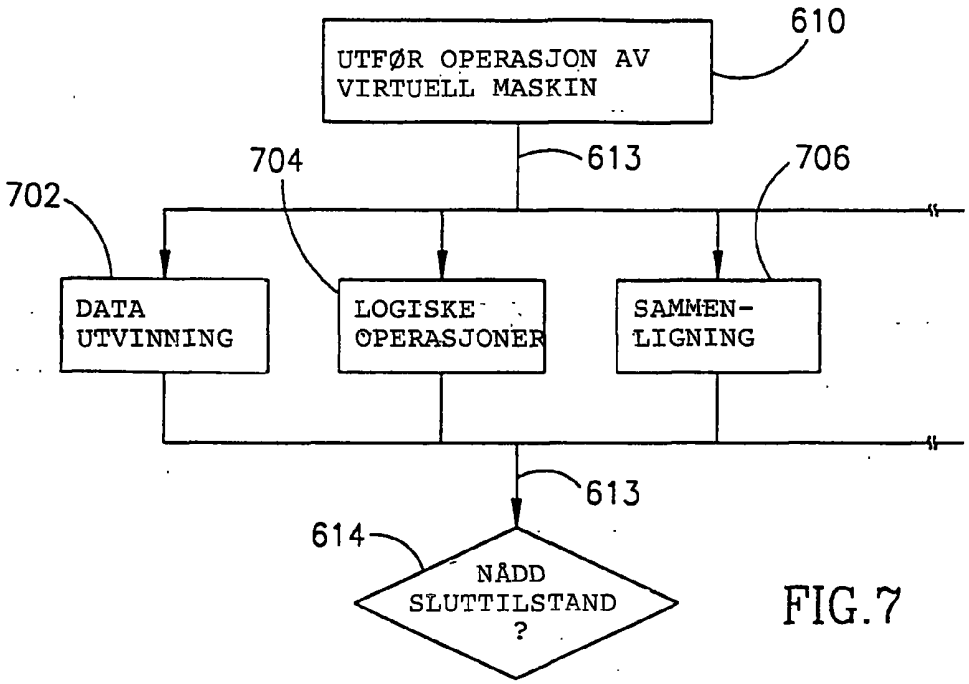

610
UTFØR OPERASJON AV VIRTUELL MASKIN
613
704
706
702
DATA UTVINNING
LOGISKE OPERASJONER
SAMMEN-LIGNING
613
614
NÅDD SLUTTILSTAND ?

FIG.7

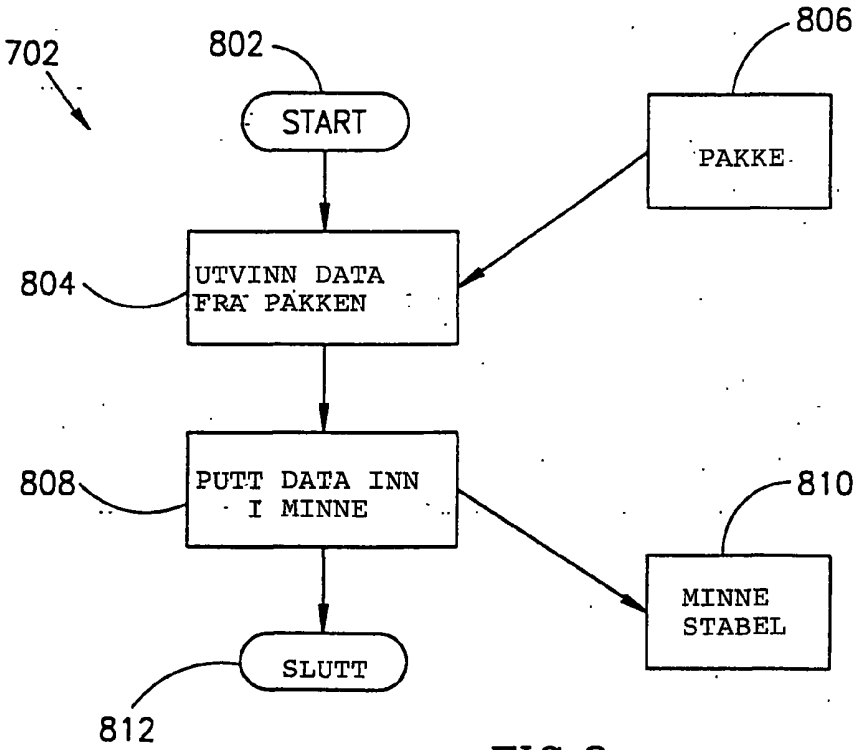

702
802
START
806
PAKKE
804
UTVINN DATA FRA PAKKEN
808
PUTT DATA INN I MINNE
810
MINNE STABEL
SLUTT
812

FIG.8

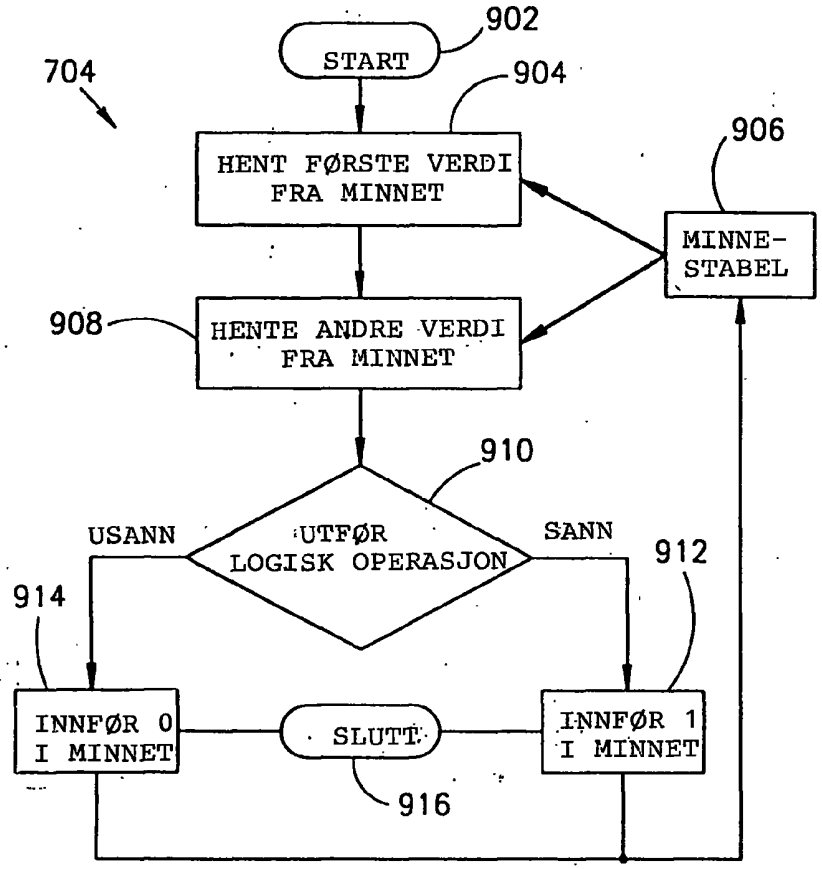

704
START
902
HENT FØRSTE VERDI FRA MINNET
904
MINNE-STABEL
906
HENTE ANDRE VERDI FRA MINNET
908
UTFØR LOGISK OPERASJON
910
USANN
SANN
912
914
INNFØR 0 I MINNET
SLUTT
INNFØR 1 I MINNET
916

FIG.9

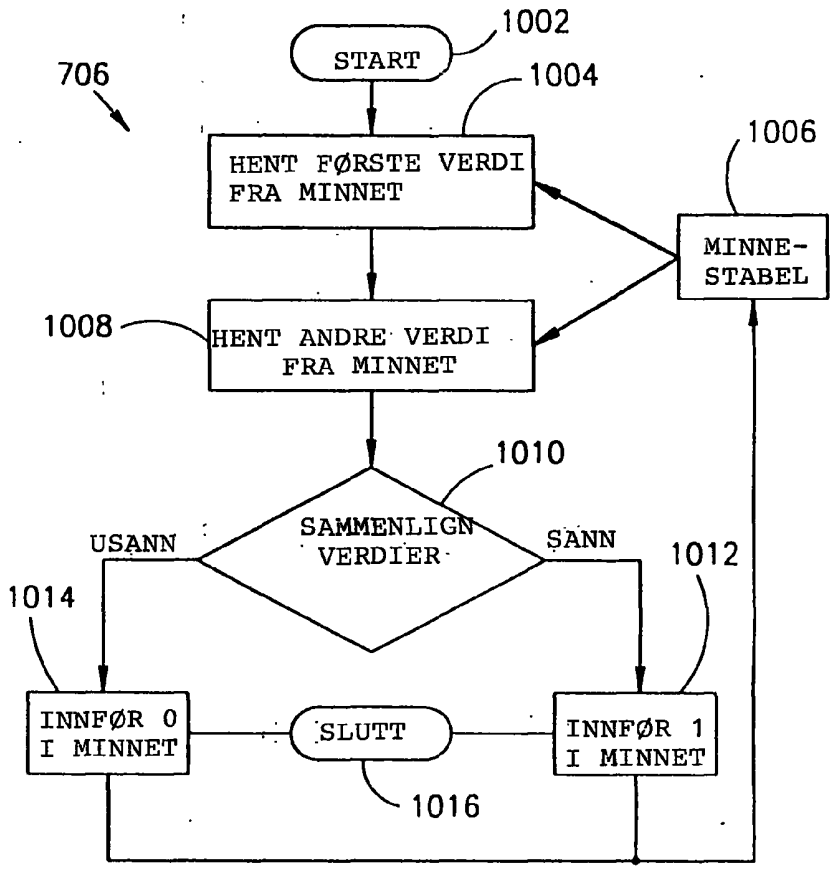

706
START
1002
HENT FØRSTE VERDI FRA MINNET
1004
MINNE-STABEL
1006
HENT ANDRE VERDI FRA MINNET
1008
SAMMENLIGN VERDIER
1010
USANN
SANN
1012
1014
INNFØR 0 I MINNET
SLUTT
INNFØR 1 I MINNET
1016

FIG.10

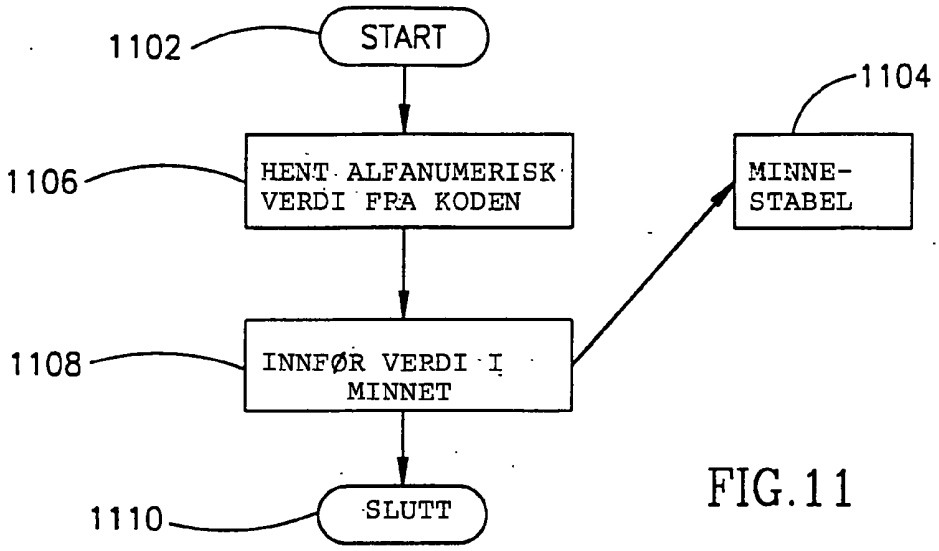

1102
START
1104
1106
HENT ALFANUMERISK VERDI FRA KODEN
MINNE-STABEL
1108
INNFØR VERDI I MINNET
1110
SLUTT

FIG.11

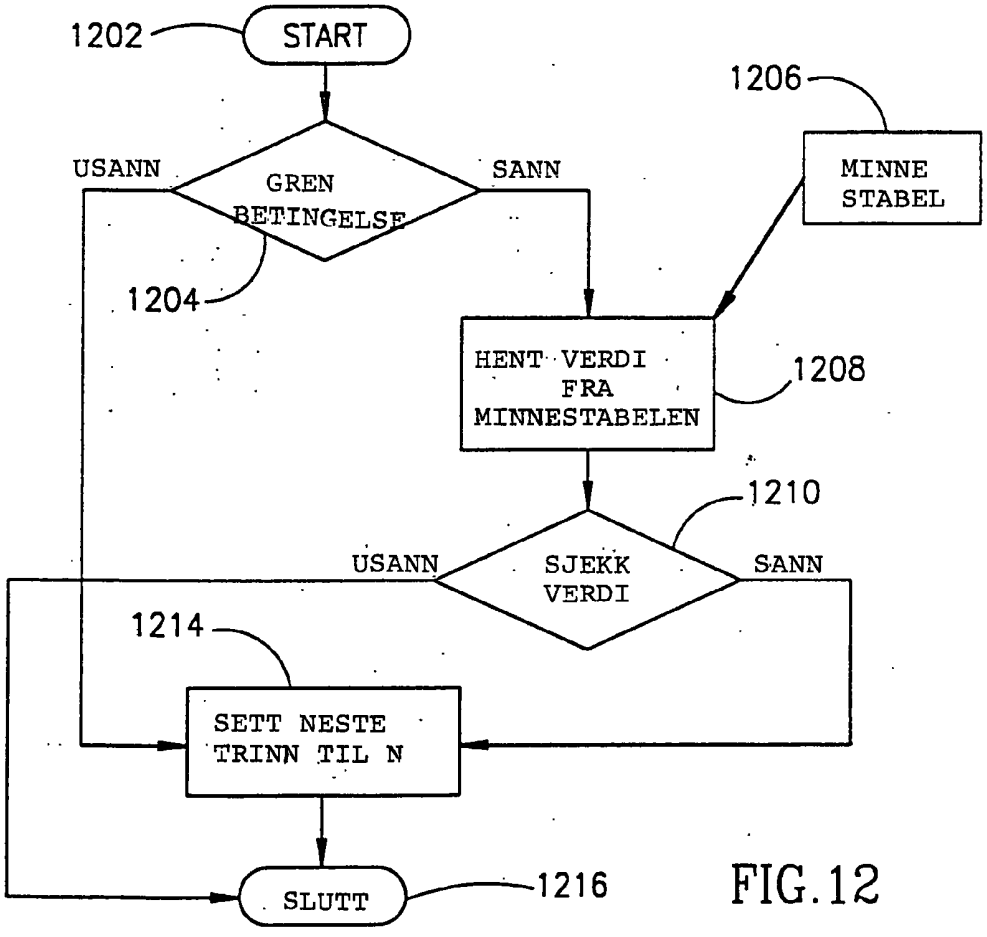

1202
START
1206
USANN
GREN BETINGELSE
SANN
MINNE STABEL
1204
HENT VERDI FRA MINNESTABELEN
1208
1210
USANN
SJEKK VERDI
SANN
1214
SETT NESTE TRINN TIL N
SLUTT
1216

FIG.12

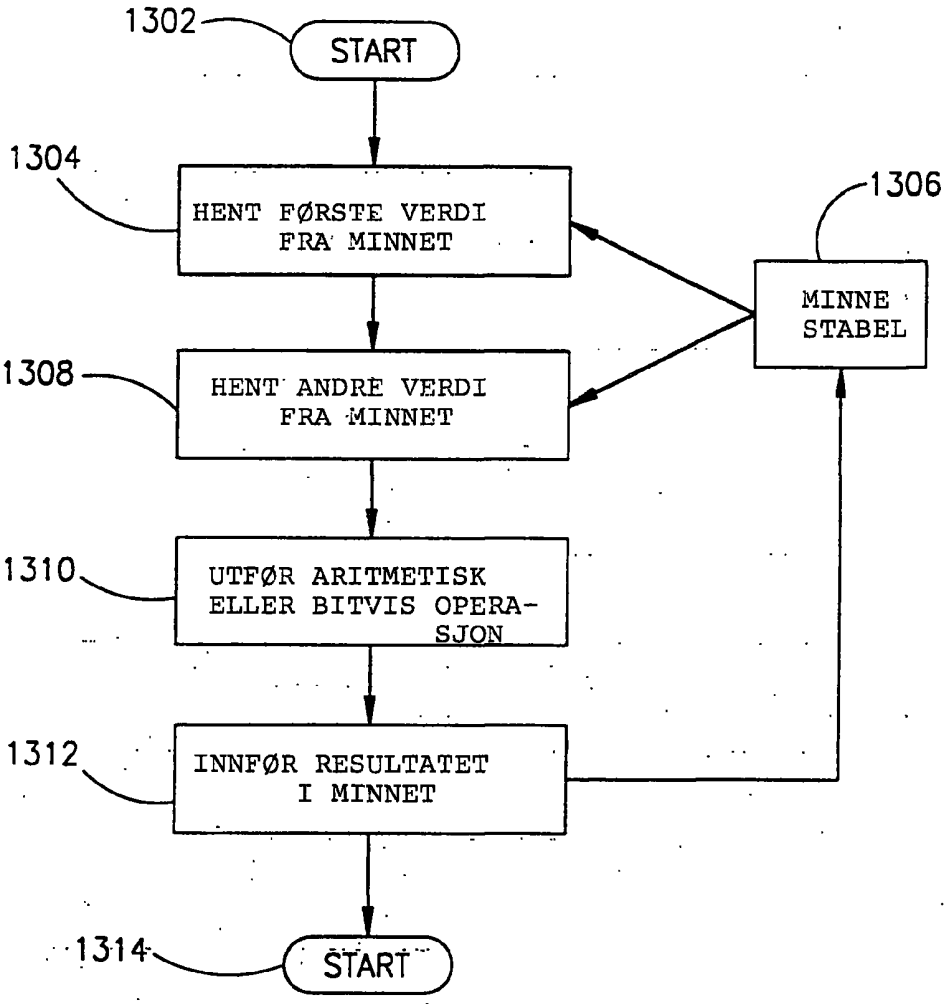

1302
START
1304
HENT FØRSTE VERDI FRA MINNET
1306
MINNE STABEL
1308
HENT ANDRE VERDI FRA MINNET
1310
UTFØR ARITMETISK ELLER BITVIS OPERA-SJON
1312
INNFØR RESULTATET I MINNET
1314
START

FIG.13

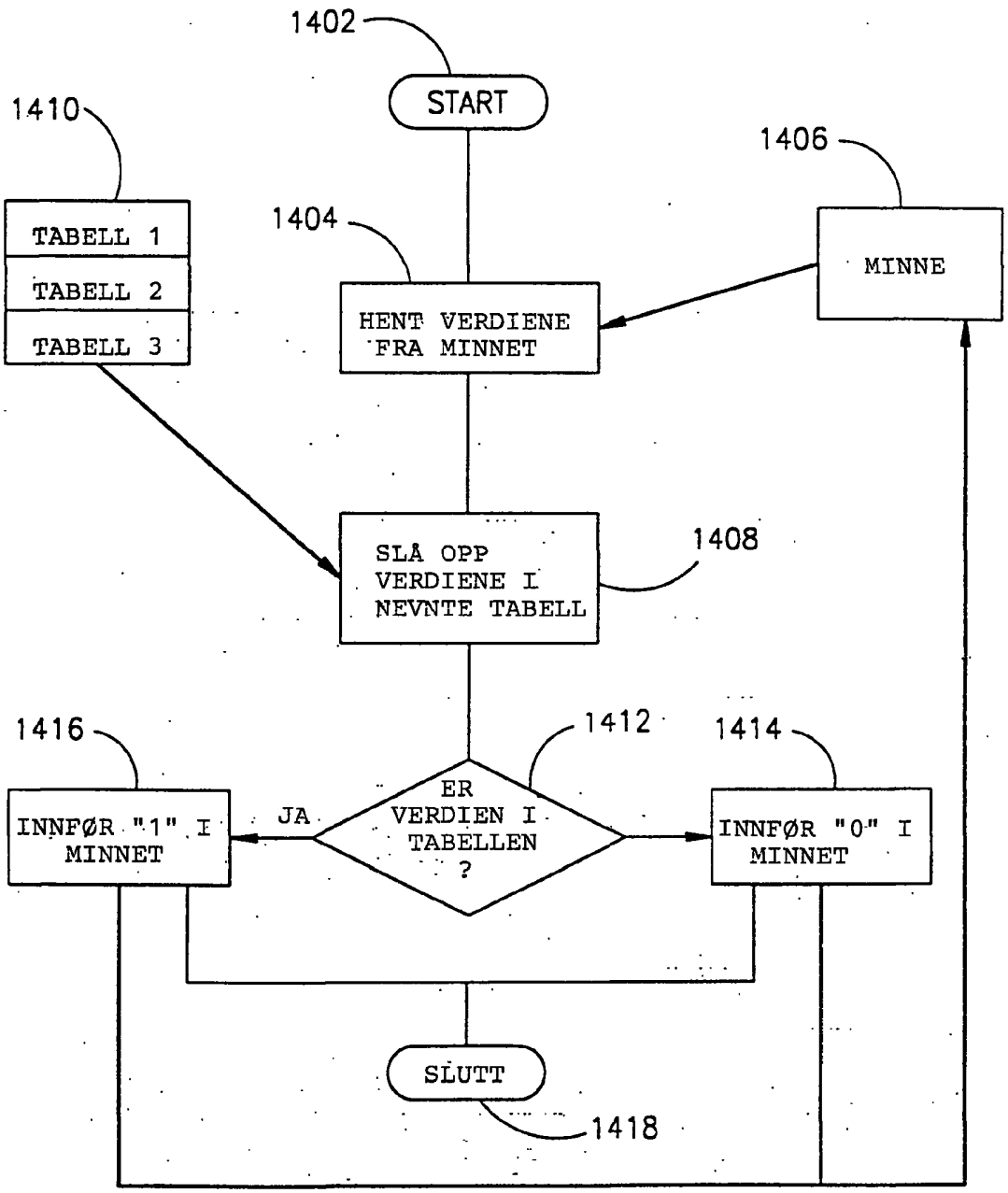

1402
START
1410
TABELL 1
TABELL 2
TABELL 3
1406
MINNE
1404
HENT VERDIENE FRA MINNET
SLÅ OPP VERDIENE I NEVNTE TABELL
1408
1416
1412
1414
INNFØR "1" I MINNET
JA
ER VERDIEN I TABELLEN ?
INNFØR "0" I MINNET
SLUTT
1418

FIG.14

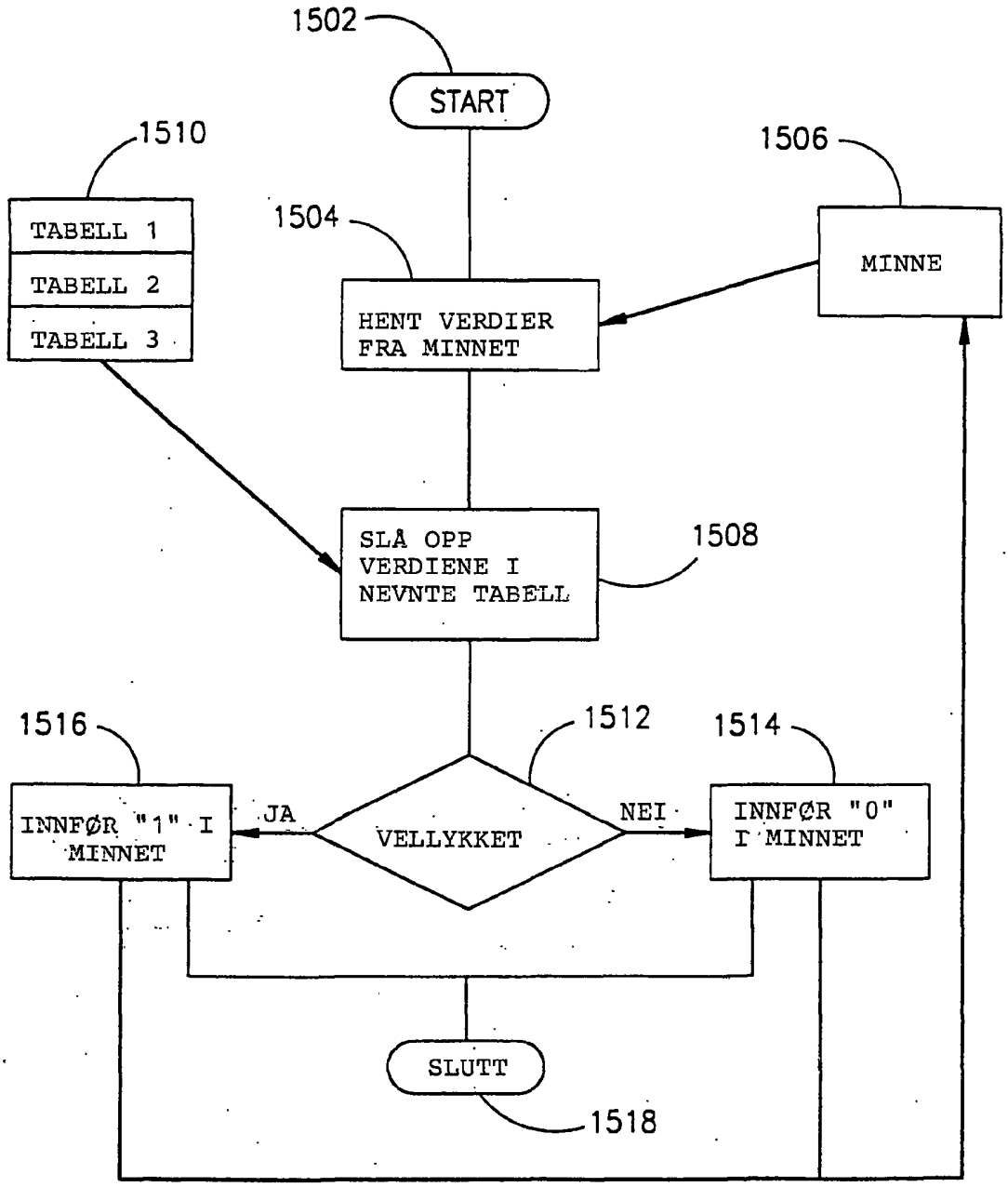

1502
START
1510
TABELL 1
TABELL 2
TABELL 3
1504
HENT VERDIER FRA MINNET
1506
MINNE
SLÅ OPP VERDIENE I NEVNTE TABELL
1508
1516
INNFØR "1" I MINNET
JA
1512
VELLYKKET
NEI
1514
INNFØR "0" I MINNET
SLUTT
1518

FIG.15

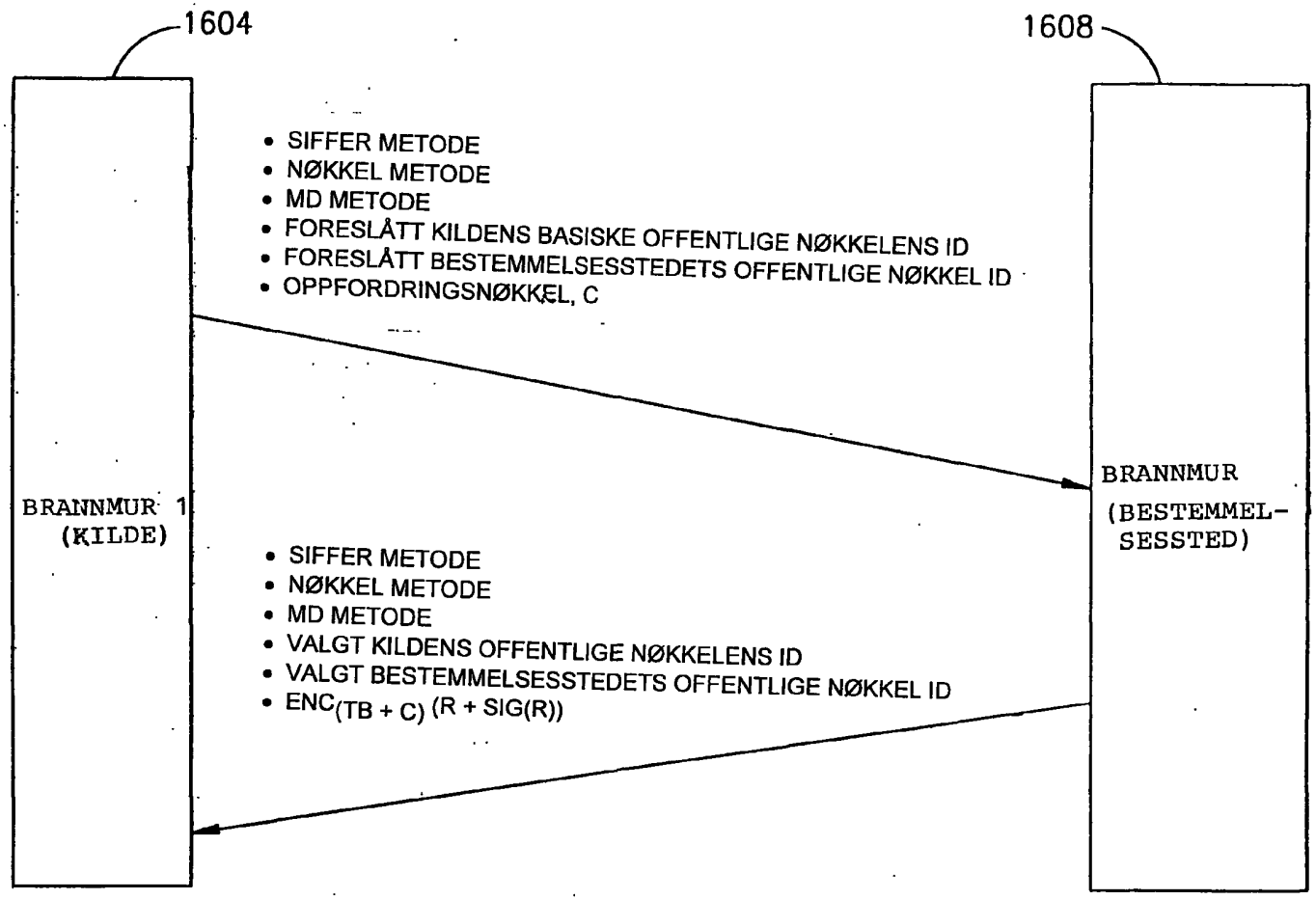

1604
1608
• SIFFER METODE
• NØKKEL METODE
• MD METODE
• FORESLÅTT KILDENS BASISKE OFFENTLIGE NØKKELENS ID
• FORESLÅTT BESTEMMELSESSTEDETS OFFENTLIGE NØKKEL ID
• OPPFORDRINGSNØKKEL, C
BRANNMUR 1
(KILDE)
BRANNMUR
(BESTEMMEL-
SESSTED)
• SIFFER METODE
• NØKKEL METODE
• MD METODE
• VALGT KILDENS OFFENTLIGE NØKKELENS ID
• VALGT BESTEMMELSESSTEDETS OFFENTLIGE NØKKEL ID
• ENC(TB + C) (R + SIG(R))

FIG.17

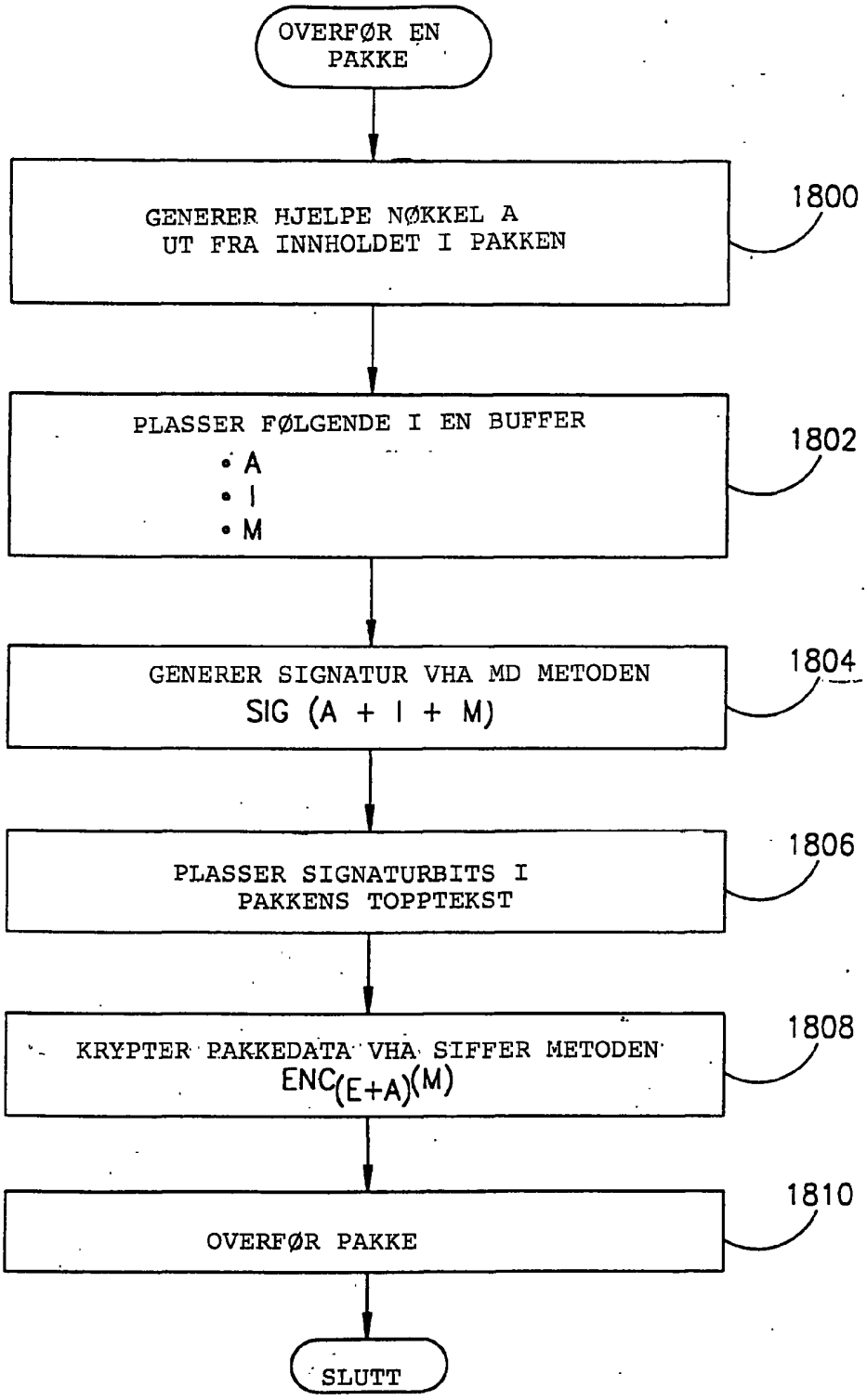

OVERFØR EN PAKKE
GENERER HJELPE NØKKEL A UT FRA INNHOLDET I PAKKEN
1800
PLASSER FØLGENDE I EN BUFFER
• A
• I
• M
1802
GENERER SIGNATUR VHA MD METODEN
SIG (A + I + M)
1804
PLASSER SIGNATURBITS I PAKKENS TOPPTEKST
1806
KRYPTER PAKKEDATA VHA SIFFER METODEN
ENC(E+A)(M)
1808
OVERFØR PAKKE
1810
SLUTT

FIG.18

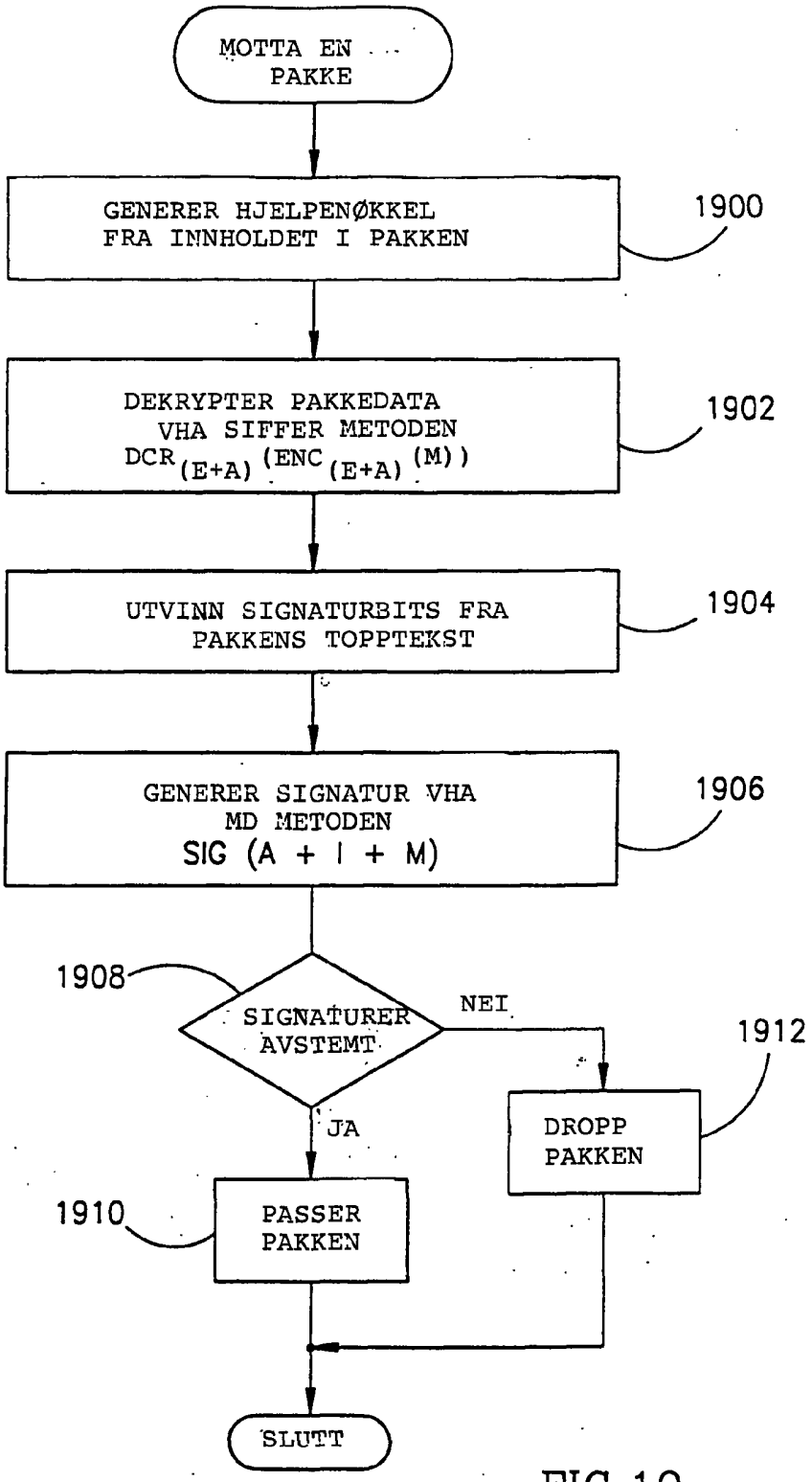

MOTTA EN PAKKE
GENERER HJELPENØKKEL FRA INNHOLDET I PAKKEN
1900
DEKRYPTER PAKKEDATA VHA SIFFER METODEN
DCR(E+A) (ENC(E+A) (M))
1902
UTVINN SIGNATURBITS FRA PAKKENS TOPPTEKST
1904
GENERER SIGNATUR VHA MD METODEN
SIG (A + I + M)
1906
1908
SIGNATURER AVSTEMT
NEI
1912
JA
DROPP PAKKEN
1910
PASSER PAKKEN
SLUTT

FIG.19

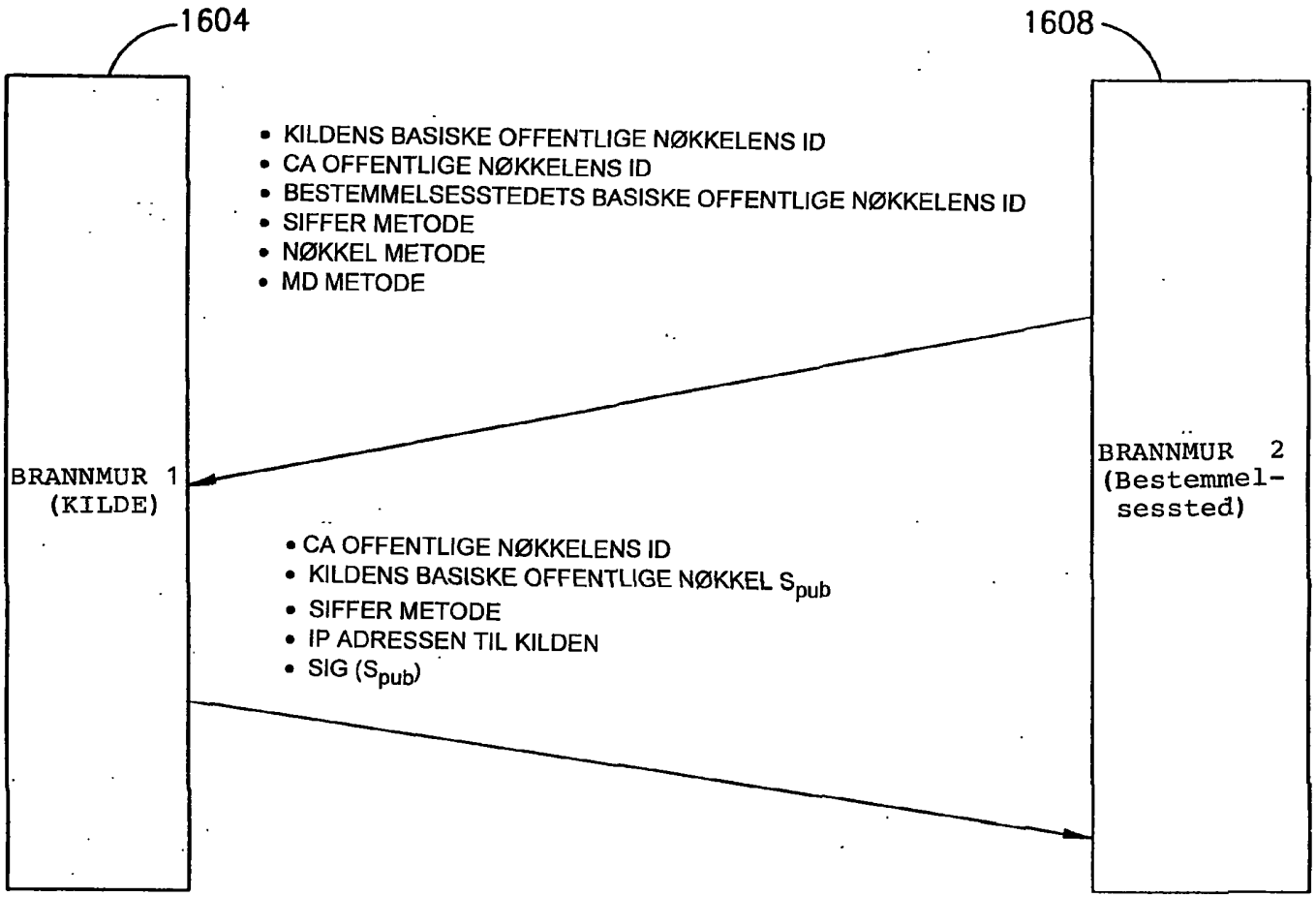

1604
1608
• KILDENS BASISKE OFFENTLIGE NØKKELENS ID
• CA OFFENTLIGE NØKKELENS ID
• BESTEMMELSESSTEDETS BASISKE OFFENTLIGE NØKKELENS ID
• SIFFER METODE
• NØKKEL METODE
• MD METODE
BRANNMUR 1 (KILDE)
BRANNMUR 2 (Bestemmel-sessted)
• CA OFFENTLIGE NØKKELENS ID
• KILDENS BASISKE OFFENTLIGE NØKKEL S_{pub}
• SIFFER METODE
• IP ADRESSEN TIL KILDEN
• SIG (S_{pub})

FIG.20

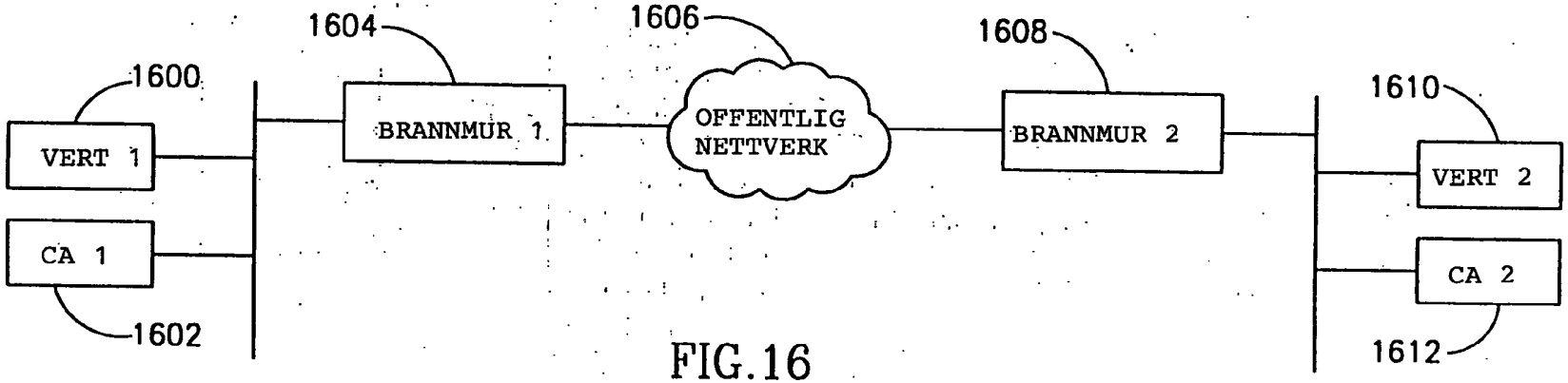

1600
1604
1606
1608
1610
VERT 1
BRANNMUR 1
OFFENTLIG NETTVERK
BRANNMUR 2
VERT 2
CA 1
CA 2
1602
1612

FIG.16

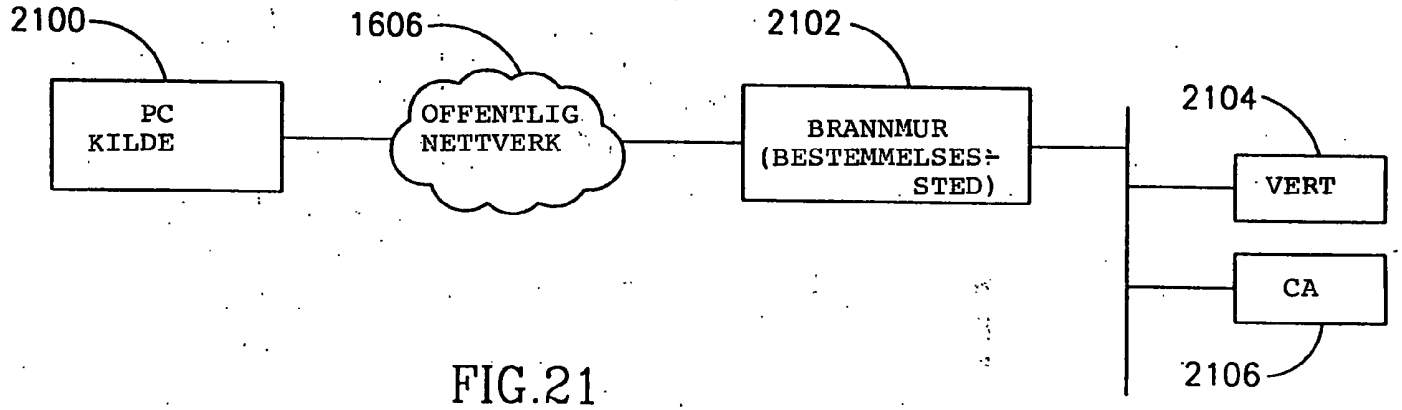

2100
1606
2102
2104
PC KILDE
OFFENTLIG NETTVERK
BRANNMUR (BESTEMMELSES-STED)
VERT
CA
2106

FIG.21

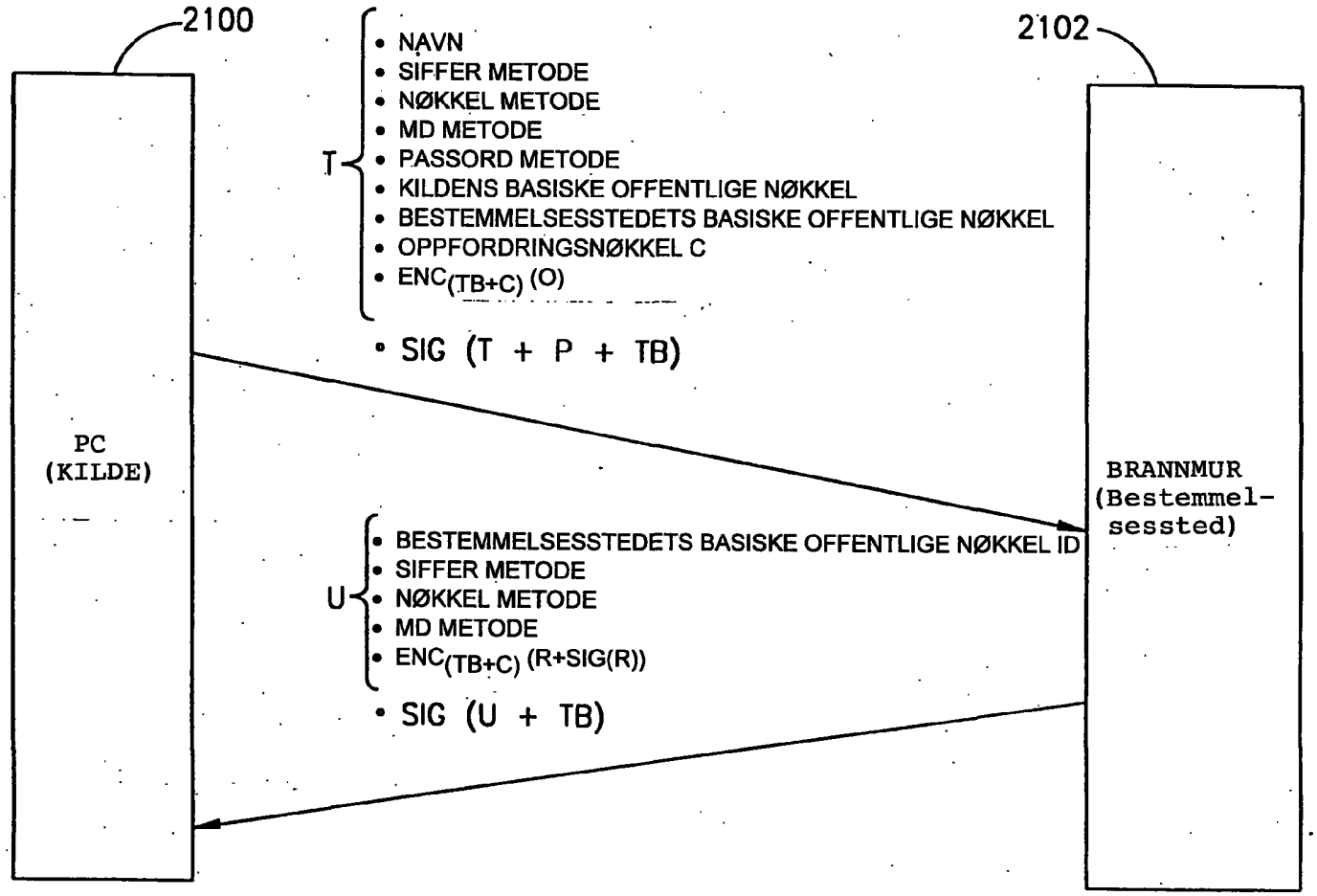

2100
2102
T
• NAVN
• SIFFER METODE
• NØKKEL METODE
• MD METODE
• PASSORD METODE
• KILDENS BASISKE OFFENTLIGE NØKKEL
• BESTEMMELSESSTEDETS BASISKE OFFENTLIGE NØKKEL
• OPPFORDRINGSNØKKEL C
• ENC(TB+C) (O)
• SIG (T + P + TB)
PC
(KILDE)
BRANNMUR
(Bestemmel-
sessted)
U
• BESTEMMELSESSTEDETS BASISKE OFFENTLIGE NØKKEL ID
• SIFFER METODE
• NØKKEL METODE
• MD METODE
• ENC(TB+C) (R+SIG(R))
• SIG (U + TB)

FIG.22

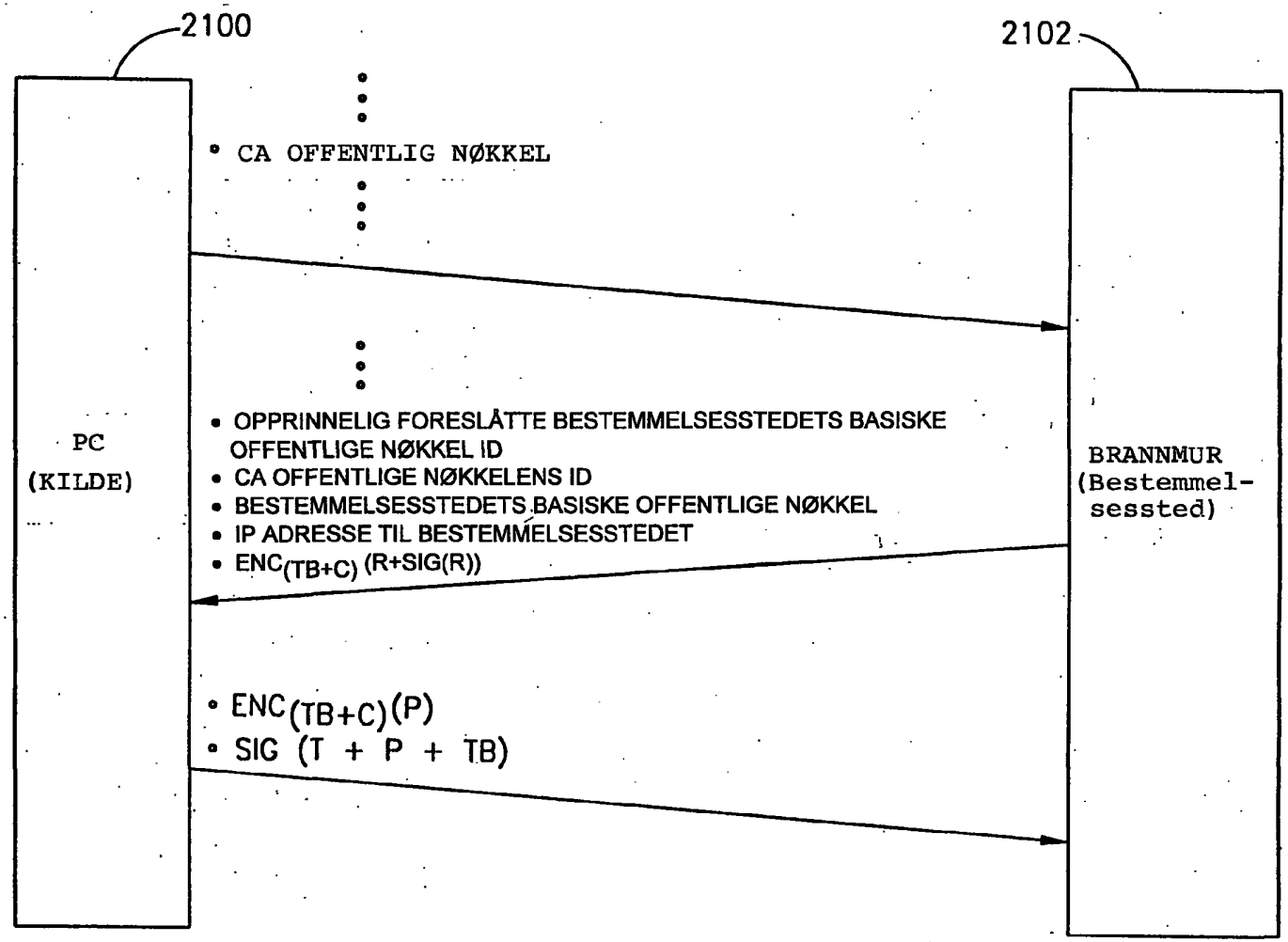

2100
2102
• CA OFFENTLIG NØKKEL
PC
(KILDE)
• OPPRINNELIG FORESLÅTTE BESTEMMELSESSTEDETS BASISKE OFFENTLIGE NØKKEL ID
• CA OFFENTLIGE NØKKELENS ID
• BESTEMMELSESSTEDETS BASISKE OFFENTLIGE NØKKEL
• IP ADRESSE TIL BESTEMMELSESSTEDET
• ENC(TB+C) (R+SIG(R))
BRANNMUR
(Bestemmel-
sessted)
• ENC(TB+C)(P)
• SIG (T + P + TB)

FIG.23