beim ISP **11** zu Beginn einer Kommunikationssitzung eingeloggt wurde. Wenn daher die Vorrichtung **12**(m) ein Nachrichtenpaket an eine Vorrichtung, z. B. einen Server **31**(s), in dem virtuellen privaten Netzwerk **15** unter Verwendung einer Klartext-Internetadresse übertragen möchte, welche z. B. durch einen Bediener bereitgestellt bzw. eingegeben wurde, greift die Vorrichtung **12**(m) zu Beginn auf den Namen-Server **17** zu, wie es oben beschrieben wurde, um zu versuchen, die zu der Klartext-Internetadresse zugehörige Zahlen-Internetadresse zu erhalten. Da der Namen-Server **17** außerhalb des virtuellen privaten Netzwerkes **15** ist und die durch die Vorrichtung **12**(m) angeforderten Information nicht besitzt, sendet er ein entsprechend lautendes Antwortnachrichtenpaket. Die Vorrichtung **12**(m) wird sodann ein Anfragenachrichtenpaket zur Übertragung an den Namen-Server **32** durch die Firewall **30** und über den Sicherheitstunnel erzeugen. Falls der Namen-Server **32** eine Zahlen-Internetadresse besitzt, welche zu der Klartext-Internetadresse in dem Anfragenachrichtenpaket gehört, welches durch die Vorrichtung **12**(m) geliefert wird, stellt er die Zahlen-Internetadresse in einer Weise bereit, welche im allgemeinen derjenigen ähnlich ist, welche oben im Zusammenhang mit dem Namen-Server **17** beschrieben wurde mit der Ausnahme, daß die Zahlen-Internetadresse durch den Namen-Server **32** in einem an die Firewall **30** gerichteten Nachrichtenpaket geliefert wird, und die Firewall **30** sodann das Nachrichtenpaket über den Sicherheitstunnel an die Vorrichtung **12**(m) übermittelt. Es versteht sich, daß sich in dem Nachrichtenpaket, welches durch die Firewall **30** übertragen wird, die Zahlen-Internetadresse in dem Nachrichtenpaket im Datenabschnitt des Nachrichtenpakets befindet, welches über den Sicherheitstunnel übertragen wird und entsprechend verschlüsselt sein wird. Das Nachrichtenpaket wird durch die Vorrichtung **12**(m) in einer ähnlichen Weise verarbeitet, wie sie oben im Zusammenhang mit anderen Nachrichtenpaketen beschrieben wurde, welche durch die Vorrichtung **12**(m) über den Sicherheitstunnel empfangen werden. Das heißt, daß das Nachrichtenpaket durch den Sicherheits-Paketprozessor **26** vor dem Übermitteln an den Paketempfänger und -prozessor **23** zur Verarbeitung entschlüsselt wird. Die Zahlen-Internetadresse für den Server **31**(s) kann in einem Cache in einer Zugriffskontrolliste (ACL) in dem IP-Parameterspeicher **25** gespeichert werden, zusammen mit der Zuordnungsinformation bezüglich der zugehörigen Klartext-Internetadresse, einer Angabe, daß der Server **31**(s), der dieser Klartext-Internetadresse zugeordnet ist, über die Firewall **30** des virtuellen privaten Netzwerkes **15** zugänglich ist, und die Identifizierungen der Verschlüsselungs- und Entschlüsselungsalgorithmen und -schlüssel, welche für eine Verschlüsselung und Entschlüsselung der geeigneten Abschnitte der Nachrichtenpakete zu verwenden sind, welche an den Server **31**(s) übertragen und von diesem erhalten werden.

Es versteht sich, daß in Reaktion auf ein Nachrichtenpaket von der Vorrichtung **12**(m), welches beim Namen-Server **32** die Bereitstellung einer Zahlen-Internetadresse für eine durch die Vorrichtung **12**(m) angegebene Klartext-Internetadresse anfordert, falls der Namen-Server **32** keine Zuordnungsinformation zwischen der Klartext-Internetadresse und einer Zahlen-Internetadresse besitzt, der Namen-Server **32** ein Antwortnachrichtenpaket, das entsprechend lautet, übertragen kann. Falls die Vorrichtung **12**(m) eine Identifizierung von anderen Namen-Servern besitzt, welche z. B. mit anderen virtuellen privaten Netzwerken (nicht gezeigt) verbunden sein können und zu welchen die Vorrichtung **12**(m) Zugriff hat, dann kann die Vorrichtung **12**(m) versuchen, auf die anderen Namen-Server in einer ähnlichen Weise, wie es oben beschrieben ist, zuzugreifen. Falls die Vorrichtung **12**(m) nicht in der Lage ist, eine Zahlen-Internetadresse, welche der Klartext-Internetadresse zugeordnet ist, von irgendeinem der Namen-Server zu erhalten, zu welchem sie Zugriff hat und welche im allgemeinen im IP-Parameterspeicher **25** der Vorrichtung **12**(m) identifiziert sind, wird sie allgemein nicht in der Lage sein, auf eine Vorrichtung mit der vorgegebenen Klartext-Internetadresse zuzugreifen und wird den Bediener oder ein Programm, welche den Zugriff angefordert haben, dementsprechend unterrichten.

Mit diesem Hintergrund werden nun Operationen, welche durch die Vorrichtung **12**(m) und das virtuelle private Netzwerk **15** in Verbindung mit der vorliegenden Erfindung durchgeführt werden, im Detail beschrieben. Im allgemeinen laufen die Operationen in zwei Phasen ab. In einer ersten Phase arbeiten die Vorrichtung **12**(m) und das virtuelle private Netzwerk **15** zusammen, um einen Sicherheitstunnel durch das Internet **14** aufzubauen. In dieser ersten Phase liefert das virtuelle private Netzwerk **15**, insbesondere die Firewall **30**, die Identifizierung eines Namen-Servers **32**, und es kann auch die den Verschlüsselungs- und Entschlüsselungsalgorithmus und -schlüssel betreffende Information bereitstellen, wie es oben beschrieben wurde. In der zweiten Phase, nachdem der Sicherheitstunnel eingerichtet wurde, kann die Vorrichtung **12**(m) die während der ersten Phase gelieferten Information im Zusammenhang mit der Erzeugung und Übertragung von Nachrichtenpaketen an einen oder mehrere Server **31**(s) in dem virtuellen privaten Netzwerk **15** und bei dem notwendigen Umwandlungsvorgang der Klartext-Internetadressen zu Zahlen-Internetadressen aus dem Namen-Server **32**, welcher durch die Firewall **30** während der ersten Phase identifiziert wurde, verwenden.

Folglich erzeugt die Vorrichtung **12**(m) in der ersten (Sicherheitstunnelaufbau)phase zu Beginn ein Nachrichtenpaket zur Übertragung an die Firewall **30**, welches einen Aufbau eines Sicherheitstunnels anfordert. Das Nachrichtenpaket enthält eine Zahlen-Internetadresse für die Firewall, (welche durch den Bediener der Vorrichtung oder ein Programm bereitgestellt werden kann, welches durch die Vorrichtung **12**(m) verarbeitet wird, oder durch den Namen-Server **17** bereitgestellt werden kann, nachdem eine Klartext-Internetadresse durch den Bediener oder ein Programm bereitgestellt wurde), und welche insbesondere dazu dient, die Firewall **30** zu veranlassen, mit der Vorrichtung **12**(m) einen Sicherheitstunnel aufzubauen. Falls die Firewall **30** die Anfrage bezüglich des Sicherheitstunnelaufbaus akzeptiert und falls die Firewall **30** die Verschlüsselungs- und Entschlüsselungsalgorithmen und -schlüssel bereitstellt, so wie es oben angegeben wurde, erzeugt die Firewall **30** ein Antwortnachrichtenpaket zur Übertragung an die Vorrichtung **12**(m), welches die Verschlüsselungs- und Entschlüsselungsalgorithmen und -schlüssel identifiziert. Wie oben beschrieben, wird dieses Antwortnachrichtenpaket nicht verschlüsselt. Wenn die Vorrichtung **12**(m) die Antwort empfängt, werden die Identifizierungen der Verschlüsselungs- und Entschlüsselungsalgorithmen und -schlüssel in dem IP-Parameterspeicher **25** gespeichert.

Zu einem späteren Zeitpunkt in der ersten Phase erzeugt die Firewall **30** auch ein Nachrichtenpaket zur Übertragung an die Vorrichtung **12**(m), welches die Zahlen-Internetadresse des Namen-Servers **32** enthält. Bei diesem Nachrichtenpaket wird der Abschnitt des Nachrichtenpakets, welcher die Zahlen-Internetadresse des Namen-Servers **32** enthält, unter Verwendung eines Verschlüsselungsalgorithmus und Verschlüsselungsschlüssels verschlüsselt, und dies kann unter Verwendung des Entschlüsselungsalgorithmus und -schlüssels, die durch das zuvor beschriebene Antwortnachrichtenpaket geliefert wurden, wieder entschlüsselt werden. Diese Nachricht hat im allgemeinen die folgende Struktur:

"<IIA(FW),IIA(DEV12(m)><SEC_TUN>
<ENCR<<IIA(FW),IIA(DEV_12(m)><(DNS_ADRS:IIA(-
NS_2>>>"

wobei

(i) "IIA(FW)" die Quellenadresse darstellt, d. h. eine Zahlen-Internetadresse der Firewall **30**,
(ii) "IIA(DEV_12(m))" die Zieladresse darstellt, d. h. die Zahlen-Internetadresse der Vorrichtung **12** (m),
(iii) "DNS_ADRS:IIA(NS)" angibt, daß "IIA(NS_32)" die Zahlen-Internetadresse des Namen-Servers **32** darstellt, für dessen Benutzung die Vorrichtung **12**(m) authorisiert ist, und
(iv) "ENCR<. . . .>" bedeutet, daß die Information, zwischen den Klammern "<" und ">" verschlüsselt ist.

Der Anfangsabschnitt der Nachricht "IIA(FW),IIA(DEV_12(m))>" bildet wenigstens einen Teil des Kopfabschnitts der Nachricht, und "<ENCR<<IIA(FW),IIA(DEV_12(m))><IIA(NS>>>" stellt wenigstens einen Teil des Datenabschnitts der Nachricht dar. "<SEC_TUN>" stellt einen Hinweis in dem Kopfabschnitt dar, welcher angibt, daß die Nachricht über den Sicherheitstunnel übertragen wird, wodurch auch angezeigt wird, daß der Datenabschnitt der Nachricht verschlüsselte Information enthält.

Nachdem die Vorrichtung **12**(m) die Nachricht von der Firewall **30** empfängt, wie es oben beschrieben wurde, und weil das Nachrichtenpaket den <SEC_TUN> Hinweis enthält, überträgt deren Netzwerkschnittstelle **21** den verschlüsselten Abschnitt "<ENCR<<IIA(FW),IIA(DEV_12(m)><DNS_ADRS:IIA(-NS_32)>>>" an den Sicherheits-Paketprozessor **26** zur Verarbeitung. Der Sicherheits-Paketprozessor **26** entschlüsselt den verschlüsselten Abschnitt, bestimmt weiter, daß der Abschnitt "IIA(NS_32)" die Zahlen-Internetadresse des Namen-Servers darstellt, insbesondere des Namen-Servers **32**, für dessen Benutzung die Vorrichtung **12**(m) authorisiert ist, und speichert diese Adresse in dem IP-Parameterspeicher **25** zusammen mit einer Angabe, daß die dorthin gerichteten Nachrichtenpakete zu der Firewall **30** zu übertragen sind, und daß die Daten in den Nachrichtenpaketen unter Verwendung des Verschlüsselungsalgorithmus und -schlüssels, die davor durch die Firewall **30** übermittelt wurden, zu verschlüsseln sind. Es versteht sich, daß aufgrund der Tatsache, daß die Zahlen-Internetadresse des Namen-Servers **32** von der Firewall an die Vorrichtung **12**(m) in verschlüsselter Form übertragen wird, diese vertraulich bleibt, selbst wenn das Paket durch einen Dritten abgefangen wird.

In Abhängigkeit des speziellen Protokolls, welches für den Aufbau des Sicherheitstunnels verwendet wird, können die Firewall **30** und die Vorrichtung **12**(m) auch Nachrichtenpakete austauschen, welche andere Information enthalten als die oben beschriebenen.

Wie oben erwähnt wurde, kann die Vorrichtung **12**(m) in der zweiten Phase nach der Einrichtung des Sicherheitstunnels die Information, welche während der ersten Phase bereitgestellt wurde, im Zusammenhang mit dem Erzeugen und Übertragen von Nachrichtenpaketen zu einem oder mehreren der Server **31**(s) in dem virtuellen privaten Netzwerk **15** nutzen. Falls bei diesen Operationen der Bediener einer Vorrichtung **12**(m) oder ein Programm, welches durch eine Vorrichtung **12**(m) verarbeitet wird, möchte, daß die Vorrichtung **12**(m) ein Nachrichtenpaket an einen Server **31**(s) in dem virtuellen privaten Netzwerk **15** überträgt, und falls der Bediener durch die Bedienerschnittstelle **20** oder das Programm eine Klartext-Internetadresse bereitstellt, wird zunächst die Vorrichtung **12**(m), insbesondere der Paketgenerator **22**, bestimmen, ob der IP-Parameterspeicher **25** dort in einem Cache eine Zahlen-Internetadresse gespeichert hat, welche zu der Klartext-Internetadresse gehört. Falls dies nicht der Fall ist, erzeugt der Paketgenerator **22** ein Anfragenachrichtenpaket zur Übertragung an den Namen-Server **17**, um von diesem die zu der Klartext-Internetadresse gehörige Zahlen-Internetadresse anzufordern. Falls der Namen-Server **17** eine zu der Klartext-Internetadresse gehörige Zahlen-Internetadresse besitzt, wird dieser die Zahlen-Internetadrese an die Vorrichtung **12**(m) liefern. Es versteht sich, daß dies nur erfolgen kann, wenn die Klartext-Internetadresse im Anfragenachrichtenpaket sowohl einer Vorrichtung **13** außerhalb des virtuellen privaten Netzwerkes **15** als auch einem Server **32**(s) in dem virtuellen privaten Netzwerk **15** zugeordnet wurde. Danach kann die Vorrichtung **12**(m) die Zahlen-Internetadresse verwenden, um Nachrichtenpakete zur Übertragung über das Internet zu erzeugen, wie es oben beschrieben wurde.

Falls andererseits angenommen wird, daß der Namen-Server **17** keine der Klartext-Internetadresse zugeordnete Zahlen-Internetadresse besitzt, wird der Namen-Server **17** ein entsprechend lautendes Antwortnachrichtenpaket an die Vorrichtung **12**(m) übermitteln. Sodann erzeugt der Paketgenerator **22** der Vorrichtung **12**(m) ein Anfragenachrichtenpaket zur Übertragung an den nächsten Namen-Server, der in ihrem IP-Parameterspeicher **25** identifiziert ist, um von diesem Namen-Server die der Klartext-Internetadresse zugeordnete Zahlen-Internetadresse anzufordern. Falls dieser nächste Namen-Server der Namen-Server **32** ist, liefert der Paketgenerator **22** das Nachrichtenpaket an den Sicherheits-Paketprozessor **26** zur weiteren Verarbeitung. Der Sicherheits-Paketprozessor **26** erzeugt daraufhin ein Anfragenachrichtenpaket zur Übertragung über den Sicherheitstunnel an die Firewall **30**. Diese Nachricht hat im allgemeinen folgende Struktur:

"<IIA(DEV_12(m)),IIA(FW)><SEC_TUN>
<ENCR<<IIA(DEV_12(m)),IIA(NS_32))><IIA_REQ>>>-
"

wobei

(i) "IIA(DEV_12(m))" die Quellenadresse darstellt, d. h. die Zahlen-Internetadresse der Vorrichtung **12**(m),
(ii) "IIA(FW)" die Zieladresse darstellt, d. h. die Zahlen-Internetadresse der Firewall **30**,
(iii) "IIA(NS_32)" die Adresse des Namen-Servers **32** darstellt,
(iv) "<<IIA(DEV_12(m)),IIA(NS_32))><IIA_REQ>->>" das Anfragenachrichtenpaket darstellt, welches durch den Paketgenerator **22** erzeugt wird, wobei "<IIA(DEV_12(m)),IIA(NS_32)>" den Kopfabschnitt des Anfragenachrichtenpakets und "<IIA_REQ>" den Datenabschnitt des Anfragenachrichtenpakets darstellt,
(v) "ENCR<. . . .>" angibt, daß die Information zwischen den Klammern "<" und ">" verschlüsselt ist, und
(vi) "<SEC_TUN>" einen Hinweis in dem Kopfabschnitt des Nachrichtenpakets darstellt, welches durch den Sicherheitspaketgenerator **26** erzeugt wird und angibt, daß die Nachricht über den Sicherheitstunnel übertragen wird, wobei hierdurch angegeben wird, daß der Datenabschnitt der Nachricht verschlüsselte Information enthält.

Wenn die Firewall **30** das durch den Sicherheitspaketgenerator **26** erzeugte Anfragenachrichtenpaket empfängt, wird diese den verschlüsselten Abschnitt des Nachrichtenpakets entschlüsseln, um "<<IIA(DEV_12(m)),IIA(NS_32))><IIA_REQ>>" zu erhalten. Dies stellt das Anfragenachrichtenpaket dar, welches durch den Paketgenerator **22** erzeugt wird. Nachdem das Anfragenachrichtenpaket erhalten wurde, überträgt die Firewall **30** dieses über die Übertragungsverbindung **33** an den Namen-Server **32**. In Abhängigkeit von dem Protokoll zur Übertragung von Nachrichtenpaketen über die Übertragungsverbindung **33** kann es bei diesem Prozeß für die Firewall **30** notwendig sein, das Anfragenachrichtenpaket zu modifizieren, damit es dem Protokoll der Übertragungsverbindung **33** entspricht.

Nachdem der Namen-Server **32** das Anfragenachrichtenpaket erhalten hat, wird dieser das Anfragenachrichtenpaket verarbeiten, um zu bestimmen, ob er eine der Klartext-Internetadresse, welche in dem Anfragenachrichtenpaket gesendet wird, zugeordnete Zahlen-Internetadresse besitzt. Falls der Namen-Server feststellt, daß er eine solche Zahlen-Internetadresse aufweist, wird dieser ein Antwortnachrichtenpaket zur Übertragung an die Firewall erzeugen, welches die Zahlen-Internetadresse enthält. Im allgemeinen hat das Antwortnachrichtenpaket die folgende Struktur:

"<<IIA(NS_32),IIA(DEV_12(m)><IIA_RESP>>"

wobei

(i) "IIA(NS_32)" die Quellenadresse darstellt, d. h. die Zahlen-Internetadresse des Namen-Servers **32**,
(ii) "IIA(DEV_12(m))" die Zieladresse darstellt, d. h. die Zahlen-Internetadresse der Vorrichtung **12**(m), und
(iii) "IIA_RESP" die Zahlen-Internetadresse darstellt, welche der Klartext-Internetadresse zugeordnet ist.

Nachdem die Firewall **30** das Antwortnachrichtenpaket empfangen hat, und weil die Kommunikation mit der Vorrichtung **12**(m) über den dazwischenliegenden Sicherheitstunnel stattfindet, verschlüsselt die Firewall **30** das von dem Namen-Server **32** empfangene Antwortnachrichtenpaket und erzeugt ein Nachrichtenpaket zur Übertragung an die Vorrichtung **12**(m), welches das verschlüsselte Antwortnachrichtenpaket enthält. Im allgemeinen hat das durch die Firewall **30** erzeugte Nachrichtenpaket die folgende Struktur:

"<IIA(FW),IIA(DEV12(m)><SEC_TUN)>
<ENCR<<IIA(NS_32),IIA(DEV_12(m))><IIA_RESP>>>"

wobei

(i) "IIA(FW)" die Quellenadresse darstellt, d. h. die Zahlen-Internetadresse der Firewall **30**,
(ii) "IIA(DEV_12(m))" die Zieladresse darstellt, d. h. die Zahlen-Internetadresse der Vorrichtung **12**(m),
(iii) "SEC_TUN" einen Hinweis in dem Kopfabschnitt des Nachrichtenpakets darstellt, welches durch den Sicherheitspaketgenerator **26** erzeugt wird, und angibt, daß die Nachricht über den Sicherheitstunnel übertragen wird, und wobei auch angegeben wird, daß der Datenabschnitt der Nachricht verschlüsselte Information enthält,
(iv) "ENCR<. . . .>" angibt, daß die Information zwischen den Klammern "<" und ">" (was dem von dem Namen-Server **32** empfangenen Antwortnachrichtenpaket entspricht) verschlüsselt ist.

Zusätzlich kann es je nach dem Protokoll zur Übertragung von Nachrichtenpaketen über die Übertragungsverbindung **33** für die Firewall **30** notwendig sein, das Nachrichtenpaket zu bearbeiten und/oder zu modifizieren, damit dieses dem Protokoll des Internets **14** entspricht.

Wenn die Vorrichtung **12**(m) das Nachrichtenpaket von der Firewall **30** empfängt, wird das Nachrichtenpaket an den Sicherheits-Paketprozessor **26** geliefert. Der Sicherheitspaketprozessor **26** entschlüsselt daraufhin den verschlüsselten Abschnitt des Nachrichtenpakets, um die der Klartext-Internetadresse zugeordnete Zahlen-Internetadresse zu erhalten und lädt diese Information in den IP-Parameterspeicher **25**. Danach kann die Vorrichtung diese Zahlen-Internetadresse beim Erzeugen von Nachrichtenpaketen zur Übertragung an den Server **31**(s) verwenden, welcher zu der Klartext-Internetadresse gehört.

Es versteht sich, daß, falls der Namen-Server **32** keine Zahlen-Internetadresse besitzt, welche der durch die Vorrichtung **12**(m) in dem Anfragenachrichtenpaket gelieferte Klartext-Internetadresse zugeordnet ist, dies der Namen-Server **32** in dem durch ihn erzeugten Antwortnachrichtenpaket entsprechend anzeigen. Die Firewall **30** erzeugt dann in Reaktion auf das durch den Namen-Server **32** gelieferte Antwortnachrichtenpaket auch ein Nachrichtenpaket zur Übertragung an die Vorrichtung **12**(m), welches einen verschlüsselten Abschnitt enthält, der das Antwortnachrichtenpaket umfaßt, das durch den Namen-Server **32** erzeugt wurde. Nachdem die Vorrichtung **12**(m) das Nachrichtenpaket empfangen hat, wird der verschlüsselte Abschnitt durch den Sicherheitspaketprozessor **26** entschlüsselt, welcher daraufhin den Paketgenerator **22** darüber informiert, daß der Namen-Server **32** keine der Klartext-Internetadresse zugeordnete Zahlen-Internetadresse besitzt. Falls der IP-Parameterspeicher **25** die Identifizierung eines anderen Namen-Servers enthält, erzeugt sodann der Paketgenerator **22** der Vorrichtung **12**(m) ein Anfragenachrichtenpaket zur Übertragung an den nächsten Namen-Server, der in deren IP-Parameterspeicher **25** identifiziert ist, um von diesem Namen-Server die Zahlen-Internetadresse anzufordern, welche der Klartext-Internetadresse zugeordnet ist. Falls andererseits der IP-Parameterspeicher **25** keine Identifizierung eines anderen Namen-Servers enthält, kann der Paketgenerator **22** die Bedienerschnittstelle **20** oder ein Programm darüber informieren, daß er nicht in der Lage ist, ein Nachrichtenpaket zur Übertragung an eine Vorrichtung zu erzeugen, welche der Klartext-Internetadresse zugeordnet ist, welche durch die Bedienerschnittstelle **20** oder ein Programm eingegeben bzw. bereitgestellt wurde.

Die Erfindung liefert eine Anzahl von Vorteilen. Insbesondere schafft die Erfindung ein System zum Vereinfachen der Kommunikation zwischen Vorrichtungen, welche mit einem öffentlichen Netzwerk verbunden sind, z. B. mit dem Internet **14**, und Vorrichtungen, welche mit privaten Netzwerken verbunden sind, z. B. mit dem virtuellen privaten Netzwerk **15**, indem die Umwandlung von Klartextadressen in Netzwerkadressen durch einen Namen-Server, der bevorzugt über einen Sicherheitstunnel mit den privaten Netzwerken verbunden ist, ermöglicht wird.

Es versteht sich, daß eine Vielzahl von Modifikationen an der im Zusammenhang mit **Fig. 1** beschriebenen Anordnung durchgeführt werden können. Obwohl das Netzwerk **10** so beschrieben wurde, daß die Identifizierung der Verschlüsselungs- und Entschlüsselungsalgorithmen und -schlüssel durch die Vorrichtung **12**(m) und die Firewall **30** während des Dialogs, währenddessen der Sicherheitstunnel eingerichtet wird, ausgetauscht wird, versteht es sich, daß beispielsweise Information durch die Vorrichtung **12**(m) und die Firewall **30** getrennt von dem Aufbau eines solchen Sicherheitstunnels bereitgestellt werden können.

Obwohl die Erfindung im Zusammenhang mit dem Internet beschrieben wurde, versteht es sich ferner, daß die Erfindung in Verbindung mit jedem, insbesondere globalen, Netzwerk verwendet werden kann. Obwohl die Erfindung im Zusammenhang mit einem Netzwerk beschrieben wurde, welches ein System von Klartext-Netzwerkadressen bereitstellt, versteht es sich ferner, daß die Erfindung nicht darauf beschränkt ist sondern in Verbindung mit jedem Netzwerk verwendet werden kann, welches irgendeine Form einer – den systemeigenen Netzwerkadressen übergeordnete Sekundär-Netzwerkadresseneinrichtung oder vergleichbare nicht-formeller Netzwerkadresseneinrichtung vorsieht.

Es versteht sich ferner, daß ein erfindungsgemäßes System als ganzes oder in Teilen aus speziell hierfür geeigneter Hardware oder einem allgemein geeigneten Computersystem oder jeder Kombination davon aufgebaut werden kann, wobei jeder Abschnitt davon durch ein geeignetes Programm gesteuert werden kann. Jedes Programm kann als ganzes oder in Teilen einen Teil des Systems umfassen oder auf dem System in einer konventionellen Weise gespeichert sein, oder es kann als ganzes oder in Teilen in das System über ein Netzwerk oder andere Mechanismen zur Übertragung von Information in einer konventionellen Weise bereitgestellt werden. Zusätzlich versteht es sich, daß das System betrieben und/oder auf andere Art und Weise mittels Information gesteuert werden kann, welche durch einen Bediener mittels Bedienereingabeelementen (nicht gezeigt) bereitgestellt wird, welche direkt an das System angeschlossen sein können oder welche die Information über ein Netzwerk oder ändere Mechanismen zur Übertragung von Information in einer konventionellen Weise übertragen können.

Die vorstehende Beschreibung hat sich auf ein spezifisches Ausführungsbeispiel der Erfindung bezogen. Es versteht sich jedoch, daß verschiedene Variationen und Modifikationen der Erfindung gemacht werden können, bei welchen einige oder alle der Vorteile der Erfindung erreicht werden. Diese und andere Variationen und Modifikationen fallen in den Schutzbereich der vorliegenden Erfindung, der durch die nachfolgenden Ansprüche bestimmt ist.

Patentansprüche

1. System umfassend ein virtuelles privates Netzwerk (**15**) und eine externe Vorrichtung (12 (m)), welche über ein digitales Netzwerk (**14**) kommunizieren, wobei:
das virtuelle private Netzwerk (**15**) eine Firewall (**30**), wenigstens eine interne Vorrichtung (**31**(s)) und einen Namen-Server (**32**) aufweist, welche jeweils eine Netzwerkadresse besitzen, wobei die interne Vorrichtung (**31**(s)) auch eine Sekundäradresse besitzt und der Namen-Server (**32**) derart konfiguriert ist, daß er eine Zuordnung zwischen der Sekundäradresse und der Netzwerkadresse bereitstellt,
die Firewall (**30**) derart konfiguriert ist, daß sie der externen Vorrichtung (**12**(m)) in Reaktion auf deren Anfrage zum Aufbau einer Verbindung zur Firewall (**30**) die Netzwerkadresse des Namen-Servers (**32**) liefert, und
die externe Vorrichtung (**12**(m)) derart konfiguriert ist, daß sie in Reaktion auf eine Anfrage zum Zugriff auf die interne Vorrichtung (**31**(s)), welche die Sekundäradresse der internen Vorrichtung (**31**(s)) enthält, eine Netzwerkadressen-Anfragenachricht zur Übertragung über die Verbindung an die Firewall (**30**) erzeugt, welche eine Auflösung der der Sekundäradresse zugeordneten Netzwerkadresse anfordert, wobei die Firewall (**30**) derart konfiguriert ist, daß sie die Adressenauflösungsanfrage an den Namen-Server (**32**) übermittelt, der Namen-Server (**32**) derart konfiguriert ist, daß er die der Sekundäradresse zugeordnete Netzwerkadresse bereitstellt, und die Firewall (**30**) daraufhin die Netzwerkadresse in einer Netzwerkadressen-Antwortnachricht zur Übertragung über die Verbindung an die externe Vorrichtung (**12**(m)) bereitstellt.

2. System nach Anspruch 1, bei welchem die externe Vorrichtung (**12**(m)) derart konfiguriert ist, daß sie die in der Netzwerkadressen-Antwortnachricht bereitgestellte Netzwerkadresse beim Erzeugen von wenigstens einer Nachricht zur Übertragung an die interne Vorrichtung (**31**(s)) verwendet.

3. System nach Anspruch 1 oder 2, bei welchem die externe Vorrichtung (**12**(m)) derart konfiguriert ist, daß sie mit dem Netzwerk (**14**) durch einen Netzwerk-Service-Provider (**11**) verbunden wird.

4. System nach Anspruch 3, bei welchem die externe Vorrichtung (**12**(m)) derart konfiguriert ist, daß sie eine Kommunikationssitzung mit dem Netzwerk-Service-Provider (**11**) aufbaut, wobei der Netzwerk-Service-Provider (**11**) der externen Vorrichtung (**12**(m)) die Identifizierung eines weiteren Namen-Servers übermittelt, wobei der weitere Namen-Server derart konfiguriert ist, daß er eine Zuordnung zwischen einer Sekundäradresse und einer Netzwerkadresse für wenigstens eine Vorrichtung bereitstellt.

5. System nach einem der vorstehenden Ansprüche, bei welchem die externe Vorrichtung (**12**(m)) derart konfiguriert ist, daß sie eine Liste von Namen-Servern erhält, welche der externen Vorrichtung (**12**(m)) identifiziert wurden, und die externe Vorrichtung (**12**(m)) die Namen-Server in der Liste nacheinander in Reaktion auf eine Anfrage zum Zugriff auf eine andere Vorrichtung abfragt, wobei die Anfrage eine Sekundäradresse der anderen Vorrichtung enthält, solange bis die externe Vorrichtung (**12**(m)) eine Netzwerkadresse empfängt, wobei die externe Vorrichtung (**12**(m)) in jedem Abfragevorgang eine Netzwerkadressen-Anfragesnachricht zur Übertragung über das Netzwerk (**14**) erzeugt, welche durch einen der Namen-Server in der Liste zu beantworten ist, und von diesem eine Netzwerkadressen-Antwortnachricht empfängt.

6. System nach einem der vorstehenden Ansprüche, bei welchem die Verbindung zwischen der externen Vorrichtung (**12**(m)) und der Firewall (**30**) ein Sicherheitstunnel ist, in welchem wenigstens ein der zwischen der externen Vorrichtung (**12**(m)) und der Firewall (**30**) übertragenen Nachrichten verschlüsselt ist.

7. Verfahren zum Betreiben eines Systems umfassend ein virtuelles privates Netzwerk (**15**) und eine externe Vorrichtung (**12**(m)), welche durch ein digitales Netzwerk (**14**) miteinander verbunden sind, wobei das virtuelle private Netzwerk (**15**) eine Firewall (**30**), wenigstens eine interne Vorrichtung (**31**(s)) und einen Namen-Server (**32**) aufweist, welche jeweils eine Netzwerkadresse besitzen, wobei die interne Vorrichtung (**31**(s)) auch eine Sekundäradresse besitzt, und der Namen-Server (**32**) derart konfiguriert ist, daß er eine Zuordnung zwischen der Sekundäradresse und der Netzwerkadresse bereitstellt, wobei:

A. in Reaktion auf eine Anfrage der externen Vorrichtung (**12**(m)) zum Aufbau einer Verbindung zur Firewall (**30**) die Firewall (**30**) der externen Vorrichtung (**12**(m)) die Netzwerkadresse des

Namen-Servers (32) übermittelt; und

B. (i) in Reaktion auf eine Anfrage zum Zugriff auf die interne Vorrichtung (31(s)), welche die Sekundäradresse der internen Vorrichtung (31(s)) enthält, die externe Vorrichtung (12(m)) eine Netzwerkadressen-Anfragenachricht zur Übertragung über die Verbindung an die Firewall (30) erzeugt, welche eine Auflösung der Netzwerkadresse, welche der Sekundäradresse zugeordnet ist, anfordert,

(ii) die Firewall (30) die Adressenauflösungsanfrage an den Namen-Server (32) übermittelt, (iii) der Namen-Server (32) die der Sekundäradresse zugeordnete Netzwerkadresse bereitstellt, und

(iv) die Firewall (30) die Netzwerkadresse in einer Netzwerkadressen-Antwortnachricht zur Übertragung über die Verbindung an die externe Vorrichtung (12(m)) bereitstellt.

8. Verfahren nach Anspruch 7, bei welchem die externe Vorrichtung (12((m) ferner die in der Netzwerkadressen-Antwortnachricht bereitgestellte Netzwerkadresse beim Erzeugen von wenigstens einer Nachricht zur Übertragung an die interne Vorrichtung (31(s)) verwendet.

9. Verfahren nach Anspruch 7 oder 8, bei welchem die externe Vorrichtung (12(m)) mit dem Netzwerk (14) durch einen Netzwerk-Service-Provider (11) verbunden werden kann.

10. Verfahren nach Anspruch 9, bei welchem die externe Vorrichtung (12(m)) eine Kommunikationssitzung mit dem Netzwerk-Service-Provider (11) aufbaut, wobei der Netzwerk-Service-Provider (11) der externen Vorrichtung (12(m)) die Identifizierung eines weiteren Namen-Servers übermittelt, wobei der weitere Namen-Server eine Zuordnung zwischen einer Sekundäradresse und einer Netzwerkadresse für wenigstens eine Vorrichtung bereitstellt.

11. Verfahren nach einem der Ansprüche 7 bis 10, bei welchem die externe Vorrichtung (12(m)) eine Liste von Namen-Servern erhält, welche der externen Vorrichtung (12(m)) identifiziert wurden, und die externe Vorrichtung (12(m)) die Namen-Server in der Liste nacheinander in Reaktion auf eine Anfrage zum Zugriff auf eine andere Vorrichtung abfragt, wobei die Anfrage eine Sekundäradresse der anderen Vorrichtung enthält, solange bis die externe Vorrichtung (12(m)) eine Netzwerkadresse empfängt, wobei die externe Vorrichtung (12(m)) in jedem Abfragevorgang eine Netzwerkadressen-Anfragenachricht zur Übertragung über das Netzwerk (14) erzeugt, welche durch einen der Namen-Server in der Liste zu beantworten ist, und von diesem eine Netzwerkadressen-Antwortnachricht empfängt.

12. Verfahren nach einem der Ansprüche 7 bis 11, bei welchem die Verbindung zwischen der externen Vorrichtung (12(m)) und der Firewall (30) ein Sicherheitstunnel ist, in welchem wenigstens ein Abschnitt der zwischen der externen Vorrichtung (12(m)) und der Firewall (30) übertragenen Nachrichten verschlüsselt ist.

13. Computerprogramm-Produkt zur gemeinsamen Verwendung mit einem virtuellen privaten Netzwerk (15) und einer externen Vorrichtung (12(m)), welche durch ein digitales Netzwerk (14) miteinander verbunden sind, wobei das virtuelle private Netzwerk eine Firewall (30), wenigstens eine interne Vorrichtung (31(s)) und einen Namen-Server (32) aufweist, welche jeweils eine Netzwerkadresse besitzen, wobei die interne Vorrichtung (31(s)) auch eine Sekundäradresse besitzt, und der Namen-Server (32) derart konfiguriert ist, daß er eine Zuordnung zwischen der Sekundäradresse und der Netzwerkadresse bereitstellt, wobei das Computerprogrammprodukt ein maschinenlesbares Medium mit folgenden Codes aufweist:

A. ein Namen-Server-Identifizierungscodemodul, welches veranlaßt, daß die Firewall (30) der externen Vorrichtung (12(m)) in Reaktion auf deren Anfrage zum Aufbau einer Verbindung zur Firewall (30) die Netzwerkadresse des Namen-Servers (32) übermittelt,

B. ein Codemodul zur Erzeugung einer Netzwerkadressen-Anfragenachricht, welches veranlaßt, daß die externe Vorrichtung (12(m)) in Reaktion auf eine Anfrage zum Zugriff auf die interne Vorrichtung (31(s)), welche die Sekundäradresse der internen Vorrichtung (31(s)) enthält, eine Netzwerkadressen-Anfragenachricht zur Übertragung über die Verbindung an die Firewall (30) erzeugt, welche die Auflösung der der Sekundäradresse zugeordneten Netzwerkadresse anfordert,

C. ein Modul zur Übermittlung einer Adressenauflösungsanfrage, welches veranlaßt, daß die Firewall (30) die Adressenauflösungsanfrage an den Namen-Server (32) übermittelt,

D. ein Namen-Server-Steuerungsmodul, welches veranlaßt, daß der Namen-Server (32) die der Sekundäradresse zugeordnete Netzwerkadresse bereitstellt, und

E. ein Modul zur Übermittlung einer Netzwerkadressen-Antwortnachricht, welches veranlaßt, daß die Firewall (30) die Netzwerkadresse in einer Netzwerkadressen-Antwortnachricht zur Übertragung über die Verbindung an die externe Vorrichtung (12(m)) bereitstellt.

14. Computerprogramm-Produkt nach Anspruch 13, welches ferner ein Netzwerkadressenverwendungsmodul aufweist, welches veranlaßt, daß die externe Vorrichtung (12(m)) die in der Netzwerkadressen-Antwortnachricht übermittelte Netzwerkadresse beim Erzeugen von wenigstens einer Nachricht zur Übertragung an die interne Vorrichtung (31(s)) verwendet.

15. Computerprogramm-Produkt nach Anspruch 13 oder 14, welches ferner ein Netzwerk-Service-Provider-Steuerungsmodul aufweist, welches veranlaßt, daß die externe Vorrichtung (12(m)) mit dem Netzwerk (14) durch einen Netzwerk-Service-Provider (11) verbunden wird.

16. Computerprogramm-Produkt nach Anspruch 15, bei welchem das Netzwerk-Service-Provider-Steuerungsmodul ein Kommunikationssitzungsaufbaumodul umfaßt, welches veranlaßt, daß die externe Vorrichtung (12(m)) mit dem Netzwerk-Service-Provider (11) eine Kommunikationssitzung aufbaut und von diesem eine Identifizierung von einem weiteren Namen-Server empfängt.

17. Computerprogramm-Produkt nach einem der Ansprüche 13 bis 16, welches ferner ein Namen-Server-Abfragesteuerungsmodul aufweist, welches veranlaßt, daß die externe Vorrichtung (12(m)) eine Liste von Namen-Servern erhält, welche der externen Vorrichtung (12(m)) identifiziert wurden, und die Namen-Server in der Liste nacheinander in Reaktion auf eine Anfrage zum Zugriff auf eine andere Vorrichtung abfragt, wobei die Anfrage eine Sekundäradresse der anderen Vorrichtung enthält, solange bis die externe Vorrichtung (12(m)) eine Netzwerkadresse empfängt, und wobei die externe Vorrichtung (12(m)) in jedem Abfragevor-

gang eine Netzwerkadressen-Anfragesnachricht zur Übertragung über das Netzwerk (**14**) erzeugt, welche durch einen der Namen-Server in der Liste zu beantworten ist, und von diesem eine Netzwerkadressen-Antwortnachricht empfängt.

18. Computerprogramm-Produkt nach einem der Ansprüche 13 bis 17, bei welchem die Verbindung zwischen der externen Vorrichtung (**12**(m)) und der Firewall (**30**) ein Sicherheitstunnel ist, in welchem wenigstens ein Abschnitt der zwischen der externen Vorrichtung (**12**(m)) und der Firewall (**30**) übertragenen Nachrichten verschlüsselt ist.

Hierzu 1 Seite(n) Zeichnungen

Nummer: DE 199 24 575 A1
Int. Cl.^{6}: H 04 L 29/06
Offenlegungstag: 2. Dezember 1999

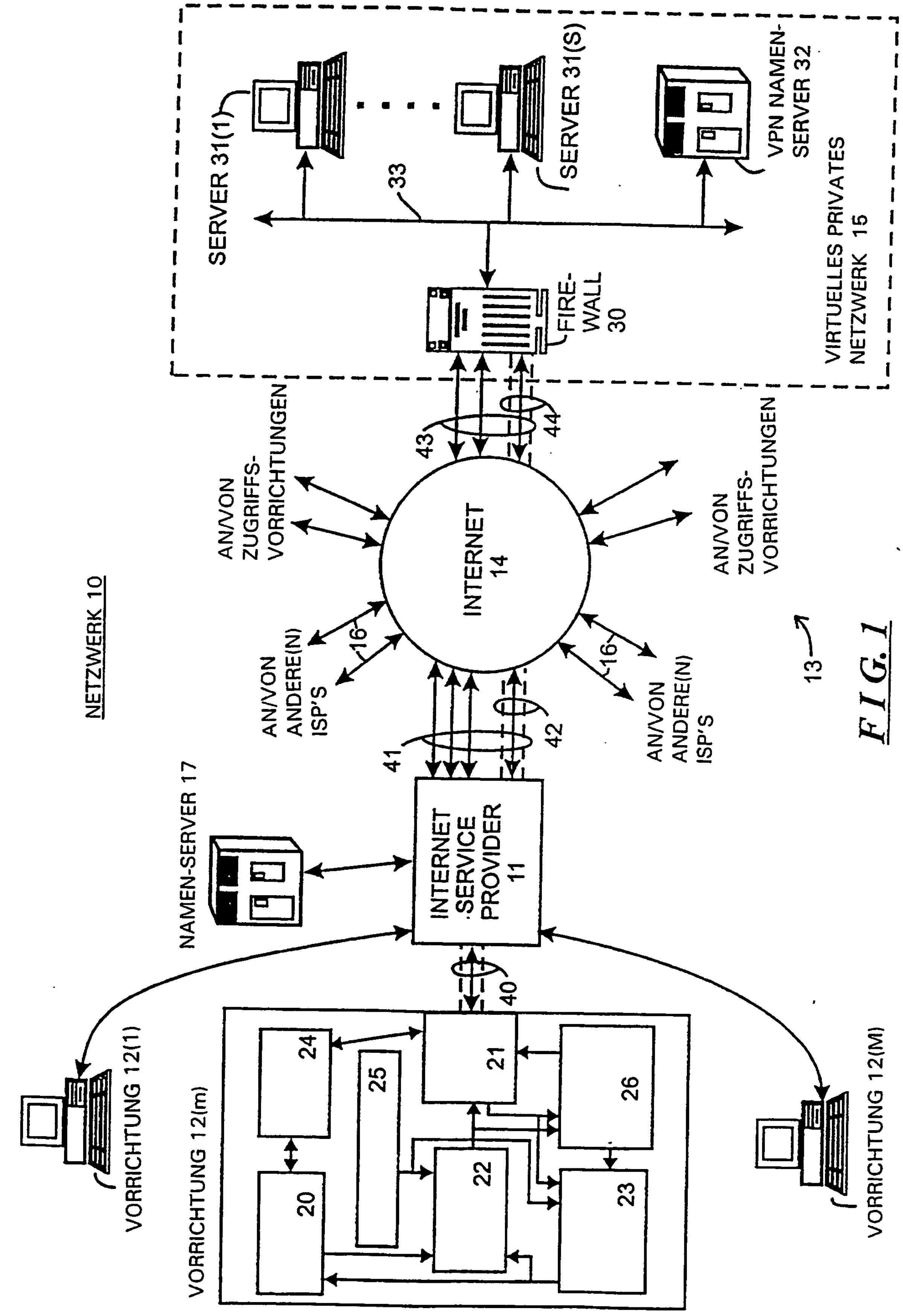


FIG. 1

EAST Search History

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L1	1	("6725268").PN.	US-PGPUB; USPAT	OR	OFF	2006/06/17 18:45
L2	0	("10714849").PN.	US-PGPUB; USPAT	OR	OFF	2006/06/17 18:45
L3	1	("6502135").PN.	US-PGPUB; USPAT	OR	OFF	2006/06/17 18:59
L4	10	secure near4 domain near5 name near5 service	US-PGPUB; USPAT	OR	ON	2006/06/17 19:02
L5	698	secure same domain same name	US-PGPUB; USPAT	OR	ON	2006/06/17 19:03
L6	93	l5 and portal	US-PGPUB; USPAT	OR	ON	2006/06/17 19:04
L7	63	l6 and (authenticat$ or cryptographic)	US-PGPUB; USPAT	OR	ON	2006/06/17 19:05
L8	63	l7 and (portal or router)	US-PGPUB; USPAT	OR	ON	2006/06/17 19:05
L9	8	l8 and secure near4 network near4 address$	US-PGPUB; USPAT	OR	ON	2006/06/17 19:08
L10	23	l8 and (domain near4 name).ti,ab, clm.	US-PGPUB; USPAT	OR	ON	2006/06/17 19:12
L11	2196	DNS.ti,ab,clm.	US-PGPUB; USPAT	OR	ON	2006/06/17 19:13
L12	14	l11 and portal.ti,ab,clm.	US-PGPUB; USPAT	OR	ON	2006/06/17 19:15
L13	84	l11 and secure same network same address	US-PGPUB; USPAT	OR	ON	2006/06/17 19:16
L14	70	l13 and (portal or router)	US-PGPUB; USPAT	OR	ON	2006/06/17 19:17
L15	69	l14 and internet	US-PGPUB; USPAT	OR	ON	2006/06/17 19:17
L16	51	l15 and (authenticat$ or cryptographic$)	US-PGPUB; USPAT	OR	ON	2006/06/17 19:18

EAST Search History

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L1	185	scom.ti,ab,clm. or sorg.ti,ab,clm.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/09/02 11:11
L2	142	l1 and @ad<="19990607"	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/09/02 11:16
L3	0	l2 and sgov.ti,ab,clm. and snet.ti, ab,clm.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/09/02 11:12
L4	0	l2 and secure.ti,ab,clm. and network.ti,ab,clm.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/09/02 11:13
L5	1	l2 and secure and network	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/09/02 11:13
L6	14	l2 and (secure domain name service)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/09/02 11:18
L7	990019	(secure domain name service).ti,ab, clm.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/09/02 11:18
L8	89178	l7 and server.ti,ab,clm.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/09/02 11:19

EAST Search History

L9	57641	l8 and (secure network connection). ti,ab,clm.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/09/02 11:19
L10	7778	l9 and authenticat$.ti,ab,clm.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/09/02 11:20
L11	1809	l10 and (edge router)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/09/02 11:20
L12	1809	l11 and @ad<="199900607"	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/09/02 11:21
L13	380	l12 and cryptographic	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/09/02 11:21
L14	380	l13 and server.ti,ab,clm.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/09/02 11:22
L15	380	l14 and (virtual private network)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/09/02 11:27
L16	0	l15 and (".scom" or ".sorg" or ". snet")	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/09/02 11:23

EAST Search History

L17	39	l15 and @ad<="19990607"	US-PGPUB;
			USPAT;
			USOCR;
			EPO; JPO;
			DERWENT;
			IBM_TDB
				OR	ON	2006/09/02 11:27

INFORMATION DISCLOSURE STATEMENT BY APPLICANT
(Not for submission under 37 CFR 1.99)

Application Number	10714849
Filing Date	2003-11-18
First Named Inventor	Victor Larson
Art Unit	2143
Examiner Name	TBD
Attorney Docket Number	000479.00111

U.S.PATENTS

Examiner Initial*	Cite No	Patent Number	Kind Code1	Issue Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1	5870610	A	1999-02-09	Beyda et al.	

If you wish to add additional U.S. Patent citation information please click the Add button.

U.S.PATENT APPLICATION PUBLICATIONS

Examiner Initial*	Cite No	Publication Number	Kind Code1	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1					

If you wish to add additional U.S. Published Application citation information please click the Add button.

FOREIGN PATENT DOCUMENTS

Examiner Initial*	Cite No	Foreign Document Number3	Country Code2 i	Kind Code4	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear	T5
	1	0 838 930	EP	A	1998-04-29	Compaq Computer Corp.		□
	2	0 814 589	EP	A	1997-12-29	AT&T Corp.		□
	3	2 334 181	GB	A	1999-08-11	NEC Technologies; Globalmart Ltd.		□

INFORMATION DISCLOSURE STATEMENT BY APPLICANT **(Not for submission under 37 CFR 1.99)**	Application Number	10714849
	Filing Date	2003-11-18
	First Named Inventor	Victor Larson
	Art Unit	2143
	Examiner Name	TBD
	Attorney Docket Number	000479.00111

	4	9827783	WO	A	1998-06-25	Northern Telecom Limited; Antonio, G; Hui, Margare		□
	5	2 317 792	GB	A	1998-04-01	Secure Computing Corporation		□
	6							□

If you wish to add additional Foreign Patent Document citation information please click the Add button

NON-PATENT LITERATURE DOCUMENTS

Examiner Initials*	Cite No	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc), date, pages(s), volume-issue number(s), publisher, city and/or country where published.	T5
	1	Laurie Wells (LANCASTERBIBELMAIL MSN COM); "Subject: Security Icon" USENET NEWSGROUP, 19 October 1998, XP002200606	□
	2	Davila J et al, "Implementation of Virtual Private Networks at the Transport Layer", Information Security, Second International Workshop, ISW '99. Proceedings (Lecture Springer-Verlag Berlin, Germany, [Online] 1999, pages 85-102, XP002399276, ISBN 3-540-66695-B, retrieved from the Internet: URL: http://www.springerlink.com/ content/4uac0tb0heccma89/fulltext.pdf> (Abstract)	□
	3		□
	4		□

If you wish to add additional non-patent literature document citation information please click the Add button

EXAMINER SIGNATURE

Examiner Signature		Date Considered	

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through a citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT **(Not for submission under 37 CFR 1.99)**	Application Number	10714849
	Filing Date	2003-11-18
	First Named Inventor	Victor Larson
	Art Unit	2143
	Examiner Name	TBD
	Attorney Docket Number	000479.00111

[1] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [2] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [3] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document. [4] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [5] Applicant is to place a check mark here if English language translation is attached.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT **(Not for submission under 37 CFR 1.99)**	Application Number		10714849
	Filing Date		2003-11-18
	First Named Inventor	Victor Larson	
	Art Unit		2143
	Examiner Name	TBD	
	Attorney Docket Number		000479.00111

CERTIFICATION STATEMENT

Please see 37 CFR 1.97 and 1.98 to make the appropriate selection(s):

☒ That each item of information contained in the information disclosure statement was first cited in any communication from a foreign patent office in a counterpart foreign application not more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(1).

OR

□ That no item of information contained in the information disclosure statement was cited in a communication from a foreign patent office in a counterpart foreign application, and, to the knowledge of the person signing the certification after making reasonable inquiry, no item of information contained in the information disclosure statement was known to any individual designated in 37 CFR 1.56(c) more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(2).

□ See attached certification statement.

□ Fee set forth in 37 CFR 1.17 (p) has been submitted herewith.

□ None

SIGNATURE

A signature of the applicant or representative is required in accordance with CFR 1.33, 10.18. Please see CFR 1.4(d) for the form of the signature.

Signature	/Steve Chang/	Date (YYYY-MM-DD)	2006-11-09
Name/Print	Steve S. Chang	Registration Number	42,402

This collection of information is required by 37 CFR 1.97 and 1.98. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 1 hour to complete, including gathering, preparing and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

Privacy Act Statement

The Privacy Act of 1974 (P.L. 93-579) requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C. 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether the Freedom of Information Act requires disclosure of these record s.

2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.

3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.

4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).

5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.

6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).

7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (i.e., GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.

8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspections or an issued patent.

9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

WO9827783

Publication Title:

VIRTUAL PRIVATE NETWORK SERVICE PROVIDER FOR ASYNCHRONOUS TRANSFER MODE NETWORK

Abstract:

Abstract of WO9827783

A virtual private network service provider is used to transfer data over a data network to a final destination, with third-party billing. The method comprises the steps of: prompting the user at a data terminal to select a destination, password, and call type; sending a set-up message to the data network; selecting a virtual private network provider through the data network; the virtual pri 104d vate network provider giving an encryption key to the user, and then prompting the user for a password and a user identification; encrypting the password, and sending the user identification and the encrypted password to the virtual private network provider; the virtual private network provider decrypting the encrypted password, and verifying the password; the virtual private network provider providing an authorization code; and the data terminal transferring the data through the data network to the final destination, using the authorization code. Data supplied from the esp@cenet database - Worldwide

PCT

WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 :
H04Q 11/04, H04L 12/22

A1

(11) International Publication Number: WO 98/27783

(43) International Publication Date: 25 June 1998 (25.06.98)

(21) International Application Number: PCT/IB97/01563

(22) International Filing Date: 12 December 1997 (12.12.97)

(30) Priority Data:
08/769,649 19 December 1996 (19.12.96) US

(71) Applicant (*for all designated States except US*): NORTHERN TELECOM LIMITED [CA/CA]; World Trade Center of Montreal, 8th floor, 380 St. Antoine Street West, Montreal, Quebec H2Y 3Y4 (CA).

(72) Inventors; and
(75) Inventors/Applicants (*for US only*): TELLO, Antonio, G. [US/US]; 114 Fountain Hills Drive, Garland, TX 75044 (US). HUI, Margaret [US/US]; 9920 Forest Lane #208, Dallas, TX 75243 (US). HOLMES, Kim [US/US]; 5409 Scenic Drive, Rowlett, TX 75088 (US).

(74) Agents: MCCOMBS, David et al.; Haynes and Boone, L.L.P., Suite 3100, 901 Main Street, Dallas, TX 75202-3789 (US).

(81) Designated States: AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GE, GH, HU, IL, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, UA, UG, US, UZ, VN, YU, ZW, ARIPO patent (GH, GM, KE, LS, MW, SD, SZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, ML, MR, NE, SN, TD, TG).

Published
With international search report.

(54) Title: VIRTUAL PRIVATE NETWORK SERVICE PROVIDER FOR ASYNCHRONOUS TRANSFER MODE NETWORK

(57) Abstract

A virtual private network service provider is used to transfer data over a data network to a final destination, with third-party billing. The method comprises the steps of: prompting the user at a data terminal to select a destination, password, and call type; sending a set-up message to the data network; selecting a virtual private network provider through the data network; the virtual private network provider giving an encryption key to the user, and then prompting the user for a password and a user identification; encrypting the password, and sending the user identification and the encrypted password to the virtual private network provider; the virtual private network provider decrypting the encrypted password, and verifying the password; the virtual private network provider providing an authorization code; and the data terminal transferring the data through the data network to the final destination, using the authorization code.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania
AM	Armenia
AT	Austria
AU	Australia
AZ	Azerbaijan
BA	Bosnia and Herzegovina
BB	Barbados
BE	Belgium
BF	Burkina Faso
BG	Bulgaria
BJ	Benin
BR	Brazil
BY	Belarus
CA	Canada
CF	Central African Republic
CG	Congo
CH	Switzerland
CI	Côte d'Ivoire
CM	Cameroon
CN	China
CU	Cuba
CZ	Czech Republic
DE	Germany
DK	Denmark
EE	Estonia
ES	Spain
FI	Finland
FR	France
GA	Gabon
GB	United Kingdom
GE	Georgia
GH	Ghana
GN	Guinea
GR	Greece
HU	Hungary
IE	Ireland
IL	Israel
IS	Iceland
IT	Italy
JP	Japan
KE	Kenya
KG	Kyrgyzstan
KP	Democratic People's Republic of Korea
KR	Republic of Korea
KZ	Kazakstan
LC	Saint Lucia
LI	Liechtenstein
LK	Sri Lanka
LR	Liberia
LS	Lesotho
LT	Lithuania
LU	Luxembourg
LV	Latvia
MC	Monaco
MD	Republic of Moldova
MG	Madagascar
MK	The former Yugoslav Republic of Macedonia
ML	Mali
MN	Mongolia
MR	Mauritania
MW	Malawi
MX	Mexico
NE	Niger
NL	Netherlands
NO	Norway
NZ	New Zealand
PL	Poland
PT	Portugal
RO	Romania
RU	Russian Federation
SD	Sudan
SE	Sweden
SG	Singapore
SI	Slovenia
SK	Slovakia
SN	Senegal
SZ	Swaziland
TD	Chad
TG	Togo
TJ	Tajikistan
TM	Turkmenistan
TR	Turkey
TT	Trinidad and Tobago
UA	Ukraine
UG	Uganda
US	United States of America
UZ	Uzbekistan
VN	Viet Nam
YU	Yugoslavia
ZW	Zimbabwe

VIRTUAL PRIVATE NETWORK SERVICE PROVIDER FOR ASYNCHRONOUS TRANSFER MODE NETWORK

Technical Field

The invention relates generally to asynchronous transfer mode ("ATM") networks and virtual private networks ("VPN"), such as those offered by MCI and Sprint, and, more particularly, to a method of using a VPN to transfer data over a data network, with third-party billing.

Background of the Invention

Telephone service providers offer third-party billing. For example, local and long distance telephone companies offer calling cards for third party billing.

VPNs exist to provide the sense of a private network among a company's locations. The lines/trunks of a VPN are actually shared among several companies, to reduce costs, yet to each company the VPN appears to be that company's own private network. However, a user at a remote data terminal, such as a portable computer in a hotel room, can not immediately charge his company for the access time to a data net, such as the Internet. Instead, his access time is charged to his hotel room, and so he must pay the inflated rates that hotels charge for phone service.

What is needed is a VPN service provider that offers remote access for users belonging to a VPN, user authorizations to prevent delinquent access into the VPN, and convenient third-party billing.

Summary of the Invention

The present invention, accordingly, provides a system and method for using a VPN service provider to transfer data over a data network to a final destination, with third-party billing. The method comprises the steps of: prompting the user at a data terminal to select a destination, password, and call type; selecting a VPN through the data network; giving an encryption key to the user, and then prompting the user for a password and a user identification; verifying the password, and providing an authorization code to the user; and allowing the user to transfer the data through the data network to the final destination, using the authorization code.

In another feature of the invention, the method further comprises negotiating for more bandwidth for the user, and including within the authorization code a grant of additional bandwidth.

In another feature of the invention, the method further comprises encrypting the user's password, and sending the user identification and the encrypted password to the VPN service provider.

In another feature of the invention, the method further comprises a step of sending a set-up message to the data network.

In another feature of the invention, the method further comprises a step of the VPN service provider decrypting the encrypted password.

A technical advantage achieved with the invention is that it shifts or defers costs from an end user to a bulk purchaser of data network services. Another technical advantage achieved with the invention is that it permits end users mobility while attaining a virtual appearance on a corporate intranet.

Brief Description of the Drawings

Fig. 1 is a system block diagram of a VPN service provider of the present invention.

Fig. 2 is a flow chart depicting the method of the present invention, as implemented by application software on a user terminal.

Fig. 3 is the initial screen display of the user interface of the application software.

Figs. 4A and 4B are call flow diagrams, illustrating the preferred sequence of steps of the method of the present invention.

Figs. 5A, 5B, 5C, 5D, 5E, and 5F comprise a flow chart depicting the method of the present invention, as implemented by switching control point software.

Description of the Preferred Embodiment

In Fig. 1, the VPN service provider system of the present invention is designated generally by a reference numeral 10. The VPN service provider system 10 includes a VPN 12. The VPN 12 may be a corporate, government, association, or other organization's telephone/data line `network. The VPN service provider system 10 also includes access lines 13 from the VPN 12 to a data network 14, such as the Internet, or an ATM network. The VPN service provider system 10 also includes access lines 16 from the data network 14 to a long distance phone company 18, such as AT&T, MCI, or Sprint. The VPN service provider system 10 also includes access lines 20 from the data network 14 to a called party 22, such as, for example, American Express reservations service. The VPN service provider system 10 also includes access lines 24 from the data network 14 to a remote user terminal 26, such as a portable computer in a hotel room. The user terminal 26 includes user application software 28, which provides the interface for the user to enter the number to be called, the user identification number, and the user's authorization code. The VPN service provider system 10 also includes VPN service provider software 30, located in a switching control point (SCP) device 32, which, in the preferred embodiment may be physically located anywhere. The SCP 32 connects to the data network 14 via access lines 36. One possible physical location for the SCP 32 is on the premises of a local phone company central switch building 34. However, even when located within the building 34, the SCP 32 connects to the local phone company switches via the data network 14. The local phone company switches connect to the data network 14 via access lines 38.

In an alternate embodiment, the VPN service provider software 30 and the SCP device 32 may be located on the premises of an independent provider of local phone service, or on the premises of an independent VPN service provider.

Referring now to Fig. 2, the application software 28 begins the data transfer process in step 50. In step 52, the user is presented with a screen display.

Referring now to Fig. 3, a screen display 100 displays the following information requests: whether the call is a direct call 102 or a VPN call 104, the number the user desires to call 106, the VPN user ID 108, and the user password 110. The user is also presented with the option to make the call 112, or to quit 114.

Referring back to Fig. 2, in step 54 the user terminal sends to the SCP 32 the information captured through the graphical user interface ("GUI") in step 52 within a user network interface ("UNI") setup message. In step 56 the user terminal 26 waits for a connect message from the SCP 32. In step 58 the user terminal 26 determines if a connection was made. If no connection was made, then in step 60 the user application software 28 displays an error message to the user, and returns to step 50 to begin again the data transfer process.

If a connection was made, then in step 62 the user terminal 26 sends the VPN user ID to the SCP 32. In step 64 the user terminal 26 waits for an encryption key from the SCP 32. In step 66, having received the encryption key from the SCP 32, the user application software 28 encrypts the user's password, and sends it to the SCP 32. In step 68 the user terminal 26 waits for authentication of the user. In step 70 the user application software 28 determines if the SCP 32 authorizes the user to make the call.

If the user is not authorized, then in step 72 the user terminal 26 displays an error message, terminates the connection, blanks the screen display 100, and returns to step 50 to begin again the data transfer process. If the user is authorized, then in step 74 the VPN service provider software 30 sets up the billing, and authorizes it. In step 76 the user terminal 26 sends a "release", meaning to terminate or disconnect the connection, to the SCP 32. In step 78 the user terminal 26 sends a setup message to the number listed by the user as the "number to call", that is, to the final destination. In step 80 the user terminal 26 waits for a connection. In step 82 the user terminal 26 determines if a connection was made.

If a connection to the final destination was not made, then the user application software 28 returns to step 72, in which step the user terminal 26 displays an error message, terminates the connection, blanks the screen display 100, and returns to step 50 to begin again the data transfer process. If a connection to the final destination was made, then in step 84 the user terminal 26 exchanges user data, services, and/or value added or user specific applications with the computer at the address, that is, the telephone number, of the final destination. In step 86 the user selects the option presented to him to release, or terminate, the call. In step 88 the user terminal 26 sends a release message to the final destination. In step 90 the data network 14 sends billing information to the SCP 32. In step 92 the application software 28 ends the data transfer process.

Fig. 4A and Fig. 4B are call flow diagrams, showing the sequence of messages in the method of the preferred embodiment. These diagrams present the same method as the flow chart of Fig. 2. The horizontal arrows represent the messages sent and received. The vertical lines represent the various devices involved in sending and receiving the messages. For example, the top left arrow in Fig. 4A represents a message sent from the user terminal 26, labeled "Macintosh" in Fig.4A, to an interface with a public network. The user terminal 26 can be any brand of a work station computer, a desktop computer, a laptop computer, or even a notebook computer. The interface could be any interface, but in the example of Fig. 4A and Fig. 4B, the interface is imagined to be at a hotel, where a business traveler is using the method of the present invention. Thus, the interface is labeled "Hotel ATM Interface", which is not shown in Fig. 1. The vertical line labeled "Public ATM Network" is the same as the data network 14 in Fig. 1. The vertical line labeled "Moe's VPN Service" represents the VPN service provider software 30 within the SCP 32. The vertical line labeled "Travel ATM Interface" is not shown in Fig. 1, but is located between the called party 22 and the data network 14. The vertical line labeled "Travel Service" is one example of the called party 22 shown in Fig. 1. In the example of Fig. 4A and Fig. 4B, the business traveler is imagined to be using the method of the present invention to contact a travel service to make reservations for his next airline flight. In Figs. 4A and 4B the designation "Ack" represents "acknowledge", and the designation "Cmp" represents "complete".

Referring now to Fig. 5, the VPN service provider software 30 begins the data transfer process in step 300 by waiting for an event. The event it waits for is a setup message on a signaling port of the SCP 32, to be received from the user terminal 26. In step 302, having monitored the signaling ports, and the SCP 32 having received a setup message, the VPN service provider software 30 assigns a call condense block ("CCB") to the setup message, based on a call reference number. The CCB is a software data structure for tracking resources associated with the call. The call reference number is a number, internal to the SCP, for tracking calls. In step 304 the VPN service provider software 30 compiles the connect message. In step 306 the VPN service provider software 30 sends a connect message to the calling address, that is, the hotel room from which the user is calling. In step 308 the VPN service provider software 30 condenses, that is, it remains in a wait state for that call.

Referring now to Fig. 5B, in step 310 the VPN service provider software 30 waits for an event by monitoring the signaling ports of the SCP 32. After the SCP 32 receives a connect acknowledge message from the user terminal 26, then in step 312 the VPN service provider software 30 accesses the CCB, based on the call reference number. In step 314 the VPN service provider software 30 condenses.

Referring now to Fig. 5C, in step 316 the VPN service provider software 30 waits for dialog on a data port of the SCP 32. After the SCP 32 receives a

VPN ID on a data port, the VPN service provider software 30 verifies the VPN ID in step 318. In step 320 the VPN service provider software 30 determines if the VPN ID is valid. If the VPN ID is not valid, then in step 322 the SCP 32 sends a reject message over an assigned switch virtual circuit ("SVC"). The SVC is a channel over the data network 14. In step 324 the VPN service provider software 30 waits for dialog. In step 326, because the VPN ID is valid, the VPN service provider software 30 assigns an encryption key to the user terminal 26, in step 328 sends the encryption key over the assigned SVC to the user terminal 26, and in step 330 waits for dialog.

Referring now to Fig. 5D, in step 332 the VPN service provider software 30 waits for dialog. When the SCP 32 receives the encrypted password from the user terminal 26 at a data port, then in step 334 the VPN service provider software 30 verifies the password, and determines in step 336 if the password is valid. If the password is not valid, then in step 338 the SCP 32 sends a reject message over the assigned SVC to the user terminal, and in step 340 waits for dialog. If the password is valid, then in step 342 the VPN service provider software 30 assigns an authorization token to the user terminal 26, in step 344 sends the token over an assigned SVC to the user terminal 26, and in step 346 waits for dialog.

Referring now to Fig. 5E, in step 348 the VPN service provider software 30 waits for an event. When the VPN service provider software 30 senses that the SCP 32 has received on a signaling port a release message from the user terminal 26, then in step 350 the VPN service provider software 30 accesses the CCB, based on the call reference number of the user terminal 26, in step 352 compiles a release complete message, in step 354 sends a release complete message to the user terminal 26, and in step 356 condenses.

Referring now to Fig. 5F, in step 358 the VPN service provider software 30 waits for an event. When the VPN service provider software 30 senses that the SCP 32 has received on a signaling port a third-party billing setup message from the user terminal 26, then in step 360 the VPN service provider software 30 verifies the token just received from the user terminal 26, to determine, in step 362, if it is the same token that the VPN service provider software 30 sent to the user terminal 26 in step 344. If the token is not valid, then in step 364 the SCP 32 sends a release message to the terminal 26, and in step 366 condenses. If the token is valid, then in step 368 the SCP 32 sends a modified third-party billing setup message to the data network 14, and in step 370 condenses.

Although an illustrative embodiment of the invention has been shown and described, other modifications, changes, and substitutions are intended in the foregoing disclosure. Accordingly, it is appropriate that the appended claims be construed broadly and in a manner consistent with the scope of the invention.

WHAT IS CLAIMED IS:

1. A computerized method of a virtual private network service provider with third party billing, using a virtual private network to transfer data over a data network to a final destination, the method comprising the steps of:

   a. prompting the user at a data terminal to select a destination, password, and call type;

   b. selecting a virtual private network through the data network;

   c. giving an encryption key to the user, and then prompting the user for a password and a user identification;

   d. verifying the password, and providing an authorization code to the user; and

   e. allowing the user to transfer the data through the data network to the final destination, using the authorization code.

2. The method of claim 1, wherein step (d) further comprises negotiating for more bandwidth for the user, and including within the authorization code a grant of additional bandwidth.

3. The method of claim 2, wherein step (c) further comprises encrypting the user's password, and sending the user identification and the encrypted password to the virtual private network service provider.

4. The method of claim 3, further comprising, after step (a), the step of sending a set-up message to the data network.

5. The method of claim 4, further comprising, after step (c), the step of the virtual private network service provider decrypting the encrypted password.

6. An apparatus for providing a datalink connection from a user terminal to a data network and to a virtual private network, with third party billing, comprising:

   a. an interface between the user terminal and the data network;

b. a switching control point device connected to the data network, the switching control point device connected to a computer; and

c. a computer-readable medium encoded with a method of using the virtual private network and the data network, with third party billing, the computer-readable medium accessible by the computer.

7. The apparatus of claim 6, wherein the method comprises negotiating for more bandwidth for the user, and including within an authorization code a grant of additional bandwidth.

8. The apparatus of claim 7, wherein the method further comprises encrypting a user's password, and temporarily storing the user identification and the encrypted password.

9. The apparatus of claim 8, wherein the method further comprises sending a set-up message to the data network.

10. The apparatus of claim 9, wherein the method further comprises decrypting the encrypted password.

11. A computer-readable medium encoded with a method of using a virtual private network, with third party billing, the method comprising the steps of:

a. prompting the user at a data terminal to select a destination, password, and call type;

b. selecting a virtual private network through the data network;

c. giving an encryption key to the user, and then prompting the user for a password and a user identification;

d. verifying the password, and providing an authorization code to the user; and

e. allowing the user to transfer the data through the data network to the final destination, using the authorization code.

12. The computer-readable medium of claim 11 wherein step (d) further comprises negotiating for more bandwidth for the user, and including within the authorization code a grant of additional bandwidth.

13. The computer-readable medium of claim 12 wherein step (c) further comprises encrypting the user's password, and sending the user identification and the encrypted password to the virtual private network service provider.

14. The computer-readable medium of claim 13 further comprising, after step (a), the step of sending a set-up message to the data network.

15. The computer-readable medium of claim 14 further comprising, after step (c), the step of the virtual private network service provider decrypting the encrypted password.

16. An apparatus for providing a datalink connection from a user terminal to a data network and to a virtual private network, with third party billing, comprising:

a. means for prompting a user at the data terminal to select a destination, password, and call type;

b. means for selecting the virtual private network through the data network;

c. means for giving an encryption key to the user, and then prompting the user for a password and a user identification;

d. means for verifying the password, and providing an authorization code to the user; and

e. means for allowing the user to transfer data through the data network to a final destination, using the authorization code.

17. The apparatus of claim 16, further comprising means for negotiating for more bandwidth for the user, and including within the authorization code a grant of additional bandwidth.

18. The apparatus of claim 17, further comprising means for encrypting the user's password, and sending the user identification and the encrypted password to the virtual private network service provider.

19. The apparatus of claim 18, further comprising means for sending a set-up message to the data network.

20. The apparatus of claim 19, further comprising means for decrypting the encrypted password.

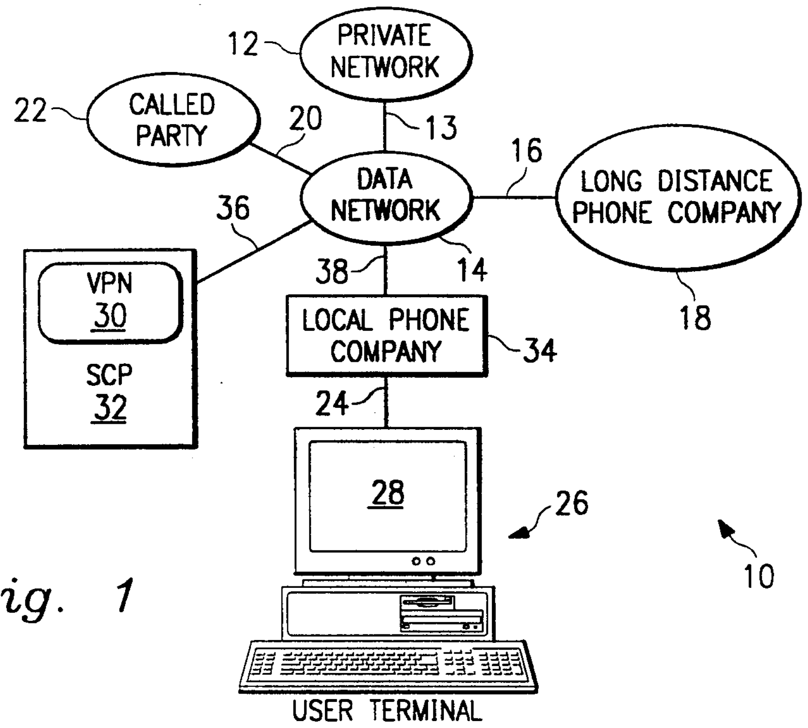

12
PRIVATE NETWORK
22
CALLED PARTY
20
13
16
DATA NETWORK
LONG DISTANCE PHONE COMPANY
36
38
14
18
VPN
30
SCP
32
LOCAL PHONE COMPANY
34
24
28
26
10
USER TERMINAL

Fig. 1

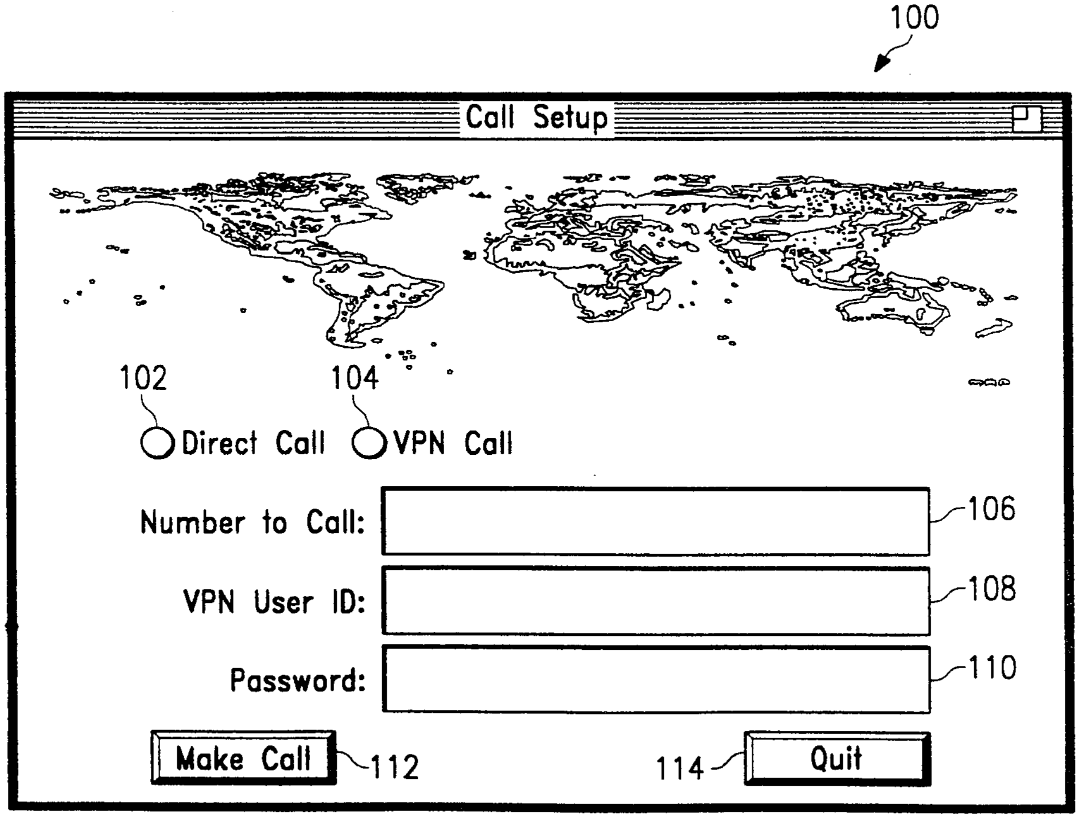

100
Call Setup
102
104
○ Direct Call
○ VPN Call
Number to Call:
106
VPN User ID:
108
Password:
110
Make Call
112
114
Quit

Fig. 3

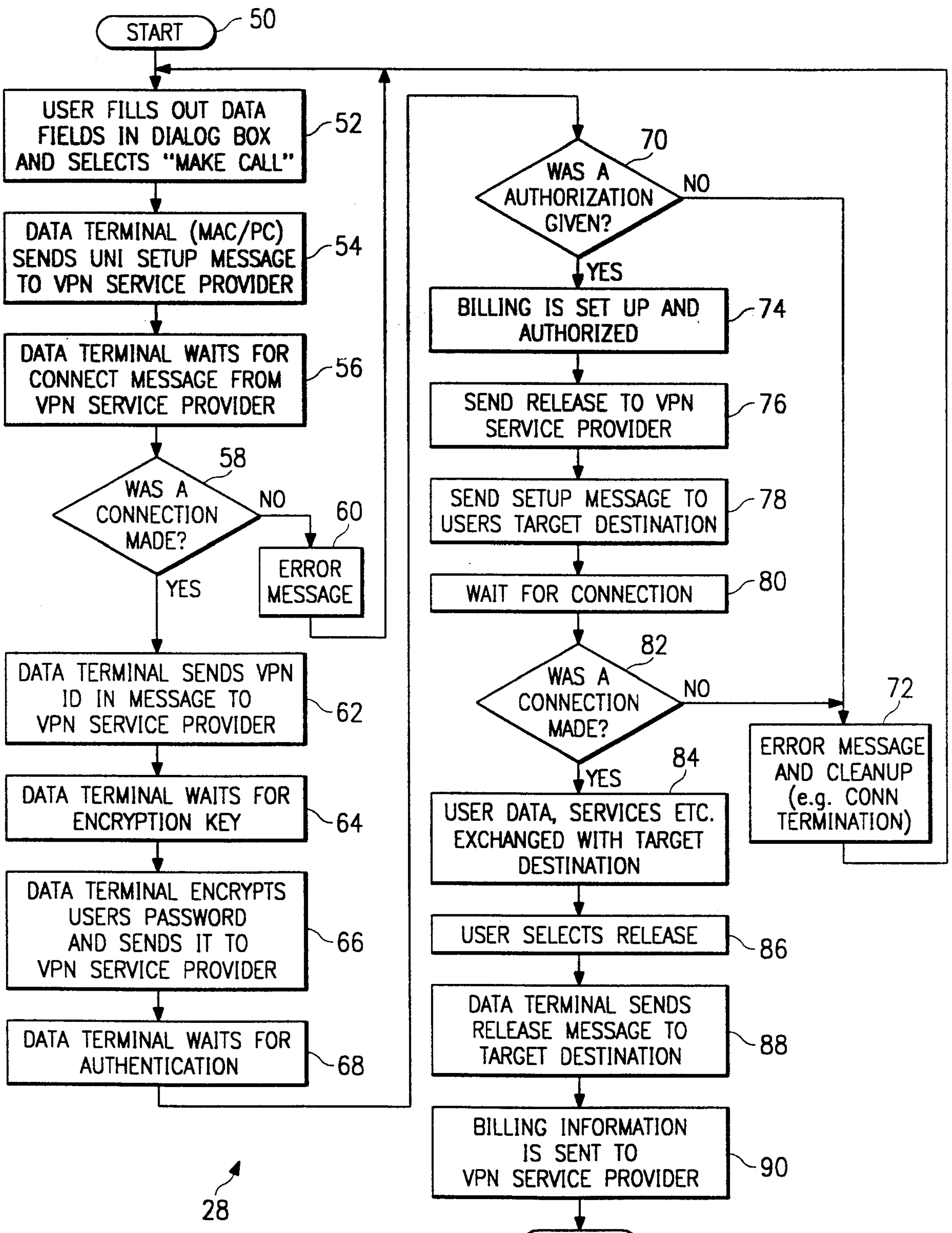

START
50
USER FILLS OUT DATA FIELDS IN DIALOG BOX AND SELECTS "MAKE CALL"
52
DATA TERMINAL (MAC/PC) SENDS UNI SETUP MESSAGE TO VPN SERVICE PROVIDER
54
DATA TERMINAL WAITS FOR CONNECT MESSAGE FROM VPN SERVICE PROVIDER
56
WAS A CONNECTION MADE?
58
NO
60
ERROR MESSAGE
YES
DATA TERMINAL SENDS VPN ID IN MESSAGE TO VPN SERVICE PROVIDER
62
DATA TERMINAL WAITS FOR ENCRYPTION KEY
64
DATA TERMINAL ENCRYPTS USERS PASSWORD AND SENDS IT TO VPN SERVICE PROVIDER
66
DATA TERMINAL WAITS FOR AUTHENTICATION
68
WAS A AUTHORIZATION GIVEN?
70
NO
YES
BILLING IS SET UP AND AUTHORIZED
74
SEND RELEASE TO VPN SERVICE PROVIDER
76
SEND SETUP MESSAGE TO USERS TARGET DESTINATION
78
WAIT FOR CONNECTION
80
WAS A CONNECTION MADE?
82
NO
72
ERROR MESSAGE AND CLEANUP (e.g. CONN TERMINATION)
YES
84
USER DATA, SERVICES ETC. EXCHANGED WITH TARGET DESTINATION
USER SELECTS RELEASE
86
DATA TERMINAL SENDS RELEASE MESSAGE TO TARGET DESTINATION
88
BILLING INFORMATION IS SENT TO VPN SERVICE PROVIDER
90
END
92
28

Fig. 2

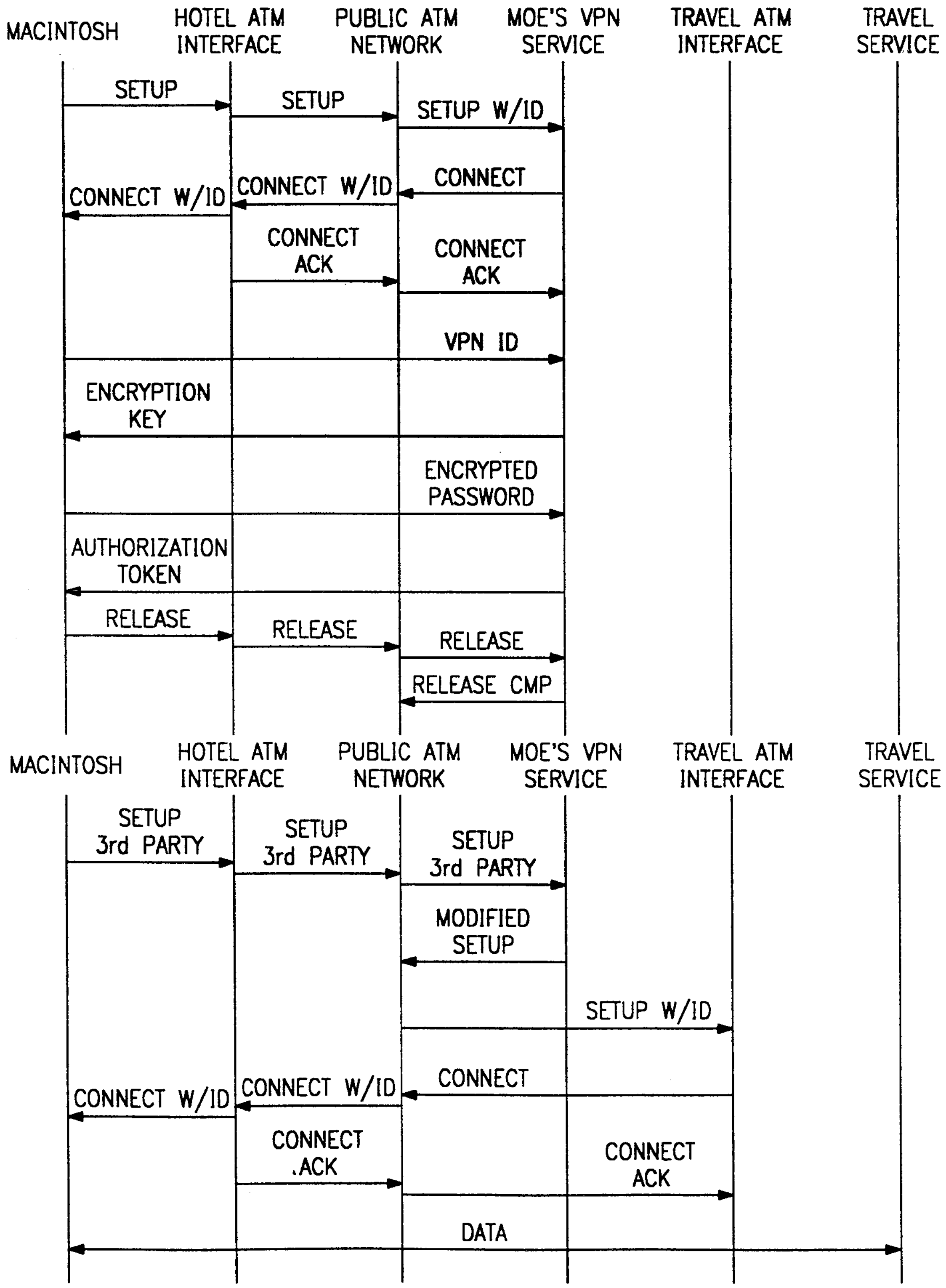

MACINTOSH
HOTEL ATM INTERFACE
PUBLIC ATM NETWORK
MOE'S VPN SERVICE
TRAVEL ATM INTERFACE
TRAVEL SERVICE
SETUP
SETUP
SETUP W/ID
CONNECT
CONNECT W/ID
CONNECT W/ID
CONNECT ACK
CONNECT ACK
VPN ID
ENCRYPTION KEY
ENCRYPTED PASSWORD
AUTHORIZATION TOKEN
RELEASE
RELEASE
RELEASE
RELEASE CMP
MACINTOSH
HOTEL ATM INTERFACE
PUBLIC ATM NETWORK
MOE'S VPN SERVICE
TRAVEL ATM INTERFACE
TRAVEL SERVICE
SETUP 3rd PARTY
SETUP 3rd PARTY
SETUP 3rd PARTY
MODIFIED SETUP
SETUP W/ID
CONNECT
CONNECT W/ID
CONNECT W/ID
CONNECT .ACK
CONNECT ACK
DATA

Fig. 4A

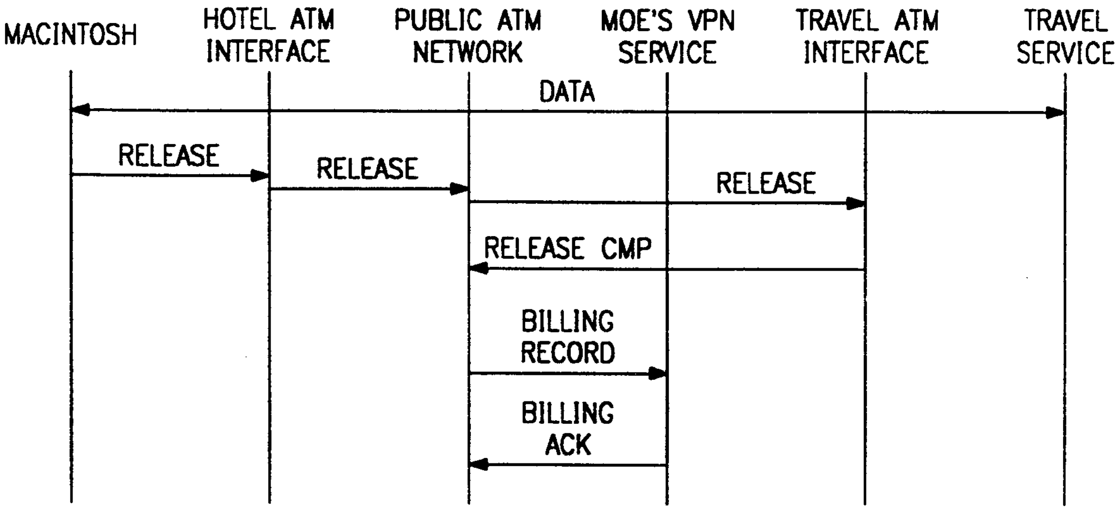


Fig. 4B

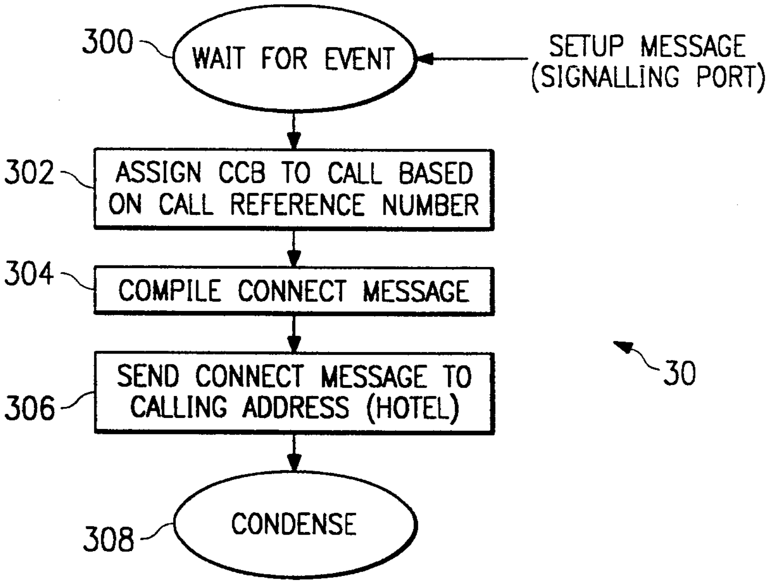

300
WAIT FOR EVENT
SETUP MESSAGE (SIGNALLING PORT)
302
ASSIGN CCB TO CALL BASED ON CALL REFERENCE NUMBER
304
COMPILE CONNECT MESSAGE
306
SEND CONNECT MESSAGE TO CALLING ADDRESS (HOTEL)
308
CONDENSE
30

Fig. 5A

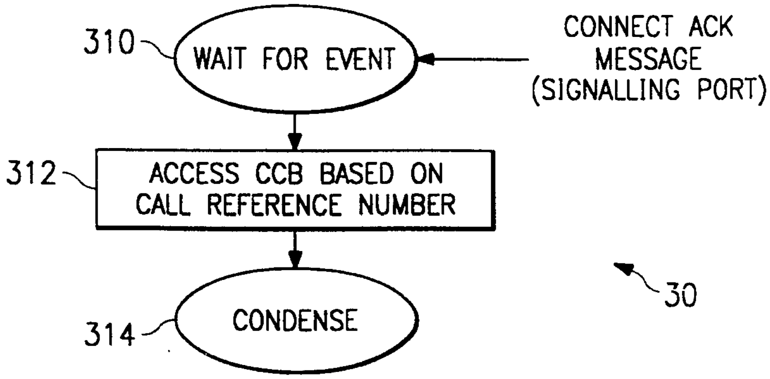

310
WAIT FOR EVENT
CONNECT ACK MESSAGE (SIGNALLING PORT)
312
ACCESS CCB BASED ON CALL REFERENCE NUMBER
314
CONDENSE
30

Fig. 5B

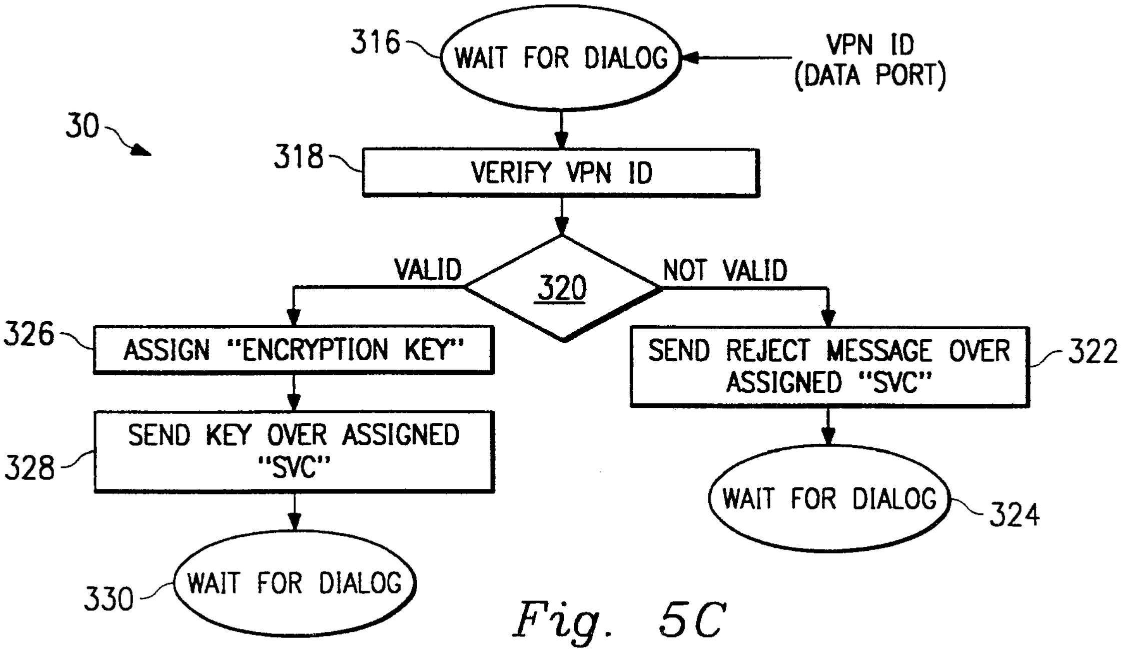

316
WAIT FOR DIALOG
VPN ID
(DATA PORT)
30
318
VERIFY VPN ID
VALID
320
NOT VALID
326
ASSIGN "ENCRYPTION KEY"
SEND REJECT MESSAGE OVER ASSIGNED "SVC"
322
328
SEND KEY OVER ASSIGNED "SVC"
WAIT FOR DIALOG
324
330
WAIT FOR DIALOG

Fig. 5C

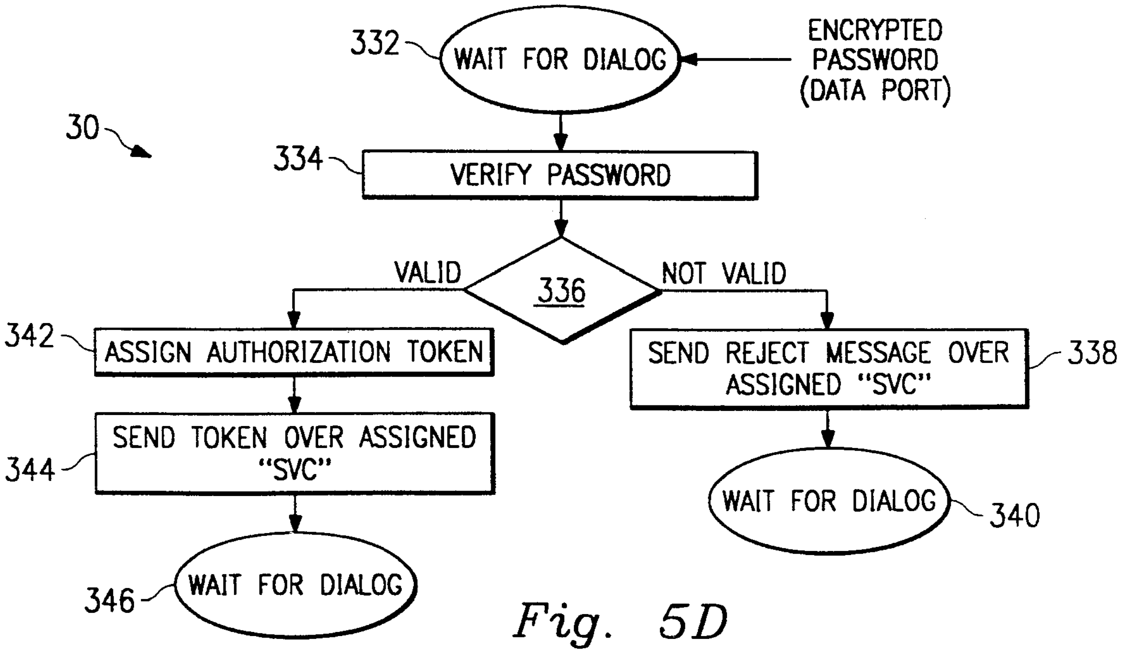

332
WAIT FOR DIALOG
ENCRYPTED
PASSWORD
(DATA PORT)
30
334
VERIFY PASSWORD
VALID
336
NOT VALID
342
ASSIGN AUTHORIZATION TOKEN
SEND REJECT MESSAGE OVER ASSIGNED "SVC"
338
344
SEND TOKEN OVER ASSIGNED "SVC"
WAIT FOR DIALOG
340
346
WAIT FOR DIALOG

Fig. 5D

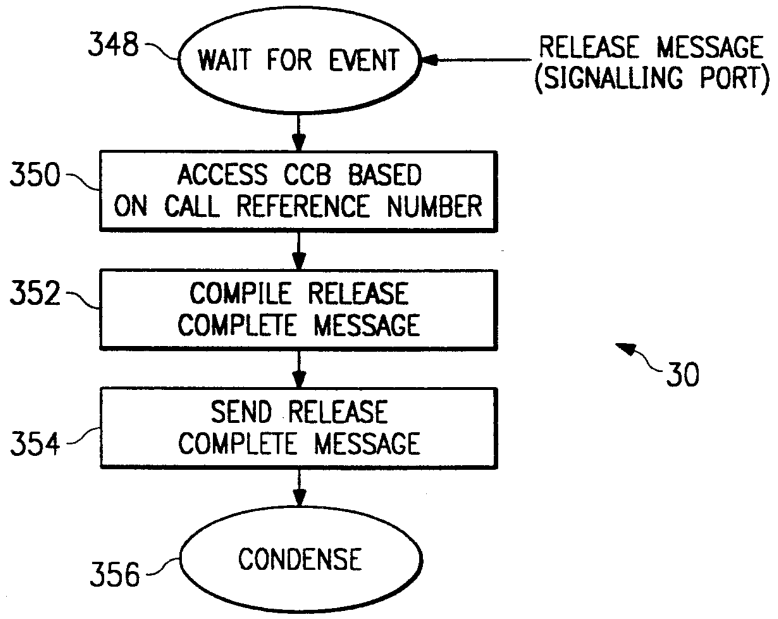

348
WAIT FOR EVENT
RELEASE MESSAGE (SIGNALLING PORT)
350
ACCESS CCB BASED ON CALL REFERENCE NUMBER
352
COMPILE RELEASE COMPLETE MESSAGE
354
SEND RELEASE COMPLETE MESSAGE
356
CONDENSE
30

Fig. 5E

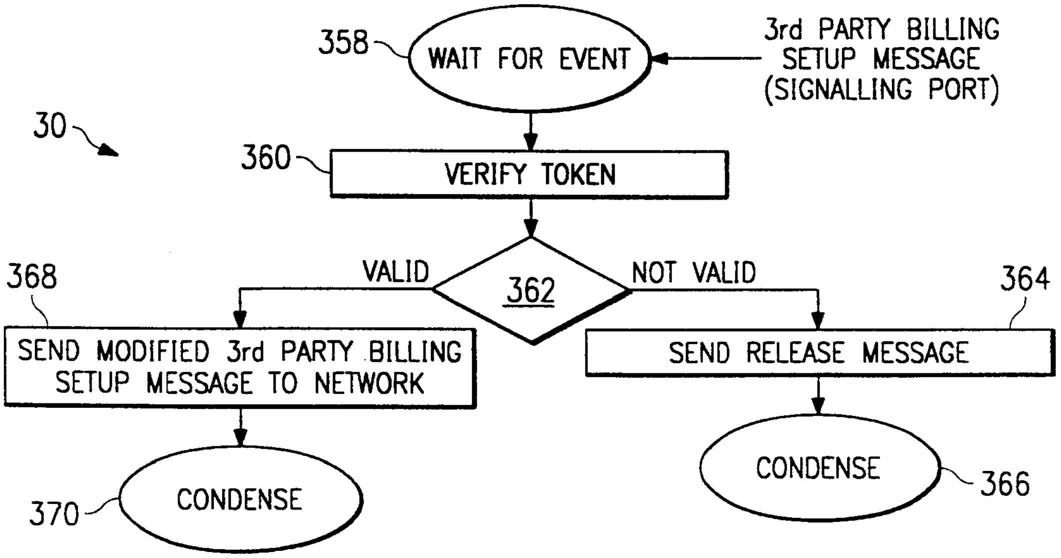

358
WAIT FOR EVENT
3rd PARTY BILLING SETUP MESSAGE (SIGNALLING PORT)
30
360
VERIFY TOKEN
362
VALID
NOT VALID
368
SEND MODIFIED 3rd PARTY BILLING SETUP MESSAGE TO NETWORK
364
SEND RELEASE MESSAGE
CONDENSE
370
CONDENSE
366

Fig. 5F

INTERNATIONAL SEARCH REPORT

Inte ɔnal Application No
PCT/IB 97/01563

A. CLASSIFICATION OF SUBJECT MATTER
IPC 6 H04Q11/04 H04L12/22

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 6 H04Q H04L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	MUN CHOON CHAN ET AL: "AN ARCHITECTURE FOR BROADBAND VIRTUAL NETWORKS UNDER CUSTOMER CONTROL" NOMS '96 IEEE NETWORK OPERATIONS AND MANAGEMENT SYMPOSIUM , vol. 1, 15 April 1996, KYOTO, JP, pages 135-144, XP000641086 see abstract ---	1-20
A	BIC V: "VOICE PERIPHERALS IN THE INTELLIGENT NETWORK" TELECOMMUNICATIONS, vol. 28, no. 6, June 1994, page 29/30, 32, 34 XP000600293 see the whole document --- -/--	1-20

☒ Further documents are listed in the continuation of box C.

☒ Patent family members are listed in annex.

° Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

1

Date of the actual completion of theinternational search 19 March 1998	Date of mailing of the international search report 02/04/1998
Name and mailing address of the ISA European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Authorized officer Staessen, B

INTERNATIONAL SEARCH REPORT

Inte ›nal Application No
PCT/IB 97/01563

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication,where appropriate, of the relevant passages	Relevant to claim No.
A	EP 0 729 256 A (NEDERLAND PTT) 28 August 1996 see abstract figures of pages 136 and 140 ---	1-20
A	CROCETTI P ET AL: "ATM VIRTUAL PRIVATE NETWORKS: ALTERNATIVES AND PERFORMANCES COMPARISONS" SUPERCOMM/ICC '94, 1 May 1994, NEW ORLEANS, US, pages 608-612, XP000438985 see abstract -----	1-20

1

INTERNATIONAL SEARCH REPORT

ınformation on patent family members

Inte ɔnal Application No
PCT/IB 97/01563

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
EP 0729256 A	28-08-96	NL 9500339 A	01-10-96

GB2334181

Publication Title:

Over-the-air re-programming of radio transceivers

Abstract:

Abstract of GB2334181

A method of downloading reprogramming data from a network for installation in a mobile station makes use of a dedicated small bandwidth pilot channel. The mobile station obtains from the base station the radio access parameters of a second channel. The second channel is a large bandwidth (bootstrap) channel suitable for fast transfer of data. The bootstrap channel is logically mapped onto a local transmission mode such as DECT or GSM by the mobile station and re-programming data may be downloaded from the base station via the bootstrap channel.

(12) UK Patent Application (19) GB (11) 2 334 181 (13) A

(43) Date of A Publication 11.08.1999

(21) Application No 9802545.5

(22) Date of Filing 06.02.1998

(71) Applicant(s)
NEC Technologies (UK) Ltd
(Incorporated in the United Kingdom)
Castle Farm Campus, Priorslee, TELFORD, Shropshire, TF2 9SA, United Kingdom

(72) Inventor(s)
Charles Marie Herve Noblet

(74) Agent and/or Address for Service
J W White
NEC Technologies (UK) Ltd, Level 3, The Imperium,
Imperial Way, READING, Berks, RG2 0TD,
United Kingdom

(51) INT CL^6
H04Q 7/32

(52) UK CL (Edition Q)
H4L LDSC

(56) Documents Cited
US 5613204 A US 5109403 A

(58) Field of Search
UK CL (Edition P) H4L LDSC LDSU LECC LECX
INT CL^6 H04Q 7/32 7/38
Online: WPI

(54) Abstract Title

Over-the-air re-programming of radio transceivers

(57) A method of downloading reprogramming data from a network for installation in a mobile station makes use of a dedicated small bandwidth pilot channel. The mobile station obtains from the base station the radio access parameters of a second channel. The second channel is a large bandwidth (bootstrap) channel suitable for fast transfer of data. The bootstrap channel is logically mapped onto a local transmission mode such as DECT or GSM by the mobile station and re-programming data may be downloaded from the base station via the bootstrap channel.

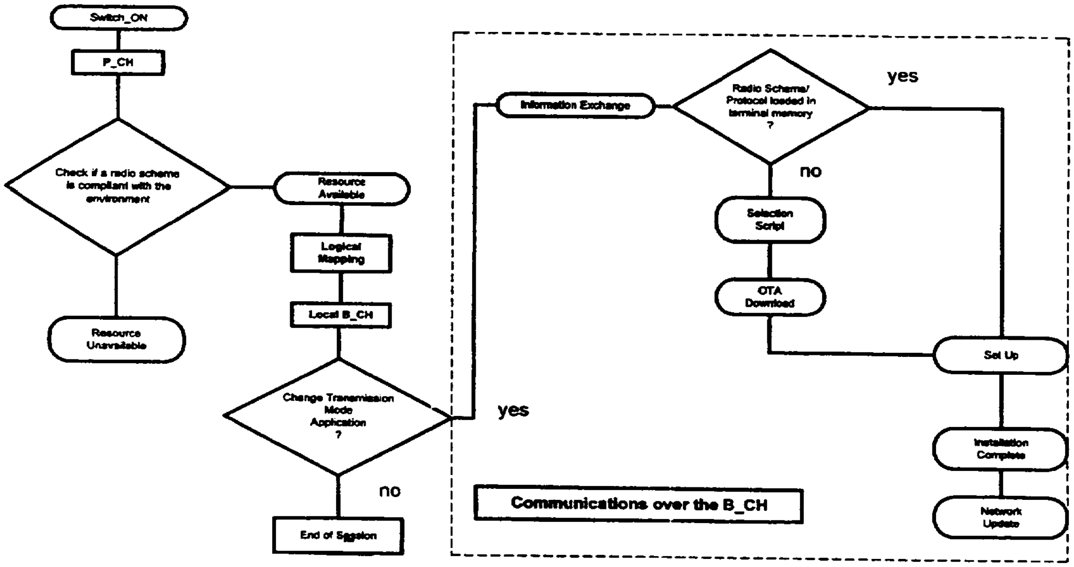


Figure 2

GB 2 334 181 A

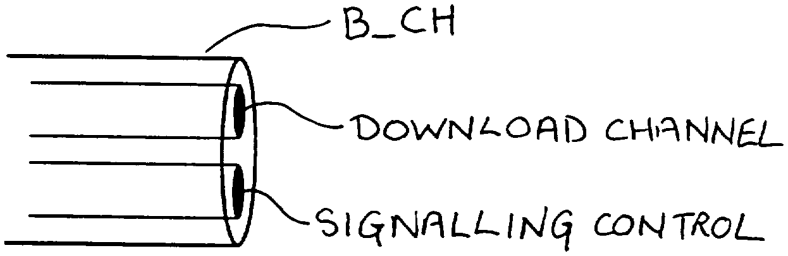


Figure: 1

2/3

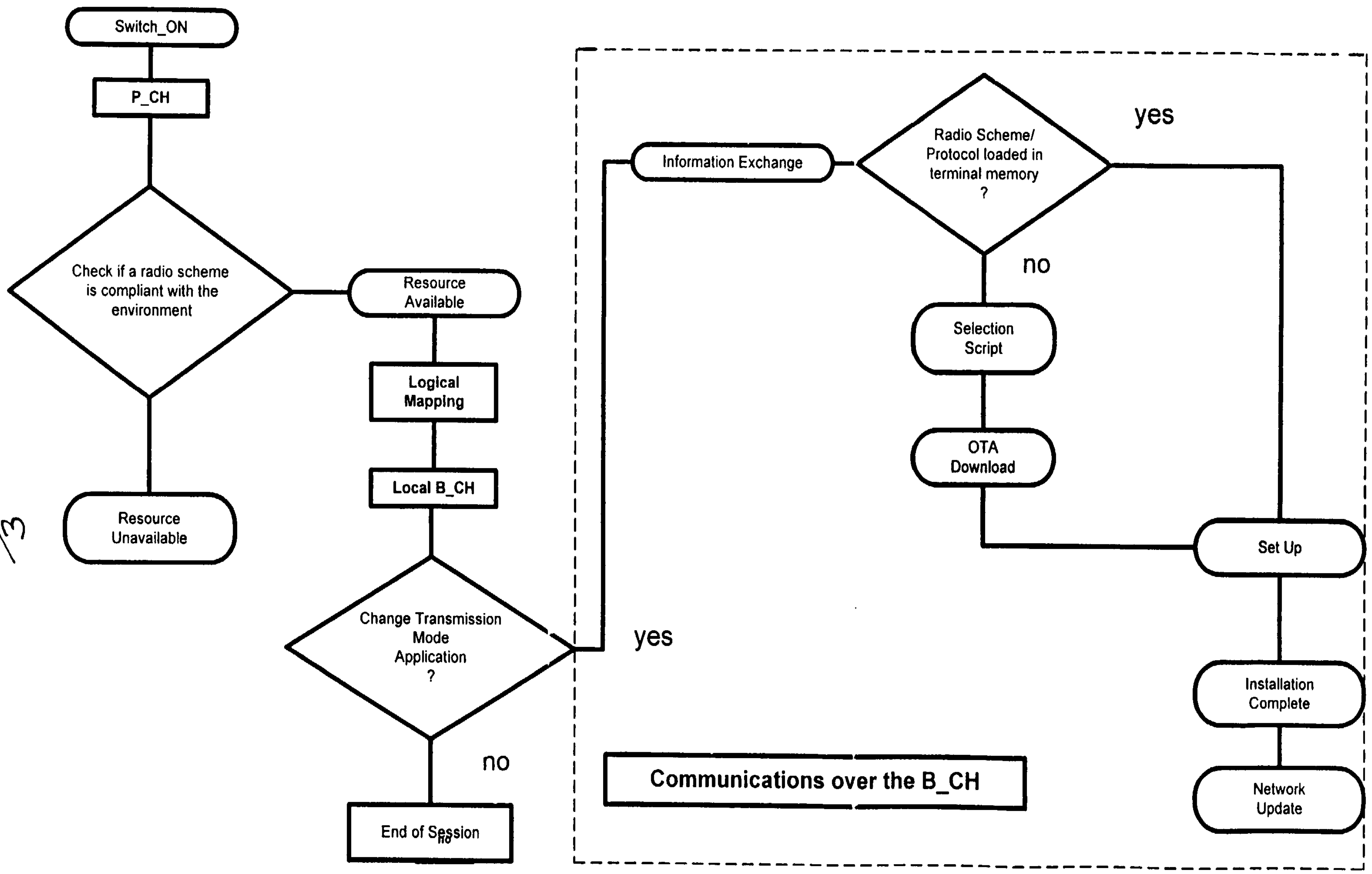

Switch_ON
P_CH
Check if a radio scheme is compliant with the environment
Resource Unavailable
Resource Available
Logical Mapping
Local B_CH
Change Transmission Mode Application ?
no
End of Session
yes
Information Exchange
Radio Scheme/ Protocol loaded in terminal memory ?
yes
no
Selection Script
OTA Download
Set Up
Installation Complete
Network Update
Communications over the B_CH

Figure 2

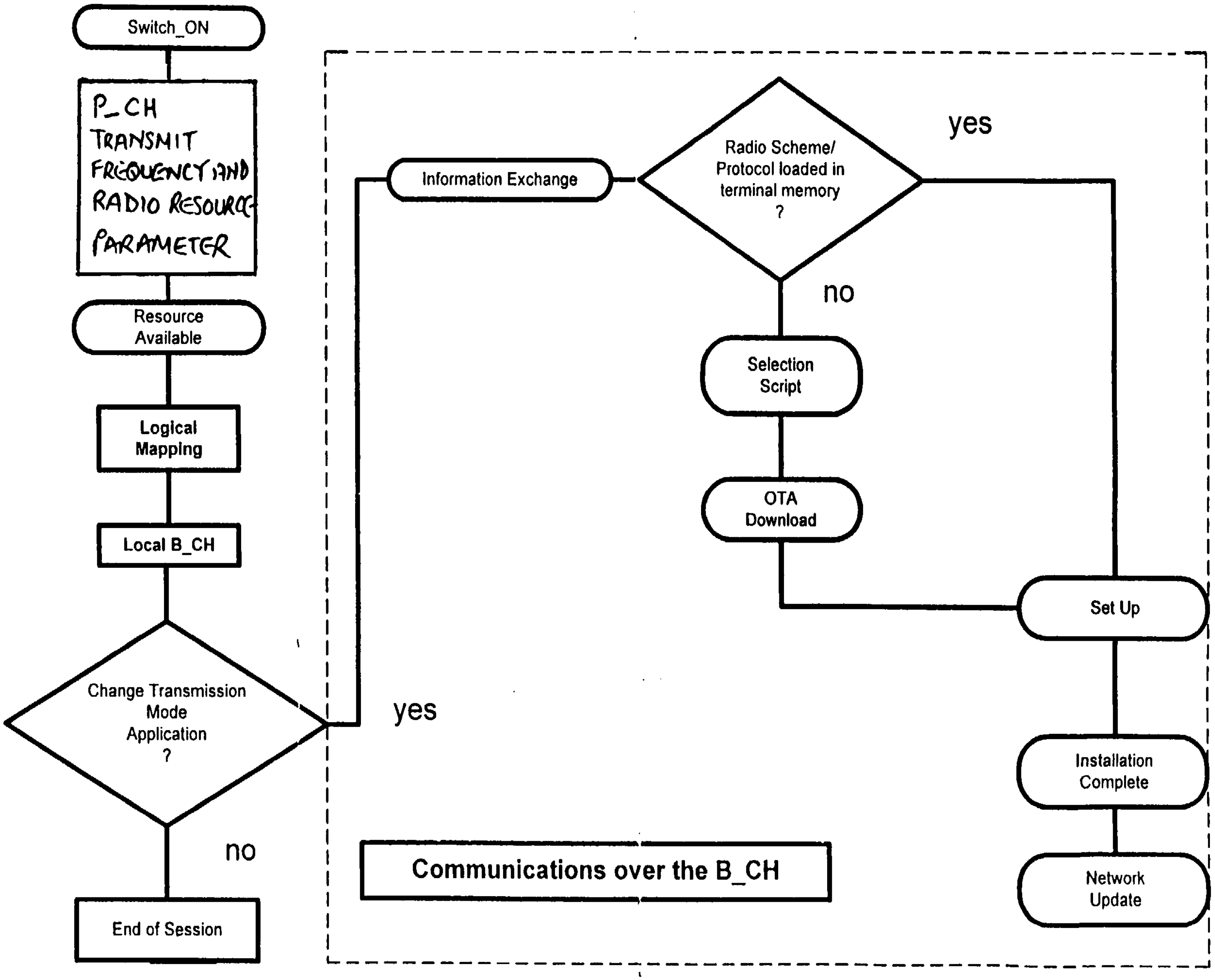

Switch_ON
P_CH TRANSMIT FREQUENCY AND RADIO RESOURCE PARAMETER
Resource Available
Logical Mapping
Local B_CH
Change Transmission Mode Application ?
no
End of Session
yes
Information Exchange
Radio Scheme/ Protocol loaded in terminal memory ?
yes
no
Selection Script
OTA Download
Set Up
Installation Complete
Network Update
Communications over the B_CH

Figure 3

Over-the-air re-programming of radio transceivers

This invention relates to radio transmitter/receivers and in particular it relates to a method of re-programming radio transmitter/receivers over-the - air.

A radio transmitter/receiver (transceiver) such as a radiotelephone is designed for operation with particular types of networks such as GSM 900 or DCS 1800. Intended use of the radiotelephone with a particular network(s) in a restricted geographical area, however, requires that the telephone be configured so as properly to communicate with the particular network (s). The user of a radiotelephone will usually have a telephone which has been configured for communication with a so called "home network". The home network is the local network usually most used by the subscriber.

The area within which a user of e.g. a GSM radiotelephone may operate, however, is considerable and is not limited to the home network but may be used on many other networks throughout the world. Use of a handset outside the home network is known as "roaming".

When the radiotelephone is to be used in roaming it is often necessary for it to have a configuration different to that for use with the home network. It is possible for re-configuration of radio transmitter/receivers to be effected by means of signals received across the air interface.

It is also convenient for the radio to be re-configurable over the air interface so as to support different types of communication and user applications e.g. addition of address book manager, whether or not it is located in the home network.

Over the air re-programming of radio receivers is well known in the art and reference may be made to US patent 5 381 138 for example. The capability to obtain programming data from a network is particularly useful for a roaming radio transmitter/receiver.

When beginning operation in an area for which the radiotelephone is not configured and it is required to download the data for reconfiguration from one of the available networks, a communication link must first be established with the network of interest. It has been proposed that a pilot channel be established in all areas from which the roaming radiotelephone may obtain the data necessary for reconfiguration.

A pilot channel of this type, however, will require a relatively large bandwidth to allow a sufficiently fast transfer of the data required.

According to the invention there is provided a method of downloading reprogramming data from a network for installation in a radio transmitter/receiver comprising initial communication from a first dedicated channel of relatively small bandwidth broadcasting at least the frequency and radio access parameters of a second channel of relatively large bandwidth from which reprogramming data may be downloaded.

Examples of the invention will now be described in more detail with reference to the accompanying figures in which

figure 1 Illustrates the logical structure of the bootstrap channel

:igure 2 Is a flow diagram of a reconfiguration process

figure 3 Is a flow diagram of an alternative reconfiguration process

A roaming radio transmitter/receiver (mobile) is located in a region served by one or more networks and the user wishes to communicate with a network from which he can obtain reprogramming data and subsequently begin communicating with the network in the communication mode selected.

A pilot channel broadcast is maintained in the region and contained in the pilot channel broadcast there is at least sufficient information for the mobile to connect to a second channel which we shall call the bootstrap channel. Conveniently the pilot channel will be broadcast in all regions over a standardised radio interface. Only a small bandwidth is required for the pilot channel because of the small amount of information contained in the broadcast.

The small bandwidth requirement makes the task of standardisation much easier with respect to the pilot channel. The wider bandwidth channels are more conveniently assigned locally for ease of implementation.

The Pilot Channel (P_CH) broadcasts a list of sets of parameters corresponding to networks available in the region. The mobile receives the network transmission through the P_CH. If the existing configuration of the mobile is matched to the available regional radio schemes, then a second channel the bootstrap channel (B_CH) is logically mapped onto the selected transmission mode. The base station and mobile exchange information over this dedicated logical channel.

The Bootstrap channel is logically mapped on top of one of the default modes of the terminal; a mapping of a logical B_CH onto the physical GSM channel for instance may be implemented. Once the mapping has been effected the terminal may download data from the base station. The bootstrap channels provided by each operator may accommodate differing services with regard to the applications available for downloading.

The flow diagram shown at fig 3 depicts a reconfiguration procedure.

When the mobile is switched on, it reads the Pilot Channel broadcast. The mobile must be configured to support the (standardised) radio interface of the Pilot Channel. The Pilot Channel carries local radio parameters (standards supported in the regional environment in which the mobile is located). After processing the received information, the mobile communicates with the base station through the Bootstrap Channel, provided that the mobile has the minimum resources required by its local radio environment. Prior to the change of channel, P_CH to B_CH, a logical mapping of the Bootstrap Channel is performed within the mobile on the selected air interface.

When operation on a local B_CH transmission has been established, the user may wish to change some properties or the performance of his mobile and can request supply of the desired services from the network. If no changes are required then the mobile adopts the default transmission mode in stand-by and releases the allocated B_CH.

If the user requests a change then communication between the base station and mobile is maintained for the exchange, the nature of which will depend on the capabilities of both mobile and network. At least 3 conditions can affect the nature of this information exchange.

Firstly, the mobile may not be able to support the required software. Where the mobile is not able to support the required software, no communication channel is available to the mobile from the existing network resources and use of the mobile within the region will therefore not be possible.

Secondly, the required software may be stored already in the mobile's memory. In this situation there is no need to download a software module but the allocated B_CH connection is maintained for further operations as described.

Thirdly, the software module required to support a different type of communication or user application may need to be downloaded from the base station. Where the download of a software module is required, initially a selection script is downloaded to the mobile followed by downloading and installation of the required software.

When the installation of the required software into the mobile has been completed, the mobile signals to the network the achievement of correct reconfiguration. On receipt of the "correct reconfiguration" signal from the mobile details of the mobile identity and its present configuration are entered on the network database (to license the product for instance) .

With reference to figure 1, the logical structure of the bootstrap channel will include 2 logical sub-channels : a download channel and a signalling control channel (S_CH). The signalling control channel assists in the reduction of errors in transmission so as to allow correct software download.

In the above example, the first channel, the Pilot Channel, is standardised and the mobile must be configured to support the radio interface for the Pilot Channel. The second (bootstrap) channel may be subject to local definition through logical mapping on a local transmission mode e.g. GSM, DECT and the mobile is not initially configured to support the radio interface for the bootstrap channel..

An example of a method of reprogramming providing greater flexibility will now be given. In this example the mobile is configured to support the radio interfaces for both the first, dedicated relatively small bandwidth (Pilot) channel and the second relatively large bandwidth (bootstrap) channel. That is to say that when the mobile is switched on in most and preferably all regions, the network can communicate with the mobile via both pilot and bootstrap channels.

In order for the mobile always to have the appropriate radio interface for the bootstrap channel then this channel would need also to be standardised (in addition to the Pilot Channel). The parameters of the bootstrap channels provided in different regions may have local variations in terms of e.g. allocated frequency, data rate and available user applications.

With reference to figure 3 which is a flow diagram of the reconfiguration process for this example, the mobile when switched on reads the Pilot Channel broadcast. The allocated frequency and radio resource parameters for the bootstrap channel contained in the pilot channel broadcast are processed and any required logical mapping effected. After processing the received information, the mobile communicates with the base station through the Bootstrap Channel.

The condition likely to be experienced in the previous example whereby the mobile is not able to support the required software and no communication channel is available to the mobile from the existing network resources does not apply in this arrangement. The communication via the bootstrap channel allows the request for and supply of the software module necessary to establish communication with the network. The transfer to the bootstrap channel does not depend on the existing configuration of the mobile since the bootstrap channel is standardised in this example and the mobile is equipped to interface, via the pilot channel, with the bootstrap channel.

The services and structure offered by the Bootstrap Channel are common for both of the above examples, however, the requirements on the terminals and networks differ.

The bootstrap channel will provide the following services by means of over -the-air (OTA) reconfiguration :
capability Exchange - the terminal provides some information to the network on its current configuration and capabilities.
module Selection : at this stage the user specifies the software that his terminal requires to download. This operation could be compared to an installation script.
data download : transfer of the data. In some cases software code will have to be downloaded whilst in other cases the software may already be implemented in the mobile. In the latter case, a set-up mechanism would be sufficient to initiate the reconfiguration.

Once the mobile and the base station are synchronised on the bootstrap channel, information exchange can begin.

Claims

1. A method of downloading reprogramming data from a network for installation in a radio transmitter/receiver comprising initial communication from a first dedicated channel of relatively small bandwidth broadcasting at least the frequency and radio access parameters of a second channel of relatively large bandwidth from which reprogramming data may be downloaded.

2. A method of downloading reprogramming data from a network as in claim 1 where first, dedicated relatively small bandwidth channel has a standard radio interface common to many network locations.

3. A method of downloading reprogramming data from a network as in claim 2 where second relatively large bandwidth channel has a standard radio interface common to many network locations.

4. A method of downloading reprogramming data from a network as in claims 1 to 3 where first, dedicated relatively small bandwidth channel broadcasts a list of sets of parameters corresponding to networks available in the region.

5. A method of downloading reprogramming data from a network as in claim 1 where the radio transmitter/receiver is configured to support the radio interfaces for both the first, dedicated relatively small bandwidth channel and the second relatively large bandwidth channel.

CHARTER MARK

The Patent Office

10

Application No:	GB 9802545.5	**Examiner:**	Glyn Hughes
Claims searched:	1 to 5	**Date of search:**	17 August 1998

Patents Act 1977
Search Report under Section 17

Databases searched:

UK Patent Office collections, including GB, EP, WO & US patent specifications, in:

UK Cl (Ed.P): H4L (LDSC, LDSU, LECC, LECX)

Int Cl (Ed.6): H04Q 7/32, 7/38

Other: Online: WPI

Documents considered to be relevant:

Category	Identity of document and relevant passage		Relevant to claims
X	US 5613204	(HABERMAN ET AL) see in particular column 15 lines 48 to 50	1
X	US 5109403	(SUTPHIN) see abstract	1

X	Document indicating lack of novelty or inventive step	A	Document indicating technological background and/or state of the art.
Y	Document indicating lack of inventive step if combined with one or more other documents of same category.	P	Document published on or after the declared priority date but before the filing date of this invention.
&	Member of the same patent family	E	Patent document published on or after, but with priority date earlier than, the filing date of this application.

EP0814589

Publication Title:

System and method for automated network reconfiguration

Abstract:

Abstract of EP0814589

A method is disclosed for providing an enhanced level of security for sensitive or proprietary information associated with information transactions in a public network, such as the Internet. In carrying out that method, an on-line information transaction is bifurcated between a generalized information access portion of such a transaction and an exchange of sensitive user information. With such a bifurcation, the generalized information access portion of the transaction, which generally would constitute the more substantial (in terms of network resources) portion of the transaction, would be handled via a non-secure network, usually a public network such as the Internet. The portion of the transaction involving sensitive user information, on the other hand, would be handled by a separate secure connection, such as a private network, or in 10a7 tranetwork. An important characteristic of this bifurcation arrangement is the provision of a means for automated reconfiguration of a user terminal as between accessing the Ageneralized information via the non-secure network and access to the secure communications network for the exchange of sensitive user information. Such an automated reconfiguration will be carried out without the necessity for any action on the part of the user, and indeed will be largely invisible to the user.

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 0 814 589 A2

(12) EUROPEAN PATENT APPLICATION

(43) Date of publication:
29.12.1997 Bulletin 1997/52

(51) Int. Cl.^{6}: H04L 29/06

(21) Application number: 97109792.8

(22) Date of filing: 16.06.1997

(84) Designated Contracting States:
AT BE CH DE DK ES FR GB GR IE IT LI LU MC NL
PT SE

(30) Priority: 19.06.1996 US 667524

(71) Applicant: AT&T Corp.
New York, NY 10013-2412 (US)

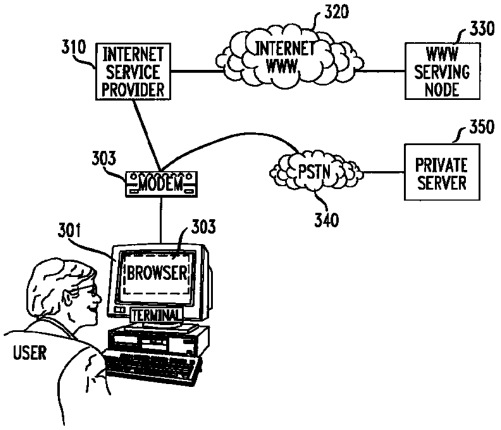
(72) Inventors:
- Harwood, Jonathan P.
Morganville, N.J. 07751 (US)
- Kimmeth, Thomas
Gladstone, N.J. 07977 (US)
- Nusbaum, Kurt
Downers Grove, Illinois 60515 (US)

(74) Representative:
KUHNEN, WACKER & PARTNER
Alois-Steinecker-Strasse 22
85354 Freising (DE)

(54) System and method for automated network reconfiguration

(57) A method is disclosed for providing an enhanced level of security for sensitive or proprietary information associated with information transactions in a public network, such as the Internet. In carrying out that method, an on-line information transaction is bifurcated between a generalized information access portion of such a transaction and an exchange of sensitive user information. With such a bifurcation, the generalized information access portion of the transaction, which generally would constitute the more substantial (in terms of network resources) portion of the transaction, would be handled via a non-secure network, usually a public network such as the Internet. The portion of the transaction involving sensitive user information, on the other hand, would be handled by a separate secure connection, such as a private network, or intranetwork. An important characteristic of this bifurcation arrangement is the provision of a means for automated reconfiguration of a user terminal as between accessing the Ageneralized information via the non-secure network and access to the secure communications network for the exchange of sensitive user information. Such an automated reconfiguration will be carried out without the necessity for any action on the part of the user, and indeed will be largely invisible to the user.

FIG. 3



Printed by Xerox (UK) Business Services
2.15.6/3.4

Description

FIELD OF THE INVENTION

This invention is related to the field of data communications, and more particularly to a method and means for establishing an automatic reconfiguration of a user terminal among alternative tasks.

BACKGROUND OF THE INVENTION

With the increasing popularity of personal computers over the last several years has come a striking growth in transaction-oriented computer-to-computer communications (as opposed to bulk-data transfers among such computers). For convenience herein such transaction-oriented computer-to-computer communications will be described by the shorthand term "information transaction". That growth in the use of computers for such information transactions has unquestionably been fueled by the existence of an international infrastructure for implementing such data communications, known as the Internet. And, driven by the burgeoning demand for such information transaction services, the Internet has itself experienced explosive growth in the amount of traffic handled.

At least partly in response to that demand, a new level of accessibility to various information sources has recently been introduced to the Internet, known as the World Wide Web ("WWW"). The WWW allows a user to access a universe of information which combines text, audio, graphics and animation within a hypermedia document. Links are contained within a WWW document which allow simple and rapid access to related documents. Using a system known as the HyperText Markup Language ("HTML"), pages of information in the WWW contain pointers to other pages, those pointers typically being a key word (commonly known as a hyperlink word). When a user selects one of those key words, a hyperlink is created to another information layer (which may be in the same, or a different information server), where typically additional detail related to that key word will be found.

In order to facilitate implementation of the WWW on the Internet, new software tools have been developed for user terminals, usually known as Web Browsers, which provide a user with a graphical user interface means for accessing information on the Web, and navigating among information layers therein. A commonly used such Web Browser is that provided by Netscape.

The substantial growth in the use of computer networks, and particularly the WWW, for such information transactions, has predictably led to significant commercialization of this communications medium. For example, with the WWW, a user is not only able to access numerous information sources, some public and some commercial, but is also able to access "catalogs" of merchandise, where individual items from such a catalog can be identified and ordered, and is able to carry out a number of banking and other financial transactions. As will be obvious, such commercial transactions will typically involve sensitive and proprietary information, such as credit card numbers and financial information of a user. Thus, with the growth of commercial activity in the Internet, has also come a heightened concern with security.

It is well known that there are persons with a high level of skill in the computer arts, commonly known as "hackers", who have both the ability and the will to intercept communications via the Internet. Such persons are thereby able to gain unauthorized access to various sensitive user information, potentially compromising or misappropriating such information.

The vulnerability of such sensitive user information to misuse when so transmitted via the Internet is a phenomena which has only recently received wide public attention. Unless such security concerns can be quickly addressed and alleviated, the commercial development of this new communications medium may be slowed or even stalled altogether.

SUMMARY OF THE INVENTION

Accordingly, it is an object of the invention to provide an acceptable level of security for sensitive or proprietary information associated with information transactions in a public network, such as the Internet. That object is realized through an arrangement whereby an on-line information transaction is bifurcated between a generalized information access portion of such a transaction and an exchange of sensitive user information. With such a bifurcation, the generalized information access portion of the transaction, which generally would constitute the more substantial (in terms of network resources) portion of the transaction would be handled via a non-secure network, usually a public network such as the Internet. The portion of the transaction involving sensitive user information, on the other hand, would be handled by a separate secure connection, such as a private network, or intranetwork. An important characteristic of this bifurcation arrangement is the provision of a means for automated reconfiguration of a user terminal as between accessing the generalized information via the non-secure network and access to the secure communications network for the exchange of sensitive user information. Such an automated reconfiguration will be carried out without the necessity for any action on the part of the user, and indeed will be largely invisible to the user. In a further embodiment of the invention, a transfer of data is provided from a public to a private network, wherein data selected by a user from a public network site may be arranged and displayed at a user terminal and, subject to further user selection/confirmation activity, thereafter transferred to a private network.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 depicts an illustrative case of information transactions carried out via a public network such as the Internet.

Figure 2 shows the architecture of a browser as would typically be applied for accessing a hypermedia web page.

Figure 3 illustrates the primary elements of the reconfigurable dual-path method of the invention.

Figure 4 depicts in flow chart form the basic jump capability of the methodology of the invention.

Figures 5A & 5B (generally designated collectively herein as "Figure 5") depict in flow chart form the "shopping cart" capability of the methodology of the invention.

Figure 6A & 6B (generally designated collectively herein as "Figure 6") depict in flow chart form the stored configuration capability of the methodology of the invention.

Figure 7A & 7B (generally designated collectively herein as "Figure 7") depict in flow chart form the off-line form capability of the methodology of the invention.

DETAILED DESCRIPTION

For clarity of explanation, the illustrative embodiment of the present invention is presented as comprising individual functional blocks. The functions these blocks represent may be provided through the use of either shared or dedicated hardware, including, but not limited to, hardware capable of executing software.

Figure 1 depicts an illustrative case of information transactions carried out via the Internet. As seen in the figure, an exemplary user obtains access to the Internet by First connecting, via a Terminal **110** having an associated Browser **111**, to an Internet Service Provider **112** selected by the user. That connection between the user and the Internet Service Provider will typically be made via the Public Switched Telephone Network (PSTN) from a modem associated with the user's Terminal to a network node in the Internet maintained by the selected Internet Service Provider.

Once the user has obtained access to the selected Internet Service Provider, an address is provided for connection to another user or other termination site and such a connection is made via the Internet to that destination location. As can be seen from the figure, communication via the Internet may be either user-to-user, as from Terminal **110** to Terminal **130**, or from a user to a node representing an information source accessed via the Internet, such as Public Site **120**.

It will of course be understood that the Internet provides service to a large number of users and includes a large number of such Public Sites, but the illustration provides the essential idea of the communication paths established for such Internet communication. It will also be understood that a number of service classifications are supported by the Internet, with the World Wide Web service, which represents a preferred embodiment for the public network aspect of the method of the invention, being one of the currently most heavily trafficked of such services.

The Web Browser, such as depicted at **111**, can be seen as a software application operating in conjunction with a user terminal (such as Terminal **110**) which provides an interface between such a user terminal and the particular functionality of the WWW information site. The architecture of such a browser is generally described in terms of three main components, as illustrated in Figure 2. At the top level is the Browser **201**, which enables the acquisition of information pages from a WWW server (beginning, in all cases, with the "home page" for that server), for display at a display device associated with the terminal. The Browser also provides the necessary interface for the terminal with the HTML functionality used by the server to provide access to other linked information layers.

The second level of the browser architecture is the TCP/IP Stack **202**, which handles the communications protocols used for connecting the terminal to the WWW server. The bottom level of this architecture is the Dialer **203**, which typically handles the function of providing dialing and setup digits to a modem, as illustrated at **204**, such a modem generally being a part of the terminal. Normally, upon receiving dialing and other setup information from the dialer, the modem would cause a connection to be made via the PSTN to the Internet Service Provider selected for that terminal.

After a connection is established in this manner to the Internet Service Provider, an address would be provided for the WWW information node sought to be contacted, a connection to that node made through the Internet, and the home page for that node caused to be displayed at the terminal's display device. A user would then select a key word in that home page, typically by clicking on the word with a mouse or similar device, and, upon transmission of that selection signal to the WWW server, a hyperlink would be created to the linked information layer and the open page of that layer would be caused to be displayed at the user terminal.

As explained above, serious questions have been raised in respect to the security of communications via the public Internet. (Note, that the discussion herein is focused on the Internet, and particularly the WWW functionality of the Internet, as a preferred embodiment of such public data communication networks generally, but the methodology of the invention will be applicable to any such network.) To address this problem, the methodology of the invention begins with a bifurcation of the information transaction between a user and the selected information transaction provider into a portion related to sensitive or proprietary user information, and other information comprising that transaction. With such a bifurcation, it becomes possible to provide substantial security for that proprietary information by use of an alternative communications path for that separated portion of the transaction via a private network, or intranetwork -- *i.e.*, a connection between a user's terminal and a secure serving node on that private network. It is anticipated that a coordination means will be established in respect to the management of information among the public and private network elements of the bifurcated information transaction.

In its basic form, this methodology may be carried out by the user terminal initiating a call via the Internet to a selected WWW node, and upon establishing connection to that node, proceeding with the desired information transaction up to the point where an exchange of sensitive or proprietary information were required. At that point the user terminal would be instructed by the WWW server to terminate that connection (*i.e.*, hangup) and to place a new call to an identified private network server for the necessary exchange of sensitive information.

However, in order to accomplish such a dual-path transaction, it is necessary that the browser at the user terminal be reconfigured to provide the dialing, authorization (*i.e.*, login and password), and other needed information for accessing the alternative private network, in order to implement the proprietary portion of the transaction. It will also usually be the case that, upon completion of that private-network transaction, the original dialer, stack and browser configurations will need to be restored, in order for the terminal to retain its normal Internet access functionality. Such a reconfiguration and subsequent restoral of the necessary parameters in the browser, stack and dialer is likely to be well beyond the capabilities of the average user.

Accordingly, as a further embodiment of the inventive methodology, an automated browser reconfiguration means is provided which interoperates with the browser. This browser reconfiguration means is described in detail hereafter and will be referred to as the "Bridging Software".

Figure 3 provides an illustration of the primary elements of the reconfigurable dual-path method of the invention. As seen in the figure, a first path comparable to the Internet link shown in Figure 1, between User Terminal **301** and WWW Serving Node **330** (via Browser **302**, Modem **303**, Internet Service Provider **310**, and Internet **320**) is provided. However, an alternative path is now provided from the output of Modem **303** to Private Server **350**. That path is illustrated as being via the PSTN, which is generally regarded as being highly secure, but an alternative dedicated or other more-secure path between the User Terminal **301** and the Private Server **350** could as well be provided. In keeping with the discussion above, Browser **302** shown in Figure 3 would also include the Bridging Software installed as a helper application for implementing the automatic reconfiguration of the Browser.

In the operation of this system, a user would normally make an initial connection to an Internet application, such as the application represented by WWW Serving Node **330**, which, *e.g.*, might be a shopping application, a financial transaction, or the provision of an enrollment form for off-line preparation. After conducting all, or some portion of an information transaction short of an exchange of sensitive or proprietary information, including a capture by the user's terminal of needed information from the public site, a user provides a signal indicative of an end to that portion of that transaction. During the course of the public portion of the information transaction, specially configured files are sent from the WWW serving node to the Bridging Software associated with Browser **302**. Such files contain instructions for the Bridging Software to store information-like products -- e.g., for selected items from a catalog, forms for enrollment, or non-secure portions of a financial transaction, and reconfiguration information for dialing and logging into the private portion of the transaction. The Bridging Software then hangs up the Internet connection, edits the user terminal's browser, stack and dialer files to reconfigure the terminal to connect to the private server. Prior to automatic redialing of the new private site for the user, the Bridging Software may be instructed by the application operating at WWW Server Node **330** to display items chosen for purchase, or to display a form for the end-user to complete off-line before dialing the private application. Upon connecting to the private application and completing the transaction as to the user sensitive information in a private environment, the Bridging Software then restores the end-user software to the dialing and authorization parameters required to dial to the public Internet.

A particularly advantageous application of the automated reconfiguration and information transfer methodology of the Bridging Software is that it adds value to certain WWW servers which do not possess the Common Gateway Interface ("CGI") capability -- *i.e.*, a provision of specialized functions on the server beyond just displaying HTML files, and are accordingly unable to accomplish any transactional processing in respect to items selected by a user. In effect, such a non-CGI server, on its own, can only serve as a "billboard" for the items represented in its database.

However, with the collection and redelivery process of the Bridging Software, a data capture and processing mechanism can be implemented for servers operating in a non-CGI environment -- such servers being incapable of more than the simple delivery of static data packets corresponding to available items. The data set enabled by the Bridging Software is a mechanism for augmenting such limited server capabilities by defining a flexible mechanism for the receipt, display, and delivery of arbitrary data from one site to another.

In such a scenario, the Bridging Software receives a "shopping cart" item list from the host as a data-set defined with a static MIME data packet associated with the Bridging Software. This information comprising the data-set may be updated, displayed to the user in a "read-only" fashion, or presented to the user for order selection.

During the process of interacting with the WWW server, a user may trigger HTML links resulting in additional MIME packets for the Bridging Software being delivered to the client. These packets allow items to be added and/or removed from the specified data set or presented to the user for local confirmation. The user will interact with a pop-up screen provided by the Bridging Software which presents the items available with product information, such as part number, description, unit cost, etc. The user identifies those items which are to be placed into the "shopping cart" and the quantity of items desired. Upon completion of the form, the Bridging Software stores the order in a format suitable for subsequent delivery to the private server site.

An additional feature provided by the methodology of the Bridging Software is an automated mechanism for providing compatibility with user terminals not previously having the Bridging Software included with the terminal's browser. To that end, the Bridging Software located at an accessed public network site initially checks to see if the browser counterpart for that software is loaded at the calling user terminal. If yes, the heretofore described processes of the Bridging Software go forward. If not however, a request is sent through the public host to download the Bridging Software to the calling terminal. After such a download, a helper application loads the Bridging Software to the terminal's browser.

I. Illustrative Embodiments

A variety of browser reconfiguration applications are supported by the automated browser reconfiguration means of the invention. Four essentially diverse capabilities of this invention, which support such applications, are described hereafter as illustrative embodiments of the invention.

A. Basic Jump Capabilities

In this configuration, which is illustrated in flow chart form in Figure 4, an end-user is connected to a chosen WWW serving node (where a desired information product is made available) via a modem and an Internet browser associated with the user's terminal (**Step 401** of Figure 4). After conducting an information transaction with the selected WWW serving node for some interval (determined in relation to the specific application accessed), the user clicks on a hypertext link, or picture, to begin an automated process which will cause that public session to be terminated and a new connection established to an alternate private data network (**Step 402**).

In response to that user action, a data message containing parameter reconfiguration instructions is passed from the WWW server application to the Bridging Software at the user's terminal (**Step 403**). Upon receiving such instructions, the Bridging Software edits the user's on-line communications software parameters, reconfiguring that software to dial the alternate data network (**Step 404**). This reconfiguration is fully automatic and transparent to the user, and includes parameters such as modem dial number, login, password, and TCP/IP addresses. At that point, the Bridging Software causes the modem to disconnect the current data network connection, shutting down the browser, and to then dial the alternate private data network (**Step 405**).

With the establishment of a connection to the private server on the alternate data network, the user interacts with the alternate data network application as appropriate (**Step 406**), and after an interval completes his activity with the alternate data network and provides an indication of such completion (**Step 407**). A data message containing parameter reconfiguration instructions is then passed from the alternate data network application to the Bridging Software (**Step 408**).

At that point, the Bridging Software again edits the user's on-line communications software parameters, reconfiguring them to dial the original public data network, or another preselected network (**Step 409**). As with the first reconfiguration, this configuration is automatic and includes parameters such as modem dial number, login, password, and TCP/IP addresses. The Bridging Software automatically causes the current private data network to be disconnected by the modem (**Step 410**), and if appropriate, causes the original public data network to be redialed (**Step 411**). When such a reconnection to the public data network is established, the end-user would then continue his application in the public data network.

B. "Shopping Cart" Capability

With this configuration, illustrated in flow chart form in Figure 5, a user begins by establishing a connection to a WWW application (assuming for the moment that the application is non-CGI enabled) at a serving node for that application, using the Internet browser and modem associated with the user's terminal (**Step 501** of Figure 5). Upon finding an item in that application to be saved, or remembered for later consideration, or purchase, the user clicks on a hypertext link, or picture, representing that item (**Step 502**). That application then sends a data message to the Bridging Software containing information about the items selected (**Step 503**) and such information is stored by the Bridging Software in the "shopping cart" file in the user's terminal (**Step 504**). Such selection download and storage steps (*i.e.*, steps 502, 503 & 504) are repeated for as many items as the user chooses to select. At any point after the Bridging Software has received the first set of item selection information, the user can instruct the Bridging Software to cause those selected items about which such information has been received to be displayed locally (at the user's terminal), where the user may review or edit (including deletion if desired) the collection of items theretofore selected. The application may also control display characteristics such as color and font for such locally displayed items. Note that in the case of a CGI-enabled application, the application itself will keep track of the items selected by the user and only download the totality of the selected items at the end of the selection process, and accordingly, the described local display option will not be applicable to such a CGI-enabled application.

At the point of completion of his "shopping", the user clicks on a hyper-text link or picture to "check out" (**Step 505**), which will begin a process of causing a jump to an alternate data network for the completion of sensitive portions of the transaction. To that end, a data message containing parameter reconfiguration instructions is passed from the WWW application to the Bridging Software (**Step 506**). It is to be noted that, as a security measure, information such as the new dial number, IP address, home page, configuration data (*e.g.*, login, password, DNS address) may be passed over the public network in encrypted form.

Upon receiving such reconfiguration instructions, the Bridging Software edits the user's on-line communications software parameters, reconfiguring that software to dial the alternate data network (**Step 507**). This reconfiguration is fully automatic and transparent to the user, and includes parameters such as modem dial number, login, password, and TCP/IP addresses. At that point, the Bridging Software causes the modem to disconnect the current data network connection, shutting down the browser, and to then dial the alternate data network (**Step 508**).

The Bridging Software passes the stored "shopping cart" data captured from the WWW application to the alternate network application (**Step 509**), where that data may be displayed for the user, permitting the user to confirm and/or modify the data (**Step 510**). The user interacts with the alternate data network application as appropriate, and after an interval completes his activity with the alternate data network (**Step 511**) and thus, by providing an appropriate completion signal to the application, completing the private portion of the information transaction (**Step 512**). A data message containing parameter reconfiguration instructions is then passed from the alternate data network application to the Bridging Software (**Step 513**).

The Bridging Software, at this point, again edits the user's on-line communications software parameters, reconfiguring them to dial the original (or another pre-defined) data network (**Step 514**). As with the first reconfiguration, this configuration is automatic and includes parameters such as modem dial number, login, password, and TCP/IP addresses. The Bridging Software automatically causes the current private data network to be disconnected by the modem (**Step 515**), and if appropriate, causes the original public data network to be redialed (**Step 516**). When such a reconnection is established to the point in the public data network where the user had left off to handle the secured aspects of his information transaction, the user would then continue his application in the public data network.

C. <u>Stored</u> <u>Configuration</u> <u>Capabilities</u>

For this configuration, depicted in flow chart form in Figure 6, an end-user is connected to a chosen WWW serving node (where a desired information product is made available) via a modem and an Internet browser associated with the user's terminal (**Step 601** of Figure 6). The user selects a hypertext link or picture associated with the WWW application by clicking on such link or picture (**Step 602**). A data message containing parameter reconfiguration instructions and an application icon (related to the selected hypertext link or picture) is passed from the WWW application to the Bridging Software (**Step 603**).

The Bridging Software creates an icon for display at the user's terminal, and saves a Bridging Software configuration file that is associated with that icon (**Step 604**). Such Bridging Software actions are automatic and multiple selections may he captured in this manner. At this point the user may continue the on-line session, or, if all desired selections have been made, a signal is provided from the user that the session should be discontinued (**Step 605**). The Bridging Software then automatically disconnects the current data network connection (**Step 606**).

After disconnecting from the WWW application, and following an interval determined by the user, a new application is selected by the user by clicking on the appropriate new icon displayed at the user's terminal (**Step 607**). The Bridging Software receives the reconfiguration instructions from the file associated with the selected icon (**Step 608**).

The Bridging Software edits the user's on-line communications software parameters, reconfiguring that software to dial the alternate data network (**Step 609**). The Bridging Software then automatically starts the user's Internet browser software and causes the alternate network application to be dialed by the modem associated with that terminal (**Step 610**). Upon establishing a connection to the alternate network, the user interacts with that application and completes the transaction to the user's satisfaction (**Step 611**). After a signal is sent to the alternate network indicating such completion of the user's activity (**Step 612**), a data message containing parameter reconfiguration instructions is passed from the alternate data network application to the Bridging Software (**Step 613**). That Software then causes the user's terminal configuration parameters to be reset (**Step 614**) and the alternate data network to be automatically disconnected (**Step 615**).

D. Off-Line Form Capability

In this configuration, depicted in flow chart form in Figure 7, an end-user is connected to a chosen WWW serving node (where a desired information product is made available) via a modem and an Internet browser associated with the user's terminal (**Step 701** of Figure 7). The user selects a hypertext link or picture associated with an off-line form application -- an exemplary such form being an HTML-based form -- by clicking on such link or picture (**Step 702**). A data message containing parameter reconfiguration instructions for the Bridging Software, the selected off-line-form application, and an optional icon (related to the selected hypertext link or picture) is passed from the WWW application to the Bridging Software (**Step 703**). Note that the selected off-line form may be for either single or multiple use.

In the case of a delayed or multiple use of the selected form, the Bridging Software may create an icon for display at the user's terminal, and will save a Bridging Software configuration file that is associated with that icon (**Step 704**). The form in question is also saved on the user's terminal. Such Bridging Software actions are automatic. At this point the user may continue the on-line session, or, if all desired selections have been made, a signal is provided from the user that the session should be discontinued (**Step 705**). The Bridging Software then automatically disconnects the current data network connection (**Step 706**).

After disconnecting from the WWW application, two cases are to be considered as to the further processing of the selected form: (1) an immediate single use of the form and (2) either a delayed or multiple use of the form. In the first case, the Bridging Software edits the user's on-line communications software parameters, reconfiguring that software to dial the alternate data network. The Bridging Software then automatically starts the user's Internet browser software which is caused to display the off-line form. The user then completes the off-line form and chooses a "Submit Form" button displayed at his terminal.

In the second case, the Bridging Software will have created an icon for display at the user's terminal and saved a Bridging Software configuration file associated with that icon. Following an interval determined by the user, the off-line-form application is started by the user by clicking on the new form icon displayed at the user's terminal (**Step 707**). The Bridging Software receives the reconfiguration instructions from the file associated with the selected icon (**Step 708**).

The Bridging Software edits the user's on-line communications software parameters, reconfiguring that software to dial the alternate data network (**Step 709**). The Bridging Software then automatically starts the user's Internet browser software which is caused to display the off-line form (**Step 710**). The user then completes the off-line form and chooses a "Submit Form" button displayed at his terminal (**Step 711**).

In either the first or second case, following activation of the "Submit Form" button, the alternate network application is then caused to be dialed by the Bridging Software. Upon establishing a connection to the alternate network, the form data is passed to the alternate network (**Step 712**). The user then interacts with that application and completes the application (**Step 713**). After a signal is sent to the alternate network indicating such completion of the user's activity (**Step 714**),a data message containing parameter reconfiguration instructions is passed from the alternate data network application to the Bridging Software (**Step 715**). That Software then causes the user's terminal configuration parameters to be reset (**Step 716**) and the alternate data network to be automatically disconnected (**Step 717**).

CONCLUSION

A system and method has been described for the automatic switching of an information transaction between two or more alternate networks. This functionality, which incorporates a reconfiguration means designated herein as the Bridging Software, supports the movement of application specific data from one on-line environment to another. Among potential applications of this process for passing data between different environments are: selected items for purchase ("shopping cart"), captured data from forms, and other server captured data such as web pages visited.

The Bridging Software reconfiguration means is intended to work with various Web Browser software implementations, including the Netscape Personal Edition (NPE) Software for Windows 3.1 and 3.11, and which represents a working embodiment for the invention. The Bridging Software installs itself as a helper application within the browser application and utilizes a special MIME type configuration file to pass reconfiguration and "shopping cart" information from the server to the client software.

When an application requires a user to re-connect to a private application, a reconfiguration file is passed to the Bridging Software helper application via a CGI script or simple hyper-text link. The helper application disconnects the current data connection, reconfigures the dial parameters (dial #, login password, DNS address, and home page) and initiates the dial program so the end-user can access the private application.

When the end-user connects to the private application, the Bridging Software reconfiguration means provides the new "private server" application with data collected from the "public server", and the application resumes in a private, secure environment.

The Bridging Software allows both short term and long term storage of dial configurations. Configurations passed to the Bridging Software can be designated as single use configurations and discarded after the application has terminated, or saved and displayed to the end-user as a dial choice by the Bridging Software.

Although the present embodiment of the invention has been described in detail, it should be understood that various changes, alterations and substitutions can be made therein without departing from the spirit and scope of the invention as defined by the appended claims. In particular, it is noted that, while the invention has been primarily described in terms of a preferred embodiment based on an automatic reconfiguration between a public and a private data network, any the methodology of the invention will be equally applicable to any set of alternate networks.

Claims

1. A method for managing a transaction via a communications path between a terminal device and a serving node in a data network, said method comprising the steps of:

   establishing an initial communications path via a first connection between said terminal device and a serving node in a first data network;
   receiving information from said serving node in said first data network for effecting a reconfiguration of said communications path for said transaction from said first connection in said first data network to a second connection in a second data network; and
   automatically connecting said terminal device to a serving node in said second data network via said second connection.

2. A method for managing a transaction via a communications path between a terminal device and a serving node in a data network, said method comprising the steps of:

   establishing an initial communications path via a first connection between said terminal device and a serving node in a first data network;
   selecting at least one information item from a data base of said information items provided at said serving node in said first data network;
   causing said selected information items to be downloaded to said terminal device via said first connection;
   receiving information from said serving node in said first data network for effecting a reconfiguration of said communications path for said transaction from said first connection in said first data network to a second connection in a second data network; and
   automatically connecting said terminal device to a serving node in said second data network via said second connection.

3. A method for managing a transaction via a communications path between a terminal device and a serving node in a data network, said method comprising the steps of:

   establishing an initial communications path via a first connection between said terminal device and a serving node in a first data network;
   identifying at least one data network application from a data base of said data network applications provided at said serving node in said first data network;
   receiving information from said serving node in said first data network for reconfiguring said terminal device for implementation of a communication path via an alternate connection between said terminal device and at least one of said identified data network applications in a second data network; and
   in response to a selection signal from a user, automatically connecting said terminal device to a selected one of said identified data network applications via said alternate connection.

4. A method for managing a transaction via a communications path between a terminal device and a serving node in a data network, said method comprising the steps of:

   establishing an initial communications path via a first connection between said terminal device and a serving node in a first data network;
   selecting an off-line form application from a data base provided at said serving node in said first data network;
   receiving information from said serving node in said first data network for reconfiguring said terminal device for implementation of a communication path via a second connection between said terminal device and said

selected off-line form application in a second data network; and

in response to, a selection signal from a user, automatically connecting said terminal device to said selected off-line form application.

5. The method for managing a transaction of Claim 1 or 2 including the further step of recognizing a signal to reconfigure said communications path from said first connection to said second connection.

6. The method for managing a transaction of Claim 3 wherein said selected data network application is operated at a serving node in said second data network.

7. The method for managing a transaction of Claim 4 wherein said selected off-line form application is operated at a serving node in said second data network.

8. The method for managing a transaction of one of the Claims 1, 2, 6 or 7 wherein said serving nodes in said first and said second data networks are manifested in a common node.

9. The method for managing a transaction of Claim 1 or 2 wherein said step of receiving information includes the further step of effecting said reconfiguration of said communications path.

10. The method for managing a transaction of Claim 1 or 2 wherein said step of automatically connecting includes the step of automatically disconnecting said first connection prior to implementation of said second connection.

11. The method for managing a transaction of Claim 1 or 2 including the further steps of:

automatically disconnecting said second connection in response to a user signal; and

reconfiguring said terminal device to enable, in response to user instruction, an implementation of a connection via an identified data network.

12. The method for managing a transaction of Claim 11 wherein said step of automatically reconfiguring said terminal device includes the step of effecting said implementation of said connection via said identified data network.

13. The method for managing a transaction of Claim 2 wherein said step of causing said selected information items to be downloaded includes the further step of causing said selected information items to be displayed at said terminal device.

14. The method for managing a transaction of Claim 13 wherein said displayed selected items can be edited by a user at said terminal device.

15. The method for managing a transaction of Claim 13 wherein display characteristics for said displayed selected items can be controlled at said terminal device.

16. The method for managing a transaction of Claim 2 wherein said step of automatically connecting includes the step of uploading said selected information items from said terminal device to said service provider via said second connection.

17. The method for managing a transaction of Claim 3 including the further steps of:

automatically disconnecting said alternate connection in response to a user signal; and

reconfiguring said terminal device to enable implementation of a pre-selected connection between said terminal device and an identified data network.

18. The method for managing a transaction of Claim 17 wherein said step of automatically reconfiguring said terminal device includes the further step of effecting said implementation of said pre-selected connection.

19. The method for managing a transaction of Claim 4 including the further step of downloading from said serving node in said first data network to said terminal device of an off-line form related to said off-line form application.

20. The method for managing a transaction of Claim 4 including the further step of uploading said downloaded off-line form from said terminal device to said selected off-line form application, after processing by a user.

21. The method for managing a transaction of Claim 4 including the further steps of:

automatically disconnecting said connection to said selected off-line form application in response to a user signal; and
reconfiguring said terminal device to enable implementation of a pre-selected connection between said terminal device and an identified data network.

22. The method for managing a transaction of Claim 21 wherein said step of automatically reconfiguring said terminal device includes the further step of effecting said implementation of said pre-selected connection.

23. A method for managing connections between a terminal device and at least one information source/processor wherein at least two of said connections are implemented via separate communications networks, comprising the steps of:

recognizing a signal for connection to an information source/processor via a communications network other than a communications network for which a predetermined connection is configured;
causing said terminal device to implement a connection to said information source/processor via said other communications network; and
upon termination of said information source/processor connection via said other communications network, automatically reconfiguring a connection criteria in said terminal device to enable said terminal device to implement, in response to user instruction, a connection via an alternative one of said communications networks.

24. The method for managing connections of Claim 23 wherein said recognizing step occurs at a point when said terminal device is connected to a given source/processor.

25. The method for managing connections of Claim 23 wherein information items may be selected by a user at said terminal device from said given source/processor, and including the further step of causing said selected information items to be downloaded from said source/processor to said terminal device.

26. The method for managing connections of Claim 25 wherein said step of effecting connection includes the further step of uploading said selected information items from said terminal device to said other information source/processor.

27. The method for managing connections of Claim 26 wherein said selected information items are processed by said user at said terminal device prior to uploading to said other information source/processor.

28. The method for managing connections of Claim 24 including the further step of causing said given source/processor to download to said terminal device configuration data for enabling said step of effecting connection to said other information source/processor.

29. The method for managing connections of Claim 24 including the further step of causing said other source/processor to download to said terminal device configuration data for enabling said step of automatically restoring a prior connection criteria in said terminal device.

30. A method for enhancing security of certain data in an on-line information transaction comprising the steps of:

bifurcating said information transaction into a first portion comprising said certain data and a remaining portion, wherein said remaining portion is carried out via a public on-line communications connection between a terminal device and a public information server;
causing said first portion to be carried out via a secure private on-line communications connection between said terminal device and a private information server; and
automatically reconfiguring network access means in said terminal device to switch between said public connection and said private connection.

FIG. 1

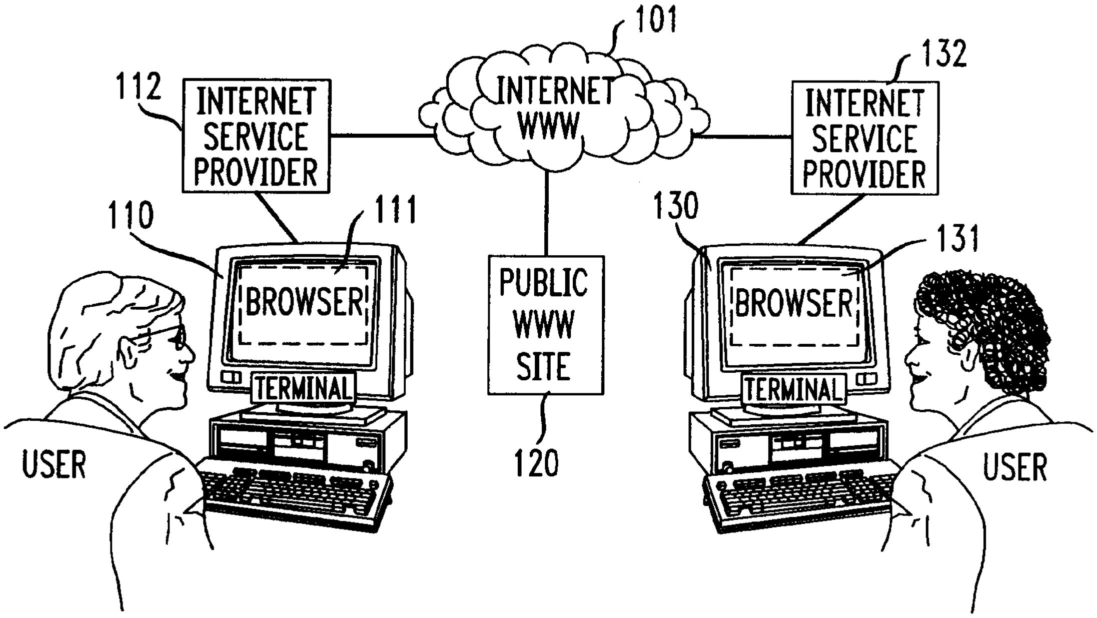

101
INTERNET WWW
112
INTERNET SERVICE PROVIDER
132
INTERNET SERVICE PROVIDER
110
111
BROWSER
TERMINAL
USER
PUBLIC WWW SITE
120
130
131
BROWSER
TERMINAL
USER

FIG. 2

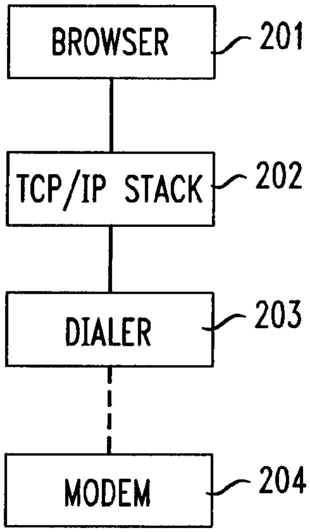

BROWSER
201
TCP/IP STACK
202
DIALER
203
MODEM
204

FIG. 3

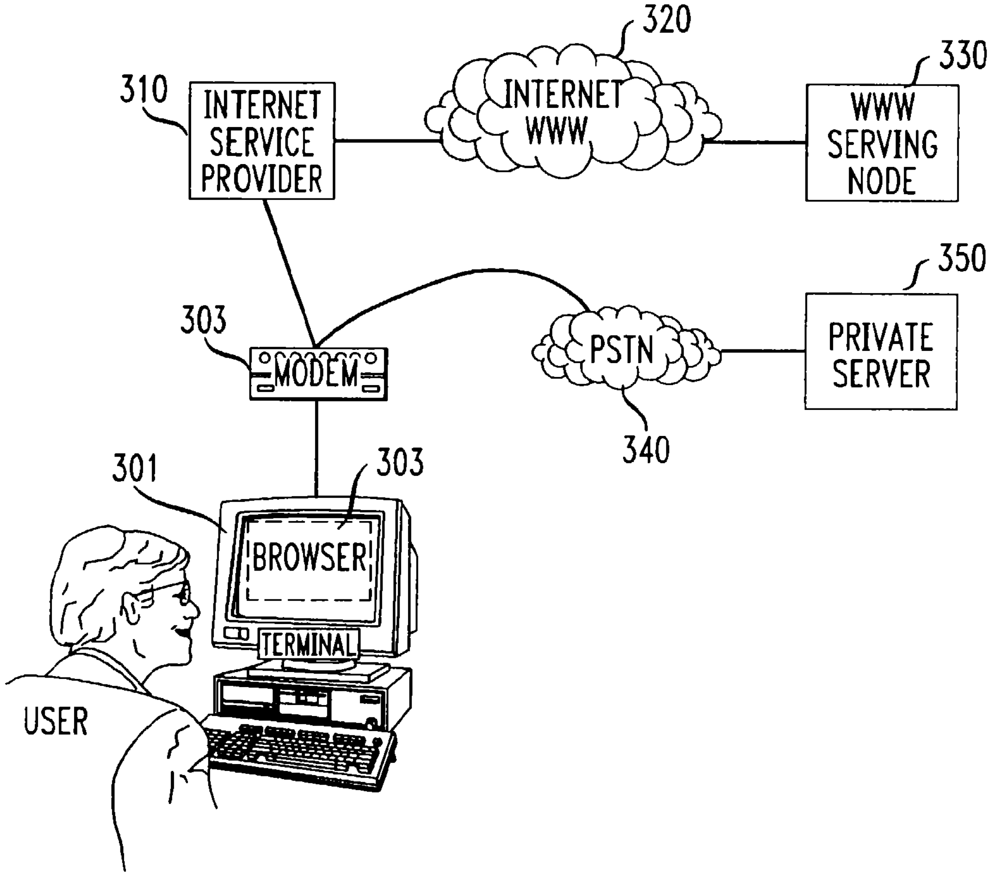
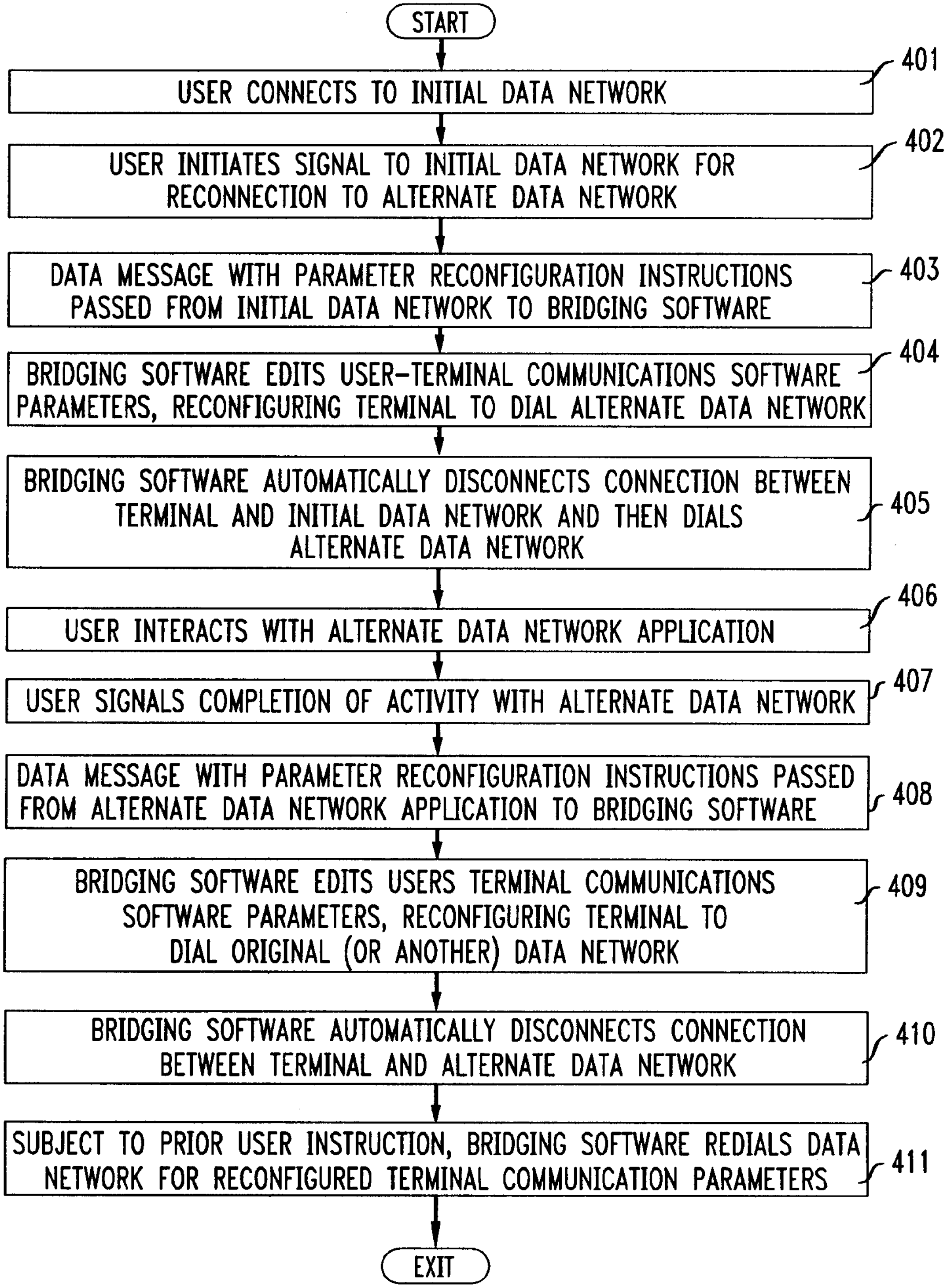

320
310
INTERNET SERVICE PROVIDER
INTERNET WWW
330
WWW SERVING NODE
350
303
MODEM
PSTN
PRIVATE SERVER
340
301
303
BROWSER
TERMINAL
USER

FIG. 4


START
USER CONNECTS TO INITIAL DATA NETWORK
401
USER INITIATES SIGNAL TO INITIAL DATA NETWORK FOR RECONNECTION TO ALTERNATE DATA NETWORK
402
DATA MESSAGE WITH PARAMETER RECONFIGURATION INSTRUCTIONS PASSED FROM INITIAL DATA NETWORK TO BRIDGING SOFTWARE
403
BRIDGING SOFTWARE EDITS USER-TERMINAL COMMUNICATIONS SOFTWARE PARAMETERS, RECONFIGURING TERMINAL TO DIAL ALTERNATE DATA NETWORK
404
BRIDGING SOFTWARE AUTOMATICALLY DISCONNECTS CONNECTION BETWEEN TERMINAL AND INITIAL DATA NETWORK AND THEN DIALS ALTERNATE DATA NETWORK
405
USER INTERACTS WITH ALTERNATE DATA NETWORK APPLICATION
406
USER SIGNALS COMPLETION OF ACTIVITY WITH ALTERNATE DATA NETWORK
407
DATA MESSAGE WITH PARAMETER RECONFIGURATION INSTRUCTIONS PASSED FROM ALTERNATE DATA NETWORK APPLICATION TO BRIDGING SOFTWARE
408
BRIDGING SOFTWARE EDITS USERS TERMINAL COMMUNICATIONS SOFTWARE PARAMETERS, RECONFIGURING TERMINAL TO DIAL ORIGINAL (OR ANOTHER) DATA NETWORK
409
BRIDGING SOFTWARE AUTOMATICALLY DISCONNECTS CONNECTION BETWEEN TERMINAL AND ALTERNATE DATA NETWORK
410
SUBJECT TO PRIOR USER INSTRUCTION, BRIDGING SOFTWARE REDIALS DATA NETWORK FOR RECONFIGURED TERMINAL COMMUNICATION PARAMETERS
411
EXIT

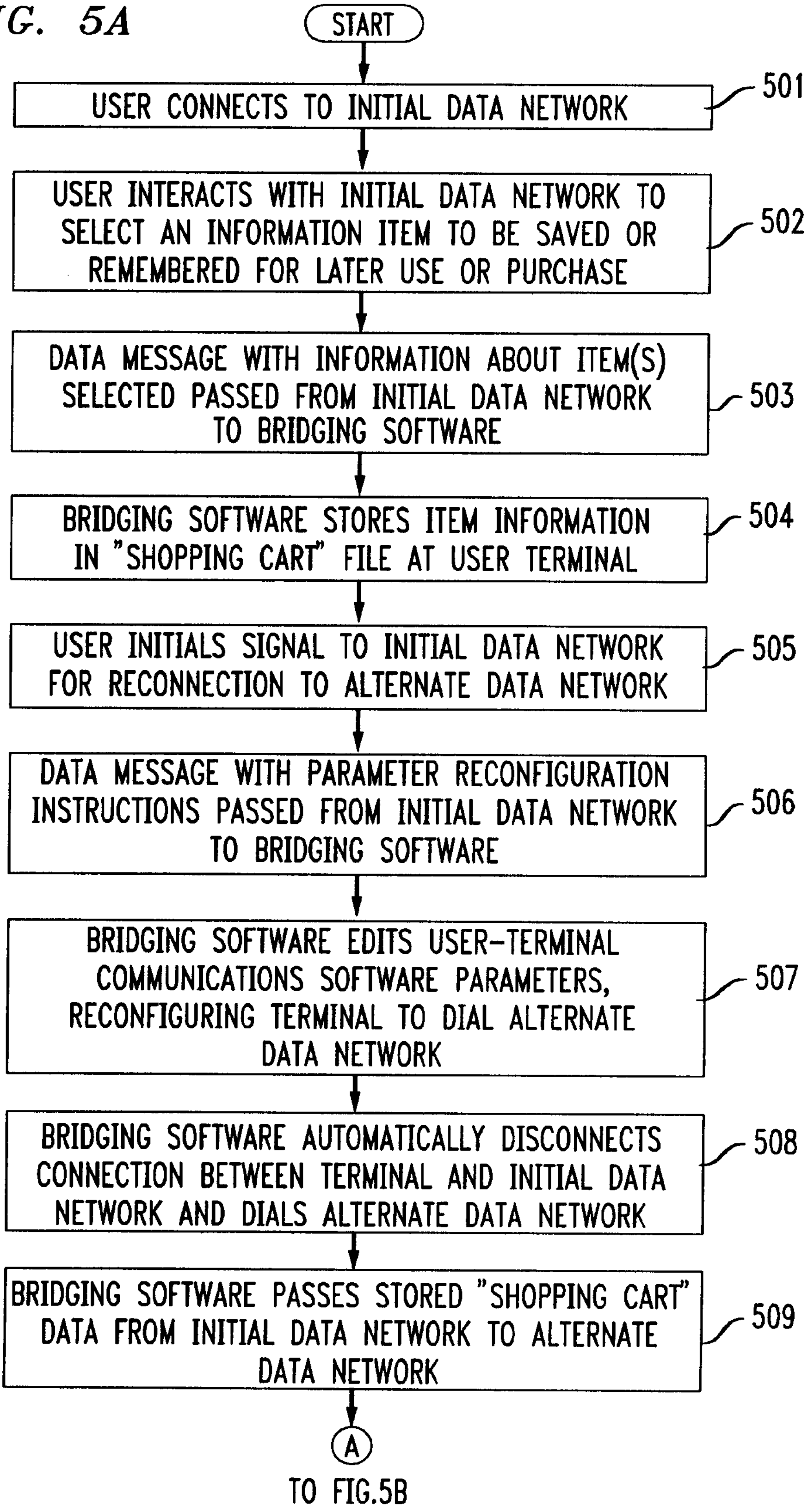


FIG. 5A
START
USER CONNECTS TO INITIAL DATA NETWORK
501
USER INTERACTS WITH INITIAL DATA NETWORK TO SELECT AN INFORMATION ITEM TO BE SAVED OR REMEMBERED FOR LATER USE OR PURCHASE
502
DATA MESSAGE WITH INFORMATION ABOUT ITEM(S) SELECTED PASSED FROM INITIAL DATA NETWORK TO BRIDGING SOFTWARE
503
BRIDGING SOFTWARE STORES ITEM INFORMATION IN "SHOPPING CART" FILE AT USER TERMINAL
504
USER INITIALS SIGNAL TO INITIAL DATA NETWORK FOR RECONNECTION TO ALTERNATE DATA NETWORK
505
DATA MESSAGE WITH PARAMETER RECONFIGURATION INSTRUCTIONS PASSED FROM INITIAL DATA NETWORK TO BRIDGING SOFTWARE
506
BRIDGING SOFTWARE EDITS USER-TERMINAL COMMUNICATIONS SOFTWARE PARAMETERS, RECONFIGURING TERMINAL TO DIAL ALTERNATE DATA NETWORK
507
BRIDGING SOFTWARE AUTOMATICALLY DISCONNECTS CONNECTION BETWEEN TERMINAL AND INITIAL DATA NETWORK AND DIALS ALTERNATE DATA NETWORK
508
BRIDGING SOFTWARE PASSES STORED "SHOPPING CART" DATA FROM INITIAL DATA NETWORK TO ALTERNATE DATA NETWORK
509
A
TO FIG.5B

FIG. 5B

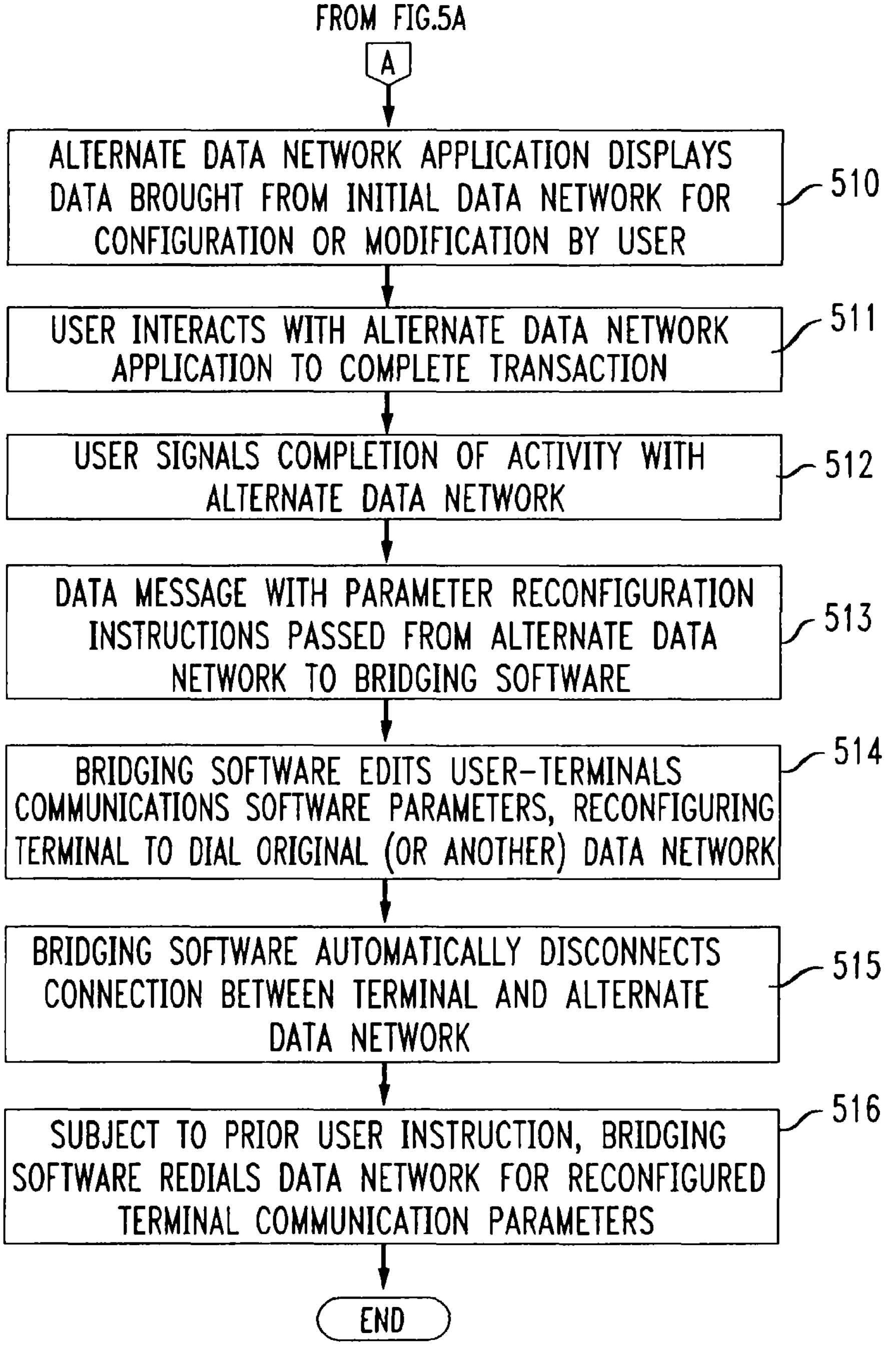

FROM FIG.5A
A
ALTERNATE DATA NETWORK APPLICATION DISPLAYS DATA BROUGHT FROM INITIAL DATA NETWORK FOR CONFIGURATION OR MODIFICATION BY USER
510
USER INTERACTS WITH ALTERNATE DATA NETWORK APPLICATION TO COMPLETE TRANSACTION
511
USER SIGNALS COMPLETION OF ACTIVITY WITH ALTERNATE DATA NETWORK
512
DATA MESSAGE WITH PARAMETER RECONFIGURATION INSTRUCTIONS PASSED FROM ALTERNATE DATA NETWORK TO BRIDGING SOFTWARE
513
BRIDGING SOFTWARE EDITS USER-TERMINALS COMMUNICATIONS SOFTWARE PARAMETERS, RECONFIGURING TERMINAL TO DIAL ORIGINAL (OR ANOTHER) DATA NETWORK
514
BRIDGING SOFTWARE AUTOMATICALLY DISCONNECTS CONNECTION BETWEEN TERMINAL AND ALTERNATE DATA NETWORK
515
SUBJECT TO PRIOR USER INSTRUCTION, BRIDGING SOFTWARE REDIALS DATA NETWORK FOR RECONFIGURED TERMINAL COMMUNICATION PARAMETERS
516
END

FIG. 6A

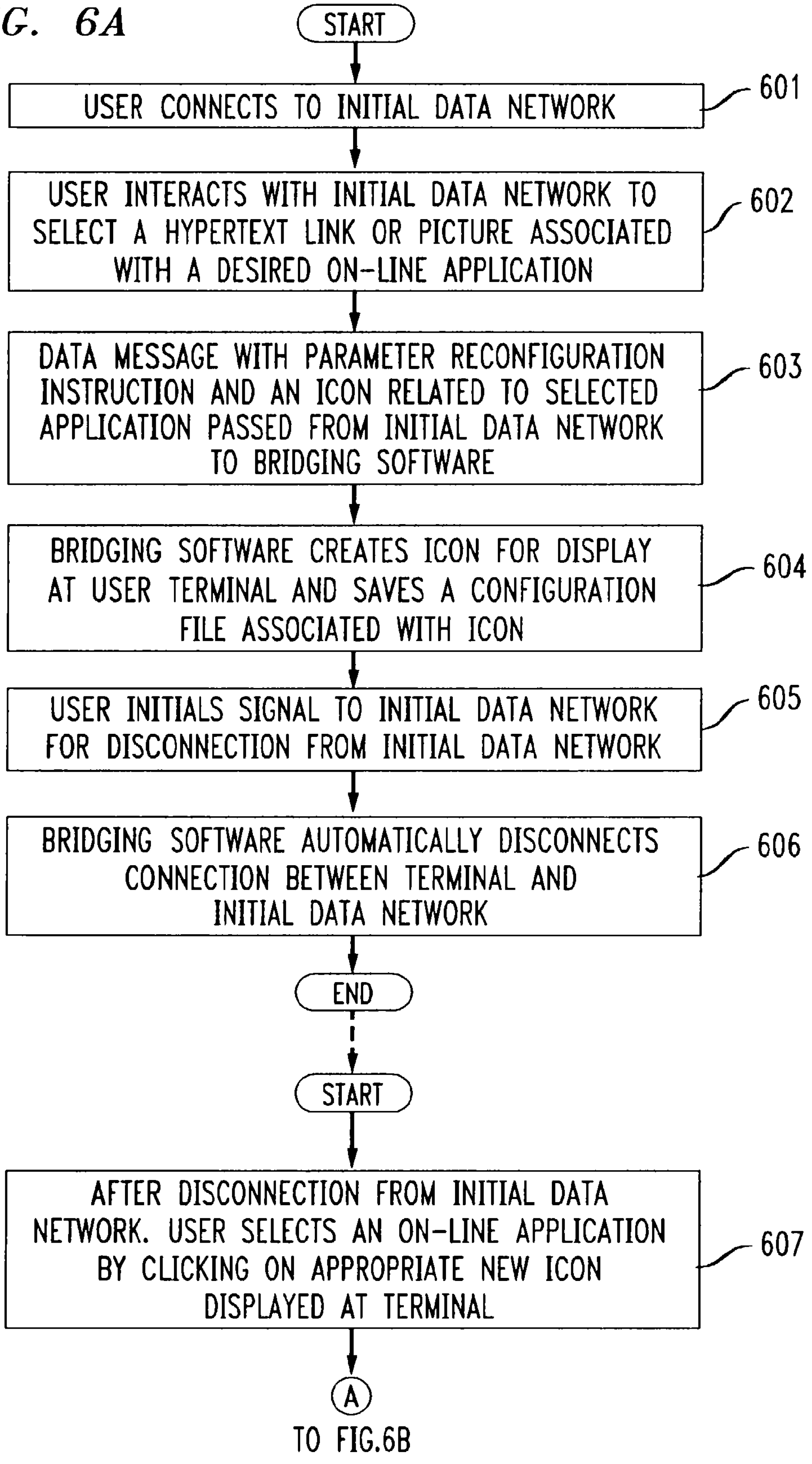

START
USER CONNECTS TO INITIAL DATA NETWORK
601
USER INTERACTS WITH INITIAL DATA NETWORK TO SELECT A HYPERTEXT LINK OR PICTURE ASSOCIATED WITH A DESIRED ON-LINE APPLICATION
602
DATA MESSAGE WITH PARAMETER RECONFIGURATION INSTRUCTION AND AN ICON RELATED TO SELECTED APPLICATION PASSED FROM INITIAL DATA NETWORK TO BRIDGING SOFTWARE
603
BRIDGING SOFTWARE CREATES ICON FOR DISPLAY AT USER TERMINAL AND SAVES A CONFIGURATION FILE ASSOCIATED WITH ICON
604
USER INITIALS SIGNAL TO INITIAL DATA NETWORK FOR DISCONNECTION FROM INITIAL DATA NETWORK
605
BRIDGING SOFTWARE AUTOMATICALLY DISCONNECTS CONNECTION BETWEEN TERMINAL AND INITIAL DATA NETWORK
606
END
START
AFTER DISCONNECTION FROM INITIAL DATA NETWORK. USER SELECTS AN ON-LINE APPLICATION BY CLICKING ON APPROPRIATE NEW ICON DISPLAYED AT TERMINAL
607
A
TO FIG.6B

FIG. 6B

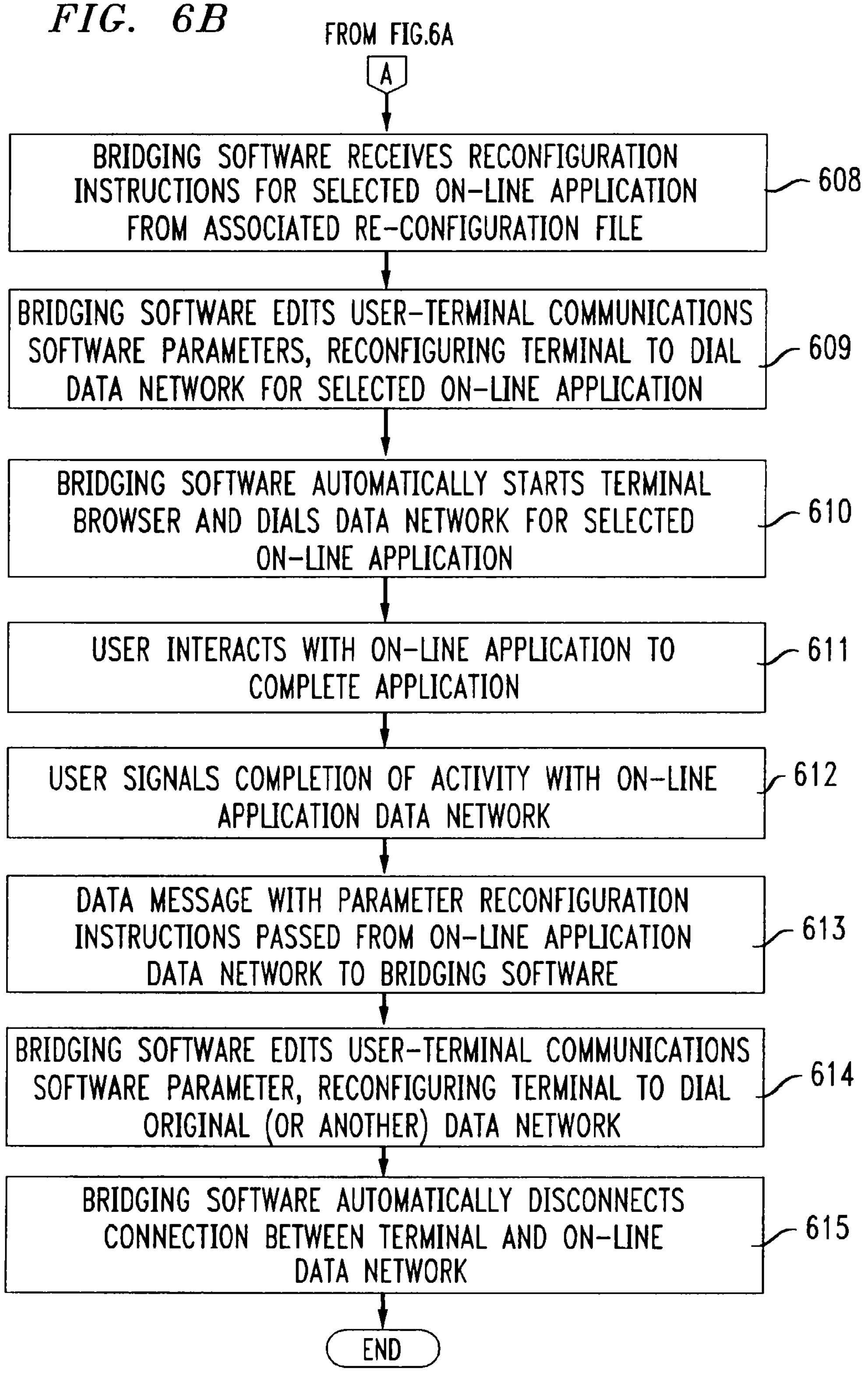

FROM FIG.6A
A
BRIDGING SOFTWARE RECEIVES RECONFIGURATION INSTRUCTIONS FOR SELECTED ON-LINE APPLICATION FROM ASSOCIATED RE-CONFIGURATION FILE
608
BRIDGING SOFTWARE EDITS USER-TERMINAL COMMUNICATIONS SOFTWARE PARAMETERS, RECONFIGURING TERMINAL TO DIAL DATA NETWORK FOR SELECTED ON-LINE APPLICATION
609
BRIDGING SOFTWARE AUTOMATICALLY STARTS TERMINAL BROWSER AND DIALS DATA NETWORK FOR SELECTED ON-LINE APPLICATION
610
USER INTERACTS WITH ON-LINE APPLICATION TO COMPLETE APPLICATION
611
USER SIGNALS COMPLETION OF ACTIVITY WITH ON-LINE APPLICATION DATA NETWORK
612
DATA MESSAGE WITH PARAMETER RECONFIGURATION INSTRUCTIONS PASSED FROM ON-LINE APPLICATION DATA NETWORK TO BRIDGING SOFTWARE
613
BRIDGING SOFTWARE EDITS USER-TERMINAL COMMUNICATIONS SOFTWARE PARAMETER, RECONFIGURING TERMINAL TO DIAL ORIGINAL (OR ANOTHER) DATA NETWORK
614
BRIDGING SOFTWARE AUTOMATICALLY DISCONNECTS CONNECTION BETWEEN TERMINAL AND ON-LINE DATA NETWORK
615
END

FIG. 7A

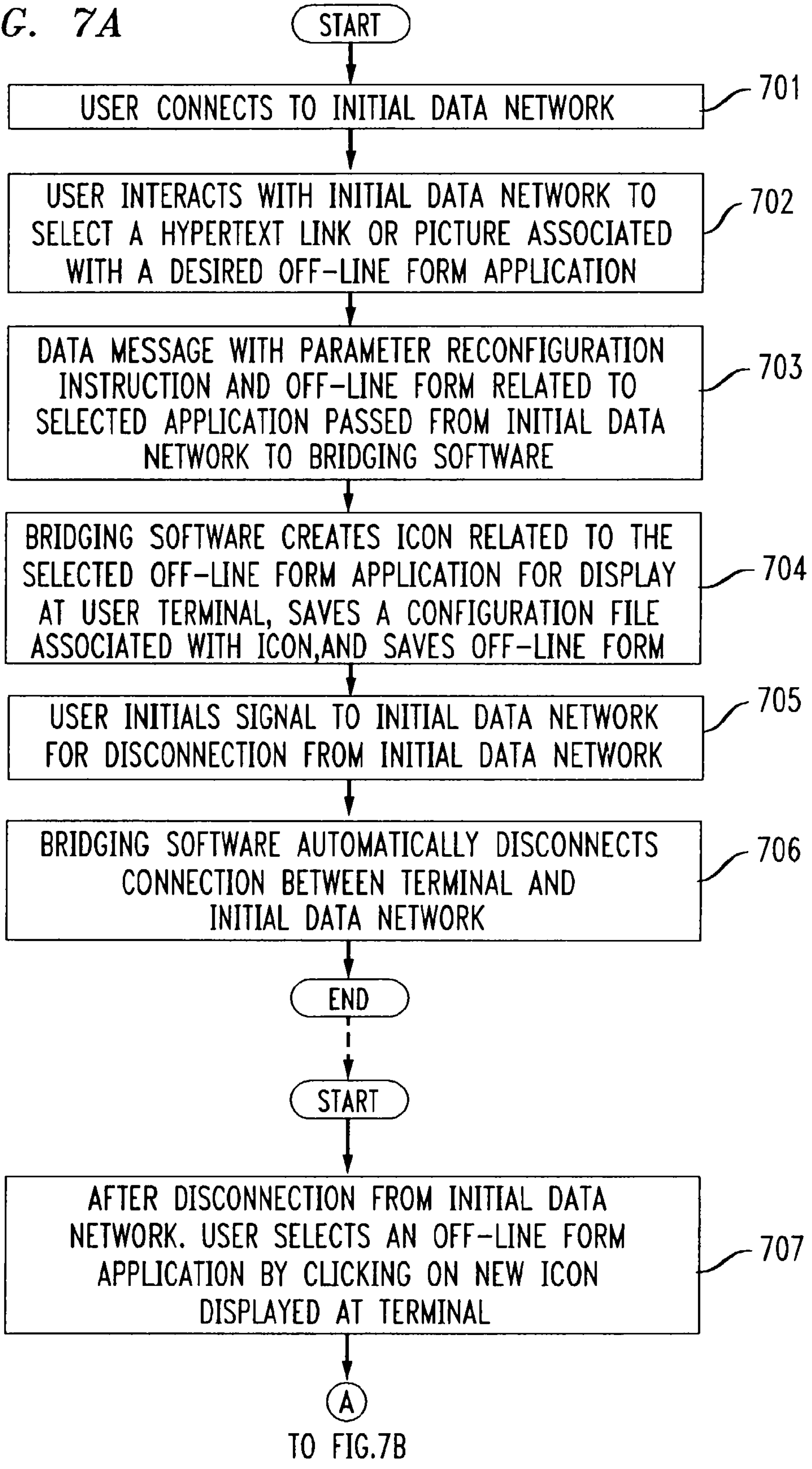
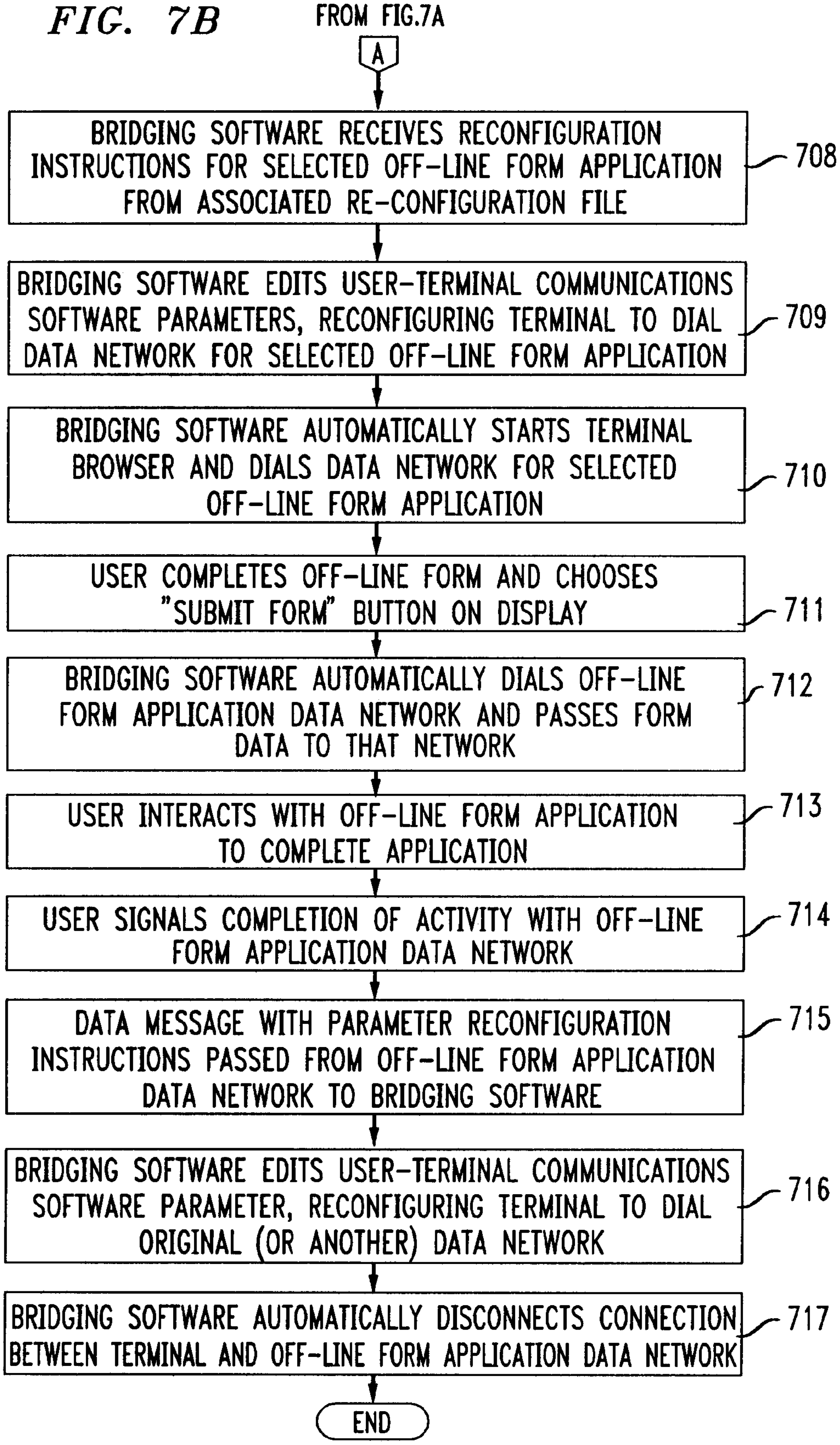

START
USER CONNECTS TO INITIAL DATA NETWORK
701
USER INTERACTS WITH INITIAL DATA NETWORK TO SELECT A HYPERTEXT LINK OR PICTURE ASSOCIATED WITH A DESIRED OFF-LINE FORM APPLICATION
702
DATA MESSAGE WITH PARAMETER RECONFIGURATION INSTRUCTION AND OFF-LINE FORM RELATED TO SELECTED APPLICATION PASSED FROM INITIAL DATA NETWORK TO BRIDGING SOFTWARE
703
BRIDGING SOFTWARE CREATES ICON RELATED TO THE SELECTED OFF-LINE FORM APPLICATION FOR DISPLAY AT USER TERMINAL, SAVES A CONFIGURATION FILE ASSOCIATED WITH ICON,AND SAVES OFF-LINE FORM
704
USER INITIALS SIGNAL TO INITIAL DATA NETWORK FOR DISCONNECTION FROM INITIAL DATA NETWORK
705
BRIDGING SOFTWARE AUTOMATICALLY DISCONNECTS CONNECTION BETWEEN TERMINAL AND INITIAL DATA NETWORK
706
END
START
AFTER DISCONNECTION FROM INITIAL DATA NETWORK. USER SELECTS AN OFF-LINE FORM APPLICATION BY CLICKING ON NEW ICON DISPLAYED AT TERMINAL
707
A
TO FIG.7B

FIG. 7B


FROM FIG.7A
A
BRIDGING SOFTWARE RECEIVES RECONFIGURATION INSTRUCTIONS FOR SELECTED OFF-LINE FORM APPLICATION FROM ASSOCIATED RE-CONFIGURATION FILE
708
BRIDGING SOFTWARE EDITS USER-TERMINAL COMMUNICATIONS SOFTWARE PARAMETERS, RECONFIGURING TERMINAL TO DIAL DATA NETWORK FOR SELECTED OFF-LINE FORM APPLICATION
709
BRIDGING SOFTWARE AUTOMATICALLY STARTS TERMINAL BROWSER AND DIALS DATA NETWORK FOR SELECTED OFF-LINE FORM APPLICATION
710
USER COMPLETES OFF-LINE FORM AND CHOOSES "SUBMIT FORM" BUTTON ON DISPLAY
711
BRIDGING SOFTWARE AUTOMATICALLY DIALS OFF-LINE FORM APPLICATION DATA NETWORK AND PASSES FORM DATA TO THAT NETWORK
712
USER INTERACTS WITH OFF-LINE FORM APPLICATION TO COMPLETE APPLICATION
713
USER SIGNALS COMPLETION OF ACTIVITY WITH OFF-LINE FORM APPLICATION DATA NETWORK
714
DATA MESSAGE WITH PARAMETER RECONFIGURATION INSTRUCTIONS PASSED FROM OFF-LINE FORM APPLICATION DATA NETWORK TO BRIDGING SOFTWARE
715
BRIDGING SOFTWARE EDITS USER-TERMINAL COMMUNICATIONS SOFTWARE PARAMETER, RECONFIGURING TERMINAL TO DIAL ORIGINAL (OR ANOTHER) DATA NETWORK
716
BRIDGING SOFTWARE AUTOMATICALLY DISCONNECTS CONNECTION BETWEEN TERMINAL AND OFF-LINE FORM APPLICATION DATA NETWORK
717
END

EP0838930

Publication Title:

Pseudo network adapter for frame capture, encapsulation and encryption

Abstract:

Abstract of EP0838930

A new pseudo network adapter provides an interface for capturing packets from a local communications protocol stack for transmission on the virtual private network, and includes a Dynamic Host Configuration Protocol (DHCP) server emulator, and an Address Resolution Protocol (ARP) server emulator. The new system indicates to the local communications protocol stack that nodes on a remote private network are reachable through a gateway that is in turn reachable through the pseudo network adapter. A transmit path in the system processes data packets from the local communications protocol stack for transmission through the pseudo network adapter. An encryption engine encrypts the data packets and an encapsulation engine encapsulates the encrypted data packets into tunnel data frames. The network adapter further includes an interface into a transport layer of the local communications protocol stack for capturing received data packets from the remote server node, and a receive path for processing received data packets captured from the transport layer of the local communications protocol stack. The receive path includes a decapsulation engine, and a decryption engine, and passes the decrypted, decapsulated data packets back to the local communications protocol stack for de ed1 livery to a user.

Data supplied from the esp@cenet database - Worldwide

Courtesy of http://v3.espacenet.com

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 0 838 930 A2

(12) **EUROPEAN PATENT APPLICATION**

(43) Date of publication:
29.04.1998 Bulletin 1998/18

(51) Int. Cl.^{6}: **H04L 29/06**

(21) Application number: **97118556.6**

(22) Date of filing: **24.10.1997**

(84) Designated Contracting States:
AT BE CH DE DK ES FI FR GB GR IE IT LI LU MC NL PT SE
Designated Extension States:
AL LT LV RO SI

(30) Priority: **25.10.1996 US 738155**

(71) Applicant:
DIGITAL EQUIPMENT CORPORATION
Maynard, Massachusetts 01754 (US)

(72) Inventors:
- **Alden, Kenneth F.**
  Boylston, Massachusetts 01505 (US)
- **Lichtenberg, Mitchell P.**
  Sunnyvale, CA 94087 (US)
- **Wobber, Edward P.**
  Menlo Park, California 94025 (US)

(74) Representative: **Betten & Resch**
Reichenbachstrasse 19
80469 München (DE)

(54) Pseudo network adapter for frame capture, encapsulation and encryption

(57) A new pseudo network adapter provides an interface for capturing packets from a local communications protocol stack for transmission on the virtual private network, and includes a Dynamic Host Configuration Protocol (DHCP) server emulator, and an Address Resolution Protocol (ARP) server emulator. The new system indicates to the local communications protocol stack that nodes on a remote private network are reachable through a gateway that is in turn reachable through the pseudo network adapter. A transmit path in the system processes data packets from the local communications protocol stack for transmission through the pseudo network adapter. An encryption engine encrypts the data packets and an encapsulation engine encapsulates the encrypted data packets into tunnel data frames. The network adapter further includes an interface into a transport layer of the local communications protocol stack for capturing received data packets from the remote server node, and a receive path for processing received data packets captured from the transport layer of the local communications protocol stack. The receive path includes a decapsulation engine, and a decryption engine, and passes the decrypted, decapsulated data packets back to the local communications protocol stack for delivery to a user.

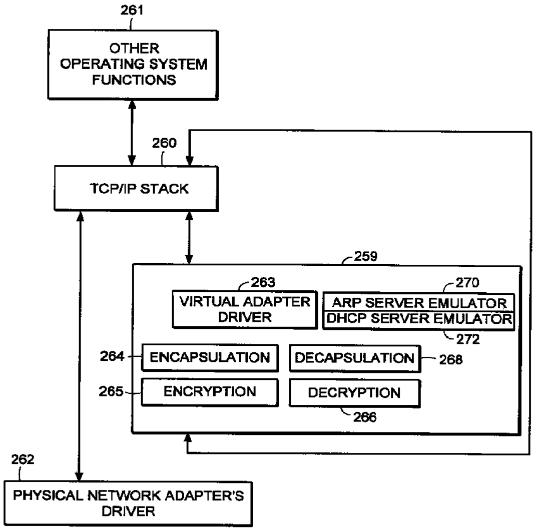


FIG. 15

Printed by Xerox (UK) Business Services
2.16.1/3.4

Description

FIELD OF THE INVENTION

The invention relates generally to establishing secure virtual private networks. The invention relates specifically to a pseudo network adapter for capturing, encapsulating and encrypting messages or frames.

BACKGROUND

In data communications it is often required that secure communications be provided between users of network stations (also referred to as "network nodes") at different physical locations. Secure communications must potentially extend over public networks as well as through secure private networks. Secure private networks are protected by "firewalls", which separate the private network from a public network. Firewalls ordinarily provide some combination of packet filtering, circuit gateway, and application gateway technology, insulating the private network from unwanted communications with the public network.

One approach to providing secure communications is to form a virtual private network. In a virtual private network, secure communications are provided by encapsulating and encrypting messages. Encapsulated messaging in general is referred to as "tunneling". Tunnels using encryption may provide protected communications between users separated by a public network, or among a subset of users of a private network.

Encryption may for example be performed using an encryption algorithm using one or more encryption "keys". When an encryption key is used, the value of the key determines how the data is encrypted and decrypted. When a public-key encryption system is used, a key pair is associated with each communicating entity. The key pair consists of an encryption key and a decryption key. The two keys are formed such that it is unfeasible to generate one key from the other. Each entity makes its encryption key public, while keeping its decryption key secret. When sending a message to node A, for example, the transmitting entity uses the public key of node A to encrypt the message, and then the message can only be decrypted by node A using node A's private key.

In a symmetric key encryption system a single key is used as the basis for both encryption and decryption. An encryption key in a symmetric key encryption system is sometimes referred to as a "shared" key. For example, a pair of communicating nodes A and B could communicate securely as follows: a first shared key is used to encrypt data sent from node A to node B, while a second shared key is to be used to encrypt data sent from node B to node A. In such a system, the two shared keys must be known by both node A and node B. More examples of encryption algorithms and keyed encryption are disclosed in many textbooks, for example "Applied Cryptography - Protocols, Algorithms, and Source Code in C", by Bruce Schneier, published by John Wiley and Sons, New York, New York, copyright 1994.

Information regarding what encryption key or keys are to be used, and how they are to be used to encrypt data for a given secure communications session is referred to as "key exchange material". Key exchange material may for example determine what keys are used and a time duration for which each key is valid. Key exchange material for a pair of communicating stations must be known by both stations before encrypted data can be exchanged in a secure communications session. How key exchange material is made known to the communicating stations for a given secure communications session is referred to as "session key establishment".

A tunnel may be implemented using a virtual or "pseudo" network adapter that appears to the communications protocol stack as a physical device and which provides a virtual private network. A pseudo network adapter must have the capability to receive packets from the communications protocol stack, and to pass received packets back through the protocol stack either to a user or to be transmitted.

A tunnel endpoint is the point at which any encryption/decryption and encapsulation/decapsulation provided by a tunnel is performed. In existing systems, the tunnel end points are pre-determined network layer addresses. The source network layer address in a received message is used to determine the "credentials" of an entity requesting establishment of a tunnel connection. For example, a tunnel server uses the source network layer address to determine whether a requested tunnel connection is authorized. The source network layer address is also used to determine which cryptographic key or keys to use to decrypt received messages.

Existing tunneling technology is typically performed by encapsulating encrypted network layer packets (also referred to as "frames") at the network layer. Such systems provide "network layer within network layer" encapsulation of encrypted messages. Tunnels in existing systems are typically between firewall nodes which have statically allocated IP addresses. In such existing systems, the statically allocated IP address of the firewall is the address of a tunnel end point within the firewall. Existing systems fail to provide a tunnel which can perform authorization based for an entity which must dynamically allocate its network layer address. This is especially problematic for a user wishing to establish a tunnel in a mobile computing environment, and who requests a dynamically allocated IP address from an Internet Service Provider (ISP).

Because existing virtual private networks are based on network layer within network layer encapsulation, they are generally only capable of providing connectionless datagram type services. Because datagram type services do not guarantee delivery of packets, existing tunnels can only easily employ encryption methods over the data contained within each transmitted packet. Encryption based on the contents of multiple packets is desirable, such as cipher block chaining or stream ciphering over multiple packets. For example, encrypted data would advantageously be formed based not only on the contents of the present packet data being encrypted, but also based on some attribute of the connection or session history between the communicating stations. Examples of encryption algorithms and keyed encryption are disclosed in many textbooks, for example "Applied Cryptography - Protocols, Algorithms, and Source Code in C", by Bruce Schneier, published by John Wiley and Sons, New York, New York, copyright 1994.

Thus there is required a new pseudo network adapter providing a virtual private network having a dynamically determined end point to support a user in a mobile computing environment. The new pseudo network adapter should appear to the communications protocol stack of the node as an interface to an actual physical device. The new pseudo network adapter should support guaranteed, in-order delivery of frames over a tunnel to conveniently support cipher block chaining mode or stream cipher encryption over multiple packets.

SUMMARY OF THE INVENTION

A new pseudo network adapter is disclosed providing a virtual private network. The new system includes an interface for capturing packets from a local communications protocol stack for transmission on the virtual private network. The interface appears to the local communications stack as a network adapter device driver for a network adapter.

The invention, in its broad form, includes a pseudo network adapter as recited in claim 1, providing a virtual network and a method therefor as recited in claim 9.

The system as described hereinafter further includes a Dynamic Host Configuration Protocol (DHCP) server emulator, and an Address Resolution Protocol (ARP) server emulator. The new system indicates to the local communications protocol stack that nodes on a remote private network are reachable through a gateway that is in turn reachable through the pseudo network adapter. The new pseudo network adapter includes a transmit path for processing data packets from the local communications protocol stack for transmission through the pseudo network adapter. The transmit path includes an encryption engine for encrypting the data packets and an encapsulation engine for encapsulating the encrypted data packets into tunnel data frames. The pseudo network adapter passes the tunnel data frames back to the local communications protocol stack for transmission to a physical network adapter on a remote server node.

Preferably, as described hereinafter, the pseudo network adapter includes a digest value in a digest field in each of the tunnel data frames. A keyed hash function is a hash function which takes data and a shared cryptographic key as inputs, and outputs a digital signature referred to as a digest. The value of the digest field is equal to an output of a keyed hash function applied to data consisting of the data packet encapsulated within the tunnel data frame concatenated with a counter value equal to a total number of tunnel data frames previously transmitted to the remote server node. In another aspect of the system, the pseudo network adapter processes an Ethernet header in each one of the captured data packets, including removing the Ethernet header.

The new pseudo network adapter further includes an interface into a transport layer of the local communications protocol stack for capturing received data packets from the remote server node, and a receive path for processing received data packets captured from the transport layer of the local communications protocol stack. The receive path includes a decapsulation engine, and a decryption engine, and passes the decrypted, decapsulated data packets back to the local communications protocol stack for delivery to a user.

Thus there is disclosed a new pseudo network adapter providing a virtual private network having dynamically determined end points to support users in a mobile computing environment. The new pseudo network adapter provides a system for capturing a fully formed frame prior to transmission. The new pseudo network adapter appears to the communications protocol stack of the station as an interface to an actual physical device. The new pseudo network adapter further includes encryption capabilities to conveniently provide secure communications between tunnel end points using stream mode encryption or cipher block chaining over multiple packets.

BRIEF DESCRIPTION OF THE DRAWINGS

A more detailed understanding of the invention may be had from the following description of a preferred embodiment, given by way of example and to be understood in conjunction with the accompanying drawing in which:

- Fig. 1 is a block diagram showing the Open Systems Interconnection (OSI) reference model;
- Fig. 2 is a block diagram showing the TCP/IP internet protocol suite;
- Fig. 3 is a block diagram showing an examplary embodiment of a tunnel connection across a public network between two tunnel servers;
- Fig. 4 is a flow chart showing an examplary embodiment of steps performed to establish a tunnel con-

nection;

- Fig. 5 is a flow chart showing an examplary embodiment of steps performed to perform session key management for a tunnel connection;
- Fig. 6 is a block diagram showing an examplary embodiment of a relay frame;
- Fig. 7 is a block diagram showing an examplary embodiment of a connection request frame;
- Fig. 8 is a block diagram showing an examplary embodiment of a connection response frame;
- Fig. 9 is a block diagram showing an examplary embodiment of a data frame;
- Fig. 10 is a block diagram showing an examplary embodiment of a close connection frame;
- Fig. 11 is a state diagram showing an examplary embodiment of a state machine forming a tunnel connection in a network node initiating a tunnel connection;
- Fig. 12 is a state diagram showing an examplary embodiment of a state machine forming a tunnel connection in a server computer;
- Fig. 13 is a state diagram showing an examplary embodiment of a state machine forming a tunnel connection in a relay node;
- Fig. 14 is a block diagram showing an examplary embodiment of a tunnel connection between a client computer (tunnel client) and a server computer (tunnel server);
- Fig. 15 is a block diagram showing an examplary embodiment of a pseudo network adapter;
- Fig. 16 is a block diagram showing an examplary embodiment of a pseudo network adapter;
- Fig. 17 is a flow chart showing steps performed by an examplary embodiment of a pseudo network adapter during packet transmission;
- Fig. 18 is a flow chart showing steps performed by an examplary embodiment of a pseudo network adapter during packet receipt;
- Fig. 19 is a data flow diagram showing data flow in an examplary embodiment of a pseudo network adapter during packet transmission;
- Fig. 20 is a data flow diagram showing data flow in an examplary embodiment of a pseudo network adapter during packet receipt;
- Fig. 21 is a diagram showing the movement of encrypted and unencrypted data in an examplary embodiment of a system including a pseudo network adapter;
- Fig. 22 is a diagram showing the movement of encrypted and unencrypted data in an examplary embodiment of a system including a pseudo network adapter; and
- Fig. 23 is a flow chart showing steps initialization of an examplary embodiment of a system including a pseudo network adapter.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

Now with reference to Fig. 1 there is described for purposes of explanation, communications based on the Open Systems Interconnection (OSI) reference model. In Fig. 1 there is shown communications 12 between a first protocol stack 10 and a second protocol stack 14. The first protocol stack 10 and second protocol stack 14 are implementations of the seven protocol layers (Application layer, Presentation layer, Session layer, Transport layer, Network layer, Data link layer, and Physical layer) of the OSI reference model. A protocol stack implementation is typically in some combination of software and hardware. Descriptions of the specific services provided by each protocol layer in the OSI reference model are found in many text books, for example "Computer Networks", Second Edition, by Andrew S. Tannenbaum, published by Prentice-Hall, Englewood Cliffs, New Jersey, copyright 1988.

As shown in Fig. 1, data 11 to be transmitted from a sending process 13 to a receiving process 15 is passed down through the protocol stack 10 of the sending process to the physical layer 9 for transmission on the data path 7 to the receiving process 15. As the data 11 is passed down through the protocol stack 10, each protocol layer prepends a header (and possibly also appends a trailer) portion to convey information used by that protocol layer. For example, the data link layer 16 of the sending process wraps the information received from the network layer 17 in a data link header 18 and a data link layer trailer 20 before the message is passed to the physical layer 9 for transmission on the actual transmission path 7.

Fig. 2 shows the TCP/IP protocol stack. Some protocol layers in the TCP/IP protocol stack correspond with layers in the OSI protocol stack shown in Fig. 1. The detailed services and header formats of each layer in the TCP/IP protocol stack are described in many texts, for example "Internetworking with TCP/IP, Vol. 1: Principles, Protocols, and Architecture", Second Edition, by Douglas E. Comer, published by Prentice-Hall, Englewood Cliffs, New Jersey, copyright 1991. The Transport Control Protocol (TCP) 22 corresponds to the Transport layer in the OSI reference model. The TCP protocol 22 provides a connection-oriented, end to end transport service with guaranteed, in-sequence packet delivery. In this way the TCP protocol 22 provides a reliable, transport layer connection.

The IP protocol 26 corresponds to the Network layer of the OSI reference model. The IP protocol 26 provides no guarantee of packet delivery to the upper layers. The hardware link level and access protocols 32 correspond to the Data link and Physical layers of the OSI reference model.

The Address Resolution Protocol (ARP) 28 is used to map IP layer addresses (referred to as "IP addresses") to addresses used by the hardware link level and access protocols 32 (referred to as "physical addresses" or "MAC addresses"). The ARP protocol layer in each network station typically contains a table of mappings between IP addresses and physical addresses (referred to as the "ARP cache"). When a mapping between an IP address and the corresponding physical address is not known, the ARP protocol 28 issues a broadcast packet (an "ARP request" packet) on the local network. The ARP request indicates an IP address for which a physical address is being requested. The ARP protocols 28 in each station connected to the local network examine the ARP request, and if a station recognizes the IP address indicated by the ARP request, it issues a response (an "ARP response" or "ARP reply" packet) to the requesting station indicating the responder's physical address. The requesting ARP protocol reports the received physical address to the local IP layer which then uses it to send datagrams directly to the responding station. As an alternative to having each station respond only for its own IP address, an ARP server may be used to respond for a set of IP addresses it stores internally, thus potentially eliminating the requirement of a broadcast request. In that case, the ARP request can be sent directly to the ARP server for physical addresses corresponding to any IP address mappings stored within the ARP server.

At system start up, each station on a network must determine an IP address for each of its network interfaces before it can communicate using TCP/IP. For example, a station may need to contact a server to dynamically obtain an IP address for one or more of its network interfaces. The station may use what is referred to as the Dynamic Host Configuration Protocol (DHCP) to issue a request for an IP address to a DHCP server. For example, a DHCP module broadcasts a DHCP request packet at system start up requesting allocation of an IP address for an indicated network interface. Upon receiving the DHCP request packet, the DHCP server allocates an IP address to the requesting station for use with the indicated network interface. The requesting station then stores the IP address in the response from the server as the IP address to associate with that network interface when communicating using TCP/IP.

Fig. 3 shows an example configuration of network nodes for which the presently disclosed system is applicable. In the example of Fig. 3, the tunnel server A is an initiator of the tunnel connection. As shown in Fig. 3, the term "tunnel relay" node is used to refer to a station which forwards data packets between transport layer connections (for example TCP connections).

For example, in the present system a tunnel relay may be dynamically configured to forward packets between transport layer connection 1 and transport layer connection 2. The tunnel relay replaces the header information of packets received over transport layer connection 1 with header information indicating transport layer connection 2. The tunnel relay can then forward the packet to a firewall, which may be conveniently programmed to pass packets received over transport layer connection 2 into a private network on the other side of the firewall. In the present system, the tunnel relay dynamically forms transport layer connections when a tunnel connection is established. Accordingly the tunnel relay is capable of performing dynamic load balancing or providing redundant service for fault tolerance over one or more tunnel servers at the time the tunnel connection is established.

Fig. 3 shows a Tunnel Server A 46 in a private network N1 48, physically connected with a first Firewall 50. The first Firewall 50 separates the private network N1 48 from a public network 52, for example the Internet. The first Firewall 50 is for example physically connected with a Tunnel Relay B 54, which in turn is virtually connected through the public network 52 with a Tunnel Relay C. The connection between Tunnel Relay B and Tunnel Relay C may for example span multiple intervening forwarding nodes such as routers or gateways through the public network 52.

The Tunnel Relay C is physically connected with a second Firewall 58, which separates the public network 52 from a private network N2 60. The second Firewall 58 is physically connected with a Tunnel Server D 62 on the private network N2 60. During operation of the elements shown in Fig. 3, the Tunnel Server D 62 provides routing of IP packets between the tunnel connection with Tunnel Server A 46 and other stations on the private network N2 60. In this way the Tunnel Server D 62 acts as a router between the tunnel connection and the private network N2 60.

During operation of the elements shown in Fig. 3, the present system establishes a tunnel connection between the private network N1 48 and the private network N2 60. The embodiment of Fig. 3 thus eliminates the need for a dedicated physical cable or line to provide secure communications between the private network 48 and the private network 60. The tunnel connection between Tunnel Server A 46 and Tunnel Server D 62 is composed of reliable, pair-wise transport layer connections between Tunnel Server A 46 (node "A"), Tunnel Relay B 54 (node "B"), Tunnel Relay C 56 (node "C"), and Tunnel Server D 62 (node "D"). For example, such pair-wise connections may be individual transport layer connections between each node A and node B, node B and node C, and node C and node D. In an alternative embodiment, as will be described below, a tunnel connection may alternatively be formed between a standalone PC in a public network and a tunnel server within a private network.

Fig. 4 and Fig. 5 show an example embodiment of steps performed during establishment of the tunnel connection between Tunnel Server A 46 (node "A") and Tunnel Server D 62 (node "D") as shown in Fig. 3. Prior to the steps shown in Fig. 4, node A selects a tunnel path to reach node D. The tunnel path includes the tunnel end points and any intervening tunnel relays. The tunnel path is for example predetermined by a system administrator for node A. Each tunnel relay along the tunnel path is capable of finding a next node in the tunnel path, for example based on a provided next node name (or "next node arc"), using a predetermined naming convention and service, for example the Domain Name System (DNS) of the TCP/IP protocol suite.

During the steps shown in Fig. 4, each of the nodes A, B and C perform the following steps:

- resolve the node name of the next node in the tunnel path, for example as found in a tunnel relay frame;
- establish a reliable transport layer (TCP) connection to the next node in the tunnel path;
- forward the tunnel relay frame down the newly formed reliable transport layer connection to the next node in the tunnel path.

As shown for example in Fig. 4, at step 70 node A establishes a reliable transport layer connection with node B. At step 72 node A identifies the next downstream node to node B by sending node B a tunnel relay frame over the reliable transport layer connection between node A and node B. The tunnel relay frame contains a string buffer describing all the nodes along the tunnel path (see below description of an example tunnel relay frame format). At step 74, responsive to the tunnel relay frame from node A, node B searches the string buffer in the relay frame to determine if the string buffer includes node B's node name. If node B finds its node name in the string buffer, it looks at the next node name in the string buffer to find the node name of the next node in the tunnel path.

Node B establishes a reliable transport layer connection with the next node in the tunnel path, for example node C. Node B further forms an association between the reliable transport layer connection between Node A and Node B, over which the relay frame was received, and the newly formed reliable transport layer connection between Node B and Node C, and as a result forwards subsequent packets received over the reliable transport layer connection with Node A onto the reliable transport layer connection with Node C, and vice versa. At step 76 node B forwards the tunnel relay frame on the newly formed reliable transport layer connection to node C.

At step 78, responsive to the relay frame forwarded from node B, node C determines that the next node in the tunnel path is the last node in the tunnel path, and accordingly is a tunnel server. Node C may actively determine whether alternative tunnel servers are available to form the tunnel connection. Node C may select one of the alternative available tunnel servers to form the tunnel connection in order to provide load balancing or fault tolerance. As a result node C may form a transport layer connections with one of several available tunnel servers, for example a tunnel server that is relatively underutilized at the time the tunnel connection is established. In the example embodiment, node C establishes a reliable transport layer connection with the next node along the tunnel path, in this case node D.

Node C further forms an association between the reliable transport layer connection between Node B and Node C, over which the relay frame was received, and the newly formed reliable transport layer connection between Node C and Node D, and as a result forwards subsequent packets received over the reliable transport layer connection with Node B to the reliable transport layer connection with Node D, and vice versa. At step 80 node C forwards the relay frame to node D on the newly formed reliable transport layer connection.

Fig. 5 shows an example of tunnel end point authentication and sharing of key exchange material provided by the present system. The present system supports passing authentication data and key exchange material through the reliable transport layer connections previously established on the tunnel path. The following are provided by use of a key exchange/authentication REQUEST frame and a key exchange/authentication RESPONSE frame:

a) mutual authentication of both endpoints of the tunnel connection;

b) establishment of shared session encryption keys and key lifetimes for encrypting/authenticating subsequent data sent through the tunnel connection;

d) agreement on a shared set of cryptographic transforms to be applied to subsequent data; and

e) exchange of any other connection-specific data between the tunnel endpoints, for example strength and type of cipher to be used, any compression of the data to be used, etc. This data can also be used by clients of this protocol to qualify the nature of the authenticated connection.

At step 90 a key exchange/authentication request frame is forwarded over the reliable transport layer connections formed along the tunnel path from node A to node D. At step 92, a key exchange/authentication response frame is forwarded from node D back to node A through the reliable transport layer connections. The attributes exchanged using the steps shown in Fig. 5 may be used for the lifetime of the tunnel connection. In an alternative embodiment the steps shown in Fig. 5 are repeated as needed for the tunnel end points to exchange sufficient key exchange material to agree upon a set of session parameters for use during the tunnel connection such as cryptographic keys, key durations, and choice of encryption/decryption algorithms.

Further in the disclosed system, the names used for authentication and access control with regard to node A and node D need not be the network layer address or physical address of the nodes. For example, in an alternative embodiment where the initiating node sending the tunnel relay frame is a stand-alone PC located within a public network, the user's name may be used for authentication and/or access control purposes. This provides a significant improvement over existing systems which base authorization on predetermined IP addresses.

Fig. 6 shows the format of an example embodiment of a tunnel relay frame. The tunnel frame formats shown in Figs. 6, 7, 8 and 9 are encapsulated within the data portion of a transport layer (TCP) frame when transmitted. Alternatively, another equivalent, connection-oriented transport layer protocol having guaranteed, in-sequence frame delivery may be used. The example TCP frame format, including TCP header fields, is conventional and not shown.

The field 100 contains a length of the frame. The field 102 contains a type of the frame, for example a type of RELAY. The field 104 contains a tunnel protocol version number. The field 106 contains an index into a string buffer field 112 at which a name of the originating node is located, for example a DNS host name of the node initially issuing the relay frame (node A in Fig. 3). The fields following the origin index field 106 contain indexes into the string buffer 112 at which names of nodes along the tunnel path are located. For example each index may be the offset of a DNS host name within the string buffer 112. In this way the field 108 contains the index of the name of the first node in the tunnel path, for example node B (Fig. 3). The field 110 contains the index of the name of the second node in the tunnel path, etc. The field 112 contains a string of node names of nodes in the tunnel path.

During operation of the present system, the initiating node, for example node A as shown in Fig. 3, transmits a tunnel relay frame such as the tunnel relay frame shown in Fig. 6. Node A sends the tunnel relay frame to the first station along the tunnel path, for example node B (Fig. 3), over a previously established reliable transport layer connection. Node B searches the string buffer in the tunnel relay frame to find its node name, for example its DNS host name. Node B finds its node name in the string buffer indexed by path index 0, and then uses the contents of path index 1 110 to determine the location within the string buffer 112 of the node name of the next node along the tunnel path. Node B uses this node name to establish a reliable transport layer connection with the next node along the tunnel path. Node B then forwards the relay frame to the next node. This process continues until the end node of the tunnel route, for example tunnel server D 62 (Fig. 3) is reached.

Fig. 7 shows the format of an example embodiment of a key exchange/key authentication request frame. The field 120 contains a length of the frame. The field 122 contains a type of the frame, for example a type of REQUEST indicating a key exchange/key authentication request frame. The field 124 contains a tunnel protocol version number. The field 126 contains an offset of the name of the entity initiating the tunnel connection, for example the name of a user on the node originally issuing the request frame. This name and key exchange material in the request frame are used by the receiving tunnel end point to authenticate the key exchange/authentication REQUEST. The name of the entity initiating the tunnel connection is also use to authorize any subsequent tunnel connection, based on predetermined security policies of the system. The field 128 contains an offset into the frame of the node name of the destination node, for example the end node of the tunnel shown as node D 62 in Fig. 3.

The field 130 contains an offset into the frame at which key exchange data as is stored, for example within the string buffer field 138. The key exchange data for example includes key exchange material used to determine a shared set of encryption parameters for the life of the tunnel connection such as cryptographic keys and any validity times associated with those keys. The key exchange data, as well as the field 132, further include information regarding any shared set of cryptographic transforms to be used and any other connection-specific parameters, such as strength and type of cipher to be used, type of compression of the data to be used, etc. The field 134 contains flags, for example indicating further information about the frame. The field 136 contains client data used in the tunnel end points to configure the local routing tables so that packets for nodes reachable through the virtual private network are sent through the pseudo network adapters. In an example embodiment, the string buffer 138 is encrypted using a public encryption key of the receiving tunnel end point.

During operation of the present system, one of the end nodes of the tunnel sends a key exchange/authentication REQUEST frame as shown in Fig. 7 to the other end node of the tunnel in order to perform key exchange and authentication as described in step 90 of Fig. 5.

Fig. 8 shows the format of an example embodiment of a key exchange/key authentication response frame, referred to as a connection RESPONSE frame. The field 150 contains a length of the frame. The field 152 contains a type of the frame, for example a type of connection RESPONSE indicating a key exchange/key authentication request frame. The field 154 contains a tunnel protocol version number.

The field 156 contains an offset into the frame at which key exchange data as is stored, for example within the string buffer field 163. The key exchange data for example includes key exchange material to be used for encryption/decryption over the life of the tunnel connection and any validity times associated with that key exchange material. The key exchange data, as well as the field 158, further includes information regarding any shared set of cryptographic transforms to be applied to subsequent data and any other connection-specific parameters, such as strength and type of cipher to be used, any compression of the data to be used, etc. The field 160 contains flags, for example indicating other information about the frame. The client data field 162 contains data used by the pseudo network adapters in the tunnel end points to configure the local routing tables so that packets for nodes in the virtual private network are sent through the pseudo network adapters. The string buffer includes key exchange material. The string buffer is for example encrypted using a public encryption key of the receiving tunnel end point, in the this case the initiator of the tunnel connection.

During operation of the present system, one of the end nodes of the tunnel sends a key exchange/authentication RESPONSE frame as shown in Fig. 7 to the other end node of the tunnel in order to perform key exchange and authentication as described in step 92 of Fig. 5.

Fig. 9 shows the format of an example embodiment of an tunnel data frame used to communicate through a tunnel connection. Fig. 9 shows how an IP datagram may be encapsulated within a tunnel frame by the present system for secure communications through a virtual private network. The field 170 contains a length of the frame. The field 172 contains a type of the frame, for example a type of DATA indicating a tunnel data frame. The field 174 contains a tunnel protocol version number.

The fields 176, 178 and 182 contain information regarding the encapsulated datagram. The field 180 contains flags indicating information regarding the frame. The field 184 contains a value indicating the length of the optional padding 189 at the end of the frame. The frame format allows for optional padding in the event that the amount of data in the frame needs to be padded to an even block boundary for the purpose of being encrypted using a block cipher. The field 186 contains a value indicating the length of the digest field 187.

The data frame format includes a digital signature generated by the transmitting tunnel end point referred to as a "digest". The value of the digest ensures data integrity, for example by detecting invalid frames and replays of previously transmitted valid frames. The digest is the output of a conventional keyed cryptographic hash function applied to both the encapsulated datagram 190 and a monotonically increasing sequence number. The resulting hash output is passed as the value of the digest field 187. The sequence number is not included in the data frame. In the example embodiment, the sequence number is a counter maintained by the transmitter (for example node A in Fig. 3) of all data frames sent to the receiving node (for example node D in Fig. 3) since establishment of the tunnel connection.

In order to determine if the data frame is invalid or a duplicate, the receiving node decrypts the encapsulated datagram 190, and applies the keyed cryptographic hash function (agreed to by the tunnel end nodes during the steps shown in Fig. 5) to both the decrypted encapsulated datagram and the value of a counter indicating the number of data frames received from the transmitter since establishment of the tunnel connection. For example the keyed hash function is applied to the datagram concatenated to the counter value. If the resulting hash output matches the value of the digest field 187, then the encapsulated datagram 190 was received correctly and is not a duplicate. If the hash output does not match the value of the digest field 187, then the integrity check fails, and the tunnel connection is closed. The field 188 contains an encrypted network layer datagram, for example an encrypted IP datagram.

The encapsulated datagram may be encrypted using various encryption techniques. An example embodiment of the present system advantageously encrypts the datagram 190 using either a stream cipher or cipher block chaining encryption over all data transmitted during the life of the tunnel connection. This is enabled by the reliable nature of the transport layer connections within the tunnel connection. The specific type of encryption and any connection specific symmetric encryption keys used is determined using the steps shown in Fig. 5. The fields in the tunnel data frame other than the encapsulated datagram 188 are referred to as the tunnel data frame header fields.

Fig. 10 is a block diagram showing an example embodiment of a "close connection" frame. The field 190 contains the length of the frame. The field 191 contains a frame type, for example having a value equal to CLOSE. Field 192 contains a value equal to the current protocol version number of the tunnel protocol. The field 193 contains a status code indicating the reason the tunnel connection is being closed.

During operation of the present system, when end point of a tunnel connection determines that the tunnel connection should be closed, a close connection frame as shown in Fig. 10 is transmitted to the other end point of the tunnel connection. When a close connection close frame is received, the receiver closes the tunnel connection and no further data will be transmitted or received through the tunnel connection.

Fig. 11 is a state diagram showing an example embodiment of forming a tunnel connection in a node initiating a tunnel connection. In Fig. 11, Fig. 12, and Fig. 13, states are indicated by ovals and actions or events are indicated by rectangles. For example the tunnel server node A as shown in Fig. 3 may act as a tunnel connection initiator when establishing a tunnel connection with the tunnel server node D. Similarly the client system 247 in Fig. 14 may act as a tunnel connection initiator when establishing a tunnel connection with the tunnel server. The tunnel initiator begins in an idle state 194. Responsive to an input from a user indicating that a tunnel connection should be established, the tunnel initiator transitions from the idle state 194 to a TCP Open state 195. In the TCP Open state 195, the tunnel initiator establishes a reliable transport layer connection with a first node along the tunnel path. For example, the tunnel initiator opens a socket interface associated with a TCP connection to the first node along the tunnel path. In Fig. 3 node A opens a socket interface associated with a TCP connection with node B.

Following establishment of the reliable transport layer connection in the TCP Open state 195, the tunnel initiator enters a Send Relay state 197. In the Send Relay state 197, the tunnel initiator transmits a relay frame at 198 over the reliable transport layer connection. Following transmission of the relay frame, the tunnel initiator enters the connect state 199. If during transmission of the relay frame there is a transmission error, the tunnel initiator enters the Network Error state 215 followed by the Dying state 208. In the Dying state 208, the tunnel initiator disconnects the reliable transport layer connection formed in the TCP Open state 195, for example by disconnecting a TCP connection with Node B. Following the disconnection at 209, the tunnel initiator enters the Dead state 210. The tunnel initiator subsequently transitions back to the Idle state 194 at a point in time predetermined by system security configuration parameters.

In the Connect state 199, the tunnel initiator sends a key exchange/authentication REQUEST frame at 200 to the tunnel server. Following transmission of the key exchange/authentication REQUEST frame 200, the tunnel initiator enters the Response Wait state 201. The tunnel initiator remains in the Response Wait state 201 until it receives a key exchange/authentication RESPONSE frame 202 from the tunnel server. After the key exchange/authentication RESPONSE frame is received at 202, the tunnel initiator enters the Authorized state 203, in which it may send or receive tunnel data frames. Upon receipt of a CLOSE connection frame at 216 in the Authorized state 203, the tunnel initiator transitions to the Dying state 208.

Upon expiration of a session encryption key at 211, the tunnel initiator enters the Reconnect state 212, and sends a CLOSE connection frame at 213 and disconnects the TCP connection with the first node along the tunnel path at 214. Subsequently the tunnel initiator enters the TCP Open state 195.

If during the authorized state 203, a local user issues an End Session command at 204, or there is a detection of an authentication or cryptography error in a received data frame at 205, the tunnel initiator enters the Close state 206. During the Close state 206 the tunnel initiator sends a CLOSE connection frame at 207 to the tunnel server. The tunnel initiator then enters the Dying state at 208.

Figure 12 is a state diagram showing the states within an example embodiment of a tunnel server, for example node D in Fig. 3 or tunnel server 253 in Fig. 14. The tunnel server begins in an Accept Wait state 217. In the Accept Wait state 217, the tunnel server receives a request for a reliable transport layer connection, for example a TCP connection reguest 218 from the last node in the tunnel path prior to the tunnel server, for example Node C in Fig. 3. In response to a TCP connection request 218 the tunnel server accepts the request and establishes a socket interface associated with the resulting TCP connection with Node C.

Upon establishment of the TCP connection with the last node in the tunnel path prior to the tunnel server, the tunnel server enters the Receive Relay state 219. In the Receive Relay state 219, the tunnel server waits to receive a relay frame at 220, at which time the tunnel server enters the Connect Wait state 221. If there is some sort of network error 234 during receipt of the relay frame at 219, the tunnel server enters the Dying state 230. During the Dying state 230 the tunnel server disconnects at 231 the transport layer connection with the last node in the tunnel path prior to the tunnel server. After disconnecting the connection, the tunnel server enters the Dead state 232.

In the Connect Wait state 221, the tunnel server waits for receipt of a key exchange/authentication REQUEST frame at 222. Following receipt of the key exchange/authentication REQUEST frame at 222, the tunnel server determines whether the requested tunnel connection is authorized at step 223. The determination of whether the tunnel connection is authorized is based on a name of the tunnel initiator, and the key exchange material within the key exchange/authentication REQUEST frame.

If the requested tunnel connection is authorized the tunnel server sends a key exchange/authentication RESPONSE frame at 224 back to the tunnel initiator. If the requested tunnel connection is not authorized, the tunnel server enters the Close state 228, in which it sends a close connection frame at 229 to the tunnel client. Following transmission of the CLOSE connection frame at 229, the tunnel server enters the Dying state 230.

If the requested tunnel connection is determined to be authorized at step 223, the tunnel server enters the Authorized state 225. In the Authorized state, the tunnel server transmits and receives tunnel data frames between itself and the tunnel initiator. If during the Authorized state 225, the tunnel server receives a CLOSE connection frame at 233, the tunnel server transitions to the Dying state 230. If during the authorized state 225, the tunnel server receives an end session command from a user at 226, then the tunnel server transitions to the Close state 228, and transmits a close connection frame at 229 to the tunnel initiator. If the tunnel server in the Authorized state 225 detects an integrity failure in a received packet, the tunnel server transitions to the Close state 228. In the close state 228 the tunnel server sends a CLOSE connection frame at 229 and subsequently enters the Dying state 230.

Fig. 13 is a state diagram showing an example embodiment of a state machine within a tunnel relay node. The tunnel relay node begins in an Accept Wait state 235. When a request is received to form a reliable transport layer connection at 236, a reliable transport layer connection is accepted with the requesting node. For example, a TCP connection is accepted between the relay node and the preceding node in the tunnel path.

The relay node then transitions to the Receive Relay state 237. During the Receive Relay state 237, the relay node receives a relay frame at 238. Following receipt of the relay frame at 238, the relay node determines what forwarding address should be used to forward frames received from the TCP connection established responsive to the TCP connect event 236. If the next node in the tunnel path is a tunnel server, the forwarding address may be selected at 239 so as to choose an underutilized tunnel server from a group of available tunnel servers or to choose an operational server where others are not operational.

Following determination of the forwarding address or addresses in step 239, the relay node enters the Forward Connect state 240. In the Forward Connect state 240, the relay node establishes a reliable transport layer connection with the node or nodes indicated by the forwarding address or addresses determined in step 239.

Following establishment of the new connection at event 241, the tunnel relay enters the Forward state 242. During the Forward state 242, the relay node forwards all frames between the connection established at 236 and those connections established at 241. Upon detection of a network error or receipt of a frame indicating a closure of the tunnel connection at 243, the tunnel relay enters the Dying state 244. Following the Dying state 244, the relay node disconnects any connections established at event 241. The relay node then enters the Dead state 246.

Fig. 14 shows an example embodiment of a virtual private network 249 formed by a pseudo network adapter 248 and a tunnel connection between a tunnel client 247 and a tunnel server 253 across a public network 251. The tunnel server 253 and tunnel client 247 are for example network stations including a CPU or microprocessor, memory, and various I/O devices. The tunnel server 253 is shown physically connected to a private LAN 256 including a Network Node 1 257 and a Network Node 2 258, through a physical network adapter 254. The tunnel server 253 is further shown physically connected with a firewall 252 which separates the private LAN 256 from the public network 251. The firewall 252 is physically connected with the public network 251. The tunnel server 253 is further shown including a pseudo network adapter 255. The client system 247 is shown including a physical network adapter 250 physically connected to the public network 251.

During operation of the elements shown in Fig. 14, nodes within the virtual private network 249 appear to the tunnel client 247 as if they were physically connected to the client system through the pseudo network adapter 248. Data transmissions between the tunnel client and any nodes that appear to be within the virtual private network are passed through the pseudo network adapter 248. Data transmissions between the tunnel client 247 and the tunnel server 253 are physically accomplished using a tunnel connection between the tunnel client 247 and the tunnel server 253.

Fig. 15 shows elements in an example embodiment of a pseudo network adapter such as the pseudo network adapter 248 in Fig. 14. In an example embodiment the elements shown in Fig. 15 are implemented as software executing on the tunnel client 247 as shown in Fig. 14. In Fig. 15 there is shown a pseudo network adapter 259 including a virtual adapter driver interface 263, an encapsulation engine 264, an encryption engine 265, a decapsulation engine 268, and a decryption engine 266. Further shown in the pseudo network adapter 259 are an ARP server emulator 270 and a Dynamic Host Configuration Protocol (DHCP) server emulator.

The pseudo network adapter 259 is shown interfaced to a TCP/IP protocol stack 260, through the virtual adapter driver interface 260. The TCP/IP protocol stack 260 is shown interfaced to other services in an operating system 261, as well as a physical network adapter's driver 262. The physical network adapter's driver 262 is for example a device driver which controls the operation of a physical network adapter such as physical network adapter 250 as shown in Fig. 14.

During operation of the elements shown in Fig. 15, the pseudo network adapter 259 registers with the network layer in the TCP/IP stack 260 that it is able to reach the IP addresses of nodes within the virtual private network 249 as shown in Fig. 14. For example, the pseudo network adapter on the client system registers that it can reach the pseudo network adapter on the server. Subsequently, a message from the tunnel client addressed to a node reachable through the virtual private network will be passed by the TCP/IP stack to the pseudo network adapter 259. The pseudo network adapter 259 then encrypts the message, and encapsulates the message into a tunnel data frame. The pseudo network adapter 259 then passes the tunnel data frame back to the TCP/IP protocol stack 260 to be sent through to the physical network adapter in the tunnel server. The tunnel server passes the received data frame to the pseudo network adapter in the server, which de-encapsulates and decrypts the message.

Fig. 16 shows a more detailed example embodiment of a pseudo network adapter 280. The pseudo network adapter 280 includes a virtual network adapter driver interface 288. The transmit path 290 includes an encryption engine 292, and an encapsulation engine 294. The encapsulation engine 294 is interfaced with a TCP/IP transmit interface 312 within a TCP/IP protocol stack, for example a socket interface associated with the first relay node in the tunnel path, or with the remote tunnel end point if the tunnel path includes no relays.

In the example embodiment of Fig. 16, the pseudo network adapter 280 appears to the TCP/IP protocol stack 282 as an Ethernet adapter. Accordingly, ethernet packets 286 for a destination addresses understood by the TCP/IP protocol stack to be reachable through the virtual private network are passed from the TCP/IP protocol stack 282 to the virtual network adapter interface 288 and through the transmit path 290. Similarly, ethernet packets 284 received through the pseudo network adapter 280 are passed from the receive path 296 to the virtual network adapter interface 288 and on to the TCP/IP protocol stack 282.

Further shown in the pseudo network adapter 280 of Fig. 16 is a receive path 296 having a decryption engine 298 interfaced to the virtual network adapter interface 288 and a decapsulation engine 300. The decapsulation engine 300 in turn is interfaced to a TCP/IP receive function 314 in the TCP/IP protocol stack 282, for example a socket interface associated with the first relay in the tunnel path, or with the remote tunnel end point if the tunnel path includes no relays. The pseudo network adapter 280 further includes an ARP server emulator 304 and a DHCP server emulator 306. ARP and DHCP request packets 302 are passed to the ARP server emulator 304 and DHCP server emulator 306 respectively. When a received packet is passed from the receive path 296 to the TCP/IP stack 282, a receive event must be indicated to the TCP/IP stack 282, for example through an interface such the Network Device Interface Specification (NDIS), defined by Microsoft™ Corporation.

Also in Fig. 16 is shown is an operating system 310 coupled with the TCP/IP protocol stack 282. The TCP/IP protocol stack 282 is generally considered to be a component part of the operating system. The operating system 310 in Fig. 16 is accordingly the remaining operating system functions and procedures outside the TCP/IP protocol stack 282. A physical network adapter 308 is further shown operated by the TCP/IP protocol stack 282.

During operation of the elements shown in Fig. 16, a user passes data for transmission to the TCP/IP protocol stack 282, and indicates the IP address of the node to which the message is to be transmitted, for example through a socket interface to the TCP layer. The TCP/IP protocol stack 282 then determines whether the destination node is reachable through the virtual private network. If the message is for a node that is reachable through the virtual private network, the TCP/IP protocol stack 282 an ethernet packet 286 corresponding to the message to the pseudo network adapter 280. The pseudo network adapter 280 then passes the ethernet packet 286 through the transmit path, in which the ethernet packet is encrypted and encapsulated into a tunnel data frame. The tunnel data frame is passed back into the TCP/IP protocol stack 282 through the TCP/IP transmit function 312 to be transmitted to the tunnel server through the tunnel connection. In an example embodiment, a digest value is calculated for the tunnel data frame before encryption within the transmit path within the pseudo network adapter.

Further during operation of the elements shown in Fig. 16, when the TCP/IP protocol stack 282 receives a packet from the remote endpoint of the TCP/IP tunnel connection, for example the tunnel server, the packet is passed to the pseudo network adapter 280 responsive to a TCP receive event. The pseudo network adapter 280 then decapsulates the packet by removing the tunnel header. The pseudo network adapter further decrypts the decapsulated data and passes it back to the TCP/IP protocol stack 282. The data passed from the pseudo network adapter 280 appears to the TCP/IP protocol stack 282 as an ethernet packet received from an actual physical device, and is the data it contains is passed on to the appropriate user by the TCP/IP protocol stack 282 based on information in the ethernet packet header provided by the pseudo network adapter.

Fig. 17 is a flow chart showing steps performed by an example embodiment of a pseudo network adapter during packet transmission, such as in the transmit path 290 of Fig. 14. The TCP/IP protocol stack determines that the destination node of a packet to be transmitted is reachable through the virtual LAN based on the destination IP address of the packet and a network layer routing table. At step 320 the packet is passed to the pseudo network adapter from the TCP/IP protocol stack. As a result, a send routine in the pseudo adapter is triggered for example in the virtual network adapter interface 288 of Fig. 16.

At step 322 the pseudo network adapter send routine processes the Ethernet header of the packet provided by the TCP/IP stack, and removes it. At step 324, the send routine determines whether the packet is an ARP request packet. If the packet is an ARP request packet for an IP address of a node on the virtual LAN, such as the pseudo network adapter of the tunnel server, then step 324 is followed by step 326. Otherwise, step 324 is followed by step 330.

At step 326, the ARP server emulator in the pseudo network adapter generates an ARP reply packet. For example, if the ARP request were for a physical address corresponding to the IP address of the pseudo network adapter on the tunnel server, the ARP reply would indicate a predetermined, reserved physical address to be associated with that IP address. At step 328 the pseudo network adapter passes the ARP response to the virtual network adapter interface. The virtual network adapter interface then indicates a received packet to the TCP/IP protocol stack, for example using an NDIS interface. The TCP/IP protocol stack then processes the ARP response as if it had been received over an actual physical network.

At step 330 the send routine determines whether the packet is a DHCP request packet requesting an IP address for the pseudo network adapter. If so, then step 330 is followed by step 332. Otherwise, step 330 is followed by step 334.

At step 334, the DHCP server emulator in the pseudo network adapter generates a DHCP response. The format of DHCP is generally described in the DHCP RFC. At step 328 the pseudo network adapter passes the DHCP response to the virtual network adapter interface, for example indicating an IP address received from the tunnel server in the client data field of the key exchange/authentication RESPONSE frame. The virtual network adapter interface then indicates a received packet to the TCP/IP protocol stack. The TCP/IP protocol stack then processes the DHCP response as if it had been received over an actual physical network.

At step 334 the pseudo network adapter encrypts the message using an encryption engine such that only the receiver is capable of decrypting and reading the message. At step 336 the pseudo network adapter encapsulates the encrypted message into a tunnel data frame. At step 338 the pseudo network adapter transmits the tunnel data frame through the tunnel connection using the TCP/IP protocol stack.

Fig. 18 is a flow chart showing steps performed by an example embodiment of a pseudo network adapter during packet receipt, such as in the receive path 296 of Fig. 14.

At step 350, the pseudo network adapter is notified that a packet has been received over the tunnel connection. At step 352 the pseudo network adapter decapsulates the received message by removing the header fields of the tunnel data frame. At step 354 the pseudo network adapter decrypts the decapsulated datagram from the tunnel data frame. At step 356, in an example embodiment, the pseudo network adapter forms an Ethernet packet from the decapsulated message. At step 358 the pseudo network adapter indicates that an Ethernet packet has been received to the TCP/IP protocol stack through the virtual network adapter interface. This causes the TCP/IP protocol stack to behave as if it had received an Ethernet packet from an actual Ethernet adapter.

Fig. 19 shows the data flow within the transmit path in an example embodiment of a pseudo network adapter. At step 1 370, an application submits data to be transmitted to the TCP protocol layer 372 within the TCP/IP protocol stack. The application uses a conventional socket interface to the TCP protocol layer 372 to pass the data, and indicates the destination IP address the data is to be transmitted to. The TCP protocol layer 372 then passes the data to the IP protocol layer 374 within the TCP/IP protocol stack. At step 2 376, the TCP/IP protocol stack refers to the routing table 378 to determine which network interface should be used to reach the destination IP address.

Because in the example the destination IP address is of a node reachable through the virtual private network, the IP layer 374 determines from the routing table 378 that the destination IP address is reachable through pseudo network adapter. Accordingly at step 3 380 the TCP/IP protocol stack passes a packet containing the data to the pseudo network adapter 382.

At step 4 384, the pseudo network adapter 382 encrypts the data packets and encapsulates them into tunnel data frames.

The pseudo network adapter 382 then passes the tunnel data frames packets back to the TCP protocol layer 372 within the TCP/IP protocol stack through a conventional socket interface to the tunnel connection with the first node in the tunnel path.

The TCP protocol layer 372 then forms a TCP layer packet for each tunnel data frame, having the tunnel data frame as its data. The TCP frames are passed to the IP layer 374. At step 5 386 the routing table 378 is again searched, and this time the destination IP address is the IP address associated with the physical network adapter on the tunnel server, and accordingly is determined to be reachable over the physical network adapter 390. Accordingly at step 6 388 the device driver 390 for the physical network adapter is called to pass the packets to the physical network adapter. At step 7 392 the physical network adapter transmits the data onto the physical network 394.

Fig. 20 is a data flow diagram showing data flow in an example embodiment of packet receipt involving a pseudo network adapter. At step 1 410 data arrives over the physical network 412 and is received by the physical network adapter and passed to the physical network driver 414. The physical network driver 414 passes the data at step 2 418 through the IP layer 420 and TCP layer 422 to the pseudo network adapter 426 at step 3 424, for example through a conventional socket interface. At step 4 428 the pseudo network adapter 426 decrypts and decapsulates the received data and passes it back to the IP layer of the TCP/IP protocol stack, for example through the TDI (Transport Layer Dependent Interface API) of the TCP/IP stack. The data is then passed through the TCP/IP protocol stack and to the user associated with the destination IP address in the decapsulated datagrams at step 5 430.

Fig. 21 shows data flow in an example embodiment of packet transmission involving a pseudo network adapter. Fig. 21 shows an example embodiment for use on a Microsoft™ Windows 95™ PC platform. In Fig. 21 a user application 450 passes unencrypted data to an interface into the TCP layer of the TCP/IP protocol, for example the WinSock API 452. The user indicates a destination IP address associated with a node reachable through a virtual private network accessible through the pseudo network adapter.

The TCP layer 454 passes the data to the IP layer 456, which in turn passes the data to the Network Device Interface Specification Media Access Control (NDIS MAC) interface 458. The pseudo network adapter 459 has previously registered with the routing layer (IP) that it is able to reach a gateway address associated with the destination IP address for the user data. Accordingly the IP layer uses the NDIS MAC layer interface to invoke the virtual device driver interface 460 to the pseudo network adapter 459. The pseudo network adapter 459 includes a virtual device driver interface 460, an ARP server emulator 462, and a DHCP server emulator 464.

In the example embodiment of Fig. 19, the pseudo network adapter 459 passes the data to a tunnel application program 466. The tunnel application program 466 encrypts the IP packet received from the IP layer and encapsulates it into a tunnel data frame. The tunnel application then passes the tunnel data frame including the encrypted data to the WinSock interface 452, indicating a destination IP address of the remote tunnel end point. The tunnel data frame is then passed through the TCP layer 454, IP layer 456, NDIS MAC layer interface 458, and physical layer 468, and transmitted on the network 470. Since the resulting packets do not contain a destination IP address which the pseudo network adapter has registered to convey, these packets will not be diverted to the pseudo network adapter.

Fig. 22 is a data flow diagram showing data flow in an example embodiment of packet transmission involving a pseudo network adapter. The embodiment shown in Fig. 22 is for use on a UNIX platform. In Fig. 20 a user application 472 passes unencrypted data to a socket interface to the TCP/IP protocol stack in the UNIX socket layer 474, indicating a destination IP address of a node reachable through the virtual private network.

The UNIX socket layer 474 passes the data through the TCP layer 476 and the IP layer 478. The pseudo network adapter 480 has previously registered with the routing layer (IP) that it is able to reach a gateway associated with the destination IP address for the user data. Accordingly the IP layer 478 invokes the virtual device driver interface 482 to the pseudo network adapter 480. The IP layer 478 passes the data to the pseudo network adapter 480. The pseudo network adapter 480 includes a virtual device driver interface 482, and a DHCP server emulator 484.

In the example embodiment of Fig. 22, the pseudo network adapter 480 passes IP datagrams to be transmitted to a UNIX Daemon 486 associated with the tunnel connection. The UNIX Daemon 486 encrypts the IP packet(s) received from the IP layer 478 and encapsulates them into tunnel data frames. The UNIX Daemon 486 then passes the tunnel data frames to the UNIX socket layer 474, through a socket associated with the tunnel connection. The tunnel data frames are then processed by the TCP layer 476, IP layer 478, data link layer 488, and physical layer 490 to be transmitted on the network 492. Since the resulting packets are not addressed to an IP address which the pseudo network adapter 480 has registered to convey, the packets will not be diverted to the pseudo network adapter 480.

Fig. 23 is a flow chart showing steps to initialize a example embodiment of a virtual private network. The steps shown in Fig. 23 are performed for example in the tunnel client 247 as shown in Fig. 14. At step 500 a tunnel application program executing in the tunnel client sends a tunnel relay frame to the tunnel server. At step 502 the tunnel application program sends a tunnel key exchange/authentication REQUEST frame to the tunnel server. The tunnel application in the tunnel server ignores the contents of the client data field in the tunnel key exchange/authentication REQUEST frame. The tunnel application in the tunnel server fills in the client data field in the tunnel key exchange/authentication RESPONSE frame with Dynamic Host Configuration Protocol (DHCP) information, for example including the following information in standard DHCP format:

1) IP Address for tunnel client Pseudo Network Adapter
2) IP Address for tunnel server Pseudo Network Adapter
3) Routes to nodes on the private network physically connected to the tunnel server which are to be reachable over the tunnel connection.

At step 504 the tunnel application receives a tunnel key exchange/authentication RESPONSE frame from the tunnel server. The client data field 508 in the tunnel connection response is made available to the pseudo network adapter in the tunnel client. The tunnel application in the tunnel client tells the TCP/IP stack that the pseudo network adapter in the tunnel client is active. The pseudo network adapter in the tunnel client is active and ready to be initialized at step 510.

The tunnel client system is configured such that it must obtain an IP address for the tunnel client pseudo network adapter dynamically. Therefore the TCP/IP stack in the tunnel client broadcasts a DHCP request packet through the pseudo network adapter. Accordingly, at step 512 the pseudo network adapter in the client receives a conventional DHCP request packet from the TCP/IP stack requesting a dynamically allocated IP address to associate with the pseudo network adapter. The pseudo network adapter passes the DHCP request packet to the DHCP server emulator within the pseudo network adapter, which forms a DHCP response based on the client data 508 received from the tunnel application. The DHCP response includes the IP address for the client pseudo adapter provided by the tunnel server in the client data. At step 514 the pseudo network adapter passes the DHCP response to the TCP/IP stack.

At step 520, the tunnel application modifies the routing tables within the tunnel client TCP/IP stack to indicate that the routes to the nodes attached to the private network to which the tunnel server is attached all are reachable only through the pseudo network adapter in the tunnel server. The IP address of the pseudo network adapter in the tunnel server provided in the client data is in this way specified as a gateway to the nodes on the private network to which the tunnel server is attached. In this way those remote nodes are viewed by the TCP/IP stack as being reachable via the virtual private network through the client pseudo network adapter.

At step 516 the pseudo network adapter in the tunnel client receives an ARP request for a physical address associated with the IP address of the pseudo network adapter in the tunnel server. The pseudo network adapter passes the ARP request to the ARP server emulator, which forms an ARP reply indicating a reserved physical address to be associated with the IP address of the pseudo network adapter in the tunnel server. At step 518 the pseudo network adapter passes the ARP response to the TCP/IP stack in the tunnel client. In response to the ARP response, the TCP/IP stack determines that packets addressed to any node on the virtual private network must be initially transmitted through the pseudo network adapter.

In an example embodiment the present system reserves two physical addresses to be associated with the pseudo network adapter in the client and the pseudo network adapter in the server respectively. These reserved physical addresses are used in responses to ARP requests passed through the pseudo network adapter for physical addresses corresponding to the IP addresses for the pseudo network adapter in the client and the pseudo network adapter in the server respectively. The reserved physical addresses should have a high likelihood of not being used in any actual network interface.

While the invention has been described with reference to specific example embodiments, the description is not meant to be construed in a limiting sense. Various modifications of the disclosed embodiments, as well as other embodiments of the invention, will be apparent to persons skilled in the art upon reference to this description. Specifically, while various embodiments have been described using the TCP/IP protocol stack, the invention may advantageously be applied where other communications protocols are used. Also, while various flow charts have shown steps performed in an example order, various implementations may use altered orders of step in order to apply the invention. And further, while certain specific software and/or hardware platforms have been used in the description, the invention may be applied on other platforms with similar advantage. It is therefore contemplated that the appended claims will cover any such modifications or embodiments which fall within the scope of the invention.

Claims

1. A pseudo network adapter providing a virtual private network, comprising:

    an interface for capturing packets from a local communications protocol stack for transmission on said virtual private network, said interface appearing to said local communications protocol stack as a network adapter device driver for a network adapter connected to said virtual private network;
    a first server emulator, providing a first reply packet responsive to a first request packet captured by said interface for capturing packets from said local communications protocol stack for transmission on said virtual private network, said first request packet requesting a network layer address for said pseudo network adapter, said first reply indicating a network layer address for said pseudo network adapter; and
    a second server emulator, providing a second reply packet responsive to an second request packet captured by said interface for capturing packets from said local communications protocol stack for transmission on said virtual private network, said second request packet requesting a physical address corresponding to a network layer address of a second pseudo network adapter, said second pseudo network adapter located on a remote server node, said second reply indicating a predetermined, reserved physical address.

2. The pseudo network adapter of claim 1, further comprising a means for indicating to said local communications protocol stack that said predetermined, reserved physical address is reachable through said pseudo network adapter, wherein said means for indicating modifies a data structure in said local communications protocol stack indicating which nodes or networks are reachable through each network interface of the local system.

3. The pseudo network adapter of claim 1, further comprising a means for indicating to said local communications protocol stack that one or more nodes on a remote private network connected to said remote server node are reachable through a gateway node equal to said second pseudo network adapter on said remote server node.

4. The pseudo network adapter of claim 1, further comprising:

a transmit path for processing data packets captured by said interface for capturing packets from said local communications protocol stack for transmission on said virtual private network;
an encryption engine, within said transmit path, for encrypting said data packets;
an encapsulation engine, within said transmit path, for encapsulating said encrypted data packets into tunnel data frames; and
a means for passing said tunnel data frames back to said local communications protocol stack for transmission to a physical network adapter on said remote server node.

5. The pseudo network adapter of claim 4, wherein said transmit path further includes means for storing a digest value in a digest field in each of said tunnel data frames, said digest value equal to an output of a keyed hash function applied to said data packet encapsulated within said tunnel data frame concatenated with a counter value equal to a total number of tunnel data frames previously transmitted to said remote server node.

6. The pseudo network adapter of claim 4, wherein said transmit path further includes means for processing an Ethernet header in each one of said captured data packets, said processing of said Ethernet header including removing said Ethernet header.

7. The pseudo network adapter of claim 1, further comprising:

an interface into a transport layer of said local communications protocol stack for capturing received data packets from said remote server node.

8. The pseudo network adapter of claim 7, further comprising:

a receive path for processing received data packets captured by said interface into said transport layer of said local communications protocol stack for capturing received data packets from said remote server node;
an decapsulation engine, within said receive path, for decapsulating said received data packets by removing a tunnel frame header;
an decryption engine, within said receive path, for decrypting said received data packets; and
a means for passing said received data packets back to said local communications protocol stack for delivery to a user.

9. A method for providing a pseudo network adapter for a virtual private network, comprising the steps of:

capturing packets from a local communications protocol stack for transmission on said virtual private network, said capturing through an interface appearing to said local communications stack as a network adapter device driver for a network adapter connected to said virtual private network;
issuing a first reply packet responsive to a first request packet captured by said interface for capturing packets from said local communications protocol stack for transmission on said virtual private network, said first request packet requesting a network layer address for said pseudo network adapter, said first reply indicating a network layer address for said pseudo network adapter; and
issuing a second reply packet responsive to a second request packet captured by said interface for capturing packets from said local communications protocol stack for transmission on said virtual private network, said second request packet requesting a physical address corresponding to a network layer address of a second pseudo network adapter, said second pseudo network adapter located on a remote server node, said ARP Reply indicating a predetermined, reserved physical address.

10. The method of claim 9, further comprising indicating to said local communications protocol stack that said predetermined, reserved physical address is reachable through said pseudo network adapter, wherein said step of indicating to said local communications protocol stack modifies a data structure in said local communications protocol stack indicating which nodes or networks are reachable through each network interface of the local system.

11. The method of claim 9, further comprising indicating to said local communications protocol stack that one or more nodes on a remote private network connected to said remote server node are reachable through a gateway node equal to said second pseudo network adapter on said remote server node, wherein said step of indicating to said local communications protocol stack that one or more nodes on said remote private network connected to said remote server node are reachable through a gateway node equal to said second pseudo network adapter on said remote server node modifies a network layer routing table in said local communications protocol stack.

12. The method of claim 9, further comprising:

processing data packets captured by said interface for capturing packets from said local communications protocol stack for transmission on said virtual private network in a transmit data path;

encrypting said data packets in an encryption engine, within said transmit path;

encapsulating said encrypted data packets into tunnel data frames by an encapsulation engine, within said transmit path; and

passing said tunnel data frames back to said local communications protocol stack for transmission to a physical network adapter on said remote server node, wherein said transmit path further includes storing a digest value in a digest field in each of said tunnel data frames, said digest value equal to an output of a keyed hash function applied to said data packet encapsulated within said tunnel data frame concatenated with a counter value equal to a total number of tunnel data frames previously transmitted to said remote server node.

13. The method of claim 12, wherein said transmit path further includes processing an Ethernet header in each one of said captured data packets, said processing of said Ethernet header including removing said Ethernet header.

14. The method of claim 9, further comprising capturing received data packets from said remote server node through an interface into a transport layer of said local communications protocol stack, further comprising:

processing received data packets captured by said interface into said transport layer of said local communications protocol stack for capturing received data packets from said remote server node in a receive path;

decapsulating said received data packets by removing a tunnel frame header in an decapsulation engine, within said receive path;

decrypting said received data packets in a decryption engine within said receive path; and

passing said received data frames packets back to said local communications protocol stack for delivery to a user.

15. The method of claim 9, wherein said network layer address for said pseudo network adapter and said predetermined, reserved physical address is communicated to said pseudo network adapter from said remote server node as client data in a connection response frame.

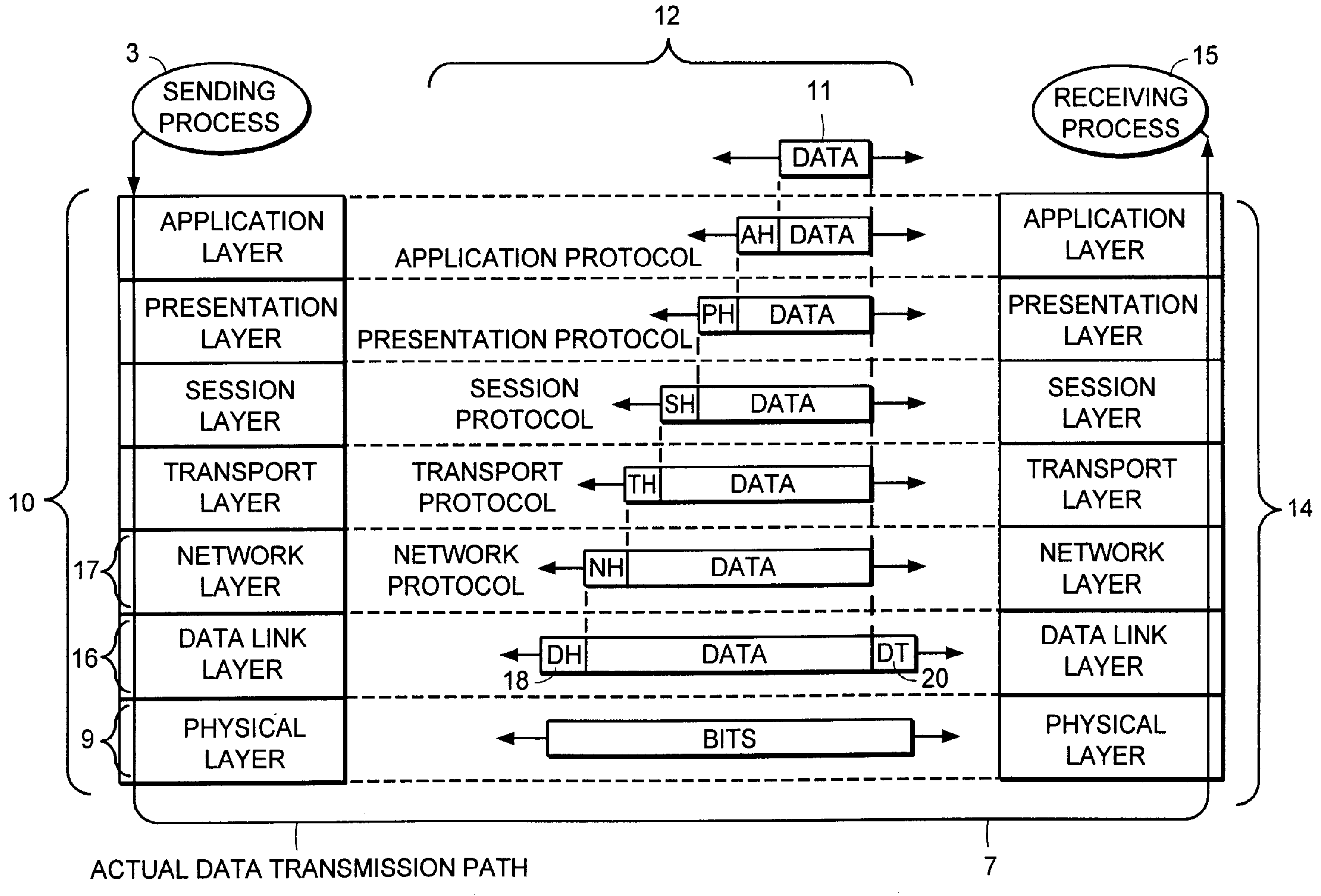

3
SENDING PROCESS
12
11
DATA
15
RECEIVING PROCESS
APPLICATION LAYER
AH DATA
APPLICATION PROTOCOL
APPLICATION LAYER
PRESENTATION LAYER
PH DATA
PRESENTATION PROTOCOL
PRESENTATION LAYER
SESSION LAYER
SESSION PROTOCOL
SH DATA
SESSION LAYER
10
TRANSPORT LAYER
TRANSPORT PROTOCOL
TH DATA
TRANSPORT LAYER
14
17
NETWORK LAYER
NETWORK PROTOCOL
NH DATA
NETWORK LAYER
16
DATA LINK LAYER
DH DATA DT
18
20
DATA LINK LAYER
9
PHYSICAL LAYER
BITS
PHYSICAL LAYER
ACTUAL DATA TRANSMISSION PATH
7

FIG. 1

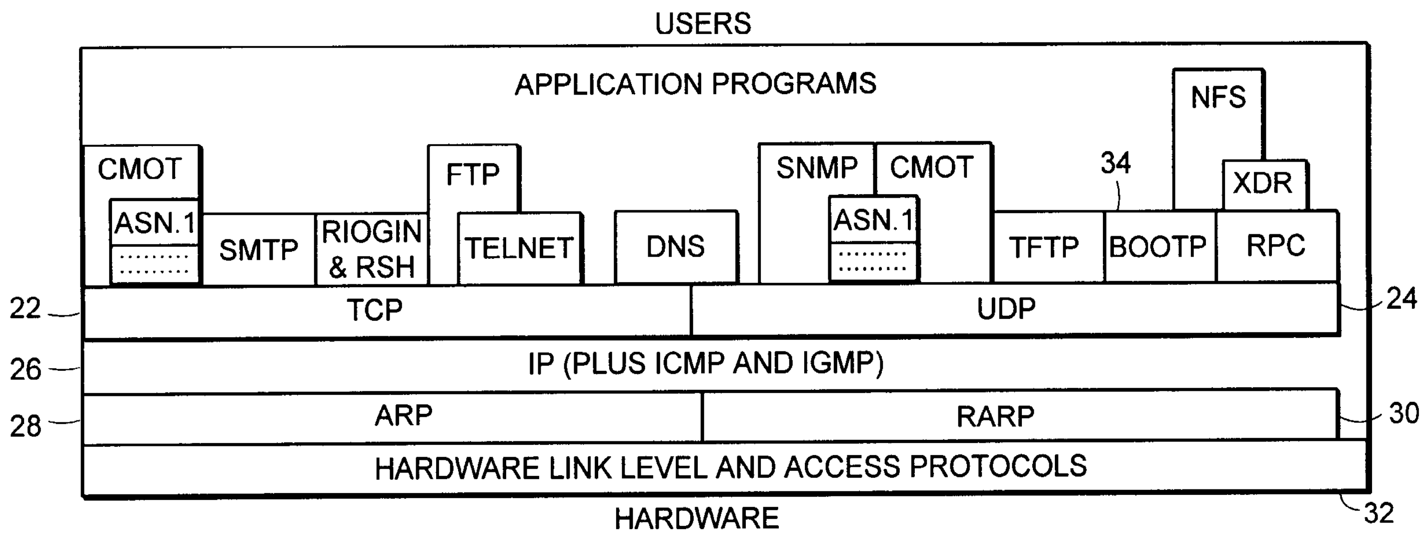

USERS
APPLICATION PROGRAMS
CMOT
ASN.1
SMTP
RIOGIN & RSH
FTP
TELNET
DNS
SNMP
CMOT
ASN.1
TFTP
34
BOOTP
NFS
XDR
RPC
22
TCP
UDP
24
26
IP (PLUS ICMP AND IGMP)
28
ARP
RARP
30
HARDWARE LINK LEVEL AND ACCESS PROTOCOLS
HARDWARE
32

FIG. 2

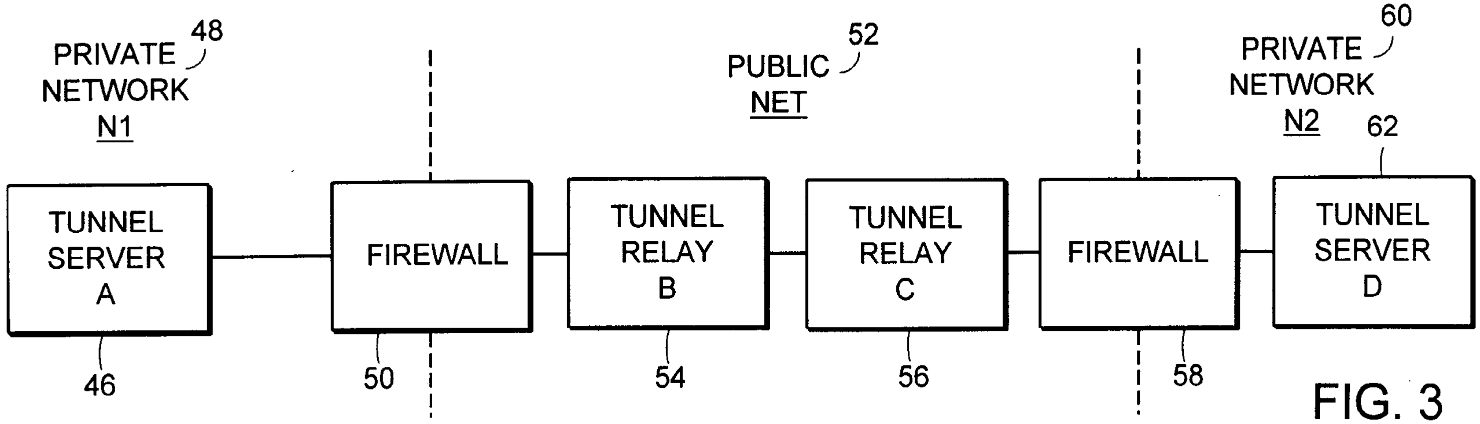

PRIVATE NETWORK N1
48
PUBLIC NET
52
PRIVATE NETWORK N2
60
62
TUNNEL SERVER A
FIREWALL
TUNNEL RELAY B
TUNNEL RELAY C
FIREWALL
TUNNEL SERVER D
46
50
54
56
58

FIG. 3

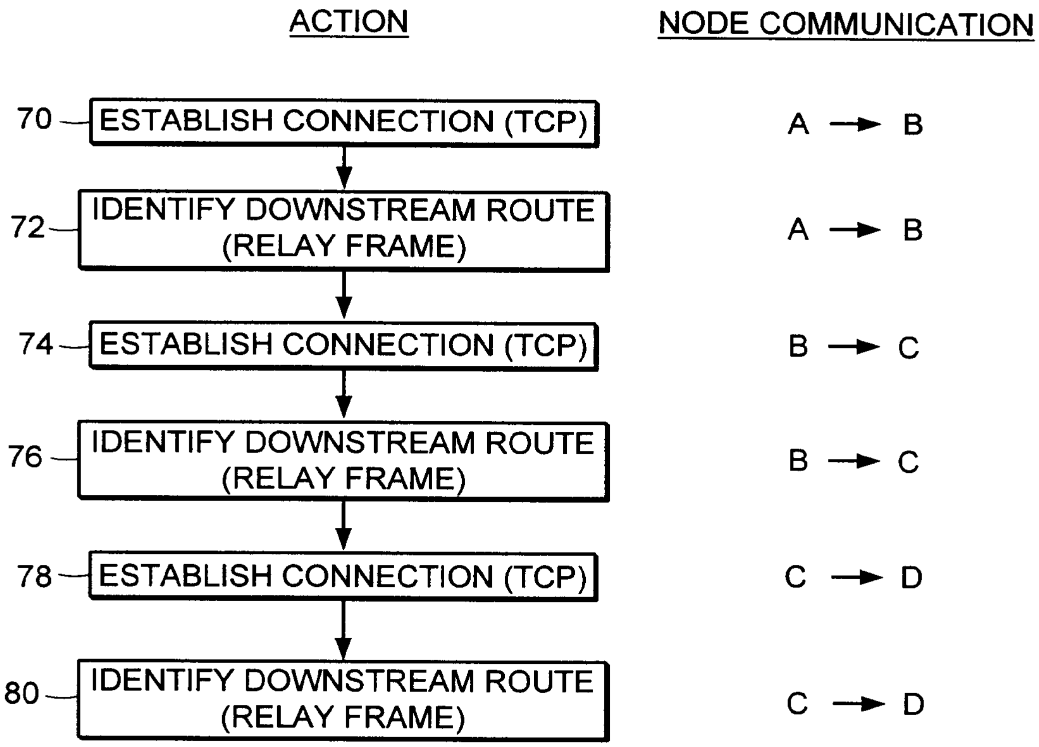

ACTION
NODE COMMUNICATION
70
ESTABLISH CONNECTION (TCP)
A → B
72
IDENTIFY DOWNSTREAM ROUTE (RELAY FRAME)
A → B
74
ESTABLISH CONNECTION (TCP)
B → C
76
IDENTIFY DOWNSTREAM ROUTE (RELAY FRAME)
B → C
78
ESTABLISH CONNECTION (TCP)
C → D
80
IDENTIFY DOWNSTREAM ROUTE (RELAY FRAME)
C → D

FIG. 4

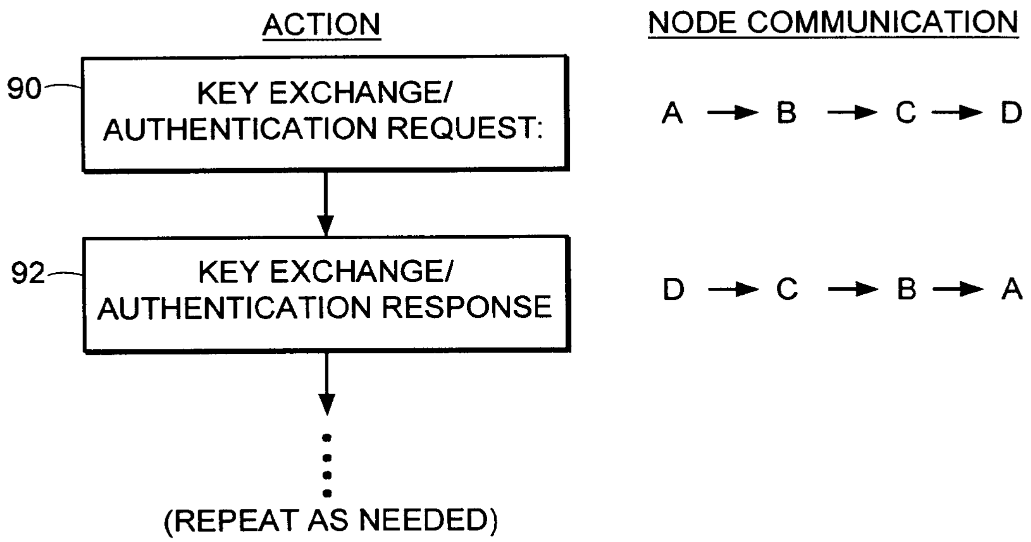

ACTION
NODE COMMUNICATION
90
KEY EXCHANGE/
AUTHENTICATION REQUEST:
A → B → C → D
92
KEY EXCHANGE/
AUTHENTICATION RESPONSE
D → C → B → A
(REPEAT AS NEEDED)

FIG. 5

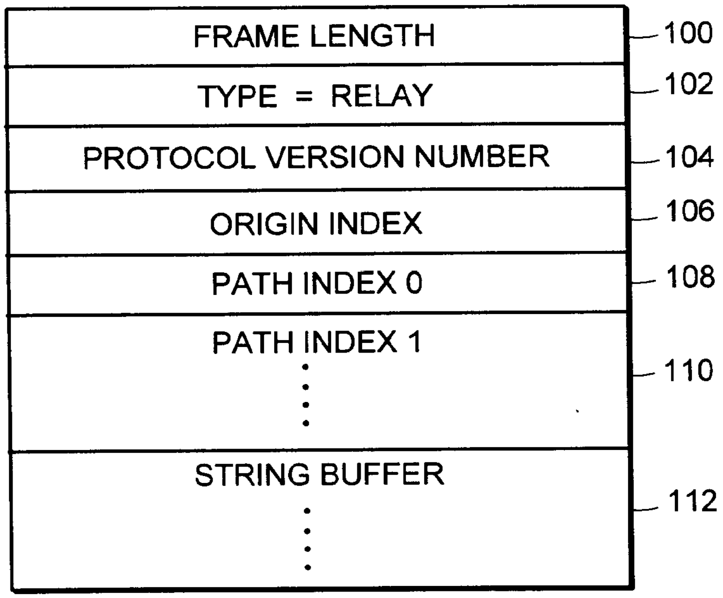

FRAME LENGTH
100
TYPE = RELAY
102
PROTOCOL VERSION NUMBER
104
ORIGIN INDEX
106
PATH INDEX 0
108
PATH INDEX 1
110
STRING BUFFER
112

FIG. 6

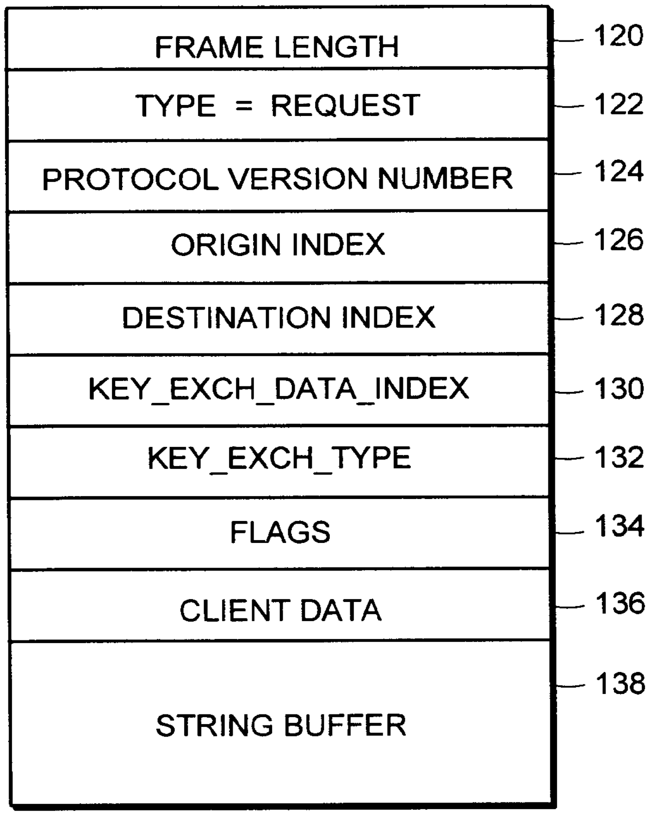

FRAME LENGTH
120
TYPE = REQUEST
122
PROTOCOL VERSION NUMBER
124
ORIGIN INDEX
126
DESTINATION INDEX
128
KEY_EXCH_DATA_INDEX
130
KEY_EXCH_TYPE
132
FLAGS
134
CLIENT DATA
136
STRING BUFFER
138

FIG. 7

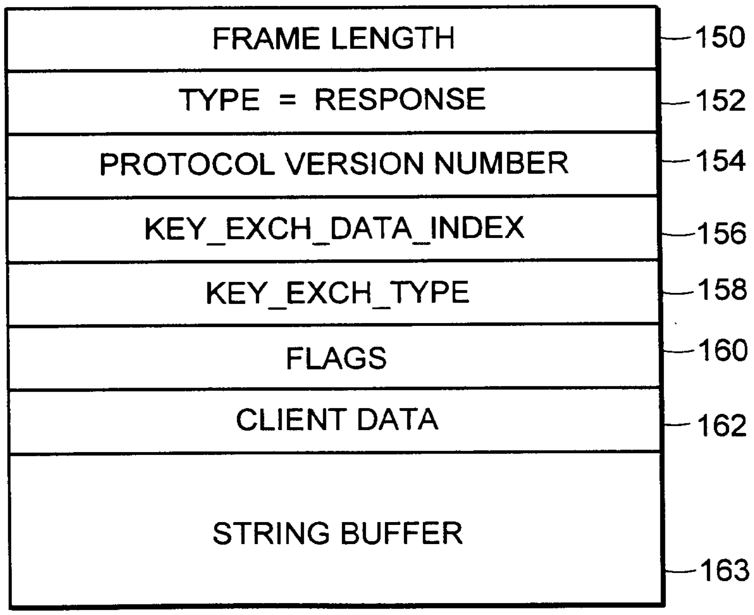

FRAME LENGTH
150
TYPE = RESPONSE
152
PROTOCOL VERSION NUMBER
154
KEY_EXCH_DATA_INDEX
156
KEY_EXCH_TYPE
158
FLAGS
160
CLIENT DATA
162
STRING BUFFER
163

FIG. 8

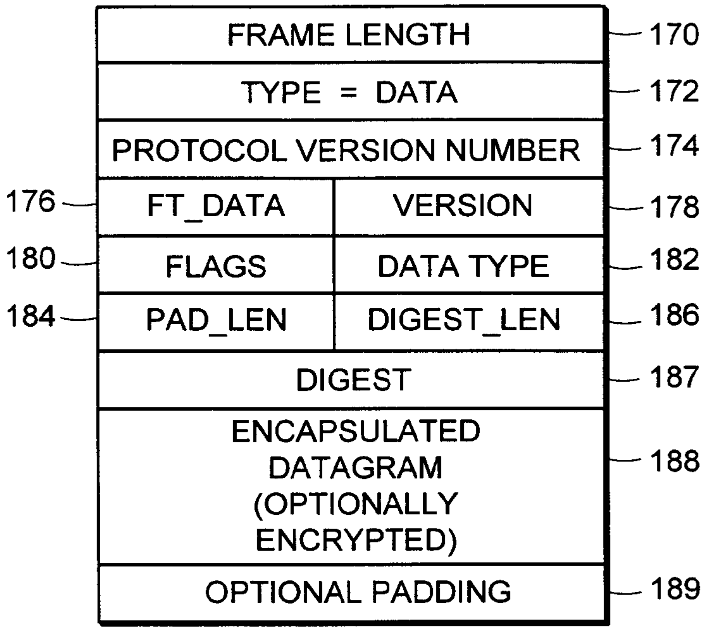

FRAME LENGTH
170
TYPE = DATA
172
PROTOCOL VERSION NUMBER
174
176
FT_DATA
VERSION
178
180
FLAGS
DATA TYPE
182
184
PAD_LEN
DIGEST_LEN
186
DIGEST
187
ENCAPSULATED DATAGRAM (OPTIONALLY ENCRYPTED)
188
OPTIONAL PADDING
189

FIG. 9

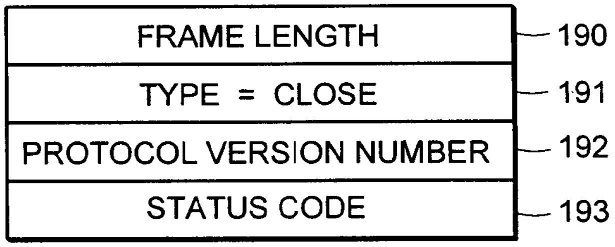

FRAME LENGTH
190
TYPE = CLOSE
191
PROTOCOL VERSION NUMBER
192
STATUS CODE
193

FIG. 10

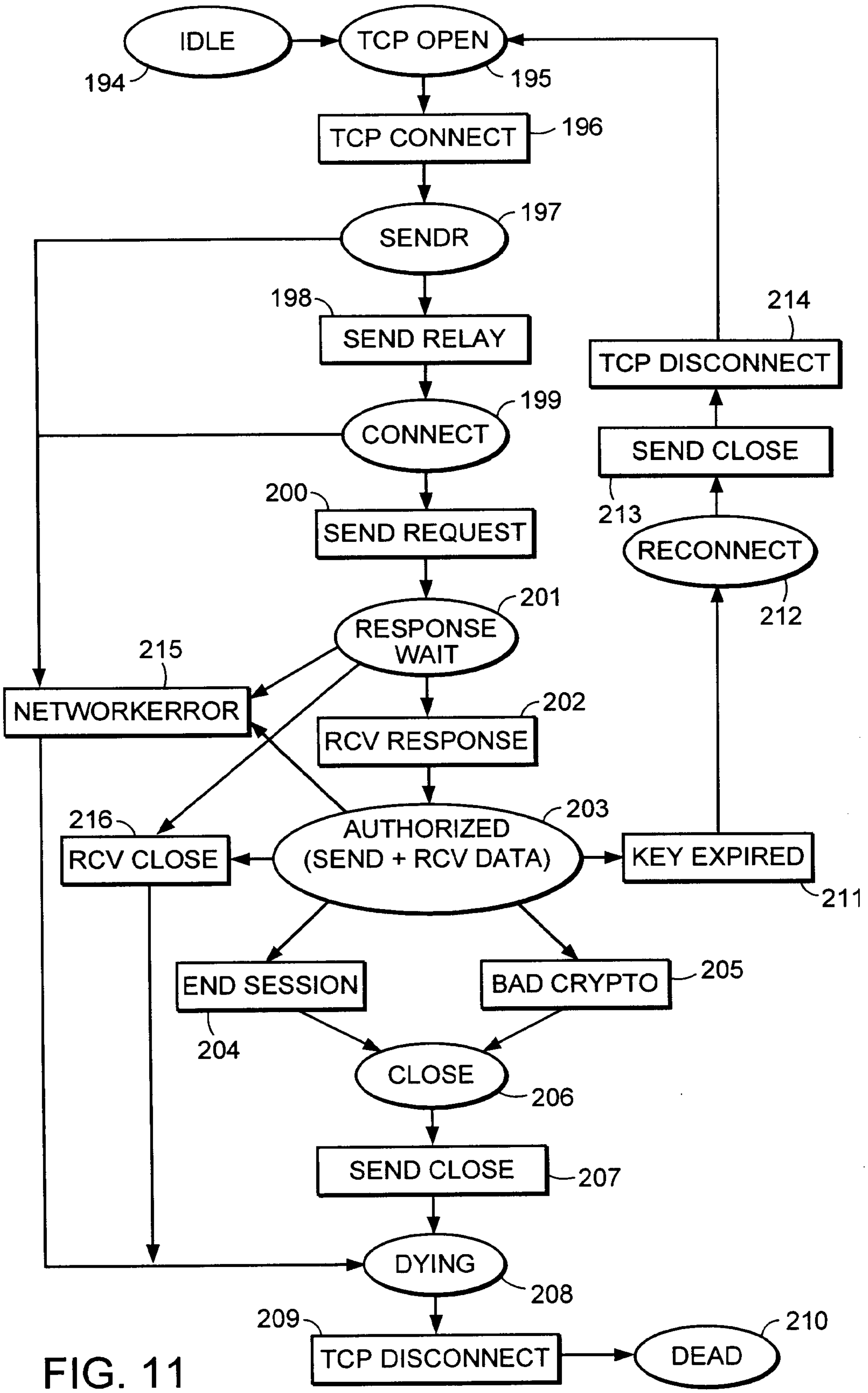

IDLE
194
TCP OPEN
195
TCP CONNECT
196
SENDR
197
198
SEND RELAY
CONNECT
199
200
SEND REQUEST
RESPONSE WAIT
201
215
NETWORKERROR
RCV RESPONSE
202
216
RCV CLOSE
AUTHORIZED (SEND + RCV DATA)
203
KEY EXPIRED
211
END SESSION
204
BAD CRYPTO
205
CLOSE
206
SEND CLOSE
207
DYING
208
209
TCP DISCONNECT
DEAD
210
214
TCP DISCONNECT
SEND CLOSE
213
RECONNECT
212

FIG. 11

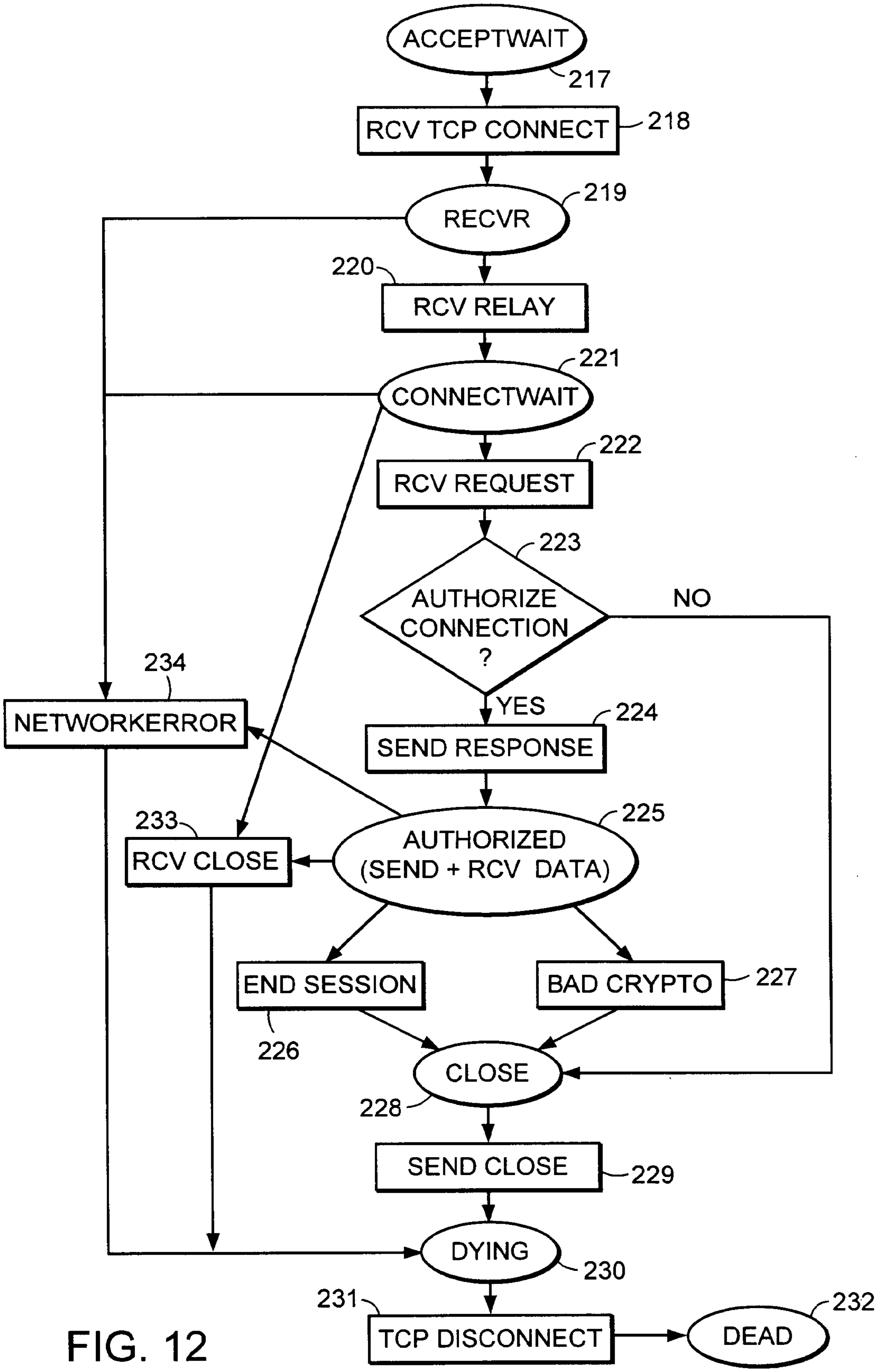

ACCEPTWAIT
217
RCV TCP CONNECT
218
RECVR
219
220
RCV RELAY
CONNECTWAIT
221
RCV REQUEST
222
223
AUTHORIZE CONNECTION ?
NO
YES
234
NETWORKERROR
SEND RESPONSE
224
233
RCV CLOSE
AUTHORIZED (SEND + RCV DATA)
225
END SESSION
226
BAD CRYPTO
227
CLOSE
228
SEND CLOSE
229
DYING
230
231
TCP DISCONNECT
DEAD
232

FIG. 12

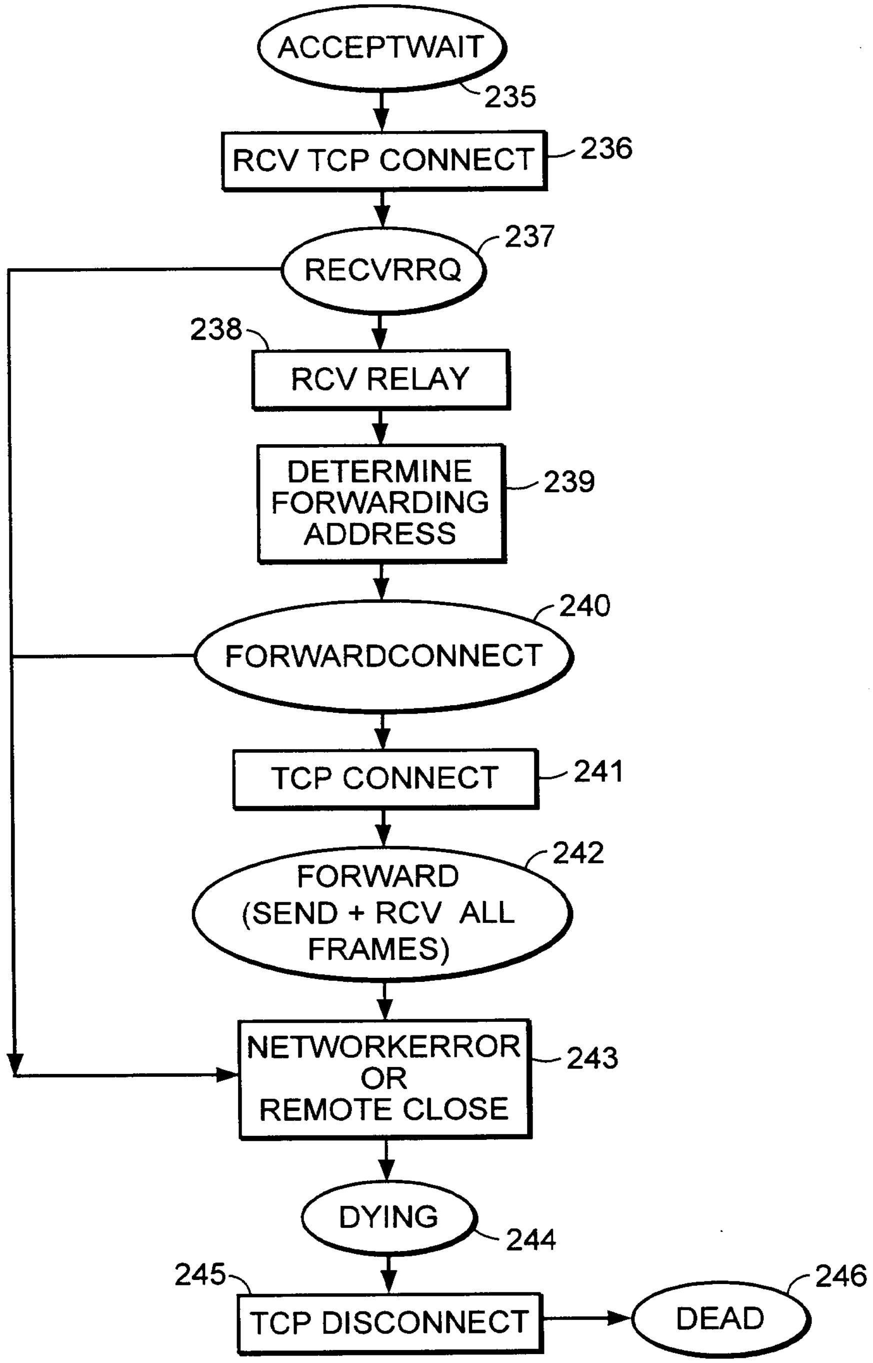

ACCEPTWAIT
235
RCV TCP CONNECT
236
RECVRRQ
237
238
RCV RELAY
DETERMINE FORWARDING ADDRESS
239
FORWARDCONNECT
240
TCP CONNECT
241
FORWARD (SEND + RCV ALL FRAMES)
242
NETWORKERROR OR REMOTE CLOSE
243
DYING
244
245
TCP DISCONNECT
DEAD
246

FIG. 13

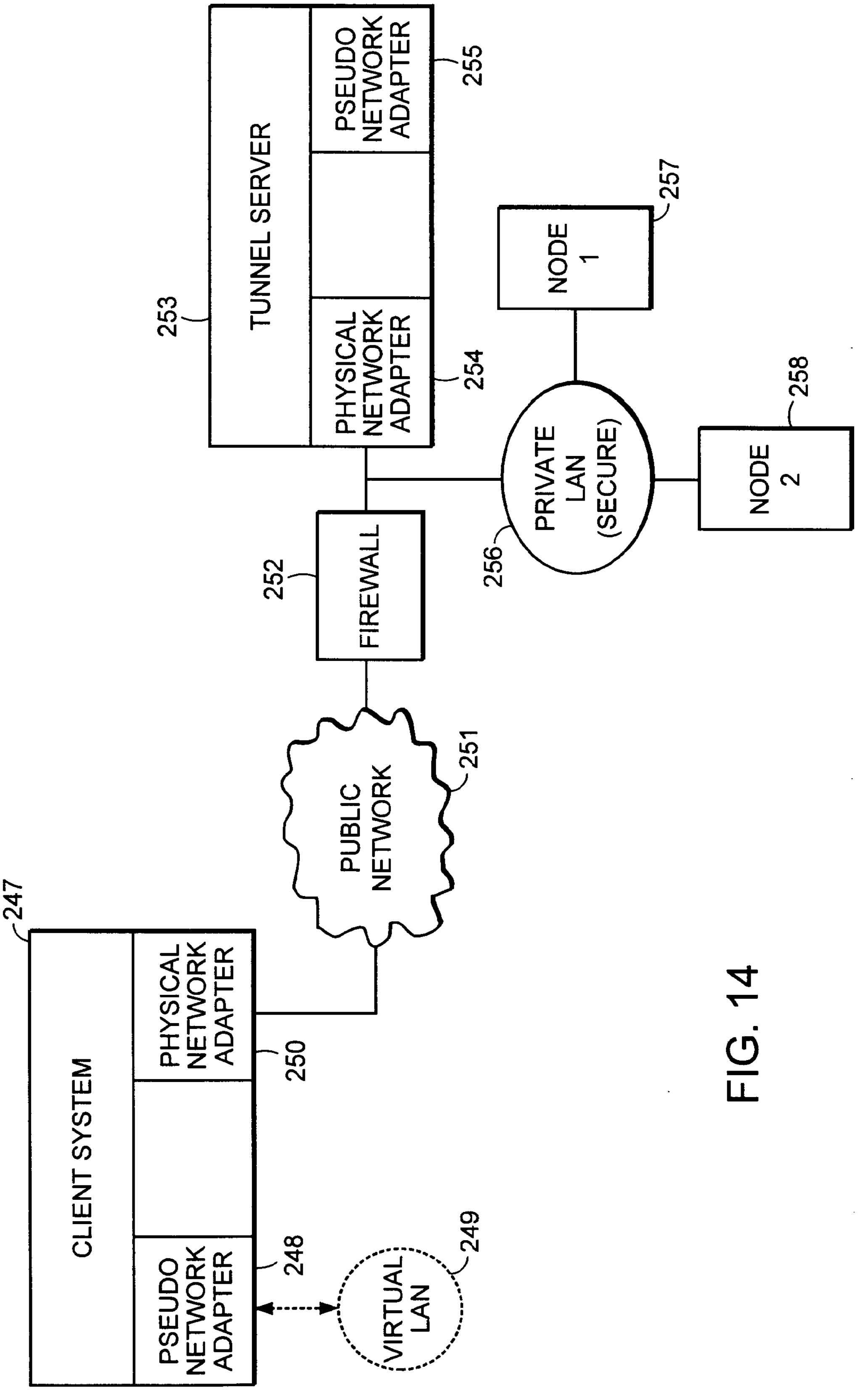

CLIENT SYSTEM
247
PSEUDO NETWORK ADAPTER
248
VIRTUAL LAN
249
PHYSICAL NETWORK ADAPTER
250
PUBLIC NETWORK
251
FIREWALL
252
TUNNEL SERVER
253
PHYSICAL NETWORK ADAPTER
254
PSEUDO NETWORK ADAPTER
255
PRIVATE LAN (SECURE)
256
NODE 1
257
NODE 2
258

FIG. 14

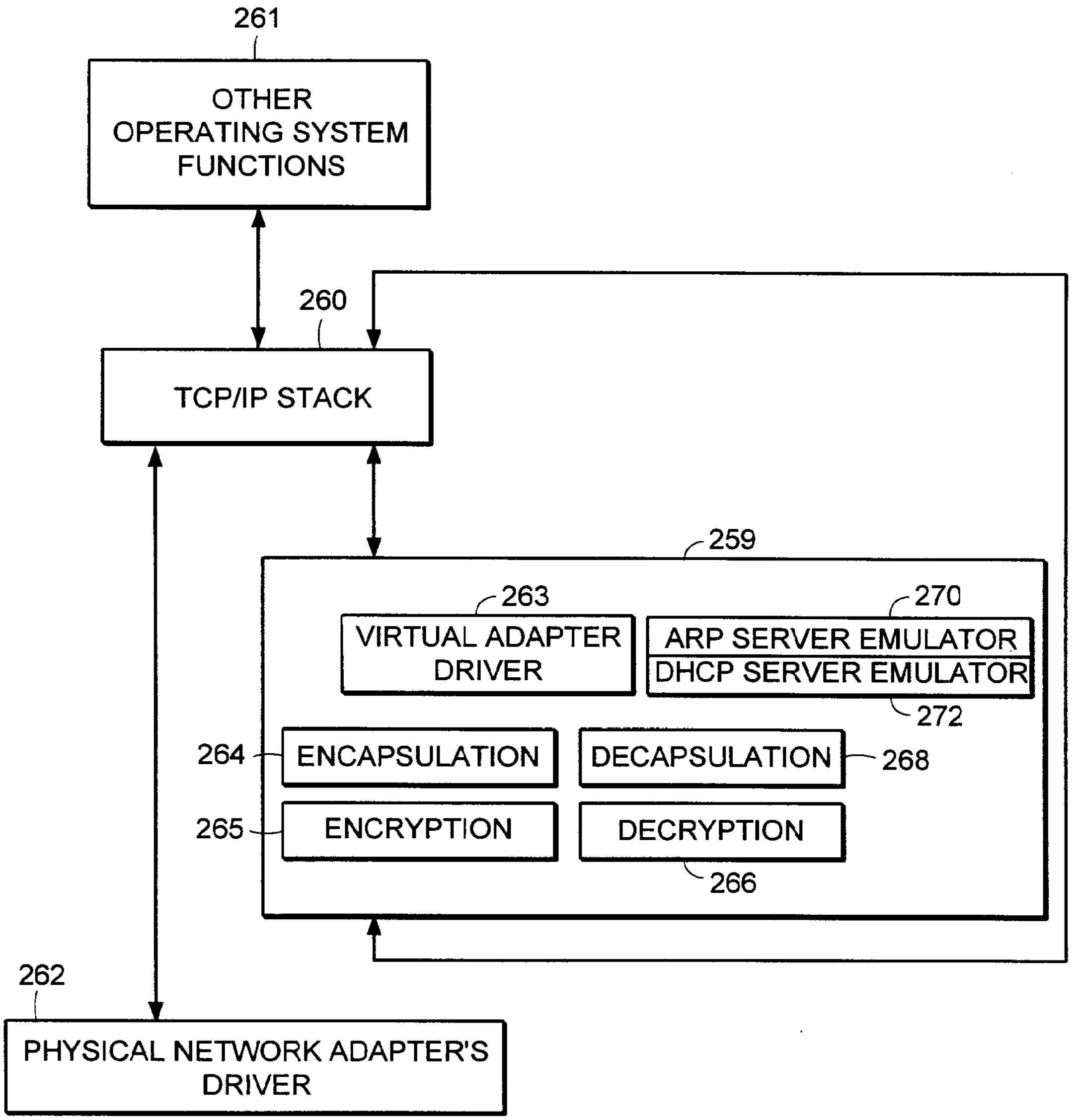

261
OTHER OPERATING SYSTEM FUNCTIONS
260
TCP/IP STACK
259
263
VIRTUAL ADAPTER DRIVER
270
ARP SERVER EMULATOR
DHCP SERVER EMULATOR
272
264
ENCAPSULATION
DECAPSULATION
268
265
ENCRYPTION
DECRYPTION
266
262
PHYSICAL NETWORK ADAPTER'S DRIVER

FIG. 15

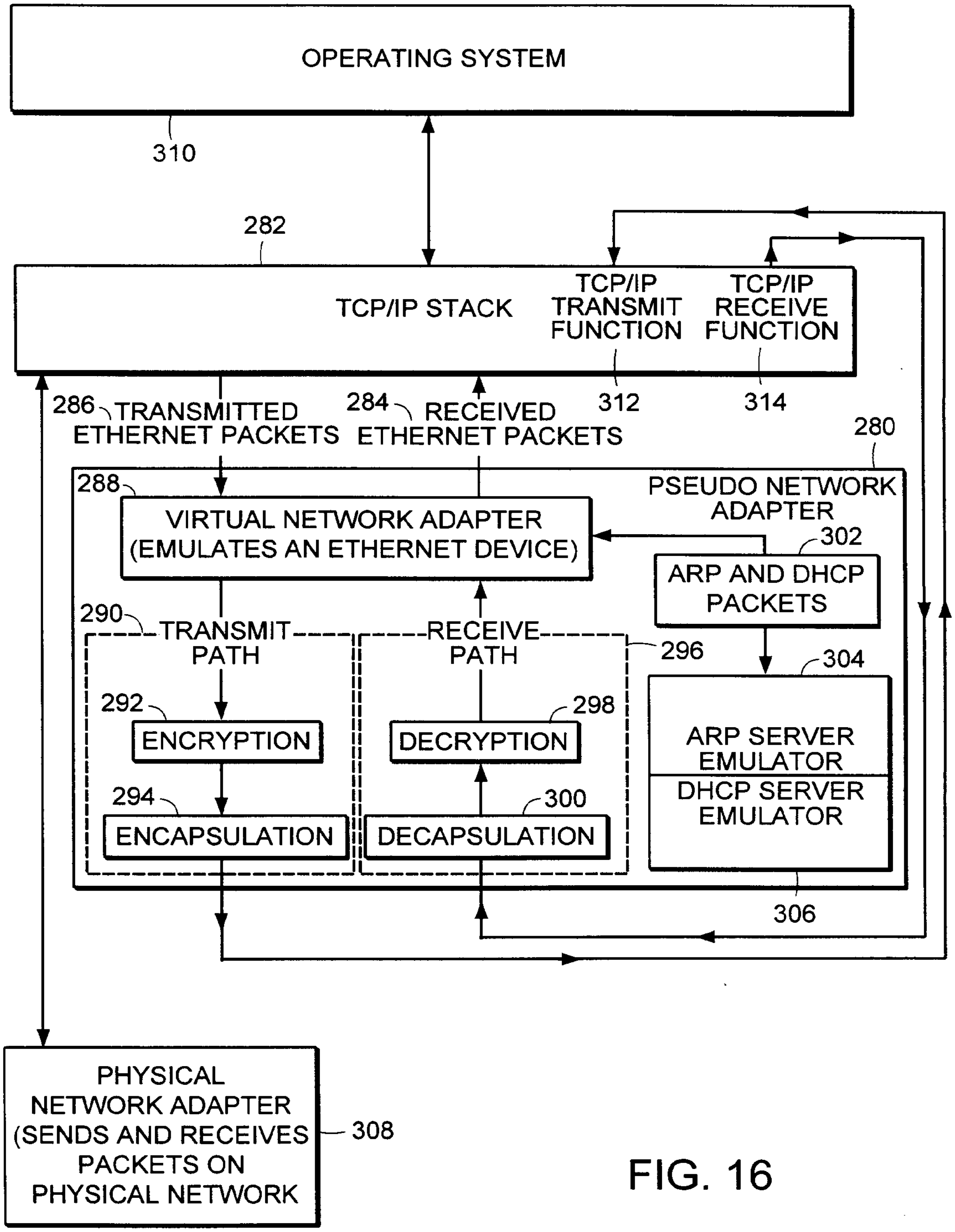

OPERATING SYSTEM
310
282
TCP/IP STACK
TCP/IP TRANSMIT FUNCTION
TCP/IP RECEIVE FUNCTION
312
314
286
TRANSMITTED ETHERNET PACKETS
284
RECEIVED ETHERNET PACKETS
280
PSEUDO NETWORK ADAPTER
288
VIRTUAL NETWORK ADAPTER (EMULATES AN ETHERNET DEVICE)
302
ARP AND DHCP PACKETS
290
TRANSMIT PATH
RECEIVE PATH
296
304
292
ENCRYPTION
298
DECRYPTION
ARP SERVER EMULATOR
DHCP SERVER EMULATOR
294
ENCAPSULATION
300
DECAPSULATION
306
PHYSICAL NETWORK ADAPTER (SENDS AND RECEIVES PACKETS ON PHYSICAL NETWORK
308

FIG. 16

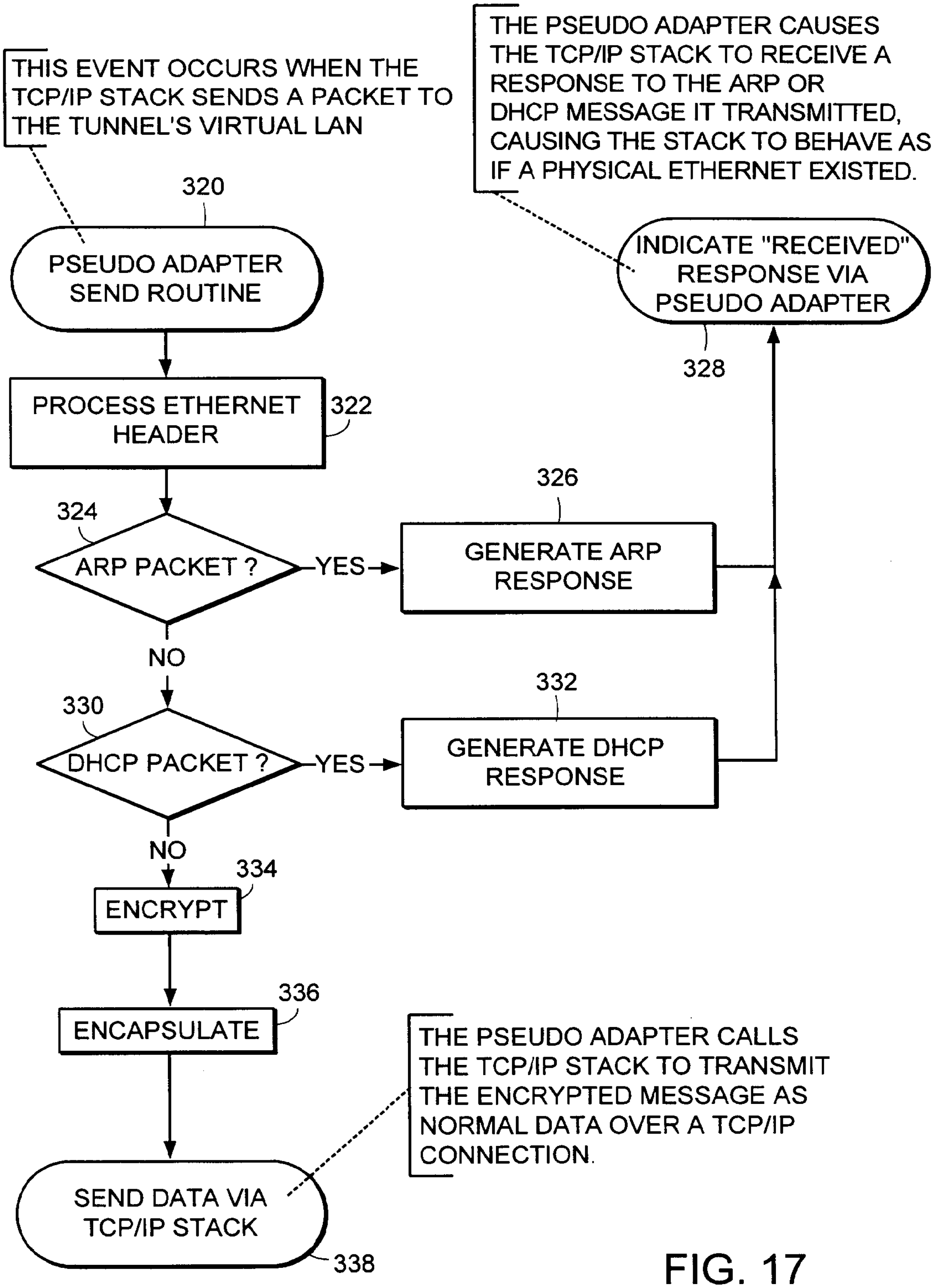

THIS EVENT OCCURS WHEN THE TCP/IP STACK SENDS A PACKET TO THE TUNNEL'S VIRTUAL LAN
320
PSEUDO ADAPTER SEND ROUTINE
PROCESS ETHERNET HEADER
322
324
ARP PACKET ?
YES
NO
326
GENERATE ARP RESPONSE
330
DHCP PACKET ?
YES
NO
332
GENERATE DHCP RESPONSE
THE PSEUDO ADAPTER CAUSES THE TCP/IP STACK TO RECEIVE A RESPONSE TO THE ARP OR DHCP MESSAGE IT TRANSMITTED, CAUSING THE STACK TO BEHAVE AS IF A PHYSICAL ETHERNET EXISTED.
INDICATE "RECEIVED" RESPONSE VIA PSEUDO ADAPTER
328
334
ENCRYPT
336
ENCAPSULATE
THE PSEUDO ADAPTER CALLS THE TCP/IP STACK TO TRANSMIT THE ENCRYPTED MESSAGE AS NORMAL DATA OVER A TCP/IP CONNECTION.
SEND DATA VIA TCP/IP STACK
338

FIG. 17

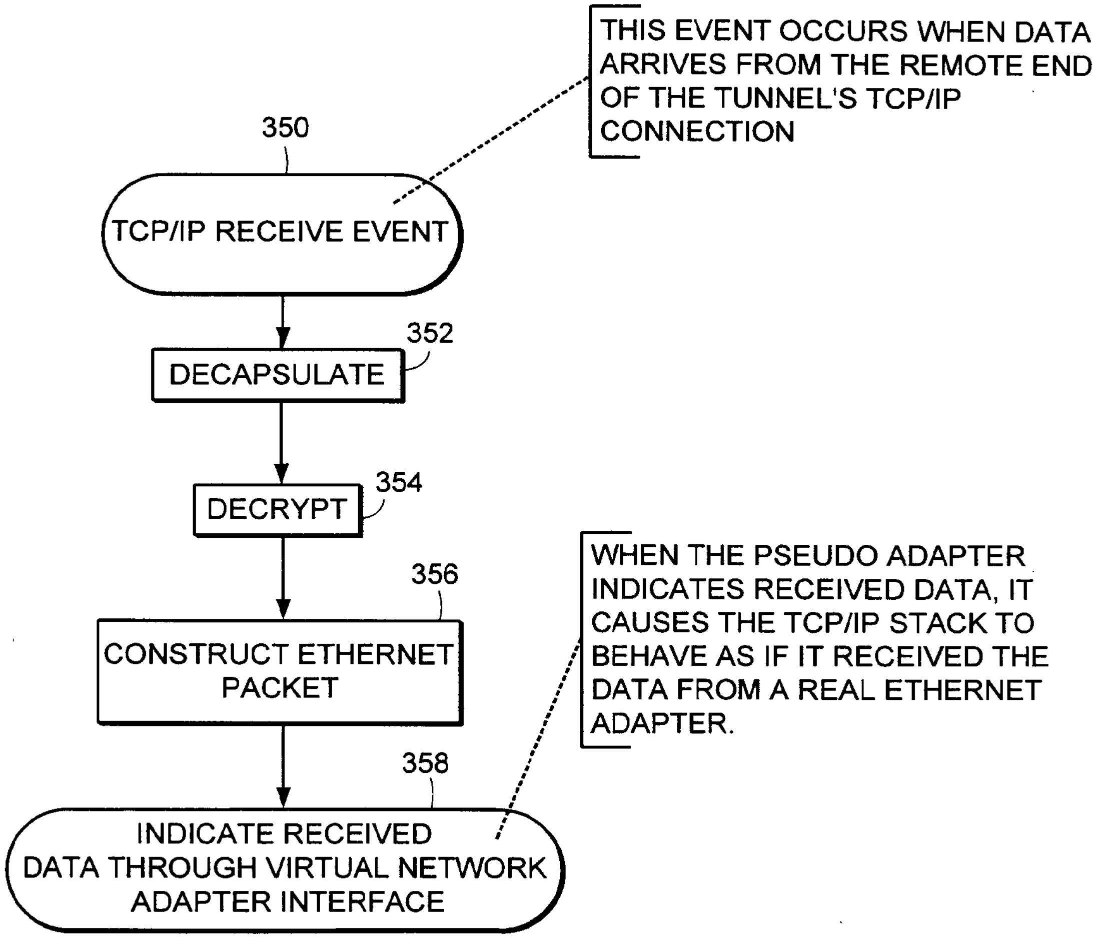

THIS EVENT OCCURS WHEN DATA ARRIVES FROM THE REMOTE END OF THE TUNNEL'S TCP/IP CONNECTION
350
TCP/IP RECEIVE EVENT
352
DECAPSULATE
354
DECRYPT
356
CONSTRUCT ETHERNET PACKET
358
INDICATE RECEIVED DATA THROUGH VIRTUAL NETWORK ADAPTER INTERFACE
WHEN THE PSEUDO ADAPTER INDICATES RECEIVED DATA, IT CAUSES THE TCP/IP STACK TO BEHAVE AS IF IT RECEIVED THE DATA FROM A REAL ETHERNET ADAPTER.

FIG. 18

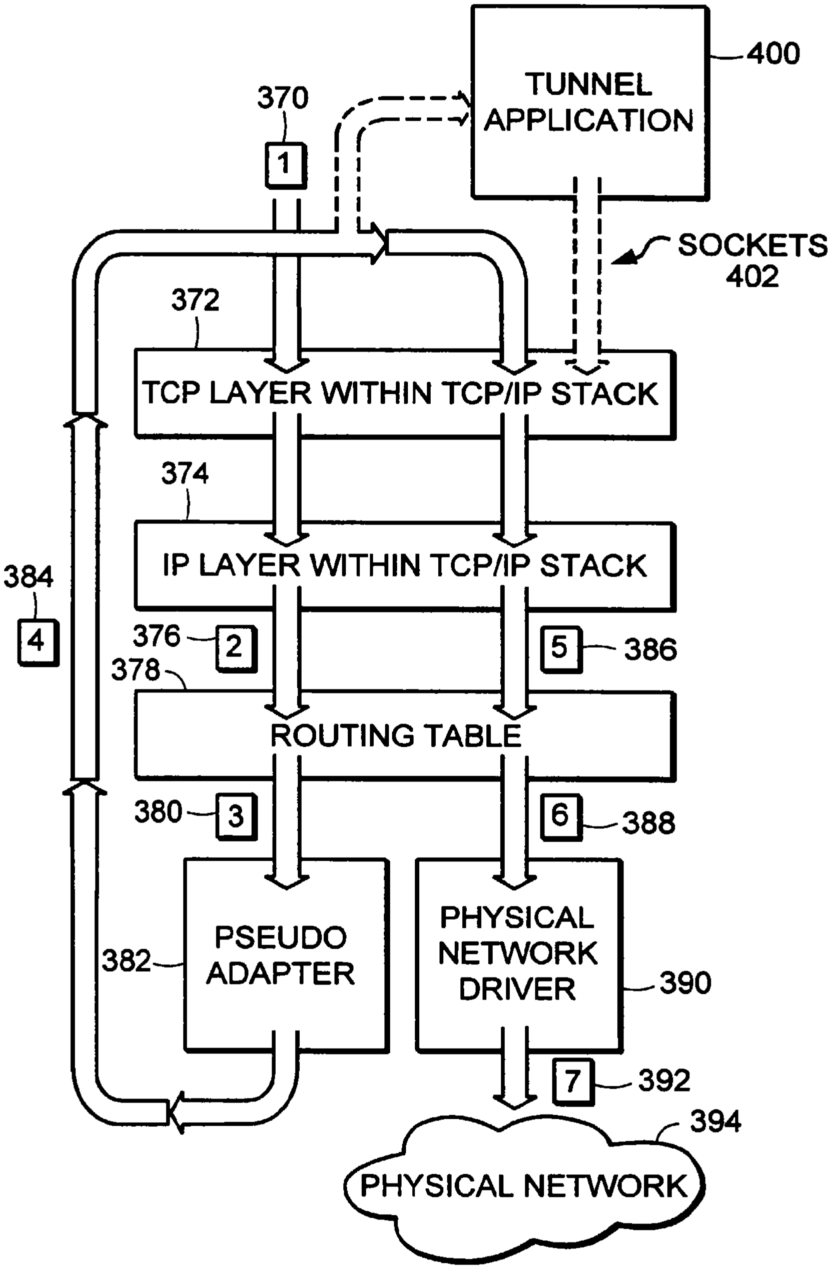

400
TUNNEL APPLICATION
370
1
SOCKETS 402
372
TCP LAYER WITHIN TCP/IP STACK
374
IP LAYER WITHIN TCP/IP STACK
384
4
376
2
378
5
386
ROUTING TABLE
380
3
6
388
382
PSEUDO ADAPTER
PHYSICAL NETWORK DRIVER
390
7
392
394
PHYSICAL NETWORK

FIG. 19

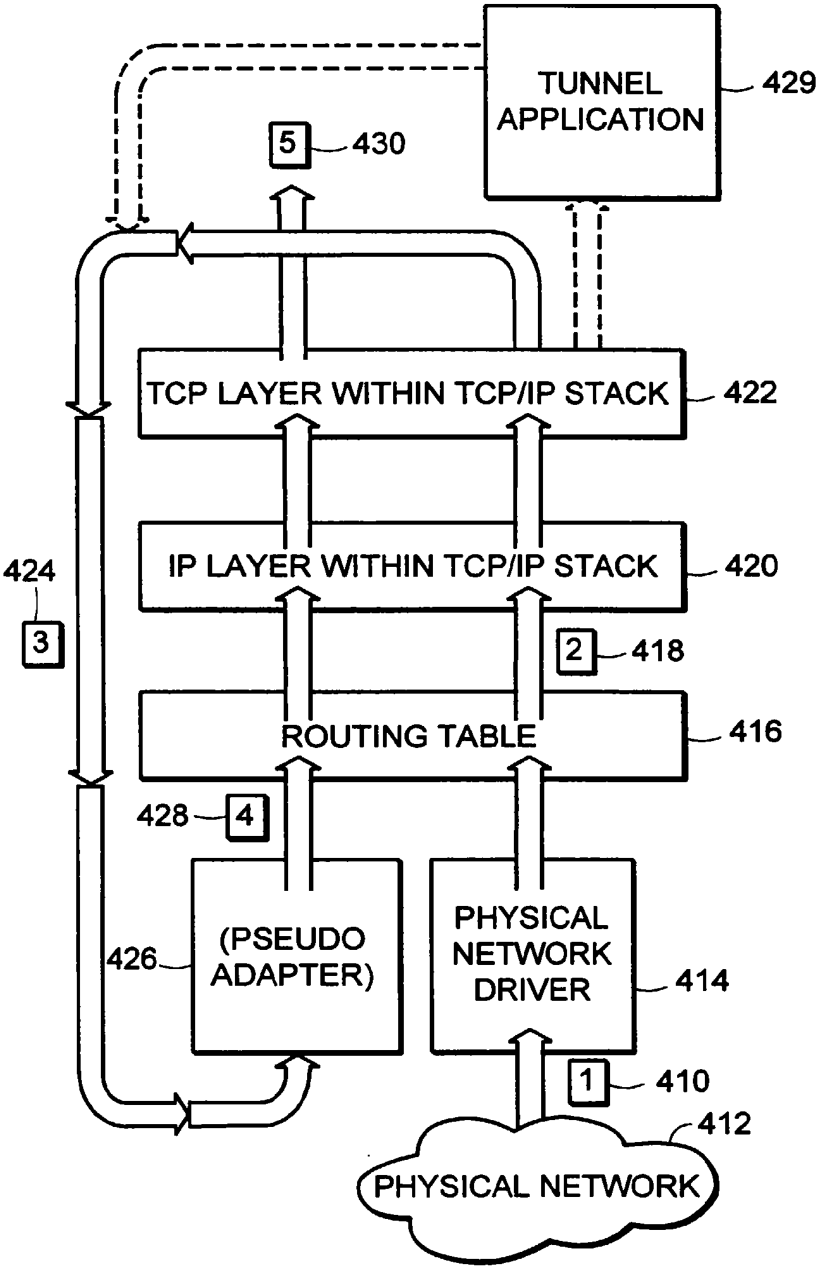

TUNNEL APPLICATION
429
5
430
TCP LAYER WITHIN TCP/IP STACK
422
IP LAYER WITHIN TCP/IP STACK
420
424
3
2
418
ROUTING TABLE
416
428
4
(PSEUDO ADAPTER)
426
PHYSICAL NETWORK DRIVER
414
1
410
412
PHYSICAL NETWORK

FIG. 20

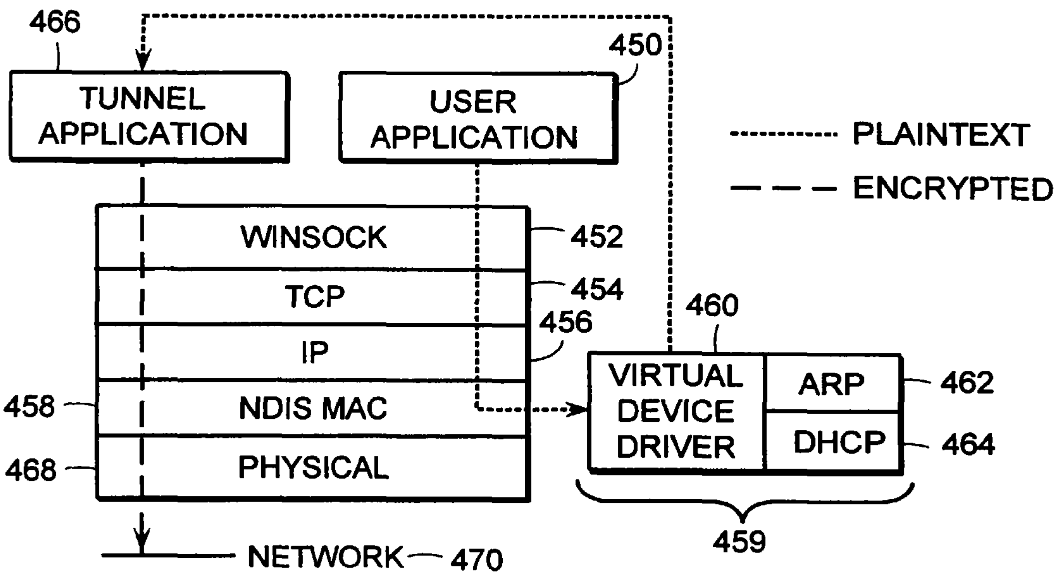

466
450
TUNNEL APPLICATION
USER APPLICATION
PLAINTEXT
ENCRYPTED
WINSOCK
452
TCP
454
456
IP
460
VIRTUAL DEVICE DRIVER
ARP
462
458
NDIS MAC
DHCP
464
468
PHYSICAL
459
NETWORK
470

FIG. 21

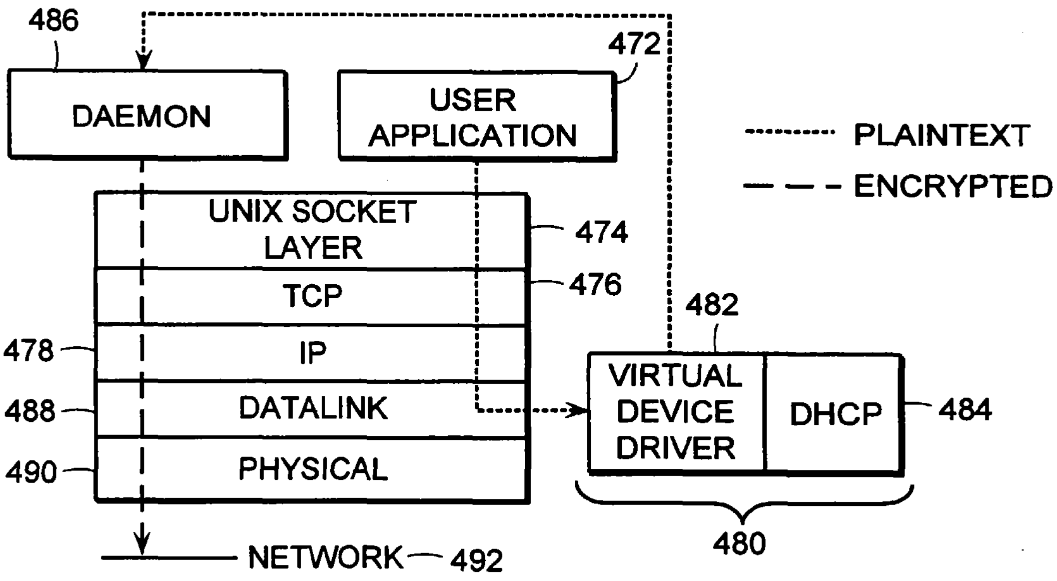

486
472
DAEMON
USER APPLICATION
PLAINTEXT
ENCRYPTED
UNIX SOCKET LAYER
474
TCP
476
482
478
IP
488
DATALINK
VIRTUAL DEVICE DRIVER
DHCP
484
490
PHYSICAL
480
NETWORK
492

FIG. 22

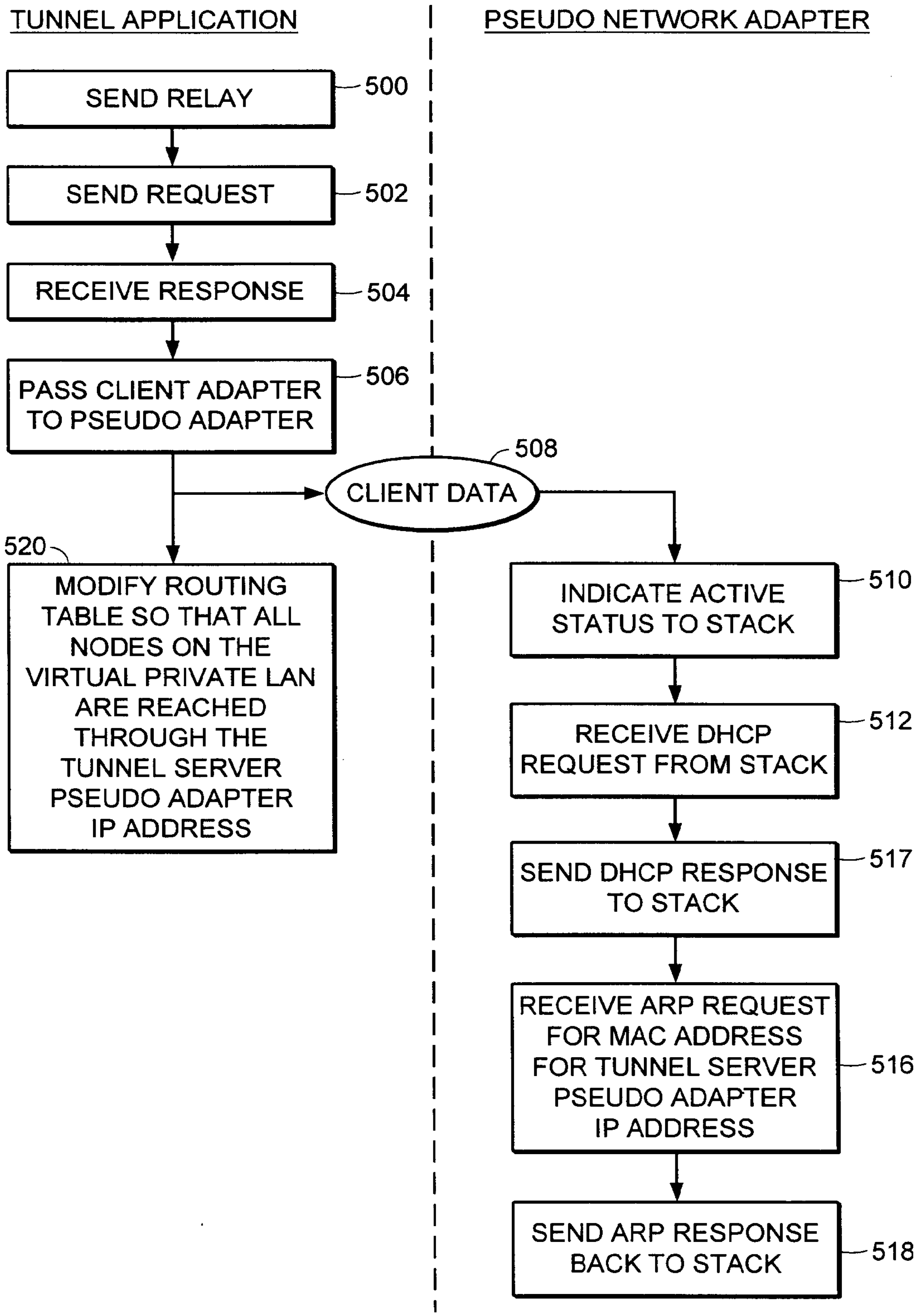

TUNNEL APPLICATION
PSEUDO NETWORK ADAPTER
SEND RELAY
500
SEND REQUEST
502
RECEIVE RESPONSE
504
PASS CLIENT ADAPTER TO PSEUDO ADAPTER
506
CLIENT DATA
508
520
MODIFY ROUTING TABLE SO THAT ALL NODES ON THE VIRTUAL PRIVATE LAN ARE REACHED THROUGH THE TUNNEL SERVER PSEUDO ADAPTER IP ADDRESS
INDICATE ACTIVE STATUS TO STACK
510
RECEIVE DHCP REQUEST FROM STACK
512
SEND DHCP RESPONSE TO STACK
517
RECEIVE ARP REQUEST FOR MAC ADDRESS FOR TUNNEL SERVER PSEUDO ADAPTER IP ADDRESS
516
SEND ARP RESPONSE BACK TO STACK
518

FIG. 23

GB2317792

Publication Title:

Virtual Private Network for encrypted firewall

Abstract:

Abstract of GB2317792

A system (10) for regulating the flow of messages through a firewall (18) having a network protocol stack, wherein the network protocol stack includes an Internet Protocol (IP) layer where if the message is not encrypted, it passes the unencrypted message up the network protocol stack to an application level proxy (50), and if the message is encrypted, it decrypts the message and passes the decrypted message up the network protocol stack to the application level proxy. The step of decrypting the message includes the step of executing a process at the IP layer to decrypt the message. Data supplied from the esp@cenet database - Worldwide

----
Courtesy of http://v3.espacenet.com

(12) UK Patent Application (19) GB (11) 2 317 792 (13) A

(43) Date of A Publication 01.04.1998

(21) Application No 9719816.2

(22) Date of Filing 17.09.1997

(30) Priority Data
(31) 08715343 (32) 18.09.1996 (33) US
08715668 18.09.1996

(71) Applicant(s)
Secure Computing Corporation

(Incorporated in USA - Delaware)

2675 Long Lake Road, Roseville,
Minnesota 55113-2536, United States of America

(72) Inventor(s)
Spence Minear
Edward B Stockwell
Troy De Jongh

(74) Agent and/or Address for Service
Beresford & Co
2-5 Warwick Court, High Holborn, LONDON,
WC1R 5DJ, United Kingdom

(51) INT CL^6
H04L 9/00

(52) UK CL (Edition P)
H4P PPEB
U1S S2124 S2209

(56) Documents Cited
WO 97/26735 A1 WO 97/26734 A1 WO 97/26731 A1
WO 97/23972 A1 WO 97/13340 A1

(58) Field of Search
UK CL (Edition P) H4P PDCSA PDCSC PPEB
INT CL^6 H04L 9/00 9/32 29/06 29/08
Online: WPI, INSPEC

(54) Virtual Private Network for encrypted firewall

(57) A system (10) for regulating the flow of messages through a firewall (18) having a network protocol stack, wherein the network protocol stack includes an Internet Protocol (IP) layer where if the message is not encrypted, it passes the unencrypted message up the network protocol stack to an application level proxy (50), and if the message is encrypted, it decrypts the message and passes the decrypted message up the network protocol stack to the application level proxy. The step of decrypting the message includes the step of executing a process at the IP layer to decrypt the message.

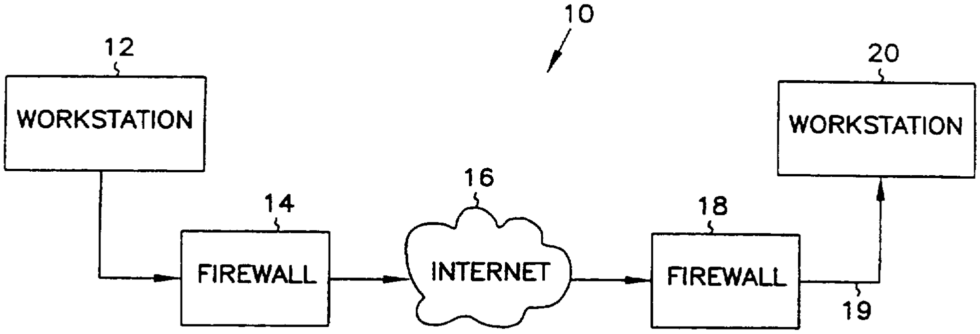


FIG. 1

GB 2 317 792 A

At least one drawing originally filed was informal and the print reproduced here is taken from a later filed formal copy.

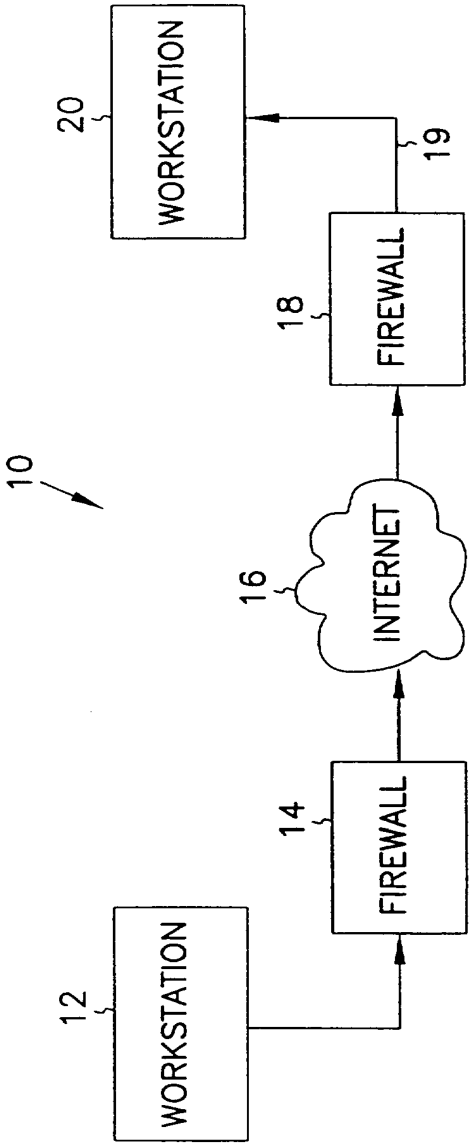
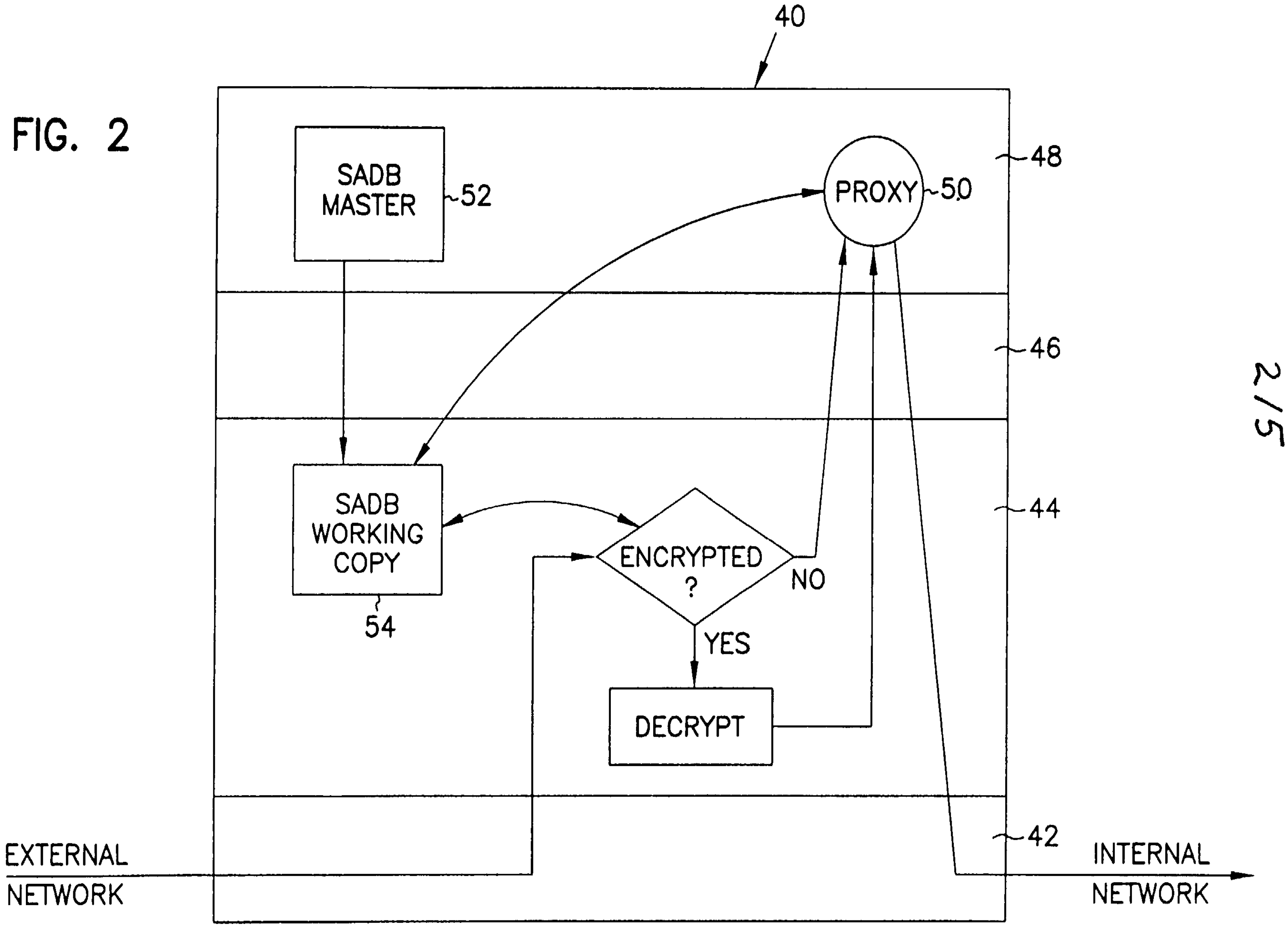

10
12
WORKSTATION
14
FIREWALL
16
INTERNET
18
FIREWALL
19
20
WORKSTATION

FIG. 1

FIG. 2


40
SADB MASTER
52
PROXY
50
48
46
SADB WORKING COPY
54
ENCRYPTED ?
NO
YES
DECRYPT
44
42
EXTERNAL NETWORK
INTERNAL NETWORK

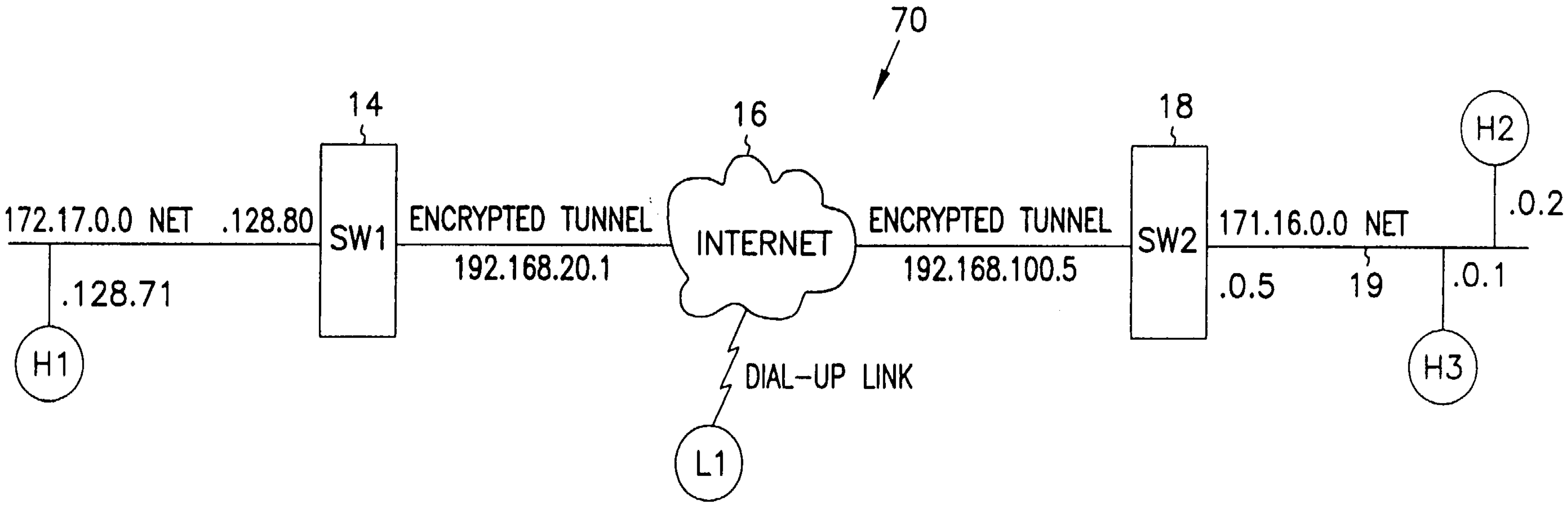

70
14
16
18
172.17.0.0 NET
.128.80
.128.71
H1
SW1
ENCRYPTED TUNNEL
192.168.20.1
INTERNET
DIAL-UP LINK
L1
ENCRYPTED TUNNEL
192.168.100.5
SW2
171.16.0.0 NET
.0.5
19
.0.1
.0.2
H2
H3

FIG. 3

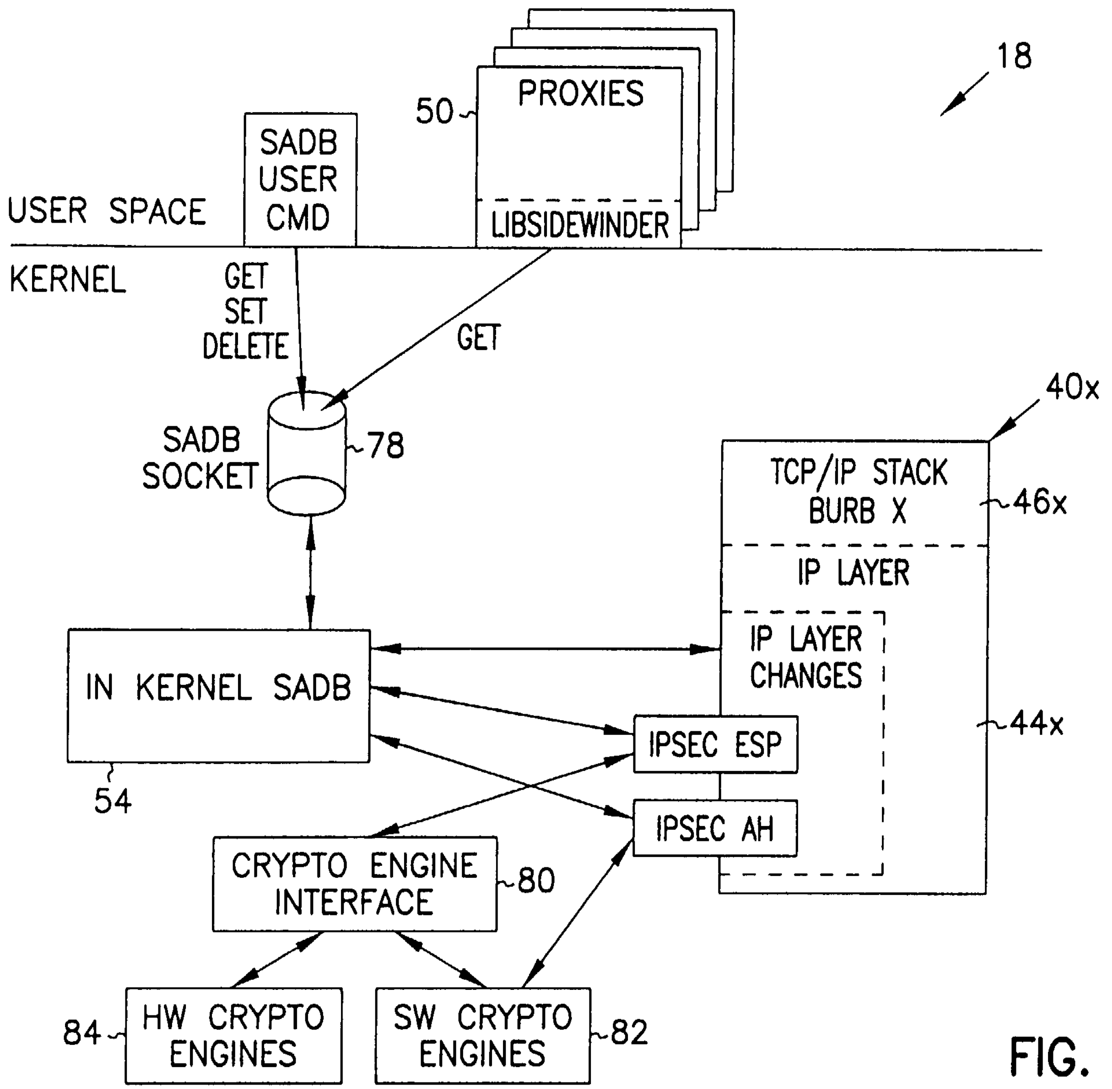

18
PROXIES
50
LIBSIDEWINDER
SADB
USER
CMD
USER SPACE
KERNEL
GET
SET
DELETE
GET
40x
SADB
SOCKET
78
TCP/IP STACK
BURB X
46x
IP LAYER
IP LAYER
CHANGES
44x
IN KERNEL SADB
54
IPSEC ESP
IPSEC AH
CRYPTO ENGINE
INTERFACE
80
HW CRYPTO
ENGINES
84
SW CRYPTO
ENGINES
82

FIG. 4

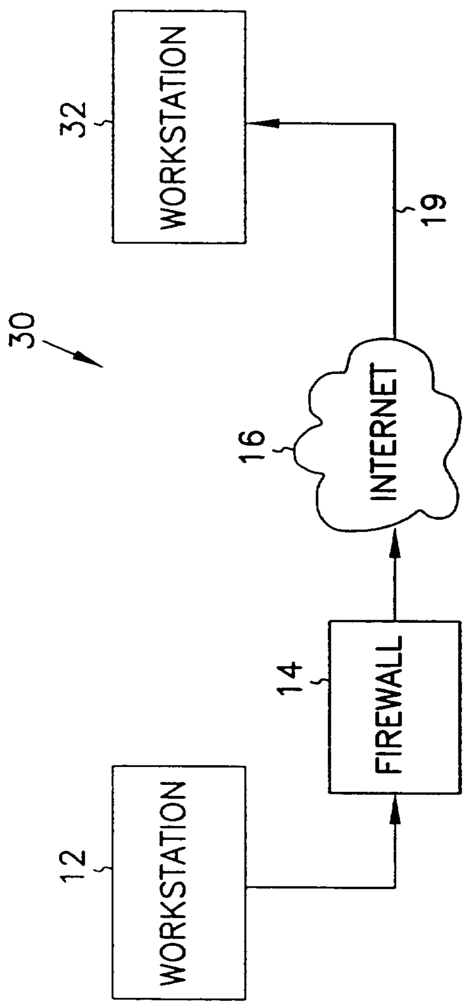

30
WORKSTATION
12
FIREWALL
14
INTERNET
16
19
WORKSTATION
32

FIG. 5

VIRTUAL PRIVATE NETWORK ON APPLICATION GATEWAY

Background of the Invention

Field of the Invention

The present invention pertains generally to network communications, and in particular to a system and method for securely transferring information between firewalls over an unprotected network.

Background Information

Firewalls have become an increasingly important part of network design. Firewalls provide protection of valuable resources on a private network while allowing communication and access with systems located on an unprotected network such as the Internet. In addition, they operate to block attacks on a private network arriving from the unprotected network by providing a single connection with limited services. A well designed firewall limits the security problems of an Internet connection to a single firewall computer system. This allows an organization to focus their network security efforts on the definition of the security policy enforced by the firewall. An example of a firewall is given in "SYSTEM AND METHOD FOR PROVIDING SECURE INTERNETWORK SERVICES" by Boebert et al. (PCT Published Application No. WO 96/13113, published on May 2, 1996), the description of which is hereby incorporated by reference. Another description of a firewall is provided by Dan Thomsen in "Type Enforcement: the new security model", *Proceedings: Multimedia: Full-Service Impact on Business, Education, and the Home*, SPIE Vol. 2617, p. 143, August 1996. Yet another such system is described in "SYSTEM AND METHOD FOR ACHIEVING NETWORK SEPARATION" by Gooderum et al. (PCT Published Application No. WO 97/29413, published on August 14, 1997), the description of which is hereby incorporated by reference. All the above systems are examples of application level gateways. Application level gateways use proxies or other such mechanisms operating at the application layer to process traffic through the firewall. As such, they can review not only the message traffic but also message content. In addition, they provide authentication and identification services, access control and auditing.

Data to be transferred on unprotected networks like the Internet is susceptible to electronic eavesdropping and accidental (or deliberate) corruption. Although a firewall can protect data within a private network from attacks launched from the unprotected network, even that data is vulnerable to both eavesdropping and corruption when transferred from the private network to an external machine. To address this danger, the Internet Engineering Task Force (IETF) developed a standard for protecting data transferred between firewalls over an unprotected network. The Internet Protocol Security (IPSEC) standard calls for encrypting data before it leaves the first firewall, and then decrypting the data when it is received by the second firewall. The decrypted data is then delivered to its destination, usually a user workstation connected to the second firewall. For this reason IPSEC encryption is sometimes called *firewall-to-firewall encryption* (FFE) and the connection between a workstation connected to the first firewall and a client or server connected to the second firewall is termed a *virtual private network*, or VPN.

The two main components of IPSEC security are data encryption and sender authentication. Data encryption increases the cost and time required for the eavesdropping party to read the transmitted data. Sender authentication ensures that the destination system can verify whether or not the encrypted data was actually sent from the workstation that it was supposed to be sent from. The IPSEC standard defines an encapsulated payload (ESP) as the mechanism used to transfer encrypted data. The standard defines an authentication header (AH) as the mechanism for establishing the sending workstation's identity.

Through the proper use of encryption, the problems of eavesdropping and corruption can be avoided; in effect, a protected connection is established from the internal network connected to one firewall through to an internal network connected to the second firewall. In addition, IPSEC can be used to provide a protected connection to an external computing system such as a portable personal computer.

IPSEC encryption and decryption work within the IP layer of the network protocol stack. This means that all communication between two IP addresses will be protected because all interfirewall communication must go through the IP layer. Such an approach is preferable over encryption and decryption at higher levels in the network protocol stack since when encryption is performed at layers higher than the IP layer more work is required to ensure that all supported communication is properly protected. In addition, since IPSEC encryption is handled below the Transport layer, IPSEC can encrypt data sent by any application. IPSEC therefore becomes a transparent add-on to such protocols as TCP and UDP.

Since, however, IPSEC decryption occurs at the IP layer, it can be difficult to port IPSEC to an application level gateway while still maintaining control at the proxy over authentication, message content, access control and auditing. Although the IPSEC specification in RFC 1825 suggests the use of a mandatory access control mechanism in a multi-level secure (MLS) network to compare a security level associated with the message with the security level of the receiving process, such an approach provides only limited utility in an application level gateway environment. In fact, implementations on application level gateways to date have simply relied on the fact that the message was IPSEC-encrypted as assurance that the message is legitimate and have simply decoded and forwarded the message to its destination. This creates, however, a potential chink in the firewall by assuming that the encrypted communication has access to all services.

What is needed is a method of handling IPSEC messages within an application level gateway which overcomes the above deficiencies. The method should allow control over access by an IPSEC connection to individual services within the internal network.

Summary of the Invention

The present invention is a system and method for regulating the flow of messages through a firewall having a network protocol stack, wherein the network protocol stack includes an Internet Protocol (IP) layer, the method comprising the steps of determining, at the IP layer, if a message is encrypted, if the message is not encrypted, passing the unencrypted message up the network protocol stack to an application level proxy, and if the message is encrypted, decrypting the message and passing the decrypted message up the network protocol stack to the application level proxy, wherein the step of decrypting the message includes the step of executing a procedure at the IP layer to decrypt the message.

According to another aspect of the present invention, a system and method is described for authenticating the sender of a message within a computer system having a network protocol stack, wherein the network protocol stack includes an Internet Protocol (IP) layer, the method comprising the steps of determining, at the IP layer, if the message is encrypted, if the message is encrypted, decrypting the message, wherein the step of decrypting the message includes the step of executing a procedure at the IP layer to decrypt the message, passing the decrypted message up the network protocol stack to an application level proxy, determining an authentication protocol appropriate for the message, and executing the authentication protocol to authenticate the sender of the message.

Brief Description of the Drawings

In the following detailed description of example embodiments of the invention, reference is made to the accompanying drawings which form a part hereof, and which is shown by way of illustration only, specific embodiments in which the invention may be practiced. It is to be understood that other embodiments may be utilized and structural changes may be made without departing from the scope of the present invention.

In the drawings, where like numerals refer to like components throughout the several views:

Figure 1 is a functional block diagram of an application level gateway-implemented firewall-to-firewall encryption scheme according to the present invention;

Figure 2 is a block diagram showing access control checking of both encrypted and unencrypted messages in network protocol stack according to the present invention;

Figure 3 is a block diagram of a representative application level gateway-implemented firewall-to-firewall encryption scheme;

Figure 4 is a block diagram of one embodiment of a network-separated protocol stack implementing IPSEC according to the present invention; and

Figure 5 is a functional block diagram of a firewall-to-workstation encryption scheme according to the present invention.

Description of the Preferred Embodiments

In the following detailed description of the preferred embodiment, references made to the accompanying drawings which form a part hereof, and in which is shown by way of illustration specific preferred embodiments in which the invention may be practiced. These embodiments are described in sufficient detail to enable those skilled in the art to practice the invention, and it is to be understood that other embodiments may be utilized and that structural, logical, physical, architectural, and electrical changes may be made without departing from the spirit and scope of the present invention. The following detailed description is, therefore, not to be taken in a limiting sense, and the scope of the present invention is defined only by the appended claims and their equivalents.

A system 10 which can be used for firewall-to-firewall encryption (FFE) is shown in Figure 1. In Figure 1, system 10 includes a workstation 12 communicating through a firewall 14 to an unprotected network 16 such as the Internet. System 10 also includes a workstation 20 communicating through a firewall 18 to unprotected network 16. In one embodiment, firewall 18 is an application level gateway.

As noted above, IPSEC encryption and decryption work within the IP layer of the network protocol stack. This means that all communications between two IP addresses will be protected because all interfirewall communication must pass through the IP layer. IPSEC takes the standard

Internet packet and converts it into a carrier packet. The carrier packet is designed to do two things: to conceal the contents of the original packet (encryption) and to provide a mechanism by which the receiving firewall can verify the source of the packet (authentication). In one embodiment of the present invention, each IPSEC carrier packet includes both an authentication header used to authenticate the sending machine and an encapsulated payload containing encrypted data. The authentication header and the encapsulated payload features of IPSEC can, however, be used independently. As required in RFC 1825, DES-CBC is provided for use in encrypting the encapsulated payload while the authentication header uses keyed MD5.

To use IPSEC, you must create a *security association* (SA) for each destination IP address. In one embodiment, each SA contains the following information:

- Security Parameters Index (SPI) - The index used to find a SA on receipt of an IPSEC datagram.
- Destination IP address - The address used to find the SA and trigger use of IPSEC processing on output.
- The peer SPI - The SPI value to put on a IPSEC datagram on output.
- The peer IP address - The destination IP address to be put into the packet header if IPSEC Tunnel mode is used.
- The Encryption Security Payload (ESP) algorithm to be used.
- The ESP key to used for decryption of input datagrams.
- The ESP key to used for encryption of output datagrams.
- The authentication (AH) algorithm to be used.
- The AH key to be used for validation of input packets.
- The AH key to be used for generation of the authentication data for output datagrams.

The combination of a given Security Parameter Index and Destination IP address uniquely identifies a particular "Security Association." In one embodiment, the sending firewall uses the sending userid and Destination Address to select an appropriate Security Association (and hence SPI value). The receiving firewall uses the combination of SPI value and Source address to obtain the appropriate Security Association.

A security association is normally one-way. An authenticated communications session between two firewalls will normally have two Security Parameter Indexes in use (one in each direction). The combination of a particular Security Parameter Index and a particular Destination Address uniquely identifies the Security Association.

More information on the specifics of an IPSEC FFE implementation can be obtained from the standards developed by the IPSEC work group and documented in *Security Architecture for IP* (RFC 1825) and in RFC's 1826-1829.

When a datagram is received from unprotected network 16 or is to be transmitted to a destination across unprotected network 16, the firewall must be able to determine the algorithms, keys, etc. that must be used to process the datagram correctly. In one embodiment, this information is obtained via a security association lookup. In one such embodiment, the lookup routine is passed several arguments: the source IP address if the datagram is being received from network 16 or the destination IP address if the datagram is to be transmitted across network 16, the SPI, and a flag that is used to indicate whether the lookup is being done to receive or transmit a datagram.

When an IPSEC datagram is received by firewall 18 from unprotected network 16, the SPI and source IP address are determined by looking in the datagram. In one embodiment a Security Association Database (SADB) stored within firewall 18 is searched for the entry with a matching SPI. In one such embodiment, security associations can be set up based on network address as well as a more granular host address. This allows the network administrator to create a security association between two firewalls with only a couple of lines in a configuration file on each machine. For such embodiments, the entry in the Security Association Database that has both the matching SPI and the longest address match is selected as the SA cntry. In another such embodiment, each SA has a prefix length value associated with the address. An address match on a SA entry means that the addresses match for the number of bits specified by the prefix length value.

There are two exceptions to this search process. First, when an SA entry is set marked as being dynamic it implies that the user of this SA may not have a fixed IP address. In this case the match is fully determined by the SPI value. Thus it is necessary that the SPI values for such SA entries be unique in the SADB. The second exception is for SA entries marked as tunnel mode entries. In this case it is normally the case that the sending entity will hide its source address so that all that is visible on the public wire is the destination address. In this case, like in the case where the SA entries are for dynamic IP addresses, the search is done exclusively on the basis of the SPI.

When transmitting a datagram across unprotected network 16 the SADB is searched using only the destination address as an input. In this case the entry which has the longest address match is selected and returned to the calling routine.

In one embodiment, if firewall 18 receives datagrams which are identified as either an IP_PROTO_IPSEC_ESP or IP_PROTO_IPSEC_AH protocol datagram, there must be a corresponding SA in the SADB or else firewall 18 will drop the packet and an audit message will be generated. Such an occurrence might indicate a possible attack or it might simply be a symptom of an erroneous key entry in the Security Association Database.

In a system such as system 10, application level gateway firewall 18 acts as a buffer between unprotected network 16 and workstations such as workstation 20. Messages coming from unprotected network 16 are reviewed and a determination is made as to whether execution of an authentication and identification protocol is warranted. In contrast to previous systems, system 10 also performs this same determination on IPSEC-encrypted messages. If desired, the same authentication and identification can be made on messages to be transferred from workstation 20 to unprotected network 16. Figure 2 illustrates one way of authenticating both encrypted and unencrypted messages in a system such as system 10.

In the system of Figure 2 a network protocol stack 40 includes a physical layer 42, an Internet protocol (IP) layer 44, a Transport layer 46 and an application layer 48. Such a protocol stack exists, for instance on application level gateway firewall 18 of Figure 1. An application executing in application layer 48 can communicate to an application executing on another system by preparing a message and transmitting it through one of the existing transport services executing on transport layer 46. Transport layer 46 in turn uses a process executing in IP layer 44 to continue the transfer. Physical layer 42 provides the software needed to transfer data through the communication hardware (e.g., a network interface card or a modem). As noted above, IPSEC executes within IP layer 44. Encryption and authentication is transparent to the host as long as the network administrator has the Security Association Database correctly configured and a key management mechanism is in place on the firewall.

In application level gateway firewall 18, a proxy 50 operating within application layer 48 processes messages transferred between internal and external networks. All network-to-network traffic must pass through one of the proxies within application layer 48 before being the transfer across networks is allowed. A message arriving from external network 16 is examined at IP layer 44 and an SADB is queried to determine if the source address and SPI are associated with an SA. In the embodiment shown in Figure 2, an SADB Master copy 52 is maintained in persistent memory at application layer 48 while a copy 54 of SADB is maintained in volatile memory within the kernel. If the message is supposed to be encrypted, the message is decrypted based on the algorithm and key associated with the particular SA and the message is transferred up through transport layer 46 to proxy 50. Proxy 50 examines the source and destination addresses and the type of service desired and decides whether authentication of the sender is warranted. If so, proxy 50 initiates an authentication protocol. The protocol may be as simple as requesting a user name and password or it may include a challenge/response authentication process. Proxy 50 also looks to see whether the message coming in was encrypted or not and may factor that into whether a particular type of authentication is needed. In Telnet, for instance, user name/password authentication may be sufficient for an FFE link while the security policy may dictate that a more stringent challenge/response protocol is needed for unencrypted links. In that case, proxy 50 will be a Telnet proxy and it will base its authentication protocol on whether the link was encrypted or not.

Since IPSEC executes within IP layer 44 there is no need for host firewalls to update their applications. Users that already have IPSEC available on their own host machine will, however, have to request that the firewall administrator set up SA's in the SADB for their traffic.

In the embodiment shown in Figure 2, a working copy 54 of the Security Association Database consisting of all currently active SA's is kept resident in memory for ready access by IP layer processing as datagrams are received and transmitted. In addition, a working master copy 52 of the SADB is maintained in a file in nonvolatile memory. During system startup and initialization processing the content of all of the required SA's in master SADB 52 is added to the working copy 54 stored in kernel memory.

In one embodiment, firewall 18 maintains different levels of security on internal and external network interfaces. It is desirable for a firewall to have different levels of security on both the internal and external interfaces. In one embodiment, firewall 18 supports three different levels, numbered 0 through 2. These levels provide a simple policy mechanism that controls permission for both in-bound and out-bound packets.

- Level 0 - do not allow any in-bound or out-bound traffic unless there is a security association between the source and destination.

- Level 1 - Allow both in-bound and out-bound non-IPSEC traffic but force the use of IPSEC if a SA exists for the address. (To support this firewall 18 must look for a SA for each in-bound datagram.)

- Level 2 - allow NULL security associations to exist. NULL associations are just like normal security associations, except no encryption or authentication transform is performed on in-bound or out-bound packets that correspond to this NULL association. With Level 2 enabled, the machine will still receive unprotected traffic, but it will not transmit unless Level 1 is enabled.

The default protection level established when the Security Association Database (SADB) is initialized at boot time is 1 for in-bound traffic and 2 for out-bound traffic.

An Access Control List, or ACL, is a list of rules that regulate the flow of Internet connections through a firewall. These rules control how a firewall's servers and proxies will react to connection attempts. When a server or proxy receives an incoming connection, it performs an ACL check on that connection.

An ACL check compares a set of parameters associated with the connection against a list of ACL rules. The rules determine whether the connection is allowed or denied. A rule can also have one or more side effects. A side effect causes the proxy to change its behavior in some fashion. For example, a common side effect is to redirect the destination IP address to an alternate machine. In addition to IP connection attempts, ACL checks can also made on the console logins and on logins made from serial ports. Finally, ACL checks can also be made on behalf of IP access devices, such as a Cisco box, through the use of the industry standard TACACS+ protocol.

In one embodiment, the ACL is managed by an acld daemon running in the kernel of firewalls 10 and 30. The acld daemon receives two types of requests, one to query the ACL and one to administer it. In one such embodiment, the ACL is stored in a relational database such as the Oracle database for fast access. By using such a database, query execution is asynchronous and many queries can be executing concurrently. In addition, these types of databases are designed to manipulate long lists of rules quickly and efficiently. These qualities ensure that a given query cannot hang up the process that issued the query for any appreciable time (> 1-2 seconds).

In one such embodiment, the database can hold up to 100,000 users and up to 10,000 hosts but can be scaled up to the capacity of the underlying database engine. The results of an ACL check is cached, allowing repeated checks to be turned around very quickly.

Applications on firewalls 10 and 30 can query acld to determine if a given connection attempt should be allowed to succeed. In one embodiment, the types of applications (i.e. "agents") that can make ACL queries can be divided into four classes:

1) Proxies. These allow connections to pass through firewall 10 or 30 in order to provide access to a remote service. They include tnauthp (authenticated telnet proxy), pftp (FTP proxy), httpp (HTTP proxy), and tcpgsp (TCP generic service proxy).

2) Servers. These provide a service on the firewall itself. They include ftpd and httpd.

3) Login agents. Login agent is a program on the firewall that can create a Unix shell. It is not considered a server because it cannot receive IP connections. One example is /usr/bin/login when used to create a dialup session or a console session on firewall 10 or 30. Another example is the command *srole*.

4) Network Access Servers (NAS). NAS is a remote IP access device, typically a dialup box manufactured by such companies as Cisco or Bridge. The NAS usually provides dialup telnet service and may also provide SLIP or PPP service.

Proxies, servers, login agents, and NASes make queries to acld to determine if a given connection attempt should be allowed to succeed. All of the agents except NAS make their queries directly. NAS, because it is remote, must communicate via an auxiliary daemon that typically uses an industry standard protocol such as RADIUS or TACACS+. The auxiliary daemon (e.g., tacradd) in turn forwards the query to local acld.

As a side effect of the query, acld tells the agent if authentication is needed. If no authentication is needed, the connection proceeds immediately. Otherwise acld provides (as another side effect) a list of allowed authentication methods that the user can choose from. The agent can present a menu of choices or simply pick the first authentication method by default. Typical authentication methods include plain password, SNK DSS, SDI SecurID, LOCKout DES, and LOCKout FORTEZZA. In one embodiment, the list of allowed authentication methods varies depending on the host name, user name, time of day, or any combination thereof.

In the case of a Level 0 policy, it would be safe to assume that all incoming traffic is encrypted or authenticated. In the case of Levels 1 through 2, a determination must be made whether or not a security association exists for a given peer. Otherwise an application may believe that in-bound traffic has been authenticated when it really has not. (That is why it is necessary to look for an SA on input of each non-IPSEC datagram.)

In one embodiment, a flag which accompanies the message as it is sent from IP layer 44 to proxy 50 indicates whether the incoming message was or was not encrypted. In another embodiment, proxy 50 accesses Security Association Database 54 (the table in the kernel can be queried via an SADB routing socket (PF-SADB)) to determine whether or not a security association exists for a given peer.. The SADB socket is much like a routing socket found in the stock BSD 4.4 kernel (protocol family PF-ROUTE) except that PF-SADB sockets are used to maintain the Security Association Database (SADB) instead of the routing table. Because the private keys used for encryption, decryption, and keyed authentication are stored in this table, access must be strictly prohibited and allowed to only administrators and key management daemons. Care must be taken when allowing user-level daemons access to `/dev/mem` or `/dev/kmem` as well, since the keys are stored in kernel memory and could be exposed with some creative hacking.

In one embodiment, a command-line tool called `sadb` is used to support the generation and maintenance of in-kernel version 54 of SADB. The primary interface between this tool and the SADB is the PF-SADB socket. The kernel provides socket processing to receive client requests to add, update, or change entries in in-kernel SADB 54. As noted above, the default protection level established when the Security Association Database (SADB) is initialized at boot time is 1 for in-bound traffic and 2 for out-bound traffic. This may be changed by the use of the sadb command.

The existing sadb command was derived from the NIST implementation of IPSEC. As noted above, this tool is much like route in that it uses a special socket to pass data structures in and out of the kernel. There are three commands recognized by the sadb command: *get, set, delete.* The following simple shell script supports adding and removing a single SA entry to SADB 54. It shows one embodiment of a parameter order for adding a SA to the SADB.

```
# ! /bin/sh
if [ $# -ne 1 ]
then
     echo "usage: $0 <on>|<off>" >&2
     exit 1
fi
ONOFF=$1

addsa ()
{
IPADDRESS=$2
PEERADDRESS=0.0.0.0
PREFIXLEN=0                # Num of bits, 0 => full 32
bit match
LOCALADDRESS=0.0.0.0
REALADDRESS=0.0.0.0
PORT=0
PROTOCOL=0
UID=0
DESALG=1                   # I = DES-CBC
IVLEN=4                    # bytes
DESKEY=0b0b0b0b0b0b0b0b
DESKEYLEN=8                # bytes
AHALG=1                    # 1 = MD5
AHKEY=30313233343536373031323334353637
AHKEYLEN=16                # bytes
LOCAL_SPI=$1
```

```
PEER_SPI=$1
TUNNEL_MODE=0
AHRESULTLEN=4
COMBINED_MODE=1             # On output, 1 = ESP, then
AH; 0 = AH, then ESP
DYNAMIC_FLAG=0

if [ "$ONOFF" = "on"
then
     ./sadb add dst $IPADDRESS $PREFIXLEN $LOCAL_SPI
$UID $PEERADDRESS $PEER_SPI $TUNNEL_MODE $LOCALADDRESS
$REALADDRESS $PROTOCOL $PORT $DESALG $IVLEN $DESKEYLEN
$DESKEY $DESKEYLEN $DESKEY $AHALG $AHKEYLEN $AHKEY
$AHKEYLEN $AHKEY $AHRESULTLEN $COMBINED_MODE
$DYNAMIC_FLAG
else
     ./sadb delete dst $IPADDRESS $LOCAL-SPI
fi
}

#    Get down to work:
addsa 500 172.17.128.115            # number6.sctc.com
```

The current status of in-kernel SADB 54 can be obtained with the sadb command. The get option allows dumping the entire SADB or a single entry. In one embodiment, the complete dump approach uses /dev/kmem to find the information. The information may be presented as follows:

```
# sadb get dst

Local-SPI Address-Family Destination-Addr
Preflx_length UID
     Peer-Address Peer-SPI Transport-Type
     Local-Address Real-Address
     Protocol Port
     ESP_Alg_ID ESP_IVEC_Length
          ESP_Enc_Key_length ESP_Enc_ESP_Key
          ESP_Dec_Key_length ESP_Dec_ESP_Key
     AH_Alg_ID AH_Data_Length
          AH_Gen_Key_Length AH_Gen_Key
          AH_Check_Key_Length AH_Check_Key
     Combined_Mode  Dynamic_Flag
```

```
--------------------------------------------------------------
-
500 INET: number6.sctc.com 0 0
     0.0.0.0   500 Transport(0) 0
     0.0.0.0 0.0.0.0
     None None
     DES/CBC-RFC1829(1) 4
          8 0b0b0b0b0b0b0b0b
          8 0b0b0b0b0b0b0b0b
     MD5-RFC1828(1) 4
          16 30313233343536373031323334353637
          16 30313233343536373031323334353637
     ESP+AH(1) 0
501 INET: spokes.sctc.com 0 0
     0.0.0.0   501 Transport(0) 0
     0.0.0.0.0.0.0.0
     None None
     DES/CBC-RFC1829(1) 4
          8 0b0b0b0b0b0b0b0b
          8 0b0b0b0b0b0b0b0b
     MD5-RFC1828(1) 4
          16 30313233343536373031323334353637
          16 30313233343536373031323334353637
     ESP+AH(1) 0

End of list.
```

When a new entry is added to in-kernel SADB 54, the add process first checks to see that no existing entry will match the values provided in the new entry. If no match is found then the entry is added to the end of the existing SADB list.

To illustrate the use and administration of an FFE, we'll go through an example using FFE 70 in Figure 3. Firewalls 14 and 18 are both application level gateway firewalls implemented according to the present invention. Workstations H2 and H3 both want to communicate with H1. For the administrator of firewalls 14 and 18, this is easy to accomplish. The administrator sets up a line something like this (we'll only show the IP address part and SPI parts of the SA, since they're the trickiest values to configure. Also, assume that we are using tunnel mode):

```
#  Hypothetical SW1 Config File
```

```
#
#  Fields are laid out in the following manner:
#  srcaddrornet= localSPI= peeraddr= peerSPI=
realsrcaddr= localaddr= key=

# The following entry sets up a tunnel between hosts
behind SW1
# and hosts behind SW2.
src=172.16.0.0 localSPI=666 peer=192.168.100.5
peerSPI=777 \
     realsrcaddr=192.168.100.5 localaddrs=0.0.0.0
     key=0xdeadbeeffadebabe

#  Hypothetical SW2 Config File
#
#  Fields are laid out in the following manner:
#  srcaddrornet= localSPI= peeraddr= peerSPI=
     realsrcaddr= localaddr= key=

#  The following entry sets up a tunnel between hosts
behind SW1 and
#  hosts behind SW2.
src=172.17.0.0 localSPI=777 peer=192.168.20.1
peerSPI=666 \
     realsrcaddr=192.168.20.1 localaddr=0.0.0.0 \
     key=Oxdeadbeeffadebabe
```

With this setup, all traffic is encrypted using one key, no matter who is talking to whom. For example, traffic from H2 to H1 as well as traffic from H3 to H1 will be encrypted with one key. Although this setup is small and simple, it may not be enough.

What happens if H2 cannot trust H3? In this case, the administrator can set up security associations at the host level. In this case, we have to rely on the SPI field of the SA, since the receiving firewall cannot tell from the datagram header which host behind the sending firewall sent the packet. Since the SPI is stored in IPSEC datagrams, we can do a lookup to obtain its value. Below are the sample configuration files for both firewalls again, but this time, each host combination communicates with a different key. Moreover, H2 excludes H3 from communications with H1, and H3 excludes H2 in the same way.

```
#  Hypothetical SW1 Config File
#
#  Fields are laid out in the following manner:
#  srcaddrornet= localSPI= peeraddr= peerSPI=
realsrcaddr= localaddr= key=

# The following entry sets up a secure link between H2
and H1
src=172.16.0.2 localSPI=666 peer=192.168.100.5
peerSPI=777 \
     realsrcaddr=192.168.100.5
localaddrs=178.17.128.71 \
     key=0x0a0a0a0a0a0a0a0a

# The following entry sets up a secure link between H3
and H1
src=172.16.0.1 localSPI=555 peer=192.168.100.5
peerSPI=888 \
     realsrcaddr=192.168.100.5
localaddrs=178.17.128.71 \
     key=0x0b0b0b0b0b0b0b0b

#  Hypothetical SW2 Config File
#
#  Fields are laid out in the following manner:
#  srcaddrornet= localSPI= peeraddr= peerSPI=
realsrcaddr= localaddr= key=

# The following entry sets up a secure link between H2
and H1
src=172.17.128.71 localSPI=777 peer=192.168.20.1
peerSPI=666 \
     realsrcaddr=192.168.20.1 localaddrs=172.16.0.2 \
     key=0x0a0a0a0a0a0a0a0a

# The following entry sets up a secure link between H3
and H1
src=172.17.128.71 localSPI=888 peer=192.168.20.1
peerSPI=555 \
     realsrcaddr=192.168.20.1 localaddrs=172.16.0.1 \
     key=0x0b0b0b0b0b0b0b0b
```

Figure 4 is a block diagram showing in more detail one embodiment of an IPSEC-enabled application level gateway firewall 18. Application level gateway firewall 18 provides access control checking of both encrypted and unencrypted messages in a more secure environment due to its network-separated architecture. Network separation divides a system into a set of independent regions or burbs, with a domain and a protocol stack assigned to each burb. Each protocol stack 40x has its own independent set of data structures, including routing information and protocol information. A given socket will be bound to a single protocol stack at creation time and no data can pass between protocol stacks 40 without going through proxy space. A proxy 50 therefore acts as the go-between for transfers between domains. Because of this, a malicious attacker who gains control of one of the regions is prevented from being able to compromise processes executing in other regions. Network separation and its application to an application level gateway is described in "SYSTEM AND METHOD FOR ACHIEVING NETWORK SEPARATION", U.S. Application No. 08/599,232, filed February 9, 1996 by Gooderum et al.

In the system shown in Figure 4, the in-bound and out-bound datagram processing of a security association continues to follow the conventions defined by the network separation model. Thus all datagrams received on or sent to a given burb remain in that burb once decrypted. In one such embodiment SADB socket 78 has been defined to have the type 'sadb'. Each proxy 50 that requires access to SADB socket 78 to execute its query as to whether the received message was encrypted must have create permission to the sadb type.

The following is list of specific requirements that a system such as is shown in Figure 4 must provide. Many of the requirements were discussed in the information provided earlier in this document.

1. Firewall applications may query the IPSEC subsystem to determine if traffic with a given address is guaranteed to be encrypted.
2. Receipt of an unencrypted datagram from an address that has a SA results in the datagram being dropped and an audit message being generated.
3. On receipt of encrypted protocol datagrams the SADB searches will be done using the SPI as the primary key. The source address will a secondary key. The SA returned by the search will be the SA which matches the SPI exactly and has the longest match with the address.

4. A search of the SADB for a SPI that finds an entry that is marked as SA for a dynamic IP will not consider the address in the search process.

5. A search of the SADB for a SPI that finds an entry that is marked as a SA for a tunnel mode connection will to consider the address if it is (0.0.0.0) i.e INADDR.

6. On receipt of a non-IPSEC datagram the SADB will be searched for an entry that matches the src address. If a SA is found the datagram will be dropped and an audit message sent.

7. SADB searches on output will be done using the DST address as key. If morc than one SA entry in the SADB has that address the first one with the maximum address match will be returned.

8. The SADB must be structured so that searches are fast regardless if the search is done by SPI or by address.

9. The SADB must provide support for connections to a site with a fixed SPI but changing IP address. SA entries for such connections will be referred to as Dynamic Address Sites, or just Dynamic entries.

10. When a dynamic entry is found by a SPI search, the current datagram's SRC address, which is required to ensure that the return datagrams are properly encrypted, will be recorded in the SA only after the AH checking has passed successfully. (This is because if the address is recorded before AH passes then an attacker can cause return packets of an outgoing connection to be transmitted in the clear.)

11. A failure of an AH check on a dynamic entry results in an audit message.

12. In an embodiment where the firewall requires that all connections use both AH and ESP, on receipt the order should be AH first ESP second.

13. The processing structure on both input and output should try to minimize the number of SADB required lookups.

Returning to Figure 4, in one embodiment firewall 18 includes a crypto engine interface 80 used to encrypt an IPSEC payload. Crypto engine interface 80 may be connected to a software encryption engine 82 or to a hardware encryption engine 84. Engines 82 and 84 perform the actual encryption function using, for example, DES-CBC. In addition, software encryption engine 82 may include the keyed MD5 algorithm used for AH.

In one embodiment, crypto engine interface 80 is a utility which provides a consistent interface between the software and hardware encryption engines. As shown in Figure 4, in one such embodiment interface 80 only supports the use of the use of hardware cryptographic engine 84 for IPSEC ESP processing. The significant design issue that interface 80 must deal with is that use of a hardware encryption engine requires that the processing be down in disjoint steps operating in different interrupt contexts as engine 84 completes the various processing steps.

The required information is stored in a request structure that is bound to the IP datagram being processed. The request is of type `crypto_request_t`. This structure is quite large and definitely does not contain a minimum state set.

In addition to the definition of the request data structure, this software implementing interface 80 provides two functions which isolate the decision of which cryptographic engine to use. The `crypt_des_encrypt` function is for use by the IP output processing to encrypt a datagram. The `crypt_des_decrypt` function is for use by the IP input processing to decrypt a datagram. If hardware encryption engine 84 is present and other hardware usage criteria are met the request is enqueued on a hardware processing queue and a return code indicating that the cryptographic processing is in progress is returned. If software engine 82 is used, the return code indicates that the cryptographic processing is complete. In the former case, the continuation of the IP processing is delayed until after hardware encryption is done. Otherwise it is completed as immediately in the same processing stream.

There are two software cryptographic engines 82 provided in the IPSEC software. One provides the MD5 algorithm used by the IPSEC AH processing, and the other provides the DES algorithm used by the IPSEC ESP processing. This software can be obtained from the US Government IPSEC implementation.

In one embodiment hardware cryptographic engine 84 is provided by a Cylink SafeNode processing board. The interface to this hardware card is provided by the Cylink device driver. A significant aspect of the Cylink card that plays a major part in the design of the IPSEC Cylink driver is that the card functions much like a low level subroutine interface and requires software support to initiate each processing step. Thus to encrypt or decrypt an individual datagram there are a minimum of two steps, one to set the DES initialization vector and one to do the encryption. Since the IP processing can not suspend itself and wait while the hardware completes and then be rescheduled by the hardware interrupt handler, in one embodiment a finite state machine is used to tie sequences of hardware processing elements together. In one such embodiment the interrupt handler looks at the current state, executes a defined after state function, transitions to the state and then executes that state's start function.

One function, `cyl_enqueue_request`, is used to initiate either an encrypt or a decrypt action. This function is designed to be called by cryptographic engine interface 80. All of the information required to initiate the processing as well as the function to be performed after the encryption operation is completed is provided in the request structure. This function will enqueue the request on the hardware request queue and start the hardware processing if necessary.

A system 30 which can be used for firewall-to-workstation encryption is shown in Figure 5. In Figure 5, system 30 includes a workstation 12 communicating through a firewall 14 to an unprotected network 16 such as the Internet. System 30 also includes a workstation 32 communicating directly with firewall 14 through unprotected network 16. Firewall 14 is an application level gateway incorporating IPSEC handling as described above. (It should be noted that IPSEC security cannot be used to authenticate the personal identity of the sender for a firewall to firewall transfer. When IPSEC is used, however, on a single user machine such as a portable personal computer, IPSEC usage should be protected with a personal identification number (PIN). In these cases IPSEC can be used to help with user identification to the firewall.)

According to the IPSEC RFC's, you can use either tunnel or transport mode with this embodiment based on your security needs. In certain situations, the communications must be sent in tunnel mode to hide unregistered addresses.

Although specific embodiments have been illustrated and described herein, it will be appreciated by those of ordinary skill in the art that any arrangement which is calculated to achieve the same purpose may be substituted for the specific embodiment shown. This application is intended to cover any adaptations or variations of the present invention. Therefore, it is intended that this invention be limited only by the claims and the equivalents thereof.

What is claimed is:

1. A method of regulating the flow of messages through a firewall having a network protocol stack, wherein the network protocol stack includes an Internet Protocol (IP) layer, the method comprising the steps of:

determining, at the IP layer, if a message is encrypted;

if the message is not encrypted, passing the unencrypted message up the network protocol stack to an application level proxy; and

if the message is encrypted, decrypting the message and passing the decrypted message up the network protocol stack to the application level proxy, wherein the step of decrypting the message includes the step of executing a procedure at the IP layer to decrypt the message.

2. A method of authenticating the sender of a message within a computer system having a network protocol stack, wherein the network protocol stack includes an Internet Protocol (IP) layer, the method comprising the steps of:

determining, at the IP layer, if the message is encrypted;

if the message is encrypted, decrypting the message, wherein the step of decrypting the message includes the step of executing a process at the IP layer to decrypt the message;

passing the decrypted message up the network protocol stack to an application level proxy;

determining an authentication protocol appropriate for the message; and

executing the authentication protocol to authenticate the sender of the message.

3. The method according to claim 2 wherein the step of determining an authentication protocol appropriate for the message includes the steps of:

determining a source IP address associated with the message; and

determining the authentication protocol associated with the source IP address.

4. The method according to claim 2 wherein the message includes security parameters index and wherein the step of determining an authentication protocol appropriate for the message includes the steps of:

determining the authentication protocol associated with a dynamic IP address, wherein the step of determining the authentication protocol includes the step of looking up a security association based on the security parameters index;

determining a current address associated with the dynamic source IP address; and

binding the current address to the security parameters index.

5. A firewall, comprising:

a first communications interface;

a second communications interface;

a network protocol stack connected to the first and the second communications interfaces, wherein the network protocol stack includes an Internet Protocol (IP) layer and a transport layer;

a decryption procedure, operating at the IP layer, wherein the decryption procedure decrypts encrypted messages received at one of said first and second communications interfaces and outputs decrypted messages; and

a proxy, connected to the transport layer of said network protocol stack, wherein the proxy receives decrypted messages from the decryption procedure and executes an authentication protocol based on the content of the decrypted message.

6. A firewall, comprising:

a first communications interface;

a second communications interface;

a first network protocol stack connected to the first communications interface, wherein the first network protocol stack includes an Internet Protocol (IP) layer and a transport layer;

a second network protocol stack connected to the second communications interface, wherein the second network protocol stack includes an Internet Protocol (IP) layer and a transport layer;

a decryption procedure, operating at the IP layer of the first network protocol stack, the decryption procedure receiving encrypted messages received by said first communications interface and outputting decrypted messages; and

a proxy, connected to the transport layers of said first and second network protocol stacks, the proxy receiving decrypted messages from the decryption procedure and executing an authentication protocol based on the content of the decrypted message.

7. The firewall according to claim 6 wherein the firewall further includes:

a third communications interface; and

a third network protocol stack connected to the third communications interface and to the proxy, wherein the third network protocol stack includes an Internet Protocol (IP) layer and a transport layer and wherein the second and third network protocol stacks are restricted to first and second burbs, respectively.

8. A method of establishing a virtual private network between a first and a second network, wherein each network includes an application level gateway firewall which uses a proxy operating at the application layer to process traffic through the firewall, wherein each firewall includes a network protocol stack and wherein each network protocol stack includes an Internet Protocol (IP) layer, the method comprising the steps of:

transferring a connection request from the first network to the second network;

determining, at the IP layer of the network protocol stack of the second network's firewall, if the connection request is encrypted;

if the connection request is encrypted, decrypting the request, wherein the step of decrypting the request includes the step of executing a procedure at the IP layer of the second network's firewall to decrypt the message;

passing the connection request up the network protocol stack to an application level proxy;

determining an authentication protocol appropriate for the connection request;

executing the authentication protocol to authenticate the connection request; and

if the connection request is authentic, establishing an active connection between the first and second networks.

9. The method according to claim 8 wherein the step of executing the authentication protocol includes the step of executing program code within the firewall of the second network to mimic a challenge/response protocol executing on a server internal to the second network.

10. The method according to claim 8 wherein the step of executing the authentication protocol includes the step of executing program code to execute the authentication protocol in line to the session.

11. The method according to claim 8 wherein the step of determining an authentication protocol includes the step of determining if the connection request arrived encrypted and selecting the authentication protocol based on whether the connection request was encrypted or not encrypted.

The Patent Office

28

Application No:	GB 9719816.2	**Examiner:**	B.J.SPEAR
Claims searched:	1-11	**Date of search:**	21 January 1998

Patents Act 1977
Search Report under Section 17

Databases searched:

UK Patent Office collections, including GB, EP, WO & US patent specifications, in:

UK Cl (Ed.P): H4P (PPEB,PDCSA,PDCSC)

Int Cl (Ed.6): H04L 9/00, 9/32, 29/06, 29/08

Other: Online:WPI, INSPEC

Documents considered to be relevant:

Category	Identity of document and relevant passage		Relevant to claims
XP	WO97/26734A1	(Raptor Systems) Whole document, eg Figs 1,3 and pages 6-12	1,2.5,6,8 at least
XP	WO97/26731A1	(Raptor Systems) Whole document, eg Figs 1,3 and pages 7-12	1,2.5,6,8 at least
XP	WO97/26735A1	(Raptor Systems) Whole document, eg Figs 1,3 and pages 4-10	1,2.5,6,8 at least
XP	WO97/23972A1	(V-ONE Corp) Whole document, eg Figs 1,2 and claim 1.	1,2.5,6,8 at least
XP	WO97/13340A1	(Digital Secured Networks) Whole document, eg pages 7-13	1,2.5,6,8 at least

X	Document indicating lack of novelty or inventive step	A	Document indicating technological background and/or state of the art.
Y	Document indicating lack of inventive step if combined with one or more other documents of same category.	P	Document published on or after the declared priority date but before the filing date of this invention.
&	Member of the same patent family	E	Patent document published on or after, but with priority date earlier than, the filing date of this application.

Electronic Acknowledgement Receipt

EFS ID:	1304381
Application Number:	10714849
International Application Number:	
Confirmation Number:	3154
Title of Invention:	Agile network protocol for secure communications using secure domain names
First Named Inventor/Applicant Name:	Victor Larson
Customer Number:	22907
Filer:	Steve S. Chang
Filer Authorized By:	
Attorney Docket Number:	000479.00111
Receipt Date:	09-NOV-2006
Filing Date:	18-NOV-2003
Time Stamp:	17:32:59
Application Type:	Utility

Payment information:

Submitted with Payment	no

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)	Multi Part /.zip	Pages (if appl.)
1	Information Disclosure Statement (IDS) Filed	00111IDS.pdf	137984	no	5
Warnings:					

Information:					
This is not an USPTO supplied IDS fillable form					
2	NPL Documents	wellsref.pdf	92941	no	1
Warnings:					
Information:					
3	Foreign Reference	WO9827783.pdf	851073	no	24
Warnings:					
Information:					
4	Foreign Reference	GB2334181.pdf	434487	no	15
Warnings:					
Information:					
5	Foreign Reference	EP0814589.pdf	1098918	no	20
Warnings:					
Information:					
6	Foreign Reference	EP0838930.pdf	1731451	no	35
Warnings:					
Information:					
7	Foreign Reference	GB2317792.pdf	1262522	no	35
Warnings:					
Information:					
8	NPL Documents	Davilaarticle.pdf	807564	no	18
Warnings:					
Information:					
Total Files Size (in bytes):				6416940	

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/714,849	11/18/2003	Victor Larson	000479.00111	3154

22907 7590 12/07/2006

BANNER & WITCOFF
1001 G STREET N W
SUITE 1100
WASHINGTON, DC 20001

EXAMINER	
LIM, KRISNA	
ART UNIT	PAPER NUMBER
2153	

DATE MAILED: 12/07/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	**Application No.** 10/714,849	**Applicant(s)** LARSON ET AL.	
	Examiner Krisna Lim	**Art Unit** 2153	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *1* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)□ Responsive to communication(s) filed on ____.

2a)□ This action is **FINAL**. 2b)□ This action is non-final.

3)□ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-27* is/are pending in the application.

4a) Of the above claim(s) ____ is/are withdrawn from consideration.

5)□ Claim(s) ____ is/are allowed.

6)□ Claim(s) ____ is/are rejected.

7)□ Claim(s) ____ is/are objected to.

8)☒ Claim(s) *1-27* are subject to restriction and/or election requirement.

Application Papers

9)□ The specification is objected to by the Examiner.

10)□ The drawing(s) filed on ____ is/are: a)□ accepted or b)□ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)□ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)□ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)□ All b)□ Some * c)□ None of:

1.□ Certified copies of the priority documents have been received.

2.□ Certified copies of the priority documents have been received in Application No. ____.

3.□ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) □ Notice of References Cited (PTO-892)

2) □ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) □ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date ____.

4) □ Interview Summary (PTO-413)
Paper No(s)/Mail Date. ____ .

5) □ Notice of Informal Patent Application

6) □ Other: ____.

DETAILED ACTION

Election/Restrictions

Restriction to one of the following inventions is required under 35 U.S.C. 121:

1. Claims 1-27 are still pending for examination.

2. Restriction to one of the following inventions is required under 35 U.S.C. § 121:

I. Claims 1-12 and 26-27 drawn to a system for providing a secure domain name service over a computer network, comprising: a server and a domain name database, classified in Class 709, subclass 226..

II. Claims 13-25 drawn a method for registering a secure domain name, comprising steps of: a) receiving a request, b) verifying ownership information, and c) registering the secure domain name, classified in Class 709, subclass 223.

3. Inventions I and II are related as combination and subcombination. Inventions in this relationship are distinct if it can be shown that (1) the combination as claimed does not require the particulars of the subcombination as claimed for patentability, and (2) that the subcombination has utility by itself or in other combinations (MPEP § 806.05(c)). In the instant case, the combination as claimed of Invention I, claims 1-12 and 26-27, does not require the particular steps of the subcombination:: a) receiving a request, b) verifying ownership information, and c) registering the secure domain name, and the subcombination Group 2, claims 13-25, does not require: a server and a domain name database.

4 Moreover, the searches for these two inventions would not be co-extensive because these groups would require different searches on PTO's classification class and subclass as following:

1) The Group I search (claims 1-12 and 26-27) would require use of search class 709, subclass 226 (which would not required for the group II).

2) The Group II search (claims 13-25) would require use of search class 709, subclass 223 (which would not required for the group I).

5. Applicant is reminded that the required for response to this requirement is **30 days, not one month.**

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Krisna Lim whose telephone number is 571-272-3956 The examiner can normally be reached on Monday to Wednesday and Friday from 7:30 AM to 4:00 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Glenton Burgess, can be reached on 571-272-3949. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Kl

December 4, 2006

KRISNA LIM
PRIMARY EXAMINER

Index of Claims	Application/Control No. 10/714,849	Applicant(s)/Patent under Reexamination LARSON ET AL.	
	Examiner Krisna Lim	Art Unit 2153	

√	Rejected	–	(Through numeral) Cancelled	N	Non-Elected	A	Appeal
=	Allowed	+	Restricted	I	Interference	O	Objected

Claim		Date								
Final	Original	12/4/06								
	1	+								
	2	+								
	3	+								
	4	+								
	5	+								
	6	+								
	7	+								
	8	+								
	9	+								
	10	+								
	11	+								
	12	+								
	13	+								
	14	+								
	15	+								
	16	+								
	17	+								
	18	+								
	19	+								
	20	+								
	21	+								
	22	+								
	23	+								
	24	+								
	25	+								
	26	+								
	27									
	28									
	29									
	30									
	31									
	32									
	33									
	34									
	35									
	36									
	37									
	38									
	39									
	40									
	41									
	42									
	43									
	44									
	45									
	46									
	47									
	48									
	49									
	50									

Claim		Date								
Final	Original									
	51									
	52									
	53									
	54									
	55									
	56									
	57									
	58									
	59									
	60									
	61									
	62									
	63									
	64									
	65									
	66									
	67									
	68									
	69									
	70									
	71									
	72									
	73									
	74									
	75									
	76									
	77									
	78									
	79									
	80									
	81									
	82									
	83									
	84									
	85									
	86									
	87									
	88									
	89									
	90									
	91									
	92									
	93									
	94									
	95									
	96									
	97									
	98									
	99									
	100									

Claim		Date								
Final	Original									
	101									
	102									
	103									
	104									
	105									
	106									
	107									
	108									
	109									
	110									
	111									
	112									
	113									
	114									
	115									
	116									
	117									
	118									
	119									
	120									
	121									
	122									
	123									
	124									
	125									
	126									
	127									
	128									
	129									
	130									
	131									
	132									
	133									
	134									
	135									
	136									
	137									
	138									
	139									
	140									
	141									
	142									
	143									
	144									
	145									
	146									
	147									
	148									
	149									
	150									

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

Bib Data Sheet

CONFIRMATION NO. 3154

SERIAL NUMBER 10/714,849	FILING OR 371(c) DATE 11/18/2003 RULE	CLASS 709	GROUP ART UNIT 2153	ATTORNEY DOCKET NO. 000479.00111

APPLICANTS

Victor Larson, Fairfax, VA;
Robert Durham Short III, Leesburg, VA;
Edmund Colby Munger, Crownsville, MD;
Michael Williamson, South Riding, VA;

** CONTINUING DATA *************************

This application is a CON of 09/558,210 04/26/2000 ABN which is a CIP of 09/504,783 02/15/2000 PAT 6,502,135
which is a CIP of 09/429,643 10/29/1999 PAT 7,010,604
which claims benefit of 60/106,261 10/30/1998
and claims benefit of 60/137,704 06/07/1999

** FOREIGN APPLICATIONS ********************

IF REQUIRED, FOREIGN FILING LICENSE GRANTED **
02/12/2004

Foreign Priority claimed □ yes ☑ no 35 USC 119 (a-d) conditions met □ yes ☑ no □ Met after Allowance Verified and Acknowledged ______ Examiner's Signature ______ Initials	STATE OR COUNTRY VA	SHEETS DRAWING 40	TOTAL CLAIMS 23	INDEPENDENT CLAIMS 5

ADDRESS
22907

TITLE

Agile network protocol for secure communications using secure domain names

FILING FEE RECEIVED 1294	FEES: Authority has been given in Paper No. _________ to charge/credit DEPOSIT ACCOUNT No. _________ for following:	□ All Fees □ 1.16 Fees (Filing) □ 1.17 Fees (Processing Ext. of time) □ 1.18 Fees (Issue) □ Other ____________ □ Credit

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:	:	
	:	
Victor Larson et al.	:	Confirmation No.: 3154
	:	
Application No.: 10/714,849	:	Group Art Unit: 2153
	:	
Filed: November 18, 2003	:	Examiner: Lim, Krisna
	:	
For: An Agile Network Protocol for Secure Communications Using Secure Domain Name	: :	Atty Docket: 007170.00025

RESPONSE TO RESTRICTION REQUIREMENT

U.S. Patent and Trademark Office
Customer Service Window
Randolph Building
401 Dulany Street
Alexandria, VA 22314

Sir:

This paper is responsive to the Office Action mailed December 7, 2006, in connection with the above-identified patent application and is filed prior to the expiration of the one (1) month period for response set therein.

In response, Applicants elect without traverse, the invention of Group I, i.e., claims 1-12 and 26-27. Applicants reserve the right to file divisional applications directed to the subject matter of the non-elected claims prior to the termination of proceedings in this patent application.

No fee is believed to be associated with the filing of this paper. However, should the U.S. Patent and Trademark Office determine a fee is required, authorization is given to charge our Deposit Account No. 19-0733.

Respectfully submitted,

/John M. Fleming/
John M. Fleming
Reg. No. 56,536

January 8, 2007

Banner & Witcoff, Ltd.
Eleventh Floor
1001 G Street, N.W.
Washington, D.C. 20001-4597

Electronic Acknowledgement Receipt

EFS ID:	1422133
Application Number:	10714849
International Application Number:	
Confirmation Number:	3154
Title of Invention:	Agile network protocol for secure communications using secure domain names
First Named Inventor/Applicant Name:	Victor Larson
Customer Number:	22907
Filer:	John McClellan Fleming/Laura Sunderland
Filer Authorized By:	John McClellan Fleming
Attorney Docket Number:	007170.00025
Receipt Date:	08-JAN-2007
Filing Date:	18-NOV-2003
Time Stamp:	13:07:43
Application Type:	Utility

Payment information:

Submitted with Payment	no

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)	Multi Part /.zip	Pages (if appl.)
1	Response to Election / Restriction Filed	717000025_response_to_restriction_requirement.pdf	19236	no	1
Warnings:					

Information:	
Total Files Size (in bytes):	19236

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111
If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371
If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

PATENT APPLICATION FEE DETERMINATION RECORD

Effective October 1, 2003

Application or Docket Number

10/714,849

CLAIMS AS FILED - PART I

	(Column 1)	(Column 2)
TOTAL CLAIMS	23	
FOR	NUMBER FILED	NUMBER EXTRA
TOTAL CHARGEABLE CLAIMS	23 minus 20=	* 3
INDEPENDENT CLAIMS	5 minus 3 =	* 2
MULTIPLE DEPENDENT CLAIM PRESENT		□

* If the difference in column 1 is less than zero, enter "0" in column 2

SMALL ENTITY TYPE □ OR OTHER THAN SMALL ENTITY

RATE	FEE		RATE	FEE
BASIC FEE	385.00	OR	BASIC FEE	770.00
X$ 9=		OR	X$18=	54
X43=		OR	X86=	172
+145=		OR	+290=	
TOTAL		OR	TOTAL	996

CLAIMS AS AMENDED - PART II

AMENDMENT A	A	(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA
	Total	* 27	Minus	** 23	= 4
	Independent	* 7	Minus	*** 5	= 2
	FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM				□

SMALL ENTITY OR OTHER THAN SMALL ENTITY

RATE	ADDITIONAL FEE		RATE	ADDITIONAL FEE
X$ 9=		OR	X$18=	126
X43=		OR	X86=	172
+145=		OR	+290=	
TOTAL ADDIT. FEE		OR	TOTAL ADDIT. FEE	298

13, 15 17 21

1/18/07

AMENDMENT B		(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA
	Total	* 27	Minus	** 27	= —
	Independent	* 7	Minus	*** 7	= —
	FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM				□

RATE	ADDITIONAL FEE		RATE	ADDITIONAL FEE
X$ 9=		OR	X$18=	
X43=		OR	X86=	
+145=		OR	+290=	
TOTAL ADDIT. FEE		OR	TOTAL ADDIT. FEE	

AMENDMENT C		(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA
	Total	*	Minus	**	=
	Independent	*	Minus	***	=
	FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM				□

RATE	ADDITIONAL FEE		RATE	ADDITIONAL FEE
X$ 9=		OR	X$18=	
X43=		OR	X86=	
+145=		OR	+290=	
TOTAL ADDIT. FEE		OR	TOTAL ADDIT. FEE	

* If the entry in column 1 is less than the entry in column 2, write "0" in column 3.
** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 20, enter "20."
*** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 3, enter "3."
The "Highest Number Previously Paid For" (Total or Independent) is the highest number found in the appropriate box in column 1.

FORM PTO-875 (Rev. 10/03)

Patent and Trademark Office, U.S. DEPARTMENT OF COMMERCE

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/714,849	11/18/2003	Victor Larson	007170.00025	3154

22907 7590 03/21/2007
BANNER & WITCOFF, LTD.
1100 13th STREET, N.W.
SUITE 1200
WASHINGTON, DC 20005-4051

EXAMINER	
LIM, KRISNA	
ART UNIT	PAPER NUMBER
2153	

SHORTENED STATUTORY PERIOD OF RESPONSE	MAIL DATE	DELIVERY MODE
3 MONTHS	03/21/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

Office Action Summary	**Application No.** 10/714,849	**Applicant(s)** LARSON ET AL.	
	Examiner Krisna Lim	**Art Unit** 2153	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *08 January 2007*.

2a)□ This action is **FINAL**. 2b)☒ This action is non-final.

3)□ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-27* is/are pending in the application.

4a) Of the above claim(s) *13-25* is/are withdrawn from consideration.

5)□ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-8,26 and 27* is/are rejected.

7)☒ Claim(s) *9-12* is/are objected to.

8)□ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)□ The specification is objected to by the Examiner.

10)□ The drawing(s) filed on _____ is/are: a)□ accepted or b)□ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)□ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)□ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)□ All b)□ Some * c)□ None of:

1.□ Certified copies of the priority documents have been received.

2.□ Certified copies of the priority documents have been received in Application No. _____.

3.□ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) □ Notice of References Cited (PTO-892)

2) □ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☒ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____.

4) □ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .

5) □ Notice of Informal Patent Application

6) □ Other: _____.

1. Claims 1-27 are pending in the application. Claims 1-12 and 26-27 are elected without traverse for examination. And, non-elected claims 13-25 are withdrawn from consideration.

2. Claims 1-12 and 25-27 are rejected under 35 U.S.C. § 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

In claim 1, it is unclear where a query is coming from or who queries to the server and what is being queried. Moreover, it is unclear how the system is providing a secure domain name service.

In claim 26, it is unclear what kind of an apparatus is it. It is unclear how this apparatus is configured. And, it is unclear for what a computer network address is queried.

In claim 27, it is similar to claim 26.

3. The following is a quotation of 35 U.S.C. § 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

4. Claims 1-12 and 26-27 are rejected under 35 U.S.C. § 103(a) as being unpatentable over IP Security Chapter 13 of XP-002167283,[Hereinafter XP]. This reference was cited in 1449 Form that was submitted by the applicant.

5. XP disclosed the invention substantially as claimed. Taking claims 1-3, and 26-27 as exemplary claims, the reference disclosed a system for providing a secure domain name service (IP Security, IPSec, page 399) over a computer network, comprising:

a) a server connected to a computer network, the server authenticating a query for a secure computer network address having a top-level domain (security at IP level), page 399) reserved for secure network connection;

b) a domain name database (routers) connected to the computer network through the server the domain name database storing secure computer network addresses for the computer network.

6. While XP further disclosed IP and IPsec (IP security, page 400), XP did not explicitly call its security at IP level as a non-standard top-level domain such as .scom, .sorg, .snet, .sedu, .smil and .sint, etc. It would have been obvious to one of an ordinary skilled in the art to recognize that the domain names such as .com, .org, .net, .edu, .mil, .int, .gov, etc. are the well-known top-level domains for companies, organizations, government, etc.(see any computer dictionaries) and the additional security of HTTPS to the standard HTTP. Thus, adding additional security to these standard domain names of .com, .org, . net, .edu, .mil, .int, etc. to become .scom, .sorg, .snet, .sedu, .smil, and .sint, etc. would be obviously similar to the teaching of IP and IPsec of XP. Moreover, having these non-secured top-level domains secured would have been a desirable feature in the art because the security of the Internet is a vital element for the national security, business security, individual's privacy security, etc.

7. As to claim 4, XP disclosed the computer network includes the Internet (e.g., see Fig. 13.1, page 401).

8. As to claim 5, XP disclosed the server comprises an edge router (e.g., see routing application, a figure 13.10, page 402).

9. As to claim 6, XP disclosed the server authenticates the query using a cryptographic technique (e.g., see authentication algorithm, a figure 13.2, Encryption Algorithm and Authentication Algorithm on page 403, Cryptographic keys on page 404).

10. As to claim 7, XP disclosed the server is connectable to a virtual private network link through the computer network (e.g., see page 421).

9. As to claim 8, XP disclosed the secure communication link is one of a plurality of secure communication links in a hierarchy of secure communication links (e.g., see IP Security of Fig. 13.1 and IP security architecture on page 402 and 403).

10. Claims 9-12 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

A shortened statutory period for response to this action is set to expire 3 (three) months and 0 (zero) days from the mail date of this letter. Failure to respond within the period for response will result in **ABANDONMENT** of the application (see 35 U.S.C 133, M.P.E.P 710.02, 710.02(b)).

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Krisna Lim whose telephone number is 571-272-3956 The examiner can normally be reached on Monday to Wednesday and Friday from 7:30 AM to 4:00 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Glenton Burgess, can be reached on 571-272-3949. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Kl

March 7, 2007

KRISNA LIM
PRIMARY EXAMINER

INFORMATION DISCLOSURE STATEMENT BY APPLICANT
(Not for submission under 37 CFR 1.99)

Application Number	10714849
Filing Date	2003-11-18
First Named Inventor	Victor Larson
Art Unit	~~2143~~ 2153
Examiner Name	~~TBD~~ K. Lim
Attorney Docket Number	000479.00111

U.S.PATENTS

Examiner Initial*	Cite No	Patent Number	Kind Code1	Issue Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1	5870610	A	1999-02-09	Beyda et al.	

If you wish to add additional U.S. Patent citation information please click the Add button.

U.S.PATENT APPLICATION PUBLICATIONS

Examiner Initial*	Cite No	Publication Number	Kind Code1	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1					

If you wish to add additional U.S. Published Application citation information please click the Add button.

FOREIGN PATENT DOCUMENTS

Examiner Initial*	Cite No	Foreign Document Number3	Country Code2 j	Kind Code4	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear	T5
	1	0 838 930	EP	A	1998-04-29	Compaq Computer Corp.		□
	2	0 814 589	EP	A	1997-12-29	AT&T Corp.		□
	3	2 334 181	GB	A	1999-08-11	NEC Technologies; Globalmart Ltd.		□

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	10714849
	Filing Date	2003-11-18
	First Named Inventor	Victor Larson
	Art Unit	~~2143~~ 2153
	Examiner Name	~~TBD~~ K. Lim
	Attorney Docket Number	000479.00111

KL	4	9827783	WO	A	1998-06-25	Northern Telecom Limited; Antonio, G; Hui, Margare		□
KL	5	2 317 792	GB	A	1998-04-01	Secure Computing Corporation		□
	6							□

If you wish to add additional Foreign Patent Document citation information please click the Add button

NON-PATENT LITERATURE DOCUMENTS

Examiner Initials*	Cite No	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc), date, pages(s), volume-issue number(s), publisher, city and/or country where published.	T5
KL	1	Laurie Wells (LANCASTERBIBELMAIL MSN COM); "Subject: Security Icon" USENET NEWSGROUP, 19 October 1998, XP002200606	□
KL	2	Davila J et al, "Implementation of Virtual Private Networks at the Transport Layer", Information Security, Second International Workshop, ISW '99. Proceedings (Lecture Springer-Verlag Berlin, Germany, [Online] 1999, pages 85-102, XP002399276, ISBN 3-540-66695-B, retrieved from the Internet: URL: http://www.springerlink.com/ content/4uac0tb0heccma89/fulltext.pdf> (Abstract)	□
	3		□
	4		□

If you wish to add additional non-patent literature document citation information please click the Add button

EXAMINER SIGNATURE

Examiner Signature		Date Considered	3/7/07

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through a citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT
(Not for submission under 37 CFR 1.99)

Application Number	10714849
Filing Date	2003-11-18
First Named Inventor	Victor Larson
Art Unit	~~2143~~ 2153
Examiner Name	~~TBD~~ K. Lim
Attorney Docket Number	000479.00111

[1] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [2] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [3] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document. [4] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [5] Applicant is to place a check mark here if English language translation is attached.

PTO/SB/08a (08-03)
Approved for use through 07/31/2006. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Substitute for form 1449A/PTO

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Use as many sheets as necessary)

Sheet 1 of 2

Complete if Known	
Application Number	10/714,849
Filing Date	November 18, 2003
First Named Inventor	Victor LARSON
Art Unit	~~2143~~ 2153
Examiner Name	~~TBA~~ K. Lim
Attorney Docket Number	000479.00111

U.S. PATENT DOCUMENTS

Examiner Initials *	Cite No.[1]	Document Number Number - Kind Code[2] *(if known)*	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			

FOREIGN PATENT DOCUMENTS

Examiner Initials*	Cite No.[1]	Foreign Patent Document Country Code[3] - Number[4] - Kind Code[5] *(if known)*	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear	T[6]
/KL/		EP 0 838 930 A2	04/29/1998	Kenneth F. ALDEN, et al.		
/KL/		DE 199 24 575 A1	12/02/1999	Joseph E. PROVINO		

Examiner Signature		Date Considered	3/6/07

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant. [1] Applicant's unique citation designation number (optional). [2] See Kinds Codes of USPTO Patent Documents at www.uspto.gov or MPEP 901.04. [3] Enter Office that issued the document, by the two-letter code (WIPO Standard ST.3). [4] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document. [5] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST. 16 if possible. [6] Applicant is to place a check mark here if English language Translation is attached.

This collection of information is required by 37 CFR 1.97 and 1.98. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 2 hours to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

PTO/SB/08b(08-03)
Approved for use through 07/31/2006. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number

Substitute for form 1449B/PTO

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Use as many sheets as necessary)

Sheet	2	of	2

Complete if Known	
Application Number	10/714,849
Filing Date	November 18, 2003
First Named Inventor	Victor LARSON
Art Unit	2143
Examiner Name	TBD
Attorney Docket Number	000479.00111

NON PATENT LITERATURE DOCUMENTS

Examiner Initials *	Cite No.[1]	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc.), date, page(s), volume-issue number(s), publisher, city and/or country where published.	T[2]
lk		Donald E. EASTLAKE, III, "Domain Name System Security Extensions", Internet Draft, April 1998.	
lk		P. SRISURESH, et al., "DNS Extensions to Network Address Translators', Internet Draft, July 1998.	
lk		D.B. CHAPMAN, et al., "Building Internet Firewalls, chapters 8 and 10 (parts)", pp. 278-296 and pp. 351-375.	

Examiner Signature		Date Considered	3/6/07

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

[1] Applicant's unique citation designation number (optional). [2] Applicant is to place a check mark here if English language Translation is attached.

This collection of information is required by 37 CFR 1.98. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 120 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: **Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

Sheet 1 of 5

PTO-1449 (Modified) U.S. DEPARTMENT OF COMMERCE PATENT AND TRADEMARK OFFICE INFORMATION DISCLOSURE STATEMENT BY APPLICANT	ATTY. DOCKET NO. 000479.00111	SERIAL NUMBER ~~TBD~~ 10/714,849
	APPLICANT Victor Larson et al.	
	FILING DATE Herewith	GROUP ART UNIT ~~TBD~~ 2153

U.S. PATENT DOCUMENTS

EXAMINER INITIAL	DOCUMENT NUMBER	DATE	NAME	CLASS	SUB CLASS	FILING DATE
KL	6,119,171	9/2000	Alkhatib			
KL	5,588,060	12/24/96	Aziz			
KL	5,689,566	11/18/9	Nguyen			
KL	5,842,040	11/24/98	Hughes et al.			
KL	4,933,846	06/12/90	Humphrey et al.			

FOREIGN PATENT DOCUMENTS

EXAMINER INITIAL	DOCUMENT NUMBER	DATE	COUNTRY	CLASS	SUB CLASS	TRANSLATION YES/NO	
KL	199 24 575	12/2/99	DE				
KL	0 838 930	4/29/98	EPO				
KL	2 317 792	4/1/98	GB				
KL	0 814 589	12/29/97	EPO				
KL	WO 98/27783	6/25/98	PCT				

OTHER DOCUMENTS (Including Author, Title, Date, Pertinent Pages, Etc.)

KL	Search Report (dated 6/18/02), International Application No. PCT/US01/13260
	Search Report (dated 6/28/02), International Application No. PCT/US01/13261
	Donald E. Eastlake, "Domain Name System Security Extensions", DNS Security Working Group, April 1998, 51 pages
	D. B. Chapman et al., "Building Internet Firewalls", November 1995, pages 278-297 and pages 351-375
	P. Srisuresh et al., "DNS extensions to Network Address Translators", July 1998, 27 pages
	Laurie Wells, "Security Icon", October 19, 1998, 1 page
	W. Stallings, "Cryptography And Network Security", 2nd Edition, Chapter 13, IP Security, June 8, 1998, pages 399-440
	W. Stallings, "New Cryptography and Network Security Book", June 8, 1998, 3 pages
	FASBENDER, KESDOGAN, and KUBITZ: "Variable and Scalable Security: Protection of Location Information in Mobile IP", IEEE publication, 1996, pages 963-967

EXAMINER Krisna Lim	DATE CONSIDERED 3/6/07

EXAMINER: Initial citation if reference was considered. Draw line through citation if not in conformance to MPEP 609 and not considered. Include copy of this form with next communication to applicant.

Sheet 2 of 5

PTO-1449 (Modified) U.S. DEPARTMENT OF COMMERCE PATENT AND TRADEMARK OFFICE INFORMATION DISCLOSURE STATEMENT BY APPLICANT	ATTY. DOCKET NO. 000479.00111	SERIAL NUMBER ~~TBD~~ 10/714,849
	APPLICANT Victor Larson et al.	
	FILING DATE Herewith	GROUP ART UNIT TBD

U.S. PATENT DOCUMENTS

EXAMINER INITIAL	DOCUMENT NUMBER	DATE	NAME	CLASS	SUB CLASS	FILING DATE
KL	6,353,614	3/5/02	Borella et al.			
	6,332,158	12/18/01	Risley et al.			
	6,330,562	12/11/01	Boden et al.			
	6,286,047	9/4/01	Ramanathan et al.			
	6,243,749	6/5/01	Sitaraman et al.			
	6,226,751	5/1/01	Arrow et al.			
	6,178,505	1/23/01	Schneider et al.			
	6,119,171	9/12/00	Alkhatib			
	6,079,020	6/20/00	Liu			
	6,052,788	4/18/00	Wesinger, Jr. et al.			
	6,016,318	1/18/00	Tomoike			
	6,006,259	12/21/99	Adelman et al.			

FOREIGN PATENT DOCUMENTS

EXAMINER INITIAL	DOCUMENT NUMBER	DATE	COUNTRY	CLASS	SUB CLASS	TRANSLATION YES	NO

OTHER DOCUMENTS (Including Author, Title, Date, Pertinent Pages, Etc.)

EXAMINER Krisna Lim	DATE CONSIDERED 3/6/07

EXAMINER: Initial citation if reference was considered. Draw line through citation if not in conformance to MPEP 609 and not considered. Include copy of this form with next communication to applicant.

Sheet 3 of 5

PTO-1449 (Modified) U.S. DEPARTMENT OF COMMERCE PATENT AND TRADEMARK OFFICE INFORMATION DISCLOSURE STATEMENT BY APPLICANT	ATTY. DOCKET NO. 000479.00111	SERIAL NUMBER TBD
	APPLICANT Victor Larson et al.	
	FILING DATE Herewith	GROUP ART UNIT TBD

U.S. PATENT DOCUMENTS

EXAMINER INITIAL	DOCUMENT NUMBER	DATE	NAME	CLASS	SUB CLASS	FILING DATE
KL	5,905,859	5/18/99	Holloway et al.			
	5,898,830	4/27/99	Wesinger, Jr. et al.			
	5,892,903	4/6/99	Klaus			
	5,878,231	3/2/99	Baehr et al.			
	5,805,801	9/8/98	Holloway et al.			
	5,796,942	8/18/98	Esbensen			

FOREIGN PATENT DOCUMENTS

EXAMINER INITIAL	DOCUMENT NUMBER	DATE	COUNTRY	CLASS	SUB CLASS	TRANSLATION YES/NO	
KL	WO 00/70458	11/23/00	PCT				

OTHER DOCUMENTS (Including Author, Title, Date, Pertinent Pages, Etc.)

KL	Linux FreeS/WAN Index File, printed from http://liberty.freeswan.org/freeswan trees/freeswan-1.3/doc/ on February 21, 2002, 3 Pages
	J. Gilmore, "Swan: Securing the Internet against Wiretapping", printed from http://liberty.freeswan.org/freeswan trees/freeswan-1.3/doc/rationale.html on February 21, 2002, 4 pages
	Glossary for the Linux FreeS/WAN project, printed from http://liberty.freeswan.org/freeswan trees/freeswan-1.3/doc/glossary.html on February 21, 2002, 25 pages
	Alan O. Frier et al., "The SSL Protocol Version 3.0", November 18, 1996, printed from http://www.netscape.com/eng/ssl3/draft302.txt on February 4, 2002, 56 pages

EXAMINER KRISNA Lim	DATE CONSIDERED 3/6/07

EXAMINER: Initial citation if reference was considered. Draw line through citation if not in conformance to MPEP 609 and not considered. Include copy of this form with next communication to applicant.

Sheet 4 of 5

PTO-1449 (Modified) U.S. DEPARTMENT OF COMMERCE PATENT AND TRADEMARK OFFICE INFORMATION DISCLOSURE STATEMENT BY APPLICANT	ATTY. DOCKET NO. 000479.00111	SERIAL NUMBER ~~TBD~~ 10/714,849
	APPLICANT Victor Larson et al.	
	FILING DATE Herewith	GROUP ART UNIT ~~TBD~~ 2153

U.S. PATENT DOCUMENTS

EXAMINER INITIAL	DOCUMENT NUMBER	DATE	NAME	CLASS	SUB CLASS	FILING DATE

FOREIGN PATENT DOCUMENTS

EXAMINER INITIAL	DOCUMENT NUMBER	DATE	COUNTRY	CLASS	SUB CLASS	TRANSLATION YES	NO
KL	0 858 189	8/12/98	EPO				
	WO 01 50688	7/12/01	PCT				
	WO 98 59470	12/30/98	PCT				
	WO 99 48303	9/23/99	PCT				
	WO 99 38081	7/29/99	PCT				

OTHER DOCUMENTS (Including Author, Title, Date, Pertinent Pages, Etc.)

KL	Search Report (dated 8/20/02), International Application No. PCT/US01/04340
	Search Report (dated 8/23/02), International Application No. PCT/US01/13260
	Shree Murthy et al., "Congestion-Oriented Shortest Multipath Routing", Proceedings of IEEE INFOCOM, 1996, pages 1028-1036
	Jim Jones et al., "Distributed Denial of Service Attacks: Defenses", Global Integrity Corporation, 2000, pages 1-14
	James E. Bellaire, "New Statement of Rules – Naming Internet Domains", Internet Newsgroup, July 30, 1995, 1 page
	D. Clark, "US Calls for Private Domain-Name System", Computer, IEEE Computer Society, August 1, 1998, pages 22-25
	August Bequai, "Balancing Legal Concerns Over Crime and Security in Cyberspace", Computer & Security, Vol. 17, No. 4, 1998, pages 293-298
	Rich Winkel, "CAQ: Networking With Spooks: The NET & The Control Of Information", Internet Newsgroup, June 21, 1997, 4 pages

EXAMINER Krisna Lim	DATE CONSIDERED 3/6/07

EXAMINER: Initial citation if reference was considered. Draw line through citation if not in conformance to MPEP 609 and not considered. Include copy of this form with next communication to applicant.

Sheet 5 of 5

PTO-1449 (Modified) U.S. DEPARTMENT OF COMMERCE PATENT AND TRADEMARK OFFICE INFORMATION DISCLOSURE STATEMENT BY APPLICANT	ATTY. DOCKET NO. 000479.00111	SERIAL NUMBER TBD
	APPLICANT Victor Larson et al.	
	FILING DATE Herewith	GROUP ART UNIT TBD

U.S. PATENT DOCUMENTS

EXAMINER INITIAL	DOCUMENT NUMBER	DATE	NAME	CLASS	SUB CLASS	FILING DATE
KL	6,016,512	1/20000	Christian Huitema			

FOREIGN PATENT DOCUMENTS

EXAMINER INITIAL	DOCUMENT NUMBER	DATE	COUNTRY	CLASS	SUB CLASS	TRANSLATION YES/NO	
KL	WO 98 55930	12/10/98	PCT				

OTHER DOCUMENTS (Including Author, Title, Date, Pertinent Pages, Etc.)

KL	Search Report (dated 10/7/02), International Application No. PCT/US01/13261
	F. Halsall, "Data Communications, Computer Networks And Open Systems", Chapter 4, Protocol Basics, 1996, pages 198-203
	Reiter, Michael K. and Rubin, Aviel D. (AT&T Labs – Research), "Crowds: Anonymity for Web Transmissions", pages 1-23
	Dolev, Shlomi and Ostrovsky, Rafil, "Efficient Anonymous Multicast and Reception" (Extended Abstract), 16 pages
	Rubin, Aviel D., Greer, Daniel, and Ranum, Marcus J. (Wiley Computer Publishing), "Web Security Sourcebook", pages 82-94
	FASBENDER, KESDOGAN, and KUBITZ: "Variable and Scalable Security" Protection of Location Information in Mobile IP", IEEE publication, 1996, pages 963-967

EXAMINER Krisna Lim	DATE CONSIDERED 3/6/07
EXAMINER: Initial citation if reference was considered. Draw line through citation if not in conformance to MPEP 609 and not considered. Include copy of this form with next communication to applicant.	

Index of Claims

Application/Control No.	Applicant(s)/Patent under Reexamination
10/714,849	LARSON ET AL.
Examiner	**Art Unit**
Krisna Lim	2153

R	Rejected	–	(Through numeral) Cancelled	N	Non-Elected	A	Appeal
=	Allowed	÷	Restricted	I	Interference	O	Objected

Claim		Date								
Final	Original	3/7/07								
	1									
	2	R								
	3	R								
	4	R								
	5	R								
	6	R								
	7	R								
	8	R								
	9	O								
	10	O								
	11	O								
	12	O								
	13	N								
	14	N								
	15	N								
	16	N								
	17	N								
	18	N								
	19	N								
	20	N								
	21	N								
	22	N								
	23	N								
	24	N								
	25	N								
	26	R								
	27	R								
	28									
	29									
	30									
	31									
	32									
	33									
	34									
	35									
	36									
	37									
	38									
	39									
	40									
	41									
	42									
	43									
	44									
	45									
	46									
	47									
	48									
	49									
	50									

Claim		Date								
Final	Original									
	51									
	52									
	53									
	54									
	55									
	56									
	57									
	58									
	59									
	60									
	61									
	62									
	63									
	64									
	65									
	66									
	67									
	68									
	69									
	70									
	71									
	72									
	73									
	74									
	75									
	76									
	77									
	78									
	79									
	80									
	81									
	82									
	83									
	84									
	85									
	86									
	87									
	88									
	89									
	90									
	91									
	92									
	93									
	94									
	95									
	96									
	97									
	98									
	99									
	100									

Claim		Date								
Final	Original									
	101									
	102									
	103									
	104									
	105									
	106									
	107									
	108									
	109									
	110									
	111									
	112									
	113									
	114									
	115									
	116									
	117									
	118									
	119									
	120									
	121									
	122									
	123									
	124									
	125									
	126									
	127									
	128									
	129									
	130									
	131									
	132									
	133									
	134									
	135									
	136									
	137									
	138									
	139									
	140									
	141									
	142									
	143									
	144									
	145									
	146									
	147									
	148									
	149									
	150									

U.S. Patent and Trademark Office

Part of Paper No. 20070307

Search Notes	Application/Control No.	Applicant(s)/Patent under Reexamination
	10/714,849	LARSON ET AL.
	Examiner	Art Unit
	Krisna Lim	2153

SEARCHED

Class	Subclass	Date	Examiner
709	226, 221	3/7/2007	KL
713	201	3/7/2007	KL

INTERFERENCE SEARCHED

Class	Subclass	Date	Examiner

SEARCH NOTES (INCLUDING SEARCH STRATEGY)

	DATE	EXMR
EAST	9/2/2006	KL
Inventors	3/7/2007	

McDermott
Will&Emery

Boston Brussels Chicago Düsseldorf London Los Angeles Miami Munich
New York Orange County Rome San Diego Silicon Valley Washington, D.C.
Strategic alliance with MWE China Law Offices (Shanghai)

FACSIMILE

Date:	June 7, 2007	**Time Sent:**	

To:	**Company:**	**Facsimile No:**	**Telephone No:**
Commissioner for Patents – Group Art Unit: 2153	U.S. Patent and Trademark Office	1.571.273.8300	

From:	Toby H. Kusmer, P.C.	*Direct Phone:*	617.535.4065
E-Mail:	tkusmer@mwe.com	*Direct Fax:*	617.535.3800
Sent By:	Cynthia Joseph	*Direct Phone:*	617.535.4111
Client/Matter/Tkpr:	77580-027/5496	*Original to Follow by Mail:*	No
		Number of Pages, Including Cover:	

Re: In re Application of: Victor Larson, et al.
Serial No.: 10/714,849
Filing Date: November 18, 2003
Title: An Agile Network Protocol For Secure Communications Using Secure Domain Names
Docket No.: 77580-027 (VRNK-1CP3CN)

Message:

Please enter the attached Change of Correspondence Address And Revocation Of Power Of Attorney And Appointment of New Attorney.

The information contained in this facsimile message is legally privileged and confidential information intended only for the use of the individual or entity named above. If the reader of this message is not the intended recipient, you are hereby notified that any dissemination, distribution, or copy of this facsimile is strictly prohibited. If you have received this facsimile in error, please notify us immediately by telephone and return the original message to us at the below address by mail. Thank you.

IF YOU DO NOT RECEIVE ALL OF THE PAGES, PLEASE CALL AS SOON AS POSSIBLE.

Main Facsimile: 617.535.3800 Facsimile Operator: 617.535.4000

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re patent of:	Victor Larson, et al.
Serial No.:	10/714,849
Filing Date:	November 18, 2003
Title:	An Agile Network Protocol For Secure Communications Using Secure Domain Names
Group Art Unit:	2153
Confirmation No.:	3154
Docket No.:	77580-027 (VRNK-1CP3CN)

MAIL STOP Amendment
Commissioner for Patents
P. O. Box 1450
Alexandria, VA 22313-1450

TRANSMITTAL LETTER

Applicants transmit herewith the following document in the above-identified application:

Change Of Correspondence Address And Revocation Of Power Of Attorney And Appointment Of New Power.

The Commissioner is authorized to charge any filing fee or credit any overpayment to Deposit Account No. 50-1133.

Respectfully submitted,

Toby H. Kusmer, P.C.
Reg. No. 26,418
McDermott Will & Emery LLP
28 State Street
Boston, MA 02109-1775
Telephone: 617.535.4065
Facsimile: 617.535.3800
e-mail: tkusmer@mwe.com

CERTIFICATE OF FACSIMILE TRANSMISSION

I hereby certify that this correspondence is being facsimile transmitted. via Facsimile No. 571.273.8300, to the U.S. Patent and Trademark Office on the date indicated below and is addressed to Commissioner for Patents, P. O. Box 1450, Alexandria, VA 22313-1450.

June 11 2007
Date

Cynthia Joseph

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re patent of:	Victor Larson, et al.
Serial No.:	10/714,849
Filing Date:	November 18, 2003
Title:	An Agile Network Protocol For Secure Communications Using Secure Domain Names
Docket No.:	77580-027 (VRNK-1CP3CN)

Commissioner for Patents
P. O. Box 1450
Alexandria, VA 22313-1450

CHANGE OF CORRESPONDENCE ADDRESS AND REVOCATION OF POWER OF ATTORNEY AND APPOINTMENT OF NEW ATTORNEY

Dear Sir:

Applicants in the above-identified patent application hereby revoke all powers of attorney previously given in connection with the above-identified patent application and hereby appoint the following attorneys, with full power of substitution, to transact all business in the Patent and Trademark Office connected therewith:

All attorneys associated with CUSTOMER NUMBER 23630

It is requested that all correspondence regarding this patent application be directed to:

Toby H. Kusmer, P.C.
McDermott Will & Emery LLP
28 State Street
Boston, Massachusetts 02109-1775
Telephone: (617) 535-4065
Facsimile: (617) 535-3800
E-mail: tkusmer@mwe.com

Respectfully submitted,

Date: 6/11/07

By ____________
Name: Randall Larson
Title: President/CEO

UNITED STATES PATENT AND TRADEMARK OFFICE

MN

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/714,849	11/18/2003	Victor Larson	007170.00025	3154

22907 7590 06/21/2007
BANNER & WITCOFF, LTD.
1100 13th STREET, N.W.
SUITE 1200
WASHINGTON, DC 20005-4051

EXAMINER	
LIM, KRISNA	

ART UNIT	PAPER NUMBER
2153	

MAIL DATE	DELIVERY MODE
06/21/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

PTOL-90A (Rev. 04/07)

UNITED STATES DEPARTMENT OF COMMERCE
U.S. Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450

APPLICATION NO./ CONTROL NO.	FILING DATE	FIRST NAMED INVENTOR / PATENT IN REEXAMINATION	ATTORNEY DOCKET NO.
10714849	11/18/03	LARSON ET AL.	007170.00025

BANNER & WITCOFF, LTD.
1100 13th STREET, N.W.
SUITE 1200
WASHINGTON, DC 20005-4051

EXAMINER	
Krisna Lim	
ART UNIT	PAPER
2153	20070618

DATE MAILED:

Please find below and/or attached an Office communication concerning this application or proceeding.

Commissioner for Patents

Per the telephone request by the attorney of the record, Examiner provided the reforence.

KRISNA LIM
PRIMARY EXAMINER

PTO-90C (Rev.04-03)

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NUMBER	FILING OR 371 (c) DATE	FIRST NAMED APPLICANT	ATTY. DOCKET NO./TITLE
10/714,849	11/18/2003	Victor Larson	007170.00025

CONFIRMATION NO. 3154

OC000000024495906

22907
BANNER & WITCOFF, LTD.
1100 13th STREET, N.W.
SUITE 1200
WASHINGTON, DC 20005-4051

Date Mailed: 06/22/2007

NOTICE REGARDING POWER OF ATTORNEY

This is in response to the Power of Attorney filed 06/12/2007 . The Power of Attorney in this application is not accepted for the reason(s) listed below:

- The Power of Attorney is from an assignee and the Certificate required by 37 CFR 3.73(b) has not been received.

Office of Initial Patent Examination (571) 272-4000, or 1-800-PTO-9199
OFFICE COPY

McDermott
Will&Emery

Boston Brussels Chicago Düsseldorf London Los Angeles Miami Munich
New York Orange County Rome San Diego Silicon Valley Washington, D.C.
Strategic alliance with MWE China Law Offices (Shanghai)

FACSIMILE

Date: July 11, 2007 **Time Sent:**

To:	**Company:**	**Facsimile No:**	**Telephone No:**
Commissioner for Patents	U.S. Patent and Trademark Office	1.571.273.8300	

From:	Toby H. Kusmer, P.C.	*Direct Phone:*	617.535.4065	
E-Mail:	tkusmer@mwe.com	*Direct Fax:*	617.535.3800	
Sent By:	Cynthia Joseph	*Direct Phone:*	617.535.4111	
Client/Matter/Tkpr:	77580-042/5496	*Original to Follow by Mail:*		No
		Number of Pages, Including Cover:		21

Re: In re Application of: Victor Larsen, et al.
Serial No.: 10/714,849
Filing Date: November 18, 2003
Title: An Agile Network Protocol For Secure Communications
Docket No.: 77580-042 (VRNK-1CP3CN)

Message:

Please enter the attached Amendment "B" in response to the Office Action of March 21, 2007

The information contained in this facsimile message is legally privileged and confidential information intended only for the use of the individual or entity named above. If the reader of this message is not the intended recipient, you are hereby notified that any dissemination, distribution, or copy of this facsimile is strictly prohibited. If you have received this facsimile in error, please notify us immediately by telephone and return the original message to us at the below address by mail. Thank you.

IF YOU DO NOT RECEIVE ALL OF THE PAGES, PLEASE CALL AS SOON AS POSSIBLE.

Main Facsimile: 617.535.3800 Facsimile Operator: 617.535.4000

U.S. practice conducted through McDermott Will & Emery LLP.
28 State Street Boston, Massachusetts 02109-1775 Telephone: 617.535.4000

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:	Victor Larsen et al.
Serial No:	10/714,849
Filing Date:	November 18, 2003
Group Art Unit:	2153
Examiner:	Krisna Lim
Confirmation No.:	3154
Title:	An Agile Network Protocol for Secure Communications
Docket No:	77580-042 (VRNK-1CP3CN)

CERTIFICATE OF MAILING OR TRANSMISSION

I hereby certify that this correspondence is being deposited with the U.S. Postal Service as first class mail in an envelope addressed to: Commissioner for Patents, P. O. Box 1450, Alexandria, VA 22313-1450, or facsimile transmitted (571-273-8300) to the USPTO, on the date indicated below.

Date: 11 July 2007

Cynthia Joseph

MS Amendment
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

AMENDMENT "B"

This paper is a response to the Official action dated 21 March 2007.

The Applicants request reconsideration and further examination in view of the following:

Amendments to the Claims, as reflected in the listing of claims beginning on page 2 of this paper; and

Remarks, beginning on page 14 of this paper.

Amendments to the Claims:

This listing of claims will replace all prior versions and listings of claims in the application:

Listing of Claims:

1. (Canceled).

2. (Currently Amended) The system of claim ~~1~~29, wherein the top-level domain name is a non-standard top-level domain name.

3. (Original) The system of claim 2, wherein the non-standard top-level domain name is one of .scom, .sorg, .snet, .sgov, .sedu, .smil and .sint.

4. (Currently Amended) The system of claim ~~1~~28, wherein the ~~computer~~ communication network includes the Internet.

5. (Currently Amended) The system of claim ~~1~~28, wherein the domain name service system~~server~~ comprises an edge router.

6. (Currently Amended) The system of claim ~~1~~29, wherein the domain name service system is configured to authenticate ~~server authenticates~~ the query using a cryptographic technique.

7. (Currently Amended) The system of claim ~~1~~28, wherein the domain name service system~~server~~ is connectable to a virtual private network ~~link~~ through the ~~computer~~ communication network.

8. (Currently Amended) The system of claim 7, wherein the virtual private network~~secure communication link~~ is one of a plurality of secure communication links in a hierarchy of secure communication links.

9. (Currently Amended) The system of claim 7, wherein the virtual private network is based on inserting into each data packet communicated over a secure communication link one or more data values that vary according to a pseudo-random sequence.

10. (Currently Amended) The system of claim 7, wherein the virtual private network is based on a ~~computer~~ network address hopping regime that is used to pseudorandomly change ~~computer~~ network addresses in packets transmitted between a first ~~computer~~ device and a second ~~computer~~device.

11. (Currently Amended) The system of claim 7, wherein the virtual private network is based on comparing a value in each data packet transmitted between a first ~~computer~~ device and a second ~~computer~~ device to a moving window of valid values.

12. (Currently Amended) The system of claim 7, wherein the virtual private network is based on a comparison of a discriminator field in a header of each data packet to a table of valid discriminator fields maintained for a first ~~computer~~device.

13. (Withdrawn) A method for registering a secure domain name, comprising steps of:

receiving a request for registering a secure domain name;

verifying ownership information for an equivalent non-secure domain name corresponding to the secure domain name;

registering the secure domain name in a secure domain name service when the ownership information for the equivalent non-secure domain name is consistent with ownership information for the secure domain name.

14. (Withdrawn) The method according to claim 13, wherein the step of verifying ownership information includes steps of:

determining whether the equivalent non-secure domain name corresponding to the secure domain name has been registered in a non-secure domain name service; and querying whether the equivalent non-secure domain name should be registered in the nonsecure domain name service when the equivalent non-secure domain name has not been registered in the non-secure domain name service.

15. (Withdrawn) A computer-readable storage medium, comprising:

a storage area; and computer-readable instructions for a method for registering a secure domain name, the method comprising steps of:

receiving a request for registering a secure domain name;

verifying ownership information for an equivalent non-secure domain name corresponding to the secure domain name; and

registering the secure domain name in a secure domain name service when the ownership information for the equivalent non-secure domain name is consistent with ownership information for the secure domain name.

16. (Withdrawn) The computer-readable medium according to claim 15, wherein the step of verifying ownership information includes steps of:

determining whether the equivalent non-secure domain name corresponding to the secure domain name has been registered in a non-secure domain name service; and

querying whether the equivalent non-secure domain name should be registered in the non-secure domain name service when the equivalent non-secure domain name has not been registered in the non-secure domain name service.

17. (Withdrawn) A method for registering a domain name, comprising steps of:

(i) receiving a request for registering a first domain name;

(ii) verifying ownership information for a second domain name corresponding to the first domain name; and

(iii) registering the first domain name when the ownership information for the second domain name is consistent with ownership information for the first domain name.

18. (Withdrawn) The method of claim 17, wherein the first domain name comprises a nonstandard top-level domain and the second domain name comprises a standard top-level domain.

19. (Withdrawn) The method of claim 17, further comprising the step of storing information corresponding to the registration performed in step (iii) in a database separate from a database storing information for standard domain name registrations.

20. (Withdrawn) The method according to claim 17, wherein the step of verifying ownership information includes steps of:

(a) determining whether the second domain name has been registered in a domain name service; and

(b) querying whether the second domain name should be registered in the domain name service when the second domain name has not been registered in the domain name service.

21. (Withdrawn) A computer-readable medium, comprising computer-readable instructions for a method for registering a domain name, the method comprising steps of:

(i) receiving a request for registering a first domain name;

(ii) verifying ownership information for a second domain name corresponding to the first domain name; and

(iii) registering the first domain name when the ownership information for the second domain name is consistent with ownership information for the first domain name.

22. (Withdrawn) The computer readable medium of claim 21, wherein the first domain name comprises a non-standard top-level domain and the second domain name comprises a standard top level domain.

23. (Withdrawn) The computer-readable medium of claim 21, wherein the step of verifying ownership information includes steps of:

(a) determining whether the second domain name has been registered in a domain name service; and

(b) querying whether the second domain name should be registered in the domain name service when the second domain name has not been registered in the domain name service.

24. (Withdrawn) The method of claim 13, wherein the secure domain name has a top-level domain reserved for secure network connections.

25. (Withdrawn) The computer-readable storage medium of claim 15, wherein the secure domain name has a top-level domain reserved for secure network connections.

26. (Canceled).

27. (Canceled).

28. (New) A system for providing a domain name service for establishing a secure communication link, the system comprising:

a domain name service system configured to be connected to a communication network, to store a plurality of domain names and corresponding network addresses, to receive a query for a network address, and to comprise an indication that the domain name service system supports establishing a secure communication link.

29. (New) The system of claim 28, wherein at least one of the plurality of domain names comprises a top-level domain name.

30. (New) The system of claim 28, wherein the domain name service system is configured to respond to the query for the network address.

31. (New) The system of claim 28, wherein the domain name service system is configured to provide, in response to the query, the network address corresponding to a

domain name from the plurality of domain names and the corresponding network addresses.

32. (New) The system of claim 28, wherein the domain name service system is configured to rcceive the query initiated from a first location, the query requesting the network address associated with a domain name, wherein the domain name service system is configured to provide the network address associated with a second location, and wherein the domain name service system is configured to support establishing a secure communication link between the first location and the second location.

33. (New) The system of claim 28, wherein the domain name service system is connected to a communication network, stores a plurality of domain names and corresponding network addresses, and comprises an indication that the domain name service system supports establishing a secure communication link.

34. (New) The system of claim 28, wherein at least one of the plurality of domain names is reserved for secure communication links.

35. (New) The system of claim 28, wherein the domain name service system comprises a server.

36. (New) The system of claim 35, wherein the domain name service system further comprises a domain name database, and wherein the domain name database stores the plurality of domain names and the corresponding network addresses.

37. (New) The system of claim 28, wherein the domain name service system comprises a server, wherein the server comprises a domain name database, and wherein the domain name database stores the plurality of domain names and the corresponding network addresses.

38. (New) The system of claim 28, wherein the domain name service system is configured to store the corresponding network addresses for use in establishing secure communication links.

39. (New) The system of claim 28, wherein the domain name service system is configured to authenticate the query for the network address.

40. (New) The system of claim 28, wherein at least one of the plurality of domain names comprises an indication that the domain name service system supports establishing a secure communication link.

41. (New) The system of claim 28, wherein at least one of the plurality of domain names comprises a secure name.

42. (New) The system of claim 28, wherein at least one of the plurality of domain names enables establishment of a secure communication link.

43. (New) The system of claim 28, wherein the domain name service system is configured to enable establishment of a secure communication link between a first location and a second location transparently to a user at the first location.

44. (New) The system of claim 28, wherein the secure communication link uses encryption.

45. (New) The system of claim 28, wherein the secure communication link is capable of supporting a plurality of services.

46. (New) The system of claim 45, wherein the plurality of services comprises a plurality of communication protocols, a plurality of application programs, multiple sessions, or a combination thereof.

47. (New) The system of claim 46, wherein the plurality of application programs comprises items selected from a group consisting of the following: video conferencing, e-mail, a word processing program, and telephony.

48. (New) The system of claim 45, wherein the plurality of services comprises audio, video, or a combination thereof.

49. (New) The system of claim 28, wherein the domain name service system is configured to enable establishment of a secure communication link between a first location and a second location.

50. (New) The system of claim 49, wherein the query is initiated from the first location, wherein the second location comprises a computer, and wherein the network address is an address associated with the computer.

51. (New) The system of claim 28, wherein the domain name service system comprises a domain name database connected to a communication network and storing a plurality of domain names and corresponding network addresses for communication,

wherein the domain name database is configured so as to provide a network address corresponding to a domain name in response to a query in order to establish a secure communication link.

52. (New) A machine-readable medium comprising instructions executable in a domain name service system, the instructions comprising code for:

connecting the domain name service system to a communication network;

storing a plurality of domain names and corresponding network addresses;

receiving a query for a network address; and

supporting an indication that the domain name service system supports establishing a secure communication link.

53. (New) A method of providing a domain name service for establishing a secure communication link, the method comprising:

connecting a domain name service system to a communication network, the domain name service system comprising an indication that the domain name service system supports establishing a secure communication link;

storing a plurality of domain names and corresponding network addresses; and

receiving a query for a network address for communication.

REMARKS

Claims 2-25 and 28-53 remain in the application. Claims 1, 26 and 27 have been canceled. Claims 2 and 4-12 have been amended and claims 28-53 have been added in order to more clearly define applicant's invention. Claims 13-25, drawn to a non-elected invention, are withdrawn from consideration. Applicants note with appreciation that claims 9-12 are considered allowable if rewritten in independent form~~,~~ . These claims however, have not been rewritten in independent form in the belief that all of the claims are now considered allowable.

Claims 1-12 and 25-27 have been rejected under 35 U.S.C. §112, second paragraph. Claims 1-12 and 26-27 have also been rejected under 35 U.S.C. §103 (a). These rejections are respectfully traversed and reconsideration is requested in view of the foregoing amendments and following remarks.

Claims 1-12 and 25 (sic) -27 have been rejected under 35 U.S.C. §112, second paragraph, because the Examiner believes that these claims are indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. It is noted that claim 25 is directed to a non-elected invention, and is considered withdrawn. Applicants believe the that it was the intent of the Examiner to reject claims 1-12, 26 and 27, and not claim 25, under 35 U.S.C. § 112, second paragraph. Accordingly, Applicant's comments are directed to the more limited grouping.

The Examiner states that with regard to claim 1, "it is unclear where the query is coming from or who queries to the server and what is being queried." The Examiner also states that "it is unclear how the system is providing a secure domain name service." Claim 1 has been canceled in favor of new claim 28. It is submitted that a query can originate from any source and provided to the domain name service recited in claim 28, with the domain name service configured "to receive a query for a network address." Further, the clarifying amendments presented by new claim 28 should address the second issue raised by the Examiner.

Regarding the rejection of claims 26 and 27, the Examiner states that "it is unclear what kind of apparatus [the claimed subject matter] it is. It is unclear how this apparatus is configured. And, it is unclear for what a computer network address is queried. (sic)" Again, it is submitted that the clarifying amendments presented by new claim 28 should render the rejection moot. Claim 28 specifically describes the configuration of the domain name service system for establishing a secure communication link.

Claims 1-12, 26 and 27 have also been rejected under 35 U.S.C. § 103(a) as being unpatentable over IP Security Chapter 13 of XP-002167283 (also referred to as XP). Claims 1, 26 and 27 have been canceled. It is submitted that the reference neither anticipates nor makes obvious the claimed invention as defined by new claim 28.

New claim 28 recites a system for providing a domain name service for establishing a secure communication link. The system comprises a domain name service system configured to be connected:

to a communication network,

to store a plurality of domain names and corresponding network addresses,

to receive a query for a network address, and

to comprise an indication that the domain name service system supports establishing a secure communication link.

It is submitted that the reference XP neither anticipates nor makes obvious the claimed subject matter of claim 28, and the claims dependent thereon. The Examiner cites pages 399 and 400 of XP. These pages describe prior art approaches to IP security known at the time the reference was published (1998). XP mentions:

> ...security mechanisms in a number of application areas, including electronic mail (S/MIME, PGP), client/server Kerberos), Web access (Secure Sockets Layer), and others. However, users have some security concerns that cut across protocol layers. For example, an enterprise can run a secure, private TCP/IP network by disallowing links to untrusted sites, encrypting packets that leave the premises, and authenticating packets that enter the premises. By implementing security at the IP level, an organization can ensure secure networking not only for applications that have security mechanisms but for the many security-ignorant applications.
>
> IP-level security encompasses three functional areas: authentication, confidentiality, and key management. The authentication mechanism

> assures that a received packet was, in fact, transmitted by the party identified as the source in the packet header. In addition, this mechanism assures that the packet has not been altered in transit. The confidentiality facility enables communicating nodes to encrypt messages to prevent eavesdropping by third parties. The key management facility is concerned with the secure exchange of keys...

Thus, the reference describes the three function areas of any security system for protecting transmitted data across any network. The reference goes on to cite a 1994 report of the Internet Architecture Board that states that the Internet needs more and better security, and it identifies key areas for security mechanisms. Among these are the need to secure the network infrastructure form unauthorized monitoring and control of network traffic and the need to secure end-user-to-end-user traffic using authentication and encryption mechanisms. The report cites reported security incidents, the most serious types of attacks including IP spoofing, and various forms of eavesdropping and packet sniffing. Thus, the reference generally describes the need for secure transmission of data.

In describing IP Security (IPSec) the reference goes on to state:

> IPSec provides the capability to secure communications across a LAN, across private and public wide area networks (WANs), and across the Internet. Examples of its use include the following:
>
> Secure branch office connectivity over the Internet: A company can build a secure virtual private network over the Internet or over a public WAN This enables a business to rely heavily on the Internet and reduce its need for private networks, saving costs and network management overhead.
>
> Secure remote access over the Internet: An end user whose system is equipped with IP security protocols can make a local call to an Internet service provider (ISP) and gain secure access to a company network. This reduces the cost of toll charges for traveling employees and telecommuters...

The reference states at the bottom of page 400, "the principal feature of IPSec that enables it to support these varied applications is that it can encrypt and/or authenticate all traffic at the IP level."

The reference describes the overall architecture in Figure 13.2 (page 403). The blocks shown in the Figure are defined on pages 403 and 404. It is clear that the architecture described and shown in the reference does not describe nor even suggest a domain name service system configured to be connected to a communication network, to store a plurality of domain names and corresponding network addresses, to receive a query for a network address, and to comprise an indication that the domain name service system supports establishing a secure communication link. (cf., claim 28 of the above-identified application).

Dependent claims 2-12 and 29-51 currently under consideration in the application are dependent from independent Claim 28 discussed above and therefore are believed to be allowable over the applied reference for at least the same reasons. Because each dependent claim is deemed to define an additional aspect of the invention, the individual consideration of each on its own merits is respectfully requested.

Finally, claims 52 and 53 have been added. Claim 52 recites "a machine-readable medium comprising instructions executable in a domain name service system. The instructions comprise code for connecting the domain name service system to a communication network; storing a plurality of domain names and corresponding network addresses; receiving a query for a network address; and supporting an indication that the domain name service system supports establishing a secure communication link." Claim 53 recites "a method of providing a domain name service for establishing a secure communication link. The method comprises connecting a domain name service system to a communication network, the domain name service system comprising an indication that the domain name service system supports establishing a secure communication link; storing a plurality of domain names and corresponding network addresses; and receiving a query for a network address for communication."

Accordingly, all of the pending claims currently under consideration, claims 2-25 and 28-53, are believed to be patentable over the cited reference. An early and favorable action thereon is therefore earnestly solicited.

If a telephone conference will expedite prosecution of the application, the Examiner is invited to telephone the undersigned.

Respectfully submitted,

McDERMOTT WILL & EMERY LLP

Toby

Date: 11 July 2007

Toby H. Kusmer, P.C.
Reg. No. 26,418
Attorney for Applicants
28 State Street
Boston, MA 02109-1775
DD Telephone: (617) 535-4065
Facsimile: (617)535-3800
e-mail: tkusmer@mwe.com

COMBINED AMENDMENT & PETITION FOR EXTENSION OF TIME UNDER 37 CFR 1.136(a) (Large Entity)	Docket No. 77580-042 (VRNK-1CP3CN)

In Re Application Of: **Victor Larsen, et al.**

Application No.	Filing Date	Examiner	Customer No.	Group Art Unit	Confirmation No.
10/714,849	November 18, 2003	Krisna Lim	23630	2153	3154

Invention: **AN AGILE NETWORK PROTOCOL FOR SECURE COMMUNICATIONS**

COMMISSIONER FOR PATENTS:

This is a combined amendment and petition under the provisions of 37 CFR 1.136(a) to extend the period for filing a response to the Office Action of March 21, 2007 (*Date*) in the above-identified application.

The requested extension is as follows (check time period desired):

☒ One month ☐ Two months ☐ Three months ☐ Four months ☐ Five months

from: June 21, 2007 (*Date*) until: July 21, 2007 (*Date*)

The fee for the amendment and extension of time has been calculated as shown below:

CLAIMS AS AMENDED

	CLAIMS REMAINING AFTER AMENDMENT		HIGHEST # PREV. PAID FOR		NUMBER EXTRA CLAIMS PRESENT		RATE	ADDITIONAL FEE
TOTAL CLAIMS	37	-	27	=	10	x	$50.00	$500.00
INDEP. CLAIMS	3	-	7	=	0	x	$200.00	$0.00
FEE FOR AMENDMENT								$500.00
FEE FOR EXTENSION OF TIME								$120.00
TOTAL FEE FOR AMENDMENT AND EXTENSION OF TIME								$620.00

COMBINED AMENDMENT & PETITION FOR EXTENSION OF TIME UNDER 37 CFR 1.136(a) (Large Entity)	Docket No. 77580-042 (VRNK-1CP3CN)

The fee for the amendment and extension of time is to be paid as follows:

☐ A check in the amount of for the amendment and extension of time is enclosed.

☒ Please charge Deposit Account No. 501133 in the amount of $620.00

☒ The Director is hereby authorized to charge payment of the following fees associated with this communication or credit any overpayment to Deposit Account No. 501133

- ☒ Any additional filing fees required under 37 C.F.R. 1.16.
- ☒ Any patent application processing fees under 37 CFR 1.17.

☒ If an additional extension of time is required, please consider this a petition therefor and charge any additional fees which may be required to Deposit Account No. 501133

☐ Payment by credit card. Form PTO-2038 is attached.
WARNING: Information on this form may become public. Credit card information should not be included on this form. Provide credit card information and authorization on PTO-2038.

Signature

Dated: 7/11/07

Toby H. Kusmer, P.C.
Reg. No. 26,418
McDermott Will & Emery LLP
28 State Street
Boston, MA 02109
Telephone: 617-535-4065
Facsimile: 617-535-3800
e-mail: tkusmer@mwe.com

cc:

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to the "Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450" [37 CFR 1.8(a)] on

July 11, 2007. or by facsimile
(Date)

Cynthia Joseph
Signature of Person Mailing Correspondence

Cynthia Joseph
Typed or Printed Name of Person Mailing Correspondence

PATENT APPLICATION FEE DETERMINATION RECORD

Effective December 8, 2004

Application or Docket Number

10/714849

CLAIMS AS FILED - PART I

	(Column 1)	(Column 2)
TOTAL CLAIMS		
FOR	NUMBER FILED	NUMBER EXTRA
TOTAL CHARGEABLE CLAIMS	minus 20=	*
INDEPENDENT CLAIMS	minus 3 =	*
MULTIPLE DEPENDENT CLAIM PRESENT		□

* If the difference in column 1 is less than zero, enter "0" in column 2

SMALL ENTITY TYPE □ OR OTHER THAN SMALL ENTITY

RATE	FEE		RATE	FEE
BASIC FEE		OR	BASIC FEE	
X$ 25=		OR	X$50=	
X100=		OR	X200=	
+180=		OR	+360=	
TOTAL		OR	TOTAL	

CLAIMS AS AMENDED - PART II

AMENDMENT A

7/11/07	(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA
Total	* 50	Minus	** 27	= 23
Independent	* 7	Minus	*** 7	=
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM				□

SMALL ENTITY OR OTHER THAN SMALL ENTITY

RATE	ADDITIONAL FEE		RATE	ADDITIONAL FEE
X$ 25=		OR	X$50=	1150
X100=		OR	X200=	
+180=		OR	+360=	
TOTAL ADDIT. FEE		OR	TOTAL ADDIT. FEE	1150

PD

AMENDMENT B

	(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA
Total	*	Minus	**	=
Independent	*	Minus	***	=
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM				□

RATE	ADDITIONAL FEE		RATE	ADDITIONAL FEE
X$ 25=		OR	X$50=	
X100=		OR	X200=	
+180=		OR	+360=	
TOTAL ADDIT. FEE		OR	TOTAL ADDIT. FEE	

AMENDMENT C

	(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA
Total	*	Minus	**	=
Independent	*	Minus	***	=
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM				□

RATE	ADDITIONAL FEE		RATE	ADDITIONAL FEE
X$ 25=		OR	X$50=	
X100=		OR	X200=	
+180=		OR	+360=	
TOTAL ADDIT. FEE		OR	TOTAL ADDIT. FEE	

* If the entry in column 1 is less than the entry in column 2, write "0" in column 3.
** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 20, enter "20."
***If the "Highest Number Previously Paid For" IN THIS SPACE is less than 3, enter "3."
The "Highest Number Previously Paid For" (Total or Independent) is the highest number found in the appropriate box in column 1.

Patent and Trademark Office, U.S. DEPARTMENT OF COMMERCE

Document code: WFEE

United States Patent and Trademark Office
Sales Receipt for Accounting Date: 07/18/2007

```
ASMITH      SALE   #00000001     Mailroom Dt:   07/11/2007     501133    10714849
                        01      FC : 1202                   650.00  DA
```

IFW

STATEMENT UNDER 37 CFR 1.97(e) ACCOMPANYING INFORMATION DISCLOSURE STATEMENT	Docket No. 77580-042 (VRNK-1CP3CN)

In Re Application Of: **Victor Larson, et al.**

Application No.	Filing Date	Examiner	Customer No.	Group Art Unit	Confirmation No.
10/714,849	**November 18, 2003**	**Krisna Lim**	**23630**	**2153**	**3154**

Invention: **AN AGILE NETWORK PROTOCOL FOR SECURE COMMUNICATIONS**

COMMISSIONER FOR PATENTS:

This is a statement under the provisions of 37 CFR 1.97(e) in the above-identified application.

Check applicable statement herebelow:

Statement Under 37 CFR 1.97(e)(1)

☒ Each item of information contained in the accompanying Information Disclosure Statement was first cited in any communication from a foreign patent office in a counterpart foreign application not more than three months prior to the filing of the Information Disclosure Statement.

Statement Under 37 CFR 1.97(e)(2)

☐ No item of information contained in the accompanying Information Disclosure Statement was cited in a communication from a foreign patent office in a counterpart foreign application, and, to the knowledge of the undersigned person, after making reasonable inquiry, no item of information contained in the accompanying Information Disclosure Statement was known to any individual designated in 37 CFR 1.56(c) more than three months prior to the filing of the Information Disclosure Statement.

Signature

Dated: 7/12/07

Toby H. Kusmer, P.C.
Reg. No. 26,418
McDermott Will & Emery LLP
28 State Street
Boston, MA 02109
Telephone: 617-535-4065
Facsimile: 617-535-3800

cc:

Certificate of Mailing by First Class Mail

I certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to "Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450" [37 CFR 1.8(a)] on

7-12-07
(Date)

Cynthia Joseph
Signature of Person Mailing Correspondence

Cynthia Joseph
Typed or Printed Name of Person Mailing Correspondence

INFORMATION DISCLOSURE CITATION *(Use several sheets if necessary)*	Docket Number (Optional) 77580-042 (VRNK-1CP3CN)	Application Number 10/714,849
	Applicant(s) Victor Larson, et al.	
	Filing Date November 18, 2003	Group Art Unit 2153

U.S. PATENT DOCUMENTS

*EXAMINER INITIAL	REF	DOCUMENT NUMBER	DATE	NAME	CLASS	SUBCLASS	FILING DATE IF APPROPRIATE
		6,557,037	04/29/2003	Provino	709	227	05/29/1998

U.S. PATENT APPLICATION PUBLICATIONS

*EXAMINER INITIAL	REF	DOCUMENT NUMBER	DATE	NAME	CLASS	SUBCLASS	FILING DATE IF APPROPRIATE

FOREIGN PATENT DOCUMENTS

	REF	DOCUMENT NUMBER	DATE	COUNTRY	CLASS	SUBCLASS	Translation	
							YES	NO

OTHER DOCUMENTS *(Including Author, Title, Date, Pertinent Pages, Etc.)*

		Eastlake, D. E., "Domain Name System Security Extensions", Internet Draft, April 1998 (1998-04), XP002199931, Sections 1, 2.3 and 2.4.

EXAMINER	DATE CONSIDERED

EXAMINER: Initial if citation considered, whether or not citation is in conformance with MPEP Section 609; Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

SHEET 1 OF 1

EAST Search History

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L1	1099823	(secure domain name service).ti,ab,clm. or DSN.ti,ab,clm.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2007/09/17 07:53
L2	561599	l1 and (secure communication link).ti,ab,clm.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2007/09/17 07:54
L3	36321	l2 and (authenticat$4 or cryptograph$3)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2007/09/17 07:54
L4	5	l3 and DSN.ti,ab,clm.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2007/09/17 07:55
L5	386	l3 and DNS.ti,ab,clm.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2007/09/17 07:56
L6	303	l5 and server.ti,ab,clm.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2007/09/17 07:56
L7	6292	709/226, "221".ccls. or 713/201.ccls.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2007/09/17 07:57
L8	17	l7 and l6	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2007/09/17 07:57

EAST Search History

L9	4	l8 and @ad<="19990607"	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2007/09/17 08:00

McDermott
Will & Emery

Boston Brussels Chicago Düsseldorf London Los Angeles Miami Munich
New York Orange County Rome San Diego Silicon Valley Washington, D.C.
Strategic alliance with MWE China Law Offices (Shanghai)

FACSIMILE

Date: October 4, 2007 **Time Sent:**

To:	**Company:**	**Facsimile No:**	**Telephone No:**
Commissioner for Patents	U.S. Patent and Trademark Office	1.571.273.8300	

From:	Toby H. Kusmer, P.C.	*Direct Phone:*	617.535.4065
E-Mail:	tkusmer@mwe.com	*Direct Fax:*	617.535.3800
Sent By:	Cynthia Joseph	*Direct Phone:*	617.535.4111
Client/Matter/Tkpr:	77580-042/5496	*Original to Follow by Mail:*	No
		Number of Pages, Including Cover:	2

Re: In re Application of: Victor Larson, et al.
Serial No.: 10/714,849
Filing Date: November 18, 2003
Title: An Agile Network Protocol For Secure Communications Using Secure Domain Names
Docket No.: 77580-042 (VRNK-1CP3CN)

Message:

Please enter the attached Status Inquiry.

The information contained in this facsimile message is legally privileged and confidential information intended only for the use of the individual or entity named above. If the reader of this message is not the intended recipient, you are hereby notified that any dissemination, distribution, or copy of this facsimile is strictly prohibited. If you have received this facsimile in error, please notify us immediately by telephone and return the original message to us at the below address by mail. Thank you.

IF YOU DO NOT RECEIVE ALL OF THE PAGES, PLEASE CALL AS SOON AS POSSIBLE.

Main Facsimile: 617.535.3800 Facsimile Operator: 617.535.4000

U.S. practice conducted through McDermott Will & Emery LLP.
28 State Street Boston, Massachusetts 02109-1775 Telephone: 617.535.4000

BST99 1553858-1.077580.0042

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:	Victor Larson, et al.
Serial No:	10/714,849
Filing Date:	November 18, 2003
Title:	An Agile Network Protocol For Secure Communications Using Secure Domain Names
Group Art Unit:	2153
Confirmation No.:	3154
Docket No:	77580-042 (VRNK-1CP3CN)

Commissioner for Patents
P. O. Box 1450
Alexandria, VA 22313-1450

Sir:

STATUS INQUIRY

Applicants make a request as to the status of the above-identified application and for information as to when they might expect to receive an Office Action.

Respectfully submitted,

Toby H. Kusmer, P.C.
Registration Number 26,418
McDermott Will & Emery LLP
28 State Street
Boston, Massachusetts 02109-1775
Telephone: (617) 535-4065
Facsimile: (617) 535-3800
e-mail: tkusmer@mwe.com

CERTIFICATE OF TRANSMISSION

I hereby certify that this correspondence is being facsimile transmitted, via Facsimile No. 571.273.8300, to the U.S. Patent and Trademark Office and is addressed to: Commissioner For Patents, P. O. Box 1450, Alexander, VA 22313-1450 on the date indicated below.

Date: October 4, 2007

Cynthia Joseph

BST99 1553801-1.077580.0042

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

NOTICE OF ALLOWANCE AND FEE(S) DUE

22907 7590 10/29/2007

BANNER & WITCOFF, LTD.
1100 13th STREET, N.W.
SUITE 1200
WASHINGTON, DC 20005-4051

EXAMINER	
LIM, KRISNA	
ART UNIT	PAPER NUMBER
2153	

DATE MAILED: 10/29/2007

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/714,849	11/18/2003	Victor Larson	007170.00025	3154

TITLE OF INVENTION: AGILE NETWORK PROTOCOL FOR SECURE COMMUNICATIONS USING SECURE DOMAIN NAMES

APPLN. TYPE	SMALL ENTITY	ISSUE FEE DUE	PUBLICATION FEE DUE	PREV. PAID ISSUE FEE	TOTAL FEE(S) DUE	DATE DUE
nonprovisional	NO	$1440	$300	$0	$1740	01/29/2008

THE APPLICATION IDENTIFIED ABOVE HAS BEEN EXAMINED AND IS ALLOWED FOR ISSUANCE AS A PATENT. PROSECUTION ON THE MERITS IS CLOSED. THIS NOTICE OF ALLOWANCE IS NOT A GRANT OF PATENT RIGHTS. THIS APPLICATION IS SUBJECT TO WITHDRAWAL FROM ISSUE AT THE INITIATIVE OF THE OFFICE OR UPON PETITION BY THE APPLICANT. SEE 37 CFR 1.313 AND MPEP 1308.

THE ISSUE FEE AND PUBLICATION FEE (IF REQUIRED) MUST BE PAID WITHIN THREE MONTHS FROM THE MAILING DATE OF THIS NOTICE OR THIS APPLICATION SHALL BE REGARDED AS ABANDONED. THIS STATUTORY PERIOD CANNOT BE EXTENDED. SEE 35 U.S.C. 151. THE ISSUE FEE DUE INDICATED ABOVE DOES NOT REFLECT A CREDIT FOR ANY PREVIOUSLY PAID ISSUE FEE IN THIS APPLICATION. IF AN ISSUE FEE HAS PREVIOUSLY BEEN PAID IN THIS APPLICATION (AS SHOWN ABOVE), THE RETURN OF PART B OF THIS FORM WILL BE CONSIDERED A REQUEST TO REAPPLY THE PREVIOUSLY PAID ISSUE FEE TOWARD THE ISSUE FEE NOW DUE.

HOW TO REPLY TO THIS NOTICE:

I. Review the SMALL ENTITY status shown above.

If the SMALL ENTITY is shown as YES, verify your current SMALL ENTITY status:

A. If the status is the same, pay the TOTAL FEE(S) DUE shown above.

B. If the status above is to be removed, check box 5b on Part B - Fee(s) Transmittal and pay the PUBLICATION FEE (if required) and twice the amount of the ISSUE FEE shown above, or

If the SMALL ENTITY is shown as NO:

A. Pay TOTAL FEE(S) DUE shown above, or

B. If applicant claimed SMALL ENTITY status before, or is now claiming SMALL ENTITY status, check box 5a on Part B - Fee(s) Transmittal and pay the PUBLICATION FEE (if required) and 1/2 the ISSUE FEE shown above.

II. PART B - FEE(S) TRANSMITTAL, or its equivalent, must be completed and returned to the United States Patent and Trademark Office (USPTO) with your ISSUE FEE and PUBLICATION FEE (if required). If you are charging the fee(s) to your deposit account, section "4b" of Part B - Fee(s) Transmittal should be completed and an extra copy of the form should be submitted. If an equivalent of Part B is filed, a request to reapply a previously paid issue fee must be clearly made, and delays in processing may occur due to the difficulty in recognizing the paper as an equivalent of Part B.

III. All communications regarding this application must give the application number. Please direct all communications prior to issuance to Mail Stop ISSUE FEE unless advised to the contrary.

IMPORTANT REMINDER: Utility patents issuing on applications filed on or after Dec. 12, 1980 may require payment of maintenance fees. It is patentee's responsibility to ensure timely payment of maintenance fees when due.

PART B - FEE(S) TRANSMITTAL

Complete and send this form, together with applicable fee(s), to: Mail **Mail Stop ISSUE FEE**
Commissioner for Patents
P.O. Box 1450
Alexandria, Virginia 22313-1450
or Fax (571)-273-2885

INSTRUCTIONS: This form should be used for transmitting the ISSUE FEE and PUBLICATION FEE (if required). Blocks 1 through 5 should be completed where appropriate. All further correspondence including the Patent, advance orders and notification of maintenance fees will be mailed to the current correspondence address as indicated unless corrected below or directed otherwise in Block 1, by (a) specifying a new correspondence address; and/or (b) indicating a separate "FEE ADDRESS" for maintenance fee notifications.

CURRENT CORRESPONDENCE ADDRESS (Note: Use Block 1 for any change of address)

22907 7590 10/29/2007

BANNER & WITCOFF, LTD.
1100 13th STREET, N.W.
SUITE 1200
WASHINGTON, DC 20005-4051

Note: A certificate of mailing can only be used for domestic mailings of the Fee(s) Transmittal. This certificate cannot be used for any other accompanying papers. Each additional paper, such as an assignment or formal drawing, must have its own certificate of mailing or transmission.

Certificate of Mailing or Transmission

I hereby certify that this Fee(s) Transmittal is being deposited with the United States Postal Service with sufficient postage for first class mail in an envelope addressed to the Mail Stop ISSUE FEE address above, or being facsimile transmitted to the USPTO (571) 273-2885, on the date indicated below.

(Depositor's name)

(Signature)

(Date)

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/714,849	11/18/2003	Victor Larson	007170.00025	3154

TITLE OF INVENTION: AGILE NETWORK PROTOCOL FOR SECURE COMMUNICATIONS USING SECURE DOMAIN NAMES

APPLN. TYPE	SMALL ENTITY	ISSUE FEE DUE	PUBLICATION FEE DUE	PREV. PAID ISSUE FEE	TOTAL FEE(S) DUE	DATE DUE
nonprovisional	NO	$1440	$300	$0	$1740	01/29/2008

EXAMINER	ART UNIT	CLASS-SUBCLASS
LIM, KRISNA	2153	709-226000

1. Change of correspondence address or indication of "Fee Address" (37 CFR 1.363).

□ Change of correspondence address (or Change of Correspondence Address form PTO/SB/122) attached.

□ "Fee Address" indication (or "Fee Address" Indication form PTO/SB/47; Rev 03-02 or more recent) attached. **Use of a Customer Number is required.**

2. For printing on the patent front page, list

(1) the names of up to 3 registered patent attorneys or agents OR, alternatively,

(2) the name of a single firm (having as a member a registered attorney or agent) and the names of up to 2 registered patent attorneys or agents. If no name is listed, no name will be printed.

1 ____________________

2 ____________________

3 ____________________

3. ASSIGNEE NAME AND RESIDENCE DATA TO BE PRINTED ON THE PATENT (print or type)

PLEASE NOTE: Unless an assignee is identified below, no assignee data will appear on the patent. If an assignee is identified below, the document has been filed for recordation as set forth in 37 CFR 3.11. Completion of this form is NOT a substitute for filing an assignment.

(A) NAME OF ASSIGNEE

(B) RESIDENCE: (CITY and STATE OR COUNTRY)

Please check the appropriate assignee category or categories (will not be printed on the patent) : □ Individual □ Corporation or other private group entity □ Government

4a. The following fee(s) are submitted:

□ Issue Fee

□ Publication Fee (No small entity discount permitted)

□ Advance Order - # of Copies ____________________

4b. Payment of Fee(s): **(Please first reapply any previously paid issue fee shown above)**

□ A check is enclosed.

□ Payment by credit card. Form PTO-2038 is attached.

□ The Director is hereby authorized to charge the required fee(s), any deficiency, or credit any overpayment, to Deposit Account Number ____________ (enclose an extra copy of this form).

5. **Change in Entity Status** (from status indicated above)

□ a. Applicant claims SMALL ENTITY status. See 37 CFR 1.27.

□ b. Applicant is no longer claiming SMALL ENTITY status. See 37 CFR 1.27(g)(2).

NOTE: The Issue Fee and Publication Fee (if required) will not be accepted from anyone other than the applicant; a registered attorney or agent; or the assignee or other party in interest as shown by the records of the United States Patent and Trademark Office.

Authorized Signature ______________________________ Date ______________________________

Typed or printed name ______________________________ Registration No. ______________________________

This collection of information is required by 37 CFR 1.311. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, Virginia 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, Virginia 22313-1450.

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/714,849	11/18/2003	Victor Larson	007170.00025	3154

22907 7590 10/29/2007

BANNER & WITCOFF, LTD.
1100 13th STREET, N.W.
SUITE 1200
WASHINGTON, DC 20005-4051

EXAMINER	
LIM, KRISNA	
ART UNIT	**PAPER NUMBER**
2153	

DATE MAILED: 10/29/2007

Determination of Patent Term Adjustment under 35 U.S.C. 154 (b)

(application filed on or after May 29, 2000)

The Patent Term Adjustment to date is 663 day(s). If the issue fee is paid on the date that is three months after the mailing date of this notice and the patent issues on the Tuesday before the date that is 28 weeks (six and a half months) after the mailing date of this notice, the Patent Term Adjustment will be 663 day(s).

If a Continued Prosecution Application (CPA) was filed in the above-identified application, the filing date that determines Patent Term Adjustment is the filing date of the most recent CPA.

Applicant will be able to obtain more detailed information by accessing the Patent Application Information Retrieval (PAIR) WEB site (http://pair.uspto.gov).

Any questions regarding the Patent Term Extension or Adjustment determination should be directed to the Office of Patent Legal Administration at (571)-272-7702. Questions relating to issue and publication fee payments should be directed to the Customer Service Center of the Office of Patent Publication at 1-(888)-786-0101 or (571)-272-4200.

mn

Notice of Allowability	**Application No.** 10/714,849	**Applicant(s)** LARSON ET AL.	
	Examiner Krisna Lim	**Art Unit** 2153	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address--

All claims being allowable, PROSECUTION ON THE MERITS IS (OR REMAINS) CLOSED in this application. If not included herewith (or previously mailed), a Notice of Allowance (PTOL-85) or other appropriate communication will be mailed in due course. **THIS NOTICE OF ALLOWABILITY IS NOT A GRANT OF PATENT RIGHTS.** This application is subject to withdrawal from issue at the initiative of the Office or upon petition by the applicant. See 37 CFR 1.313 and MPEP 1308.

1. ☒ This communication is responsive to *the amendment filed 7/11/07*.

2. ☒ The allowed claim(s) is/are *2-12 and 28-53*.

3. □ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
   a) □ All b) □ Some* c) □ None of the:
   1. □ Certified copies of the priority documents have been received.
   2. □ Certified copies of the priority documents have been received in Application No. _____ .
   3. □ Copies of the certified copies of the priority documents have been received in this national stage application from the International Bureau (PCT Rule 17.2(a)).

   * Certified copies not received: _____.

Applicant has THREE MONTHS FROM THE "MAILING DATE" of this communication to file a reply complying with the requirements noted below. Failure to timely comply will result in ABANDONMENT of this application.
THIS THREE-MONTH PERIOD IS NOT EXTENDABLE.

4. □ A SUBSTITUTE OATH OR DECLARATION must be submitted. Note the attached EXAMINER'S AMENDMENT or NOTICE OF INFORMAL PATENT APPLICATION (PTO-152) which gives reason(s) why the oath or declaration is deficient.

5. □ CORRECTED DRAWINGS (as "replacement sheets") must be submitted.
   (a) □ including changes required by the Notice of Draftsperson's Patent Drawing Review (PTO-948) attached
   1) □ hereto or 2) □ to Paper No./Mail Date _____.
   (b) □ including changes required by the attached Examiner's Amendment / Comment or in the Office action of Paper No./Mail Date _____.

Identifying indicia such as the application number (see 37 CFR 1.84(c)) should be written on the drawings in the front (not the back) of each sheet. Replacement sheet(s) should be labeled as such in the header according to 37 CFR 1.121(d).

6. □ DEPOSIT OF and/or INFORMATION about the deposit of BIOLOGICAL MATERIAL must be submitted. Note the attached Examiner's comment regarding REQUIREMENT FOR THE DEPOSIT OF BIOLOGICAL MATERIAL.

Attachment(s)

1. □ Notice of References Cited (PTO-892)
2. □ Notice of Draftperson's Patent Drawing Review (PTO-948)
3. ☒ Information Disclosure Statements (PTO/SB/08), Paper No./Mail Date _____
4. □ Examiner's Comment Regarding Requirement for Deposit of Biological Material
5. □ Notice of Informal Patent Application
6. □ Interview Summary (PTO-413), Paper No./Mail Date _____ .
7. ☒ Examiner's Amendment/Comment
8. ☒ Examiner's Statement of Reasons for Allowance
9. □ Other _____.

Examiner's Amendment

An Examiner's Amendment to the record appears below. Should the changes and/or additions be unacceptable to applicant, an amendment may be filed as provided by 37 C.F.R. 1.312. To ensure consideration of such an amendment, it **MUST** be submitted no later than the payment of the Issue Fee.

In the claims:

Cancel claims 13-25.

Pursuant to 37 C.F.R 1.109 and M.P.E.P 1302.14, the following is an Examiner's Statement of Reasons for Allowance:

The prior arts of record do not teach or a domain name service system configured to be connected to a communication network, to store a plurality of domain names and corresponding network addresses, to receive a query for a network address, and to comprise an indication that the domain name service system supports establishing a secure communication link.

The examiner considers the applicants' claims 2-12 and 28-53 to be allowable based on the claim interpretation and the aforesaid prior arts of record.

Any comments considered necessary by applicant must be submitted no later than the payment of the Issue Fee and, to avoid processing delays, should preferably **accompany** the Issue Fee. Such submissions should be clearly labeled "Comments on Statement of Reasons for Allowance."

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Krisna Lim whose telephone number is 571-272-3956 The examiner can normally be reached on Monday to Friday from 9:30 AM to 6:00 PM. If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Glenton Burgess, can be reached on 571-272-3949. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

KI

September 16, 2007

KRISNA LIM
PRIMARY EXAMINER

INFORMATION DISCLOSURE CITATION *(Use several sheets if necessary)*	Docket Number (Optional) 77580-042 (VRNK-1CP3CN)	Application Number 10/714,849
	Applicant(s) Victor Larson, et al.	
	Filing Date November 18, 2003	Group Art Unit 2153

U.S. PATENT DOCUMENTS

*EXAMINER INITIAL	REF	DOCUMENT NUMBER	DATE	NAME	CLASS	SUBCLASS	FILING DATE IF APPROPRIATE
KL		6,557,037	04/29/2003	Provino	709	227	05/29/1998

U.S. PATENT APPLICATION PUBLICATIONS

*EXAMINER INITIAL	REF	DOCUMENT NUMBER	DATE	NAME	CLASS	SUBCLASS	FILING DATE IF APPROPRIATE

FOREIGN PATENT DOCUMENTS

	REF	DOCUMENT NUMBER	DATE	COUNTRY	CLASS	SUBCLASS	Translation	
							YES	NO

OTHER DOCUMENTS *(Including Author, Title, Date, Pertinent Pages, Etc.)*

KL		Eastlake, D. E., "Domain Name System Security Extensions", Internet Draft, April 1998 (1998-04), XP002199931, Sections 1, 2.3 and 2.4.

EXAMINER KRISNA LIM	DATE CONSIDERED 9/16/07

EXAMINER: Initial if citation considered, whether or not citation is in conformance with MPEP Section 609; Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

Form PTO-A820
(also form PTO-1449)

P09A/REV05

Patent and Trademark Office • U.S. DEPARTMENT OF COMMERCE

SHEET 1 OF 1

Issue Classification	Application/Control No. 10/714,849	Applicant(s)/Patent under Reexamination LARSON ET AL.
	Examiner Krisna Lim	Art Unit 2153

ISSUE CLASSIFICATION

ORIGINAL		INTERNATIONAL CLASSIFICATION								
CLASS	SUBCLASS	CLAIMED				NON-CLAIMED				
709	226	G	06	F	15 /173					/
					/					/
					/					/
					/					/
					/					/
					/					/
					/					/

CROSS REFERENCES				
CLASS	SUBCLASS (ONE SUBCLASS PER BLOCK)			
709	221			
713	201			

(Assistant Examiner) (Date)

KRISNA LIM
PRIMARY EXAMINER 9/7/07

(Primary Examiner) (Date)

(Legal Instruments Examiner) (Date)

Total Claims Allowed: 37	
O.G. Print Claim(s)	O.G. Print Fig.
1	1

□ Claims renumbered in the same order as presented by applicant □ CPA □ T.D. □ R.1.47

Final	Original	Final	Original	Final	Original	Final	Original	Final	Original	Final	Original	Final	Original
	1	15	31		61		91		121		151		181
3	2	16	32		62		92		122		152		182
4	3	17	33		63		93		123		153		183
6	4	18	34		64		94		124		154		184
7	5	19	35		65		95		125		155		185
5	6	20	36		66		96		126		156		186
8	7	21	37		67		97		127		157		187
13	8	22	38		68		98		128		158		188
9	9	23	39		69		99		129		159		189
10	10	24	40		70		100		130		160		190
11	11	25	41		71		101		131		161		191
12	12	26	42		72		102		132		162		192
	13	27	43		73		103		133		163		193
	14	28	44		74		104		134		164		194
	15	29	45		75		105		135		165		195
	16	30	46		76		106		136		166		196
	17	31	47		77		107		137		167		197
	18	32	48		78		108		138		168		198
	19	33	49		79		109		139		169		199
	20	34	50		80		110		140		170		200
	21	35	51		81		111		141		171		201
	22	36	52		82		112		142		172		202
	23	37	53		83		113		143		173		203
	24		54		84		114		144		174		204
	25		55		85		115		145		175		205
	26		56		86		116		146		176		206
	27		57		87		117		147		177		207
1	28		58		88		118		148		178		208
2	29		59		89		119		149		179		209
14	30		60		90		120		150		180		210

Index of Claims	Application/Control No. 10/714,849	Applicant(s)/Patent under Reexamination LARSON ET AL.	
	Examiner Krisna Lim	**Art Unit** 2153	

√	**Rejected**	–	**(Through numeral) Cancelled**	N	**Non-Elected**	A	**Appeal**
=	**Allowed**	÷	**Restricted**	I	**Interference**	O	**Objected**

Claim		Date								
Final	Original	9/16/07								
	1									
3	2	=								
4	3	=								
6	4	=								
7	5	=								
5	6	=								
8	7	=								
13	8	=								
9	9	=								
10	10	=								
11	11	=								
12	12	=								
	13									
	14									
	15									
	16									
	17									
	18									
	19									
	20									
	21									
	22									
	23									
	24									
	25									
	26									
	27									
1	28	=								
2	29	=								
14	30	=								
15	31	=								
16	32	=								
17	33	=								
18	34	=								
19	35	=								
20	36	=								
21	37	=								
22	38	=								
23	39	=								
24	40	=								
25	41	=								
26	42	=								
27	43	=								
28	44	=								
29	45	=								
30	46	=								
31	47	=								
32	48	=								
33	49	=								
34	50	=								

Claim		Date								
Final	Original	9/16/07								
35	51	=								
36	52	=								
37	53	=								
	54									
	55									
	56									
	57									
	58									
	59									
	60									
	61									
	62									
	63									
	64									
	65									
	66									
	67									
	68									
	69									
	70									
	71									
	72									
	73									
	74									
	75									
	76									
	77									
	78									
	79									
	80									
	81									
	82									
	83									
	84									
	85									
	86									
	87									
	88									
	89									
	90									
	91									
	92									
	93									
	94									
	95									
	96									
	97									
	98									
	99									
	100									

Claim		Date								
Final	Original									
	101									
	102									
	103									
	104									
	105									
	106									
	107									
	108									
	109									
	110									
	111									
	112									
	113									
	114									
	115									
	116									
	117									
	118									
	119									
	120									
	121									
	122									
	123									
	124									
	125									
	126									
	127									
	128									
	129									
	130									
	131									
	132									
	133									
	134									
	135									
	136									
	137									
	138									
	139									
	140									
	141									
	142									
	143									
	144									
	145									
	146									
	147									
	148									
	149									
	150									

U.S. Patent and Trademark Office

Part of Paper No. 20070917

Search Notes	Application/Control No. 10/714,849	Applicant(s)/Patent under Reexamination LARSON ET AL.	
	Examiner Krisna Lim	Art Unit 2153	

SEARCHED

Class	Subclass	Date	Examiner
709	226, 221	9/16/2007	KL
713	201	9/16/2007	KL

SEARCH NOTES (INCLUDING SEARCH STRATEGY)

	DATE	EXMR
EAST	9/16/2007	KL
Google	9/16/2007	KL

INTERFERENCE SEARCHED

Class	Subclass	Date	Examiner
709	**226, 221**	9/16/2007	**KL**
713	201	9/16/2007	KL

11-09-07

IFW

Docket No.: 077580-0042

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant	:	Victor Larson et al.
Appl. No.	:	10/714,849
Filed	:	November 18, 2003
Title	:	AN AGILE NETWORK PROTOCOL FOR SECURE COMMUNICATIONS USING SECURE DOMAIN NAMES
Grp./A.U.	:	2153
Examiner:	:	LIM, Krisna

Customer No.: 23,630
Confirmation No.: 3154

CERTIFICATE OF MAILING (37 CFR. § 1.10)

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as "Express Mail Post Office to Addressee" under 37 CFR 1.10 in an envelope addressed to Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on **November 8, 2007**.

Express Mail Mailing Label: EV 942455095 US

Cynthia Joseph
Cynthia Joseph

INFORMATION DISCLOSURE STATEMENT

Mail Stop Amendment
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

In accordance with the provisions of 37 C.F.R. 1.56, 1.97 and 1.98, the attention of the Patent and Trademark Office is hereby directed to the documents listed on the attached form PTO-1449. It is respectfully requested that the documents be expressly considered during the prosecution of this application, and that the documents be made of record therein and appear among the "References Cited" on any patent to issue therefrom.

This Information Disclosure Statement is being filed before the receipt of a Final Office Action for the above-referenced application. The commissioner is authorized to charge a submission fee of $180.00 to our Deposit Account No. 50-1133.

This Statement is not to be interpreted as a representation that the cited publications are material, that an exhaustive search has been conducted, or that no other relevant information exists. Nor shall the citation of any publication herein be construed *per se* as a representation that such publication is prior art. Moreover, the Applicant understands that the Examiner will make an independent evaluation of the cited publications.

Please charge any shortage in fees due in connection with the filing of this paper, including extension of time fees, to Deposit Account 50-1133 and please credit any excess fees to such deposit account.

Respectfully submitted,

McDERMOTT WILL & EMERY LLP

Atabak Royaee

Atabak R. Royaee
Registration No. 59,037

28 State Street
Boston, MA 02109
Phone: 617-535-4108
Facsimile: 617-535-3800
Date: November 8 , 2007

**Please recognize our Customer No. 23630
as our correspondence address.**

SHEET 1 OF 4

INFORMATION DISCLOSURE CITATION IN AN APPLICATION (PTO-1449)	ATTY. DOCKET NO. **077580-0042**	SERIAL NO. **10/714,849**
	APPLICANT **Larson et al.**	
	FILING DATE **Nov. 18, 2003**	GROUP **2153**

U.S. PATENT DOCUMENTS

EXAMINER'S INITIALS	CITE NO.	Document Number Number-Kind Code2 (if known)		Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear
	A1	US	4,933,846 A	6/12/1990	Humphrey et al.	
	A2	US	4,988,990 A	1/29/1991	Warrior	
	A3	US	5,276,735 A	1/4/1994	Boebert et al	
	A4	US	5,311,593 A	5/10/1994	Carmi	
	A5	US	5,329,521 A	7/12/1994	Walsh et al.	
	A6	US	5,341,426 A	8/23/1994	Barney et al.	
	A7	US	5,367,643 A	11/22/1994	Chang et al	
	A8	US	5,559,883 A	9/24/1996	Williams	
	A9	US	5,561,669 A	10/1/1996	Lenney et al	
	A10	US	5,588,060 A	12/24/1996	Aziz	
	A11	US	5,625,626 A	4/29/1997	Umekita	
	A12	US	5,654,695 A	8/5/1997	Olnowich et al	
	A13	US	5,682,480 A	10/28/1997	Nakagawa	
	A14	US	5,689,566 A	11/18/1997	Nguyen	
	A15	US	5,740,375 A	4/14/1998	Dunne et al.	
	A16	US	5,774,660 A	6/30/1998	Brendel et al	
	A17	US	5,787,172 A	7/28/1998	Arnold	
	A18	US	5,796,942 A	8/18/1998	Esbensen	
	A19	US	5,805,801 A	9/8/1998	Holloway et al.	
	A20	US	5,842,040 A	11/24/1998	Hughes et al.	
	A21	US	5,845,091 A	12/1/1998	Dunne et al.	
	A22	US	5,867,650 A	2/2/1998	Osterman	
	A23	US	5,870,610 A	2/9/1999	Beyda et al.	
	A24	US	5,878,231 A	5/2/1999	Baehr et al	
	A25	US	5,892,903 A	4/6/1999	Klaus	
	A26	US	5,898,830 A	4/27/1999	Wesinger, Jr. et al.	
	A27	US	5,905,859 A	5/18/1999	Holloway et al.	
	A28	US	5,918,019 A	6/29/1999	Valencia	
	A29	US	5,996,016 A	11/30/1999	Thalheimer et al.	
	A30	US	6,006,259 A	12/21/1999	Adelman et al.	
	A31	US	6,006,272 A	12/21/1999	Aravamudan et al	

EXAMINER	DATE CONSIDERED

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant.
1 Applicant's unique citation designation number (optional). 2 Applicant is to place a check mark here if English language Translation is attached.

SHEET 2 OF 4

INFORMATION DISCLOSURE CITATION IN AN APPLICATION (PTO-1449)	ATTY. DOCKET NO. **077580-0042**	SERIAL NO. **10/714,849**
	APPLICANT **Larson et al.**	
	FILING DATE **Nov. 18, 2003**	GROUP **2153**

U.S. PATENT DOCUMENTS

EXAMINER'S INITIALS	CITE NO.	Document Number Number-Kind Code2 (if known)		Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear
	A32	US	6,016,318 A	1/18/2000	Tomoike	
	A33	US	6,016,512	1/18/2000	Huitema	
	A34	US	6,041,342 A	3/21/2000	Yamaguchi	
	A35	US	6,052,788 A	4/18/2000	Wesinger, Jr. et al.	
	A36	US	6,055,574 A	4/25/2000	Smorodinsky et al.	
	A37	US	6,061,736 A	5/9/2000	Rochberger et al	
	A38	US	6,079,020 A	6/20/2000	Liu	
	A39	US	6,092,200 A	7/18/2000	Muniyappa et al.	
	A40	US	6,119,171 A	9/12/2000	Alkhatib	
	A41	US	6,119,234 A	9/12/2000	Aziz et al.	
	A42	US	6,147,976 A	11/14/2000	Shand et al.	
	A43	US	6,157,957 A	12/5/2000	Berthaud	
	A44	US	6,158,011 A	12/5/2000	Chen et al.	
	A45	US	6,168,409 B1	1/2/2001	Fare	
	A46	US	6,175,867 B1	1/16/2001	Taghadoss	
	A47	US	6,178,409 B1	1/23/2001	Weber et al.	
	A48	US	6,178,505 B1	1/23/2001	Schneider et al	
	A49	US	6,179,102 B1	1/30/2001	Weber, et al.	
	A50	US	6,222,842 B1	4/24/2001	Sasyan et al.	
	A51	US	6,226,751 B1	5/1/2001	Arrow et al	
	A52	US	6,233,618 B1	5/15/2001	Shannon	
	A53	US	6,243,360 B1	6/5/2001	Basilico	
	A54	US	6,243,749 B1	6/5/2001	Sitaraman et al.	
	A55	US	6,243,754 B1	6/5/2001	Guerin et al	
	A56	US	6,256,671 B1	7/3/2001	Strentzsch et al.	
	A57	US	6,263,445 B1	7/17/2001	Blumenau	
	A58	US	6,286,047 B1	9/4/2001	Ramanathan et al	
	A59	US	6,301,223 B1	10/9/2001	Hrastar et al	
	A60	US	6,308,274 B1	10/23/2001	Swift	
	A61	US	6,311,207 B1	10/30/2001	Mighdoll et al	
	A62	US	6,324,161 B1	11/27/2001	Kirch	
	A63	US	6,330,562 B1	12/11/2001	Boden et al.	

EXAMINER	DATE CONSIDERED

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

1 Applicant's unique citation designation number (optional). 2 Applicant is to place a check mark here if English language Translation is attached.

SHEET 3 OF 4

INFORMATION DISCLOSURE CITATION IN AN APPLICATION	ATTY. DOCKET NO. **077580-0042**	SERIAL NO. **10/714,849**
	APPLICANT **Larson et al.**	
(PTO-1449)	FILING DATE **Nov. 18, 2003**	GROUP **2153**

U.S. PATENT DOCUMENTS

EXAMINER'S INITIALS	CITE NO.	Document Number Number-Kind Code2 (if known)		Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear
	A64	US	6,332,158 B1	12/18/2001	Risley et al.	
	A65	US	6,353,614 B1	3/5/2002	Borella et al.	
	A66	US	6,430,155 B1	8/6/2002	Davie et al	
	A67	US	6,430,610 B1	8/6/2002	Carter	
	A68	US	6,487,598 Bi	11/26/2002	Valencia	
	A69	US	6,502,135 B1	12/31/2002	Munger et al	
	A70	US	6,505,232 B1	1/7/2003	Mighdoll et al	
	A71	US	6,510,154 B1	1/21/2003	Mayes et al	
	A72	US	6,549,516 B1	4/15/2003	Albert et al	
	A73	US	6,557,037 B1	4/29/2007	Provino	
	A74	US	6,571,296 B1	5/27/2002	Dillon	
	A75	US	6,571,338 B1	5/27/2003	Shaio et al.	
	A76	US	6,581,166 B1	7/17/2003	Hirst et al.	
	A77	US	6,618,761 B2	9/9/2003	Munger et al.	
	A78	US	6,671,702 B2	12/30/2003	Kruglikov et al	
	A79	US	6,687,551 B1	2/3/2004	Steindl	
	A80	US	6,714,970 B1	3/30/2004	Fiveash et al.	
	A81	US	6,717,949 B1	4/6/2004	Boden et al.	
	A82	US	6,760,766 B1	7/6/2004	Sahlqvist	
	A83	US	6,826,616 B2	11/30/2004	Larson et al.	
	A84	US	6,839,759 B2	1/4/2005	Larson et al.	
	A85	US	7,010,604 B1	3/7/2006	Munger et al.	
	A86	US	7,133,930 B2	11/7/2006	Munger et al.	
	A87	US	7,188,180 B2	3/6/2007	Larson et al.	
	A88	US	7,197,563 B2	3/27/2007	Sheymov et al.	
	A89	US	2002/0004898 A1	1/10/2002	Droge	
	A90	US	2005/0055306 A1	3/10/2005	Miller et al.	

FOREIGN PATENT DOCUMENTS

EXAMINER'S INITIALS	CITE NO.	Foreign Patent Document Country Code3 -Number 4 -Kind Codes (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines Where Relevant Figures Appear	Translation	
						Yes	No

EXAMINER	DATE CONSIDERED

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant.
1 Applicant's unique citation designation number (optional). 2 Applicant is to place a check mark here if English language Translation is attached.

SHEET 4 OF 4

INFORMATION DISCLOSURE CITATION IN AN APPLICATION (PTO-1449)	ATTY. DOCKET NO. 077580-0042	SERIAL NO. 10/714,849
	APPLICANT Larson et al.	
	FILING DATE Nov. 18, 2003	GROUP 2153

U.S. PATENT DOCUMENTS

EXAMINER'S INITIALS	CITE NO.	Document Number Number-Kind Code2 (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear

FOREIGN PATENT DOCUMENTS

EXAMINER'S INITIALS	CITE NO.	Foreign Patent Document Country Code3-Number 4-Kind Codes (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines Where Relevant Figures Appear	Translation	
						Yes	No
	B1	EP 836306A1	4/15/1998	Sasyan et al.			
	B2	WO 00/17775	3/30/2000	Miller et al.			

OTHER ART (Including Author, Title, Date, Pertinent Pages, Etc.)

EXAMINER'S INITIALS	CITE NO.	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc.), date, page(s), volume-issue number(s), publisher, city and/or country where published.	
	C1	RFC 2401 (dated Nov. 1998) Security Architecture for the Internet Protocol (RTP)	
	C2	RFC 2543-SIP (dated March 1999): Session Initiation Protocol (SIP or SIPS)	
	C3	Search Report, IPER (dataed Nov. 13, 2002), International Applicatoin No. PCT/US01/04340.	
	C4	Search Report, IPER (dated Feb. 06, 2002), International Application No. PCT/US01/13261.	
	C5	Search Report, IPER (dated Jan. 14, 2003), International Application No. PCT/US01/13260.	
	C6	Shankar, A.U. "A verified sliding window protocol with variable flow control". Proceedings of ACM SIGCOMM conferece on Communications architectures & protocols. pp. 84-91, ACM Press, NY,NY 1986.	
	C7	W. Stallings, "Cryptography And Network Security", 2nd, Edition, Chapter 13, IP Security, Jun. 8, 1998, pp. 399–440.	

EXAMINER	DATE CONSIDERED

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

1 Applicant's unique citation designation number (optional). 2 Applicant is to place a check mark here if English language Translation is attached.

B1

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 0 836 306 A1

(12) **EUROPEAN PATENT APPLICATION**

(43) Date of publication:
15.04.1998 Bulletin 1998/16

(51) Int. Cl.^{6}: **H04L 29/06**, H04Q 11/04

(21) Application number: 96410106.7

(22) Date of filing: 10.10.1996

(84) Designated Contracting States:
AT BE CH DE DK ES FI FR GB GR IE IT LI LU MC
NL PT SE
Designated Extension States:
AL LT LV SI

(71) Applicant:
Hewlett-Packard Company
Palo Alto, California 94304 (US)

(72) Inventors:
- Sasyan, Serge
38170 Seyssinet (FR)
- Roger, Denis
38760 Varces (FR)
- Terrasse, Denis
38320 Eybens (FR)

(74) Representative:
Squibbs, Robert Francis
Intellectual Property Section,
Legal Department,
Hewlett-Packard France,
Etablissements de Grenoble
F-38053 Grenoble Cédex 9 (FR)

Remarks:
The application is published incomplete as filed (Article 93 (2) EPC). A claim No.7 is missing.

(54) System providing for multiple virtual circuits between two network entities

(57) Computers sending IP datagrams over an ATM network are generally capable of operating multiple simultaneous virtual circuits over the network. However, in doing so, they normally only set up one virtual circuit to each destination IP address so that in order to test the simultaneous operation of N virtual circuits by a computer under test, N target computers are needed. To enable a single computer (T) to provide the destination endpoints for multiple virtual circuits (SVC) from a computer (M) under test, both computers (M,T) are allocated a plurality of virtual IP addresses ($I_{M(j)}$,$I_{T(i)}$) and the target computer (T) is additionally provided with a module running address-changing processes (70,71) that avoids the IP layers (20) of both computers from rejecting IP datagrams (25A,25B) addressed with the virtual IP addresses. As a result, each computer (M,T) can be addressed with any of a plurality of IP addresses and each will result in the creation of a respective virtual circuit (SVC) between the computers (M,T).

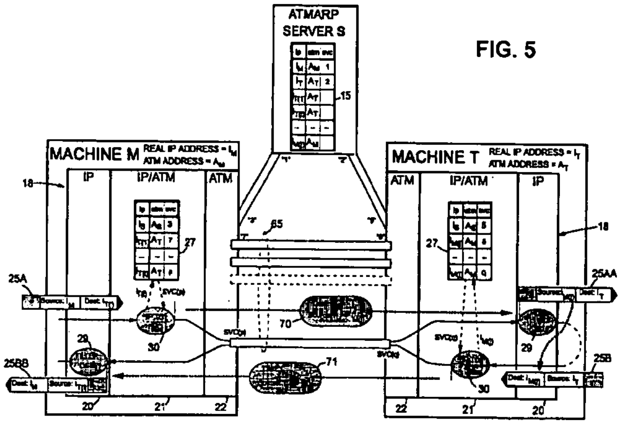


FIG. 5

Printed by Xerox (UK) Business Services
2.15.12/3.4

Description

Field of the Invention

The present invention relates to a system providing for multiple virtual circuits between two network entities for use in particular, but not exclusively, in the testing of network node apparatus providing IP messaging over an ATM network.

Background of the Invention

As is well-known, the Internet Protocol (IP) uses a scheme of IP addresses by which every connection of a node to the Internet has a unique IP address. IP addresses are high-level addresses in the sense that they are independent of the technology used for the underlying network to which a node is connected. Each node will also have a low-level, network-dependent address (often callled the MAC address) that is actually used for addressing at the network level and the IP protocol suite includes a address reolution protocol (ARP), logically positioned below the IP layer itself, that is responsible for translating between IP addresses contained in a message and the local MAC addresses.

An increasingly important technology for local area networks is ATM. ATM (Asynchronous Transfer Mode) is a multiplexing and switching technique for transferring data across a network using fixed sized cells that are synchronous in the sense that they appear strictly periodically on the physical medium. Each cell comprises a payload portion and a header, the latter including a label that associates the cell with an instance of communication between sending and receiving network end systems; this instance of communication may involve the transfer of many cells from the sending end system, possibly to multiple receiving end systems. ATM is asynchronous in the sense that cells belonging to the same instance of communication will not necessarily appear at periodic intervals.

In ATM, the labels appended to the cells are fixed-size context dependent labels, that is, they are only understandable in the light of context information already established at the interpreting network node, the label generally being replaced at one node by the label required for the next node. In other words, ATM is a virtual circuit technology requiring a set up phase for each instance of communication to establish the appropriate label knowledge at each node. Of course, to set up a desired communication, it is still necessary to identify uniquely the nodes forming the communication end points and this is achieved by using ATM addresses, generally of a significance limited to the particular ATM network concerned.

The process of sending IP messages (datagrams) over a ATM network, including the operation of the required ATM ARP system, is set out in RFC 1577 of the IETF Internet Engineering Task Force) dated January 1993. This RFC assumes an arrangement in which a sending node will only establish a single vircuit circuit to a given destination IP address (of course, this one vircuit circuit may carry multiple connections between respective pairings of high-level end points in the nodes).

Figure 1 of the accompanying drawings is a diagram illustrating the basic mechanism by which two machines M and T exchange IP datagrams over a switched virtual circuit (SVC) established across an ATM network. The machines M and T have respective IP addresses I_M and I_T and respective ATM addresses A_M and A_T; each machine knows its own addresses. An ATMARP server S knows the IP and ATM addresses of all active nodes on the network, including machines M and T; more particularly, server S maintains an ARP table 15 associating the IP address of each node with its ATM address. The server S maintains open a respective SVC (switched virtual circuit) to each active node and the identity of this SVC is held in the ARP table 15; thus, in the Figure 1 example, the server S is in communication with machine M over an SVC identified as SVC "1" at the server, and the server S is in communication with machine T over an SVC identified as SVC "2" at the server S. At machines M and T these virtual circuits are independently identified - thus at machine M its SVC to the server S is identified as SVC "3" whilst at machine T its SVC to the server S is identified as SVC "5".

The communications interface 18 in each of the machines M and T comprises three main layers, namely: an IP layer 20 responsible for forming IP datagrams (including source and destination IP addresses) for transmission ad for filtering incoming datagrams; an intermediate IP/ATM layer 21 for determining the SVC corresponding to the destination IP address of an outgoing datagram; and an ATM layer 22, including the low-level network interface hardware, for sending and receiving datagrams packaged in ATM cells over SVCs.

The IP/ATMlayer 21 maintains an ARP cache table 27 which like the table 15 of the server S contains associations between IP address, ATM address and SVC. Thus, table 27 of machine M contains an entry of the IP address I_S, ATM address A_S, and SVC identity "3" for the server S, and similarly, table 27 of machine T contains an entry of the IP address I_S, ATM address A_S, and SVC identity "5" for the server S. The cache table 27 only holds information relevant to current SVCs of the machine concerned so that during the initial establishment of a SVC to a new destination, the cache table must be updated with relevant information from the ATMARP server S; this general process will be described in more detail hereinafter with reference to Figure 2. For the present, it will be assumed that an SVC has already been established between machines M and T and that the cache tables contain the relevant information (in particular, cache table 27 of machine M contains an entry with the IP address I_T, ATM address A_T, and SVC identity "4" for machine T, and cache table

27 of machine T contains an entry with the IP address I_M, ATM address A_M, and SVC identity "9" for machine M).

Considering now the case of a high-level application in machine M wanting to send a message to machine T, this application passes the message to the IP layer 20 together with the destination IP address I_T. IP layer 20 packages the message in one (or more) datagrams 25A with a destination IP address of I_T and source I_P address of I_M. Datagram 25A is then passed to the IP/ATM layer 21 which executes an IP-to-SVC lookup task 30 to determine from table 27 the SVC to be used for sending the datagram to its destination address I_T; in the present case, table 27 returns the SVC identity "4" and the layer 21 passes this identity together with the datagram 25A to the ATM layer 22 which then sends the datagram in ATM cells on SVC "4". The datagram is in due course received by machine T and passed up by layers 22 and 21 to the IP layer 20 where a filtering task 29 determines from the datagram destination address that the datagram is indeed intended for machine T; the contents of the datagram are then passed to the relevant high-level application. In the present example, this high-level application produces a reply message which it passes to the IP layer 20 together with the required return address, namely the source IP address in the received datagram 25A. IP layer 20 generates datagram 25B with the received return address as the destination address, the IP address I_T of machine T being included as the source address. The datagram 25B is passed to IP/ATM layer 21 where IP-to-SVC lookup task 30 determines from cache table 27 that the required destination can be reached over SVC "9". This information together with datagram 25B is then passed to ATM layer 22 which transmits the datagram in ATM cells over SVC "9" to machine M. When the datagram is received at machine M it is passed up to the IP layer 20 where it is filtered by task 29 and its contents then passed on to the relevant high-level application.

Figure 2 of the accompanying drawings illustrates in more detail the functioning of the IP/ATM layers 21 of machines M and T in respect of datagram transmission from machine M to machine T, it being appreciated that the roles of the two layers 21 are reversed for transmission in the opposite direction. More particularly, upon the IP-to-SVC lookup task 30 being requested to send a datagram to IP address I_T, it first carries out a check of the cache table 27 (step 31) to determine if there is an existing entry for I_T (and thus an SVC, assuming that entries are only maintained whilst an SVC exists). Step 32 checks the result of this lookup - if an SVC already exists (in this case, SVC "4"), then step 39 is executed in which the datagram is passed together with the identity of the relevant SVC to the ATM layer 22; however, if the lookup was unsuccessful, task 30 executes steps 33 to 38 to set up an SVC to destination I_T before executing step 39.

The first step 33 of the setup process involves the sending of an ARP request to the ATMARP server S over the relevant SVC requesting the ATM address corresponding to I_T. Server responds with ATM address A_T which is received by task 30 at step 34.

Task 30 now updates the cache table 27 with the IP address I_T and ATM address A_T (step 35). Next, task 30 requests (step 36) the ATM layer 22 to establish a new SVC to ATM address A_T and this initiates an SVC setup process 28 which may be executed in any appropriate manner and will not be described in detail herein. In due course, process 28 returns the identity of the SVC that has been set up to A_T (in this case, SVC "4"), this identity being received at step 37 of task 30. Finally, cache table 30 is updated at step 38 by adding the SVC identity ("4") to the entry already containing I_T and A_T.

In machine T, the setup of the new SVC to the machine from machine M is handled by the setup process 28 of machine T. The process 28 informs the IP/ATM layer that a new SVC has been setup and this triggers execution of an update task 40 to update the cache table 27 of machine T. More particularly, on the new SVC indication being received (step 41), a first update step 42 is carried out to add an entry to the table confining the identity of the new SVC (in the present example "9"), and the ATM address A_M of the node at the other end of the SVC; at this stage, the corresponding IP address is not known to machine T. In order to obtain this IP address, an inverse ARP request is now made to machine M (step 43). In due course a response is received (step 44) containing the IP address of machine M. The cache table 27 is then updated at step 45 with the IP address I_M of machine M and the IP/ATM layer is now ready to effect IP-to-SVC translations for datagrams intended for machine M.

The inverse ARP request sent by machine T to machine M is handled by an inverse ARP task 50 that examines the request (step 51) and on finding that it contains the ATM address A_M, responds with the IP address I_M of machine M (step 52).

To facilitate explanation of the preferred embodiment of the invention hereinafter, the messages across the boundary between the IP/ATM layer 21 and the ATM layer 22 have been labelled in Figure 2 as follows where superscript "T" indicates an outgoing message (that is, from the IP/ATM layer to the ATM layer) and the superscript "R" indicates incoming messages (that is, from the ATM layer to the IP/ATM layer):

$X1^T$ -	outgoing ARP request;
$X2^R$ -	incoming ARP response;
$X3^T$ -	outgoing SVC setup request;
$X4^R$ -	incoming SVC setup done indication;
$X5^R$ -	incoming new SVC indication;
$X6^T$ -	outgoing INARP request;
$X6^R$ -	incoming INARP request;
$X7^T$ -	outgoing INARP response;
$X8^T$ -	outgoing datagram;
$X8^R$ -	incoming datagram.

It will be appreciated that machines connecting to an ATM network, such as machines M and T as well as the server S, are designed to handle a large number of virtual circuits simultanteously. If in testing such a machine (machine M in the following discussion) it is desired to fully stress the machine under test, then the design limit of concurrently operating virtual circuits must be simultaneously used. However, as already indicated, current practice is that only one virtual circuit is established to each distinct IP address. As a result, since generally each machine that might be used to test machine M has only one network connection and therefore only one IP address, if machine M is designed to operate up to N virtual circuits simultaneously, then it requires N machines to test machine M. Such an arrangement is illustrated in Figure 3 where the N machines are constituted by the server S and (N-1) other machines here represented as machines T1 to T(N-1). Such an arrangement is generally impractical as N may be as high as 1024 or more.

It is an object of the present invention to provide a mechanism that enables, inter alia, the foregoing test problem to be overcome.

Summary of the Invention

According to the present invention, there is provided a system in which a plurality of entities are connected to a network ad can exchange messages across virtual circuits set up over the network between said entities, each entity having a operative high-level address on the network, and each entity comprising:

-- high-level messaging means for handling message transmission and receipt on the basis of the aforesaid high-level addresses, the high-level messaging means comprising means for including in outgoing messages the operative high-level address of the entity as a source identifier and the operative high level address of the intended recipient entity as a destination identifier, and means for filtering incoming messages according to the destination identifier contained in the message:
-- virtual-circuit means for providing virtual circuits between the entity and other entities, there being a respective virtual circuit for each different destination identifier in use, and
-- intermediate means for passing an outgoing message from the high-level messaging means to that one of the virtual circuits provided by the virtual-circuit means which corresponds to the destination identifier of the message;

characterised in that each of a first and a second one of the entities has a plurality of virtual high-level addresses associated with it that are different from the operative high-level address of the entity, the virtual high-level addresses being usable by the messaging means of the first and second entities as destination identifiers in outgoing messages; **and in that** between the intermediate means of the first and second entities, there are provided address-changing means responsive to each of at least some of the messages sent between these entities with a said virtual high-level address as its destination identifier, to change that address to the operative high-level address of the corresponding entity and to change the operative high-level address provided as the source identifier of the message into one of the said virtual high-level addresses associated with the sending entity in dependence on the virtual high-level address initially provided as the destination identifier of the same message.

By virtue of this arrangement, it is possible to establish a plurality of virtual circuits between the first and second entities by using the different virtual high-level addresses of the entities as the destination identifiers in messages exchanged between the entities, the receiving high-level adressing means accepting such messages due to the address-changing means having changed the destination identifier to the operative high-level address of the receiving entity. By also changing the source identifier, it is possible to retain in the message information sufficient to associate any reply message with a particular one of the virtual circuits established with the sending entity (in particular, the reply message can be sent back over the same virtual circuit as the message to which it is a reply - however, if desired, it is also possible to use a separate virtual circuit for the reply messages).

Preferably, the address-changing means comprises first address-changing functionality for effecting the aforesaid changes for messages sent from the first entity to the second entity, and second address-changing functionality for effecting these changes for messages sent from the second entity to the first entity, both the first and second address-changing functionalities being provided in the second entity. This configuration is well suited for testing the ability of network node apparatus to concurrently operate a plurality of virtual circuits where the network node apparatus is operative to establish a virtual circuit for each different high-level destination address being handled; more particularly, the network node apparatus serves as the aforesaid first entity, and is caused to send messages to at least some of the vial high-level addresses associated with the second entity. By placing the address-changing means in the second entity, no modifications are needed to the network node apparatus in order for it to be able to establish a plularity of virtual circuits with the second entity.

Advantageously, the address-changing means effects a predetermined transformation on the virtual high-level address forming the initial destination identifier of a said message in order to form the virtual high-level address to be used for the source identifier of that message. For example, this transformation may simply involved changing the address by one (where the address is numeric in form).

The present invention is particularly applicable to systems in which the high-level addresses are IP addresses and the network is an ATM network.

Brief Description of the Drawings

A system embodying the invention will now be described, by way of non-limiting example, with reference to the accompanying diagrammatic drawings, in which:

. **Figure 1** is a diagram of a known system for sending IP datagrams over a ATM network between two machines M and T;

. **Figure 2** is a diagram illustrating the steps carried out by the Figure 1 system in establishing a virtual circuit between machines M and T;

. **Figure 3** is a diagram of a known test arrangement for testing the ability of a machine M to concurrently operate multiple virtual circuits;

. **Figure 4** is a diagram showing a test arrangement embodying the invention for testing the ability of a machine M to concurrently operate multiple virtual circuits;

. **Figure 5** is a diagram similar to Figure 1 but showing a system embodying the invention in which multiple virtual circuits are established between machines M and T;

. **Figure 6** is a diagram illustrating the processing effected by a module VNS disposed in machine T of the Figure 5 system when machine M initiates the opening of a new virtual circuit between machines M and T; and

. **Figure 7** is a diagram illustrating the processing effected by a module VNS disposed in machine T of the Figure 5 system when machine T initiates the opening of a new virtual circuit between machines M and T.

Best Mode of Carrying Out the Invention

The embodiment of the invention now to be described provides a system in which it is possible to establish a plurality of SVCs (switched virtual circuits) across an ATM network for the exchange of IP datagrams between two machines M and T whereby it is possible to test the ability of machine M to concurrently operate a plurality of virtual circuits without needing to provide a respective destination machine for each SVC operated by machine M. The overall test arrangement is illustrated in Figure 4 where machine M operates N SVCs over ATM network 10, one SVC being with ATMARP server S and (N-1) SVCs being with machine T. According to the preferred embodiment, the establishment of multiple concurrent SVCs between machine M and T is effected without modification to machine M.

Figure 5 shows a system embodying the present invention, this system being similar to that of Figure 1 but being operative to provide a plurality of concurrent SVCs 65 between machines M and T. In the Figure 5 system, the machines M and T and the server S are assumed to operate in the same way and have the same IP and ATM addresses as in Figure 1; in addition, in Figure 5 the same SVCs are established between the server S and the machines M and T as in Figure 1. The Figure 5 system includes, however, added functionality provided by processes 70 and 71 which in Figure 5 are shown independent of machines M and T but in practice would be provided either distributed between machines M and T or wholly in one of these machines; in a preferred embodiment, the processes 70 and 71 are provided in machine T.

In accordance with the present invention, each machine M and T is allocated a number of virtual IP addresses different from its operative (or "real") IP address (this latter address being the one which the IP layer knows about for inclusion as the source address in outgoing datagrams and upon which filtering is carried out by task 29). Thus, machine M is allocated virtual IP addresses $I_{M(1)}$, $I_{M(2)}$,......$I_{M(i)}$...; similarly, machine T is allocated virtual IP addresses $I_{T(1)}$, $I_{T(2)}$,......$I_{T(i)}$....

Each of these virtual IP addresses is entered into table 15 of ATMARP server S together with the ATM address of the corresponding one of the machines M,T; thus virtual IP address $I_{M(i)}$ is associated with ATM address A_M and virtual IP address $I_{T(i)}$ is associated with ATM address A_T.

Now, if the communications interface 18 of machine M is asked to send a message to IP address $I_{T(i)}$, IP layer 20 will construct a datagram 25A having a destination address of $I_{T(i)}$ and a source address of I_M. The IP-to-SVC task 30 of IP/ATM layer 21 then acts in the manner already described to fetch the ATM address corresponding to $I_{T(i)}$ from server S and set up an SVC (here identified by "p") towards machine T; the cache table 27 is updated appropriately. The datagram 25A is now sent by ATM layer over SVC(p) to machine T.

If no further action is taken, the datagram 25A, after receipt at machine T, will be rejected by the filter task 29 as the destination address $I_{T(i)}$ of the datagram differs from the operative IP address I_T known to task 29 of machine T. Accordingly, a process 70 is provided that recognises the destination address of datagram 25A as being a virtual IP address of machine T and substitutes the real IP address of machine T for the virtual address in the destination field of the datagram 25A. The datagram will now be allowed through by filter task 29 of machine T.

However, a further difficultly remains. If only the destination address is changed, the resultant datagram contains no indication that the datagram was not ordi-