

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第5090472号
(P5090472)

(45) 発行日 平成24年12月5日(2012.12.5)

(24) 登録日 平成24年9月21日(2012.9.21)

(51) Int.Cl. F I
H O 4 L 12/66 (2006.01) H O 4 L 12/66 A

請求項の数 20 (全 22 頁)

(21) 出願番号	特願2009-551789 (P2009-551789)	(73) 特許権者	500046438
(86) (22) 出願日	平成20年2月21日(2008.2.21)		マイクロソフト コーポレーション
(65) 公表番号	特表2010-520668 (P2010-520668A)		アメリカ合衆国 ワシントン州 9805
(43) 公表日	平成22年6月10日(2010.6.10)		2-6399 レッドモンド ワン マイ
(86) 国際出願番号	PCT/US2008/054485		クロソフト ウェイ
(87) 国際公開番号	W02008/106355	(74) 代理人	100140109
(87) 国際公開日	平成20年9月4日(2008.9.4)		弁理士 小野 新次郎
審査請求日	平成23年2月21日(2011.2.21)	(74) 代理人	100089705
(31) 優先権主張番号	11/712, 123		弁理士 社本 一夫
(32) 優先日	平成19年2月28日(2007.2.28)	(74) 代理人	100075270
(33) 優先権主張国	米国 (US)		弁理士 小林 泰
		(74) 代理人	100080137
			弁理士 千葉 昭男
		(74) 代理人	100096013
			弁理士 富田 博行

最終頁に続く

(54) 【発明の名称】 ネットワーク・アドレス変換システムと関連のあるクライアントを帯域外で存続状態に維持するメカニズム

(57) 【特許請求の範囲】

【請求項1】

接続管理を容易にするコンピューター実装システムであって、
不活性タイマを有するネットワーク・デバイスを含む接続コンポーネントであって、第1のコンピュータ・システムにおける第1のソフトウェア・エンティティと第2のコンピュータ・システムにおける第2のソフトウェア・エンティティとの間に通信接続を提供し、前記第1のソフトウェア・エンティティと前記第2のソフトウェア・エンティティとの間の前記通信接続を経由して伝達された帯域内(イン・バンド)パケットを前記通信接続に対して前記ネットワーク・デバイスによって始動された不活性タイマのタイムアウト期間の間に前記ネットワーク・デバイスが受信しない場合には前記通信接続を遮断するように構成されている接続コンポーネントと、

前記第1のソフトウェア・エンティティ及び前記第2のソフトウェア・エンティティとは異なるキープ・アライブ・コンポーネントであって、前記ネットワーク・デバイスによって前記第1のソフトウェア・エンティティと前記第2のソフトウェア・エンティティとの間の前記通信接続を経由する帯域内トラフィックとして認識される帯域外(アウト・オブ・バンド、OOB)なりすましパケットを送信し、前記不活性タイマの前記タイムアウト期間をリセットして前記第1のソフトウェア・エンティティと第2のソフトウェア・エンティティとの間の通信接続を維持するキープ・アライブ・コンポーネントと、

を備えており、前記第1のソフトウェア・エンティティから前記第2のソフトウェア・エンティティまでの前記通信接続を経由して伝達されるように見えるOOBなりすましパ

ケットは、前記第1のソフトウェア・エンティティと前記第2のソフトウェア・エンティティとの間の前記通信接続に前記キープ・アライブ・コンポーネントによって挿入され、前記ネットワーク・デバイスによって処理されて前記不活性タイムの前記タイムアウト期間をリセットし、前記ネットワーク・デバイスによって処理された後に前記第2のソフトウェア・エンティティに到達するまでの間に前記キープ・アライブ・コンポーネントによって前記通信接続から取り除かれ、

前記キープ・アライブ・コンポーネントは、前記通信接続をこれ以上望まないという指示を前記第1のソフトウェア・エンティティと前記第2のソフトウェア・エンティティとの少なくとも一方から受け取るまで前記タイムアウト期間の終了までの間O O Bなりすましパケットを前記通信接続に挿入することによって、前記第1のソフトウェア・エンティティと前記第2のソフトウェア・エンティティとの間の前記通信接続を維持する、コンピュータ実装システム。

【請求項2】

請求項1記載のシステムにおいて、前記任意の通信接続はトランスポート・レイヤー接続指向エンド・ツー・エンド・プロトコルに対応する、システム。

【請求項3】

請求項1記載のシステムにおいて、前記任意の通信接続はトランスポート・レイヤー無接続プロトコルに対応する、システム。

【請求項4】

請求項1記載のシステムにおいて、前記接続コンポーネントは個人ネットワークの公衆ネットワークとの接続を容易にするネットワーク・アドレス変換(NAT)デバイスを含む、システム。

【請求項5】

請求項1記載のシステムにおいて、前記O O Bなりすましパケットはキープ・アライブ・パケットである、システム。

【請求項6】

請求項1記載のシステムにおいて、前記キープ・アライブ・コンポーネントはTCP/IPシステム・テーブルのテーブル活動に基づいて前記O O Bなりすましパケットを挿入するアプリケーションである、システム。

【請求項7】

請求項1記載のシステムにおいて、前記キープ・アライブ・コンポーネントは、前記不活性タイムの前記タイムアウト期間を監視し、前記タイムアウト期間に基づいてキープ・アライブ・リフレッシュ期間を用い、前記リフレッシュ期間に基づいて前記O O Bなりすましパケットを発生し前記通信接続に挿入する、システム。

【請求項8】

請求項1記載のシステムにおいて、前記第1のコンピュータ・システムはクライアントとサーバーとの一方を含む、システム。

【請求項9】

請求項8記載のシステムにおいて、前記キープ・アライブ・コンポーネントはサーバー系キープ・アライブ・アプリケーションとクライアント系キープ・アライブ・アプリケーションとを備えており、前記サーバー系およびクライアント系キープ・アライブ・アプリケーションは前記接続コンポーネントを通じてO O Bなりすましパケットを送り前記接続を維持する、システム。

【請求項10】

請求項1記載のシステムにおいて、前記接続コンポーネントによって処理された前記O O Bなりすましパケットはゼロ・ペイロード・パケットである、システム。

【請求項11】

接続を管理するコンピュータ実装方法であって、

第1の計算システムの第1のネイティブ・アプリケーションと第2の計算システムの第2のネイティブ・アプリケーションとの間で不活性タイムを有するネットワーク・アドレ

10

20

30

40

50

ス変換（NAT）デバイスを経由して帯域内パケットを伝達するNAT接続を確立するステップであって、前記第1のネイティブ・アプリケーションと前記第2のネイティブ・アプリケーションとの間を前記NAT接続を経由して伝達された帯域内パケットを前記NAT接続に対して前記NATデバイスによって始動された前記不活性タイマのタイムアウト期間の間に前記NATデバイスが受信しない場合には前記NAT接続が前記NATデバイスによって遮断される、ステップと、

前記第1のネイティブ・アプリケーション及び第2のネイティブ・アプリケーションとは異なっており、前記第1のネイティブ・アプリケーションと前記第2のネイティブ・アプリケーションとの間の前記NAT接続を経由する帯域内トラフィックとして前記NATデバイスに認識されるOOBなりすましパケットを自動的に前記NATデバイスに送り、前記不活性タイマの前記タイムアウト期間をリセットして前記第1のネイティブ・アプリケーションと第2のネイティブ・アプリケーションとの間のNAT接続を維持するように構成されたキープ・アライブ・アプリケーションを起動するステップと、

10

を含んでおり、前記第1のネイティブ・アプリケーションから前記第2のネイティブ・アプリケーションまで前記NAT接続を経由して伝達されるように見えるOOBなりすましパケットは、前記第1のネイティブ・アプリケーションと前記NATデバイスとの間の前記NAT接続に前記キープ・アライブ・アプリケーションによって挿入され、前記NATデバイスによって処理されて前記不活性タイマの前記タイムアウト期間をリセットし、前記NATデバイスによって処理された後に前記第2のソフトウェア・エンティティに到達するまでの間に前記キープ・アライブ・アプリケーションによって前記NAT接続から取り除かれ、

20

前記キープ・アライブ・アプリケーションは、前記NAT接続をこれ以上望まないという指示を前記第1のネイティブ・アプリケーションと前記第2のネイティブ・アプリケーションとの少なくとも一方から受け取るまで前記タイムアウト期間の終了までの間OOBなりすましパケットを前記NAT接続に挿入することによって、前記第1のネイティブ・アプリケーションと前記第2のネイティブ・アプリケーションとの間の前記NAT接続を維持する、コンピューター実装方法。

【請求項12】

請求項11記載の方法において、所定の接続方針に基づいて前記OOBなりすましパケットを前記接続に自動的に挿入するステップを更に備えている、方法。

30

【請求項13】

請求項11記載の方法において、送信制御プロトコル（TCP）通信トランスポート技術によってNAT接続が容易化されている、方法。

【請求項14】

請求項11記載の方法において、前記第1のネイティブ・アプリケーションと前記第2のネイティブ・アプリケーションとに対して対応するキープ・アライブ・アプリケーションを起動するステップを更に備えており、前記対応するキープ・アライブ・アプリケーションはOOBなりすましパケットを自動的に送信し、TCP/IPシステム・テーブルにおける新たなエントリーに基づいて前記接続を維持する、方法。

【請求項15】

40

請求項14記載の方法において、前記対応するキープ・アライブ・アプリケーションは、対応するシステム・テーブルを監視して接続状態を調べ、複数のNAT接続に関して接続状態を処理する、方法。

【請求項16】

請求項11記載の方法において、前記ネイティブ・アプリケーションの中の少なくとも1つと関連するプロトコル番号、発信元IPアドレス、発信元ポート番号、宛先IPアドレス、または宛先ポート番号の中の少なくとも2つを含むタプルを発見するステップと、

前記タプルを用いるOOBなりすましパケットに基づいて前記不活性タイマをリセットするステップと、

50

を更に含む、方法。

【請求項 17】

請求項 11 記載の方法において、前記キープ・アライブ・アプリケーションによって前記 NAT デバイスと前記第 2 の計算システムとの間の接続から取り除かれた前記 OOB なりすましパケットは前記第 2 のネイティブ・アプリケーションと関連する宛先 IP アドレスを用いる、方法。

【請求項 18】

請求項 11 記載の方法において、前記第 1 及び第 2 のネイティブ・アプリケーションの少なくとも一方の帯域内活動に基づいてキープ・アライブ・アプリケーションを自動的に起動するステップを更に含む、方法。

10

【請求項 19】

請求項 11 記載の方法において、1 又は複数のキープ・アライブ・アプリケーションを自動的に起動し、用いられるトランスポート・プロトコルの種類に基づいて前記 1 又は複数のキープ・アライブ・アプリケーションを通じて前記 OOB なりすましパケットを挿入するステップを更に含む、方法。

【請求項 20】

処理装置とメモリとを含むコンピュータ実装システムであって、

第 1 のコンピュータ・システムの第 1 のネイティブ・アプリケーションと第 2 のコンピュータ・システムの第 2 のネイティブ・アプリケーションとの間で不活性タイマを有する NAT デバイスを經由して帯域内パケットを伝達する NAT 接続を自動的に確立するコンピュータ実装手段であって、前記 NAT 接続は、前記第 1 のネイティブ・アプリケーションと前記第 2 のネイティブ・アプリケーションとの間を前記 NAT 接続を經由して伝達された帯域内パケットを前記 NAT 接続に対して前記 NAT デバイスによって始動された前記不活性タイマのタイムアウト期間の間に前記 NAT デバイスが受信しない場合には、前記 NAT デバイスによって遮断される、コンピュータ実装手段と、

20

前記第 1 のネイティブ・アプリケーション及び第 2 のネイティブ・アプリケーションと異なる 1 又は複数のキープ・アライブ・アプリケーションを自動的に起動するコンピュータ実装手段と、

システム・テーブルに基づいて前記 NAT 接続を監視して不活性を調べるコンピュータ実装手段と、

30

不活性を感知した場合には前記第 1 のネイティブ・アプリケーションと前記第 2 のネイティブ・アプリケーションとの間の前記 NAT 接続を經由する帯域内トラフィックとして前記 NAT デバイスに認識される OOB なりすましパケットを前記 NAT デバイスに自動的に送り、前記不活性タイマの前記タイムアウト期間をリセットして前記第 1 のネイティブ・アプリケーションと第 2 のネイティブ・アプリケーションとの間の NAT 接続を維持するコンピュータ実装手段と、

を備えており、前記第 1 のネイティブ・アプリケーションから前記第 2 のネイティブ・アプリケーションまで前記 NAT 接続を經由して伝達されるように見える OOB なりすましパケットは、TCP パケットと UDP パケットとの一方に基づいて前記第 1 のネイティブ・アプリケーションと前記 NAT デバイスとの間の前記 NAT 接続にキープ・アライブ・アプリケーションによって挿入され、前記 NAT デバイスによって処理されて前記不活性タイマの前記タイムアウト期間をリセットし、前記 NAT デバイスによって処理された後に前記第 2 のソフトウェア・エンティティに到達するまでの間にキープ・アライブ・アプリケーションによって前記 NAT 接続から取り除かれ、

40

前記キープ・アライブ・アプリケーションは、前記 NAT 接続をこれ以上望まないという指示を前記第 1 のネイティブ・アプリケーションと前記第 2 のネイティブ・アプリケーションとの少なくとも一方から受け取るまで前記タイムアウト期間の終了までの間 OOB なりすましパケットを前記 NAT 接続に挿入することによって、前記第 1 のネイティブ・アプリケーションと前記第 2 のネイティブ・アプリケーションとの間の前記 NAT 接続を維持する、コンピュータ実装システム。

50

【発明の詳細な説明】**【背景技術】****【0001】**

コンピューティング・デバイスおよびネットワークにおける技術進歩により、多種多様の情報およびサービスへのアクセスが容易となり、事実上世界のどこからでもアクセスが可能となっている。その上、コンピューターや携帯デバイスの数が増大し続けるため、各デバイスがネットワーク上にあるときに、一意に識別することが接続(connectivity)に必要となることもあり得る。加入してネットワーク・デバイス毎に別個の(または固定)IPアドレスを取得するための余分な費用をかけるのではなく、ネットワーク・アドレス変換(NAT)と呼ばれる技法によって、ルータ背後にある、または内部(または個人)ネットワーク上にある多数のIPノードが、1つの公開IPアドレスを共有することが可能になっている。言い換えると、1組の未登録IPアドレスを内部ネットワーク・トラフィックに使用させ、別の1組のIPアドレスを外部トラフィック即ち公開トラフィックに使用させる規格が設けられている。

10

【0002】

通例、NATデバイスは、接続タイムアウト・タイマを採用する。接続タイムアウト・タイマは、ネイティブ・アプリケーション間において接続状態をマッピングするために、設定変更可能なタイムアウト期間を有する。テーブル・エントリーにマッピングする特定のNATポートが、タイムアウト期間よりも長い間インバウンド・トラフィックにもアウトバウンド・トラフィックにも用いられないと、その接続に対するNATタイマが時間切れとなり、エントリーはテーブルから抹消される。一旦エントリーが抹消されると、NATの背後にある共有ノードにはもはやこの接続を通じて到達することができなくなり、新たな接続を開始しなければならない(例えば、共有ノードによって)。

20

【0003】

NATタイマが終了する(即ち、時間切れとなる)のを防止する慣例的なメカニズムに、「キープ・アライブ」(KA:keep-alive)または「ハートビート」処理として知られているものがある。キープ・アライブの下では、タイマをリセットする(またはリフレッシュする)ためのNATタイムアウト期間よりも短い間隔で、接続上に無駄なトラフィックを発生させ、これによって接続をアクティブに維持する。バッテリー電力を主要電源として用いる携帯デバイス(例えば、スマート・フォン)の場合、従来のキープ・アライブ技法では電池の寿命に影響を及ぼし、接続を生かしておくために多大なワイヤレス活動が発生する。

30

【発明の概要】**【発明が解決しようとする課題】****【0004】**

NATを通じて長寿命の接続を提供するための解決策の1つは、ネイティブ・アプリケーション・プロトコルの一部としてキープ・アライブ・メカニズムを構築することである(帯域内解決策(in-band solution))。しかしながら、従来のメカニズムに対する欠点は、次のことを含む。KAメカニズムを受け入れるためには、基礎となるネイティブ・アプリケーション・プロトコルを修正しなければならない。KAメカニズムを旧来のアプリケーションに合わせて後付けすることができず、アプリケーションのアップグレードを展開しなければならない。そして、KAメカニズムに対するあらゆる更新がコア・アプリケーション・プロトコルに影響を及ぼすため、それに合わせて検査しなければならない。

40

【0005】

加えて、従来のKAメカニズムを最適化することは、帯域内制約のために困難である。帯域内制約には次のことが含まれる可能性がある。層状ネイティブ・アプリケーション・ヘッダ(例えば、大型HTTP-ハイパーテキスト・マークアップ言語およびSOAP-単純オブジェクト・アクセス・プロトコル・ヘッダ)を受け入れるために、KAパケット・サイズが不必要に増大せざるを得ない。アプリケーション・ロジックがKAロジックを束縛し、余分なネットワーク接続が必要となる可能性がある。アプリケーション・レベル

50

では、高速障害モード検出や復元をサポートできない場合がある。アプリケーション開発者は、基本的にシステム・レベルの解決策であるものを完成させるために確保すべき必要な資源、時間、または費用を賄うことができない場合がある。移動体のローミングや異なるネットワーキング環境に適応することおよび応答することが難しい。

【 0 0 0 6 】

以下に提案するのは、本明細書に記載する新規な実施形態の基本的な理解が得られるようにするための、簡略化した摘要である。この摘要は、広範囲に及び全体像ではなく、鍵となる要素/肝要な要素を特定することや、その範囲を正確に叙述することを意図するのではない。その唯一の目的は、一部の概念を、後に提案する更に詳細な説明の序文として、簡略化した形態で提案することである。

10

【 課題を解決するための手段 】

【 0 0 0 7 】

開示するアーキテクチャは、基礎となるネイティブ・アプリケーションおよび/またはアプリケーション・プロトコルには要件や修正を全く強いることなく、ネイティブ・アプリケーション接続の外部から適用することができる帯域外(OOB)技法を採用することによって、ネットワーク・アドレス変換(NAT)デバイスの接続状態を存続したまま維持する解決策を提供する。この技法は、OOBの様式でネイティブ・アプリケーションと動作する、別個のキープ・アライブ(KA)アプリケーションを設け、接続を維持するためにネイティブ・アプリケーション接続にKAパケットを注入し、その後KAパケットがネイティブ・アプリケーションに到達する前にKAパケットを除去する(例えば、欠落させる)。代替実施態様では、KAパケットを除去せずに、ネイティブ・アプリケーションによって処理する(例えば、フィルタ処理、欠落、...)。

20

【 0 0 0 8 】

本アーキテクチャは、一般的なNAT状態管理メカニズムにおける「論理ホール」(logical hole)を利用する。これらの論理ホールは「バグ」ではなく、したがってセキュリティホールとは見なされず、単にインライン・ネットワーキング・デバイスに課される制約の人為構造である。1つの論理ホールの性質は、接続活動に基づいてNAT状態を維持することに関する。NATデバイスは、アクティブな接続毎にタイムアウト・タイマを始動させ、接続にトラフィック(例えば、パケット活動)が生じる毎に、接続のタイムアウト(即ち、不活性)タイマをリセットする。所与の接続においてトラフィックが検出されない場合、タイムアウト・タイマはその接続に対して終了し、接続は作用しなくなる。更に、NATは通例トラフィックの正当性(例えば、トラフィックが来た終点)を検証しない。

30

【 0 0 0 9 】

この論理ホールを利用し、更にクライアント/サーバー環境の一例では、クライアントおよび/またはサーバー上で走る第2ネットワークKAアプリケーションをOOBメカニズムとして採用し、なりすましパケットを注入することができる。なりすましパケットは、ネイティブ接続から来ているように見える。これらの注入されたパケットは、NATデバイスを騙して、その接続に対する不活性タイマをリセットさせるが、ネイティブ・アプリケーションを騙したり混乱させることはなく、ネイティブ・アプリケーションはなりすましに気付かないことができる。このために、NATタイムアウトのために接続が終了することはなく、したがって、クライアント/サーバー・プロトコルは、接続を生かしておくために、偽の活動パケットを発生する必要がない。

40

【 0 0 1 0 】

本アーキテクチャは、NATデバイスを通じて、任意の不活性ネットワーク接続に永続性を与え、TCP(送信制御プロトコル)のような接続指向エンド・ツー・エンド・トランスポート・プロトコルおよびUDP(ユーザ・データグラム・プロトコル)のような無接続トランスポート・プロトコルと動作する。

【 0 0 1 1 】

前述の目的および関連する目的の遂行のために、開示する新規なアーキテクチャのある

50

種の例示的態様について、本明細書では以下の説明および添付図面と関連付けて記載する。しかしながら、これらの態様は、本明細書に開示する原理を採用することができる種々の方法の内の数個に過ぎず、そのような態様の全ておよびその均等物を含むことを意図している。他の利点や新規な特徴も、以下の詳細な説明を図面と合わせて考察することにより、明白となろう。

【図面の簡単な説明】

【0012】

【図1】図1は、一実施形態にしたがって接続管理を容易にするコンピューター実装システムを示す。

【図2】図2は、接続のネイティブ・アプリケーション毎にキープ・アライブ（KA）アプリケーションを採用するシステムを示す。

【図3】図3は、帯域外（OOB）KAパケットを用いて接続を維持するクライアント/サーバー・システムを示す。

【図4】図4は、多数の接続に対する接続状態を処理するためにKAアプリケーションが動作する代替システムを示す。

【図5】図5は、OOB KA管理方法を示す。

【図6】図6は、接続方針に基づく接続管理方法を示す。

【図7】図7は、トランスポート・プロトコルに基づくNAT接続不活性を管理する方法を示す。

【図8】図8は、接続管理のためにKAパケットを発生し利用する方法を示す。

【図9】図9は、開示したKAアーキテクチャを実行するように動作可能なコンピューティングシステムのブロック図を示す。

【図10】図10は、OOB KA処理を採用することができるコンピューティング環境の一例の模式ブロック図を示す。

【発明を実施するための形態】

【0013】

開示するアーキテクチャは、ネイティブ・アプリケーションに何の要件も修正も賦課することなく、外部からネイティブ・アプリケーションに適用することができる帯域外（OOB）技法を採用することによって、ネットワーク・アドレス変換（NAT）デバイスおよび/またはソフトウェアの接続を存続させるための解決策を提供する。このアーキテクチャは、一般的なNAT状態管理メカニズムにおいて「論理ホール」(logical hole)を利用して、OOB発信元（例えば、アプリケーション）からなりすまし（または、キープ・アライブ（KA））パケットを注入する。OOB発信元は、NATデバイスの観点からは、ネイティブ接続の一部であるように見える。これらの注入パケットは、NATデバイスに、その接続に対する不活性タイマをリセットさせるが、ネイティブ・アプリケーションを騙したり混乱させることはなく、ネイティブ・アプリケーションはなりすましに気が付かないことができる。したがって、接続（例えば、TCP送信制御プロトコルまたはUDP-ユーザ・データグラム・プロトコルに基づく）がNATタイムアウトによって終了することはなく、したがって、クライアント/サーバー・プロトコルは、例えば、接続を存続させておくために偽の帯域内KAパケットを発生しなくてもよい。

【0014】

これより図面を参照するが、全体を通じて同様の要素を指すために同様の参照番号を用いることとする。以下の記載では、説明の目的上、完全な理解が得られるようにするために、多数の具体的な詳細を明記する。しかしながら、新規な実施形態はこれら具体的な詳細がなくても実用可能であることは明白であろう。一方、周知の構造やデバイスは、その説明を容易にするために、ブロック図形態で示すこととする。

【0015】

最初に図面を参照すると、図1は、一実施形態による接続管理を容易にするコンピューター実装システム100を示す。システム100は、ソフトウェア・エンティティ104（ソフトウェア・エンティティ₁およびソフトウェア・エンティティ₂で示す）間に任意

10

20

30

40

50

の通信接続を提供する接続コンポーネント102(例えば、NATデバイス)を含む。また、システム100は、OOB KAパケット(または信号)を送るために接続コンポーネント102にインターフェースするキープ・アライブ・コンポーネント106も含む。OOB KAパケットは、接続を維持するために、接続コンポーネント102によって処理される。

【0016】

一実施態様では、KAコンポーネント106は、ソフトウェア・エンティティ104が位置するコンピューティングシステム上におけるTCP/IPスタック・システム・テーブルの活動(activity)を監視する。例えば、新たなテーブル・エントリは、KAパケットを挿入することができる新たな接続を示す。TCP/IPテーブルは、ソフトウェア・エンティティが位置するシステムにおいて監視することができる。例えば、クライアント・システム・テーブルを監視することができる。同様に、サーバーを伴う場合、サーバー・システムTCP/IPテーブルへの入力活動(例えば、除去または新たなエントリ)を監視することができる。

10

【0017】

代替的かつ任意の実施態様では、ソフトウェア・エンティティ104の一方または双方の接続パケット活動を、KAコンポーネント106が監視することができ、その活動に基づいて、KAパケットを接続に挿入する。KAパケットは、KAコンポーネント106によって、外部からまたは接続コンポーネント102のいずれかの側または両側から(例えば、その接続を通じてパケットを導出する他のネットワーク・デバイスによって)、接続コンポーネント102の対応する接続に挿入し、挿入されたKAパケットが接続コンポーネント102によって通常の帯域内トラフィックとして認知されるようにすることができる。次いで、KAパケットは、接続コンポーネント102によって検出された各パケットの発生時に(帯域内および/またはキープ・アライブ)その接続に対する接続タイマをリセットする。

20

【0018】

システム・テーブルは、KAコンポーネント106によって直接監視すること、および/または接続コンポーネント102を通じて間接的に監視することができる。例えば、テーブル活動が、接続を維持すべきことを示す場合、KAコンポーネント106はKAパケットをその接続に注入して、接続コンポーネント102がKAパケットを処理して、接続タイマをリセットし、接続を維持する。

30

【0019】

代替的かつ任意の実施態様では、ソフトウェア・エンティティ104の1つ以上が監視した接続パケット活動(またはそれがないこと)をKAコンポーネント106に伝達し、KAコンポーネント106にKAパケットを接続コンポーネント102に送らせることができる。次いで、接続コンポーネント102は、自己処理およびタイムアウト・タイマのリセットのために、KAパケットを接続に挿入する。更に別の例では、KAコンポーネント106によって直接監視した接続パケット不活性に基づいて、そしてそれでもなお接続を存続させておきたいという要望に基づいて、KAコンポーネント106は接続を通じて帯域内KAパケットを発生し、その接続に対するタイムアウト・タイマをリセットするように、ソフトウェア・エンティティ104の一方または双方に通知することができる。ソフトウェア・エンティティ104がKAアプリケーションである場合、KAコンポーネント106は、接続を維持するために、OOBパケットを接続に挿入するように、ソフトウェア・エンティティ104のいずれかまたは双方に通知することができる。尚、テーブルにおける不活性や帯域内パケット・トラフィックには関係なく、KAパケットを周期的に送ることができることは言うまでもない。

40

【0020】

システム100の一実施態様は、NATデバイスを接続コンポーネント102の一部として含み、ソフトウェア・エンティティ104がNATデバイスを経由して互いに通信するようになっている。ソフトウェア・エンティティ104は(ネイティブ・アプリケーシ

50

ョンとして)、NATを経由して接続を通じて違いに通信することにより、帯域内(ネーティブ・アプリケーション間のみ)パケットの通常伝達によって、アクティブ接続を作成する。NATデバイスは、タイムアウト・タイマを有し、ネーティブ・アプリケーション104からの帯域内パケットの受信に基づいて、継続的にタイマをリセットする。

【0021】

しかしながら、これまでは、タイムアウト期間以内に帯域内パケットがNATデバイスによって受信されないと、NATは接続を打ち切り、アプリケーションはNATデバイスを通じて接続を確立し直す必要があった。本アーキテクチャはこの問題を解決するにあたり、少なくとも1つのKAアプリケーション(例えば、KAコンポーネント106の一部として)を備え、これが1つ以上のネーティブ・アプリケーション104と共に起動して、KAアプリケーションがKAパケットを発生して接続に挿入し、NATデバイスに自動的にタイムアウト・タイマをリセットさせるようにすることにより、接続を維持する。テーブル・エントリがシステム・テーブルから除去されると、KAパケットはもはや接続には挿入されない。

【0022】

一実施態様では、KAコンポーネント106は、オペレーティング・システムのTCP/IPテーブルにおいて新たなエントリを検知したことに応答して、KAアプリケーションを起動する。つまり、NATデバイスを通じて動作する多数の異なる接続を管理することができる。別の実施態様では、ネーティブ・アプリケーション(またはソフトウェア・エンティティ104)毎に、1つのKAアプリケーションを起動する。ここでは、接続を切断したままにするべきであると判断するまで、KAパケットをしかるべきNAT接続に挿入することによって、ネーティブ・アプリケーションが帯域内パケット通信およびOOBパケット活動に関するKAアプリケーションのキャリアを処理する。これを図2において更に詳細に説明する。

【0023】

図2は、接続のネーティブ・アプリケーション毎にKAアプリケーションを採用するシステム200を示す。ここでは、KAコンポーネント106は2つのKAアプリケーション202、即ち、第1KAアプリケーション204(KA APP₁で示す)および第2KAアプリケーション(KA APP₂で示す)を備えている。第1KAアプリケーション204および/または第2KAアプリケーション206の一方または双方は、システム・テーブル・エントリの活動を監視する。新たなテーブル・エントリを検出すると(例えば、TCP/IPシステム・テーブルにおいて)、KAアプリケーション(204または/および206)の一方または双方は、KAパケットを接続に挿入し始める。

【0024】

代替的かつ任意の実施形態では、接続維持がテーブル活動ではなくパケット・トラフィックに基づく場合、第1KAアプリケーション204は、接続コンポーネント102(例えば、NATデバイス)と第1ネーティブ・アプリケーション(ネーティブAPP₁で示す)との間におけるネーティブ・パケット・トラフィックを監視し、および/または第2KAアプリケーション206は、接続コンポーネント102と第2ネーティブ・アプリケーション210(ネーティブAPP₂)との間におけるネーティブ・パケット・トラフィックを監視する。本明細書において記載する場合、ネーティブ・アプリケーション(208および210)は帯域内で通信すると言い、KAアプリケーション(204および206)はOOBで通信すると言う。第1ネーティブ・アプリケーション・システムと関連のあるシステム・テーブルを監視することにより、第1ネーティブ・アプリケーションのネットワーク5-タプル(network 5-tuple)(例えば、プロトコル番号、発信元IPアドレス、発信元ポート、宛先IPアドレス、および宛先ポート)を発見することができる。この情報に基づいて、第1KAアプリケーション204は、接続コンポーネント102のタイムアウトを観察し、これに基づいて所望のKAリフレッシュ期間を観察し採用することができる。KAリフレッシュ期間は、接続コンポーネント102のタイムアウト期間よりも短く、その接続のタイムアウト期間が過ぎ去る前にKAパケットが送られるようになっている。

10

20

30

40

50

例えば、タイムアウト期間が15分であるとする、リフレッシュは10分（または、タイムアウト期間よりも短く適した値であれば、他のいずれでもよい）となるように選択することができる。

【0025】

1つの代替的かつ任意の動作では、リフレッシュ期間に基づいて、第1 K Aアプリケーション204は、接続コンポーネント102の各タイムアウト期間が過ぎ去る前に、K Aパケットを接続に注入する。勿論、これは、第1 ネーティブ・アプリケーション208のパケット活動に基づく。言い換えると、第1 ネーティブ・アプリケーション208が、例えば、第1 ネーティブ・アプリケーション208から第1 K Aアプリケーション204への信号によって、第2 ネーティブ・アプリケーション210との通信がもはや望まれていないことを示した場合、第1 K Aアプリケーション204はK Aパケットを接続に注入させるのを中止する。このため、接続コンポーネント102はここで接続を終了させ、接続は作用しなくなる。

10

【0026】

一実施形態では、第1 K Aアプリケーション204がK Aパケットを送っているとき、第2 K Aアプリケーション206はK Aパケットを除去する。つまり、第2 ネーティブ・アプリケーション210はK Aアプリケーションを不必要に扱う必要がない。

【0027】

同様な任意の動作によれば、システム・デブルの代わりに、第2 ネーティブ・アプリケーション210と接続コンポーネント102との間においてパケット・トラフィックを環視することにより、第2 K Aアプリケーション206は第2 ネーティブ・アプリケーションのネットワーク5 - タプル（例えば、プロトコル番号、発信元IPアドレス、発信元ポート、宛先IPアドレス、および宛先ポート）を発見する。この情報に基づいて、第2 K Aアプリケーション206は、接続コンポーネント102の観察したタイムアウトに基づいて、適したK Aリフレッシュ期間を観察し採用することができる。リフレッシュ期間に基づいて、第2 K Aアプリケーション206は、各タイムアウト期間が過ぎ去る前に、K Aパケットを接続に注入する。勿論、これは第2 ネーティブ・アプリケーション210の活動に基づく。言い換えると、第2 ネーティブ・アプリケーション210が、第1 ネーティブ・アプリケーション208との通信がもはや望まれないことを示した場合、第2 K Aアプリケーション206は、K Aパケットを接続に注入するのを中止する。このため、接続コンポーネント102はここで接続を終了させ、接続は作用しなくなる。

20

30

【0028】

別の実施態様では、第2 K Aアプリケーション206がK Aパケットを送っているとき、第1 K Aアプリケーション204はK Aパケットを除去する。つまり、第1 ネーティブ・アプリケーション208はK Aアプリケーションを不必要に扱う必要がない。

【0029】

別の動作によれば、第1および第2 ネーティブ・アプリケーション（208および210）双方と接続コンポーネント102との間においてパケット・トラフィックを監視することによって、対応する第1および第2 K Aアプリケーション（204および206）はネーティブ・アプリケーションのネットワーク5 - タプルを発見する。この情報に基づいて、第1および第2 K Aアプリケーション（204および206）は、接続コンポーネント102の観察したタイムアウトに基づいて、所望のK Aリフレッシュ期間を観察し採用することができる。リフレッシュ期間に基づいて、第1および/または第2 K Aアプリケーション（204および206）は、各タイムアウト期間が過ぎ去る前に、K Aパケットを接続に注入する。勿論、これは、対応する第1および第2 ネーティブ・アプリケーション（208および210）の活動に基づく。更に別の実施態様では、双方のK Aアプリケーション（204および206）は、N A Tタイムアウトが過ぎ去る前のある時点において接続コンポーネント102の両側がリフレッシュされる限り、独立して動作することができる。

40

【0030】

50

言い換えると、この任意の実施態様によれば、第2ネーティブ・アプリケーション210が、例えば、第2ネーティブ・アプリケーション210から第2KAアプリケーション206への信号によって、第1ネーティブ・アプリケーション208との通信はもはや望まれていないことを示した場合、第2KAアプリケーション206はKAパケットを接続に注入するのを中止する。同様に、第1ネーティブ・アプリケーション208が、第2ネーティブ・アプリケーション210との通信がもはや望まれていないことを示す場合、第1KAアプリケーション204はKAパケットを接続に注入するのを中止する。このように、接続コンポーネント102はここで接続を終了させ、接続は作用しなくなる。第1および第2KAアプリケーション（それぞれ、204および206）の双方がKAパケットを送っているとき、反対側の第2および第1KAアプリケーション（それぞれ206および204）は、受信したKAパケットを除去することができる。つまり、第1および第2ネーティブ・アプリケーション（208および210）はKAパケットを処理するように構成しなければならない。

10

【0031】

図3は、OOB KAパケットを用いて接続を維持するクライアント/サーバー・システム300を示す。更に具体的には、クライアント/サーバーを想定した場合、クライアント304のクライアント・ネーティブ・アプリケーション302は、NATデバイス306を通じて、サーバー310のサーバー・ネーティブ・アプリケーション308への長寿命UDPまたはTCP接続を開く。このアーキテクチャは、TCPのようなエンド・ツー・エンド・トランスポート・プロトコルや、UDPのような無接続トランスポート・プロトコルを用いて動作する。また、クライアント304はファイアウォール、フィルタリング、または多重化コンポーネント312（以後、総合的にファイアウォール312と呼ぶ）も含むことができ、これを通じて通信を行う。通信は、クライアントTCP/IPスタックおよびテーブル314によって、NATデバイス306の接続、サーバーTCP/IPスタックおよびテーブル316、ならびにサーバー・ファイアウォール318を通じて進められ、サーバー・ネーティブ・アプリケーション308に到達する。スタック（314および316）には、関連するTCP/IPシステム・テーブルがあり、このTCP/IPシステム・テーブルは、新たな接続毎に新たなテーブル・エントリを用いて更新し、欠落した接続に対するテーブル・エントリを抜き取る。

20

【0032】

一実施態様によれば、KAアプリケーション（まとめてKAコンポーネント106と記載する）をクライアント304およびサーバー310双方の上で起動する。クライアントKAアプリケーション320は、クライアント304上で起動し、サーバーKAアプリケーション322はサーバー310上で起動する。尚、KAコンポーネントのアプリケーション（320および322）は、背景プロセスとして連続的に走るように、オペレーティング・システム（クライアントおよびサーバー）と共に起動することができることは言うまでもない。前述のように、ネットワーク5-タプル（例えば、プロトコル番号、発信元IPアドレス、発信元ポート、宛先IPアドレス、および宛先ポート）は、KAコンポーネント106（クライアントおよびサーバーKA（320および/または322）の一方または双方）によって、TCP/IPスタック（314および316）と関連のあるTCP/IPテーブルによって発見することができる。

30

40

【0033】

KAアプリケーション（320および322）は、協同してまたは独立して、所望のKAリフレッシュ期間を観察し採用することができる（観察したNATタイムアウトに基づく）。例えば、NATデバイス306の背後にあるクライアント304（例えば、個人側(private side)）がサーバー310（公開側）へのTCP接続を開き、その後静粛（パケット活動なし）を維持していると仮定する。この接続を確立したことにより、クライアントTCP/IPシステム・テーブルには新たなエントリが生ずる。従来の実施態様におけるように、KAコンポーネント106がないと、NATデバイス306は接続状態を終了させ、TCP接続を無効にする。説明したOOBのように動作するKAコンポーネント

50

106の利用によって、TCP接続がNATタイムアウトによって終了せず、したがってクライアント/サーバー・プロトコルは接続を生かしておくために「偽の」帯域内活動を生成する必要はないことを保証する。例えば、観察したNATタイムアウトが15分である場合、リフレッシュ期間(または値)を15分未満(例えば、10分)にすることができる。一般に、KARリフレッシュ期間は、観察したNATデバイスのタイムアウト期間未満となるように採用する。

【0034】

動作において、KAアプリケーション(320および322)は、協同してまたは独立して(トランスポート・プロトコルによって異なる)、クライアント304、サーバー310、またはクライアント304およびサーバー310双方から、KA(またはなりすまし)接続パケットを送る。KAコンポーネント106は、NATデバイス306の受信側においてなりすましパケットを除去するように機能し、これによって、受信するネイティブ・アプリケーションにおいてKAパケットを処理することによる混乱を解消する。

10

【0035】

代替実施態様では、KAコンポーネント106によるなりすましパケットの除去が行われるのは、ネイティブ・アプリケーション(クライアント・ネイティブ・アプリケーション302またはサーバー・ネイティブ・アプリケーション308)が、混乱のないように(またはエラーを生じないように)、なりすましパケットを処理できる十分なロバスト性があるときである。これは、なりすましパケットを認識し欠落させることを含むことができる。例えば、KAパケットは、受信側において、パケット・データーを調べて、なりすましパケットを一意に定める情報を得ることによって、フィルタ処理および/または除去することができる。KAパケットは、ゼロ・ペイロード・パケット(ヘッダのみを有する)とすることができる。他の方法も用いることができる。NATデバイス306は、なりすましパケットを受信する毎(およびネイティブ・パケットの受信時)に、NAT接続タイムアウトをリセットする。TCP/IPパケットの性質により、なりすましは生のIPレイヤーにおいて行われる。何故なら、UDP/TCPプロトコルは、多数のアプリケーションが送り元および受信先双方について同じ5-タプルに結集する(bind)ことを許可しないからである。

20

【0036】

KAコンポーネント106は、それぞれのファイアウォール(312および318)および/またはシステムTCP/IPスタックおよびテーブル(314および316)と通信することにより、ネイティブ・アプリケーションの5-タプルを発見することができる。加えて、KAパケットの除去は、それぞれの受信ファイアウォール(312および318)が、フィルタリング機能性を用いることによって遂行することができる。更に、システム300は、クライアント/サーバーという状況に制限されるのではなく、ピア・ツー・ピア・トポロジーにも適用可能である。

30

【0037】

図4は、KAアプリケーションが多数の接続について接続状態を処理するように動作する、代替システム400を示す。システム400は、NATデバイス306を通じて第2システム404と通信しようとしている第1システム402(例えば、携帯コンピューター)を含む。第1システム402は、2つのネイティブ・アプリケーション、即ち、第1ネイティブ・アプリケーション406および第2ネイティブ・アプリケーション408を含む。また、第1システム402は、第1KAアプリケーション410も含む。第1KAアプリケーション410は、第1および第2ネイティブ・アプリケーション(406および408)にインターフェースして、第1システム402のTCP/IPシステム・テーブル・エントリによって、第1システム402のネイティブ・アプリケーション・システム(402および404)間における活動を監視する。任意に、第1KAアプリケーション410は、NATデバイス306を通じた接続の接続状態も監視することができる。

40

【0038】

同様に、第2システム404(例えば、ウェブ・サーバー)もNATデバイス306を

50

通じて第1システム402に通信する。この特定の例では、第2システム404は、2つのネーティブ・アプリケーション、即ち、第3ネーティブ・アプリケーション412および第4ネーティブ・アプリケーション414を含む。また、第2システム404は第2KAアプリケーション416も含む。第2KAアプリケーション416は、第3および第4ネーティブ・アプリケーション(412および414)にインターフェースして、第2システム404のTCP/IPシステム・テーブル・エントリーによって、第2システム404のネーティブ・アプリケーション活動を監視する。任意に、第2KAアプリケーション416はNATデバイス306を通じたNAT接続の接続状態も監視することができる。

【0039】

この例では、第1および第3ネーティブ・アプリケーション(406および412)はNATデバイス306を通じて第1接続(CON₁で示す)を開き、第2および第4ネーティブ・アプリケーション(408および414)はNATデバイス306を通じて第2接続(CON₂で示す)を開く。第1システム402における新たなTCP/IPシステム・テーブル・エントリーおよび/または第2システム404における新たなTCP/IPシステム・テーブル・エントリーに基づいて、第1および第2KAアプリケーション(410および416)はKAパケットを対応する第1および第2接続に供給して、所望の第1および/または第2接続に対する接続状態を維持する。

【0040】

初期状態において、アクティブなネーティブ・アプリケーションがいずれのシステム(402または404)にもない場合、KAアプリケーション(410および416)を起動しない。第1ネーティブ・アプリケーション406がNATデバイス306を通じて第1接続を開くと、第1KAアプリケーション410が起動して第1接続KAパケットに対してリフレッシュ期間を採用する。例えば、第1接続は第3ネーティブ・アプリケーション412に対してアクティブであるが、しかしながら、この接続がインアクティブであり、第1接続がインアクティブでないことが望まれる場合、第1KAアプリケーション410は自動的に第1接続KAパケットを第1接続に挿入して、第1接続を維持する。第3ネーティブ・アプリケーション412において活動を検出すると、第2KAアプリケーション416は、この時点では受信側KAアプリケーションであり、受信したKAアプリケーションをパケット・ストリームから除外する。つまり、第3ネーティブ・アプリケーション412は、KAパケット処理またはフィルタリングによって損なわれないパケット・ストリームを受信することができる。

【0041】

第2ネーティブ・アプリケーション408がこの時点において活性化され、第4ネーティブ・アプリケーション414への第2接続を開くと、第1KAアプリケーション410は既にNATデバイス306のリフレッシュ期間を知っており、したがって不活性制御が望まれる場合、第2接続(CON₂)に対してKA接続維持をNATデバイス306に適用する。このため、第1KAアプリケーション410は1つのNATデバイス306によって多数の接続を管理することができる。代替動作では、第1KAアプリケーション410は第1接続を管理し、一方第2システム404の第2KAアプリケーション416は第2接続を管理する。尚、多数の接続に対して多数のポートを有するNATデバイスの典型的な実施態様では、多重接続KA管理も実行可能であることは理解できよう。

【0042】

図5は、OOB KA管理方法を示す。説明を簡単にする目的に沿って、ここに示す1つ以上の方法は、例えば、フロー・チャート即ち流れ図の形態で示し、一連の動作として示し説明するが、これらの方法は動作の順には限定されず、一部の動作は、これによれば、ここに示し説明するものとは異なる順序で、および/または他の動作と同時に進行してもよいことは言うまでもないことであり、認められよう。例えば、1つの方法は、状態図におけるように、一連の相互に関連する状態またはイベントとして表すこともできることは、当業者には理解され認められるであろう。更に、1つの方法の図示する動作全てが、

10

20

30

40

50

新規な実施態様に必要ではない場合もある。

【 0 0 4 3 】

5 0 0 において、異なるシステムのネーティブ・アプリケーション間において N A T 接続を開く。5 0 2 において、N A T 接続に対してリフレッシュ期間を決定する。言い換えると、K A アプリケーションは、リフレッシュ値のテーブルを含むことができ、これに基づいて K A パケットを送信する。リフレッシュ値は、K A アプリケーションにおいてハード・コード化することができ（例えば、3 0 秒毎）、あるいは K A アプリケーションが、予め計算した値を用いることもできる。任意に、リフレッシュ期間は、N A T タイムアウト期間に基づいて、関連する K A アプリケーションによって自動的に計算することができる。5 0 4 において、システム・テーブル活動に基づいて、接続を監視する。5 0 6 において、送出側で選択したリフレッシュ値に基づいて K A アプリケーションを用いて、K A パケットを自動的に接続に挿入する。5 0 8 において、望ましければ、受信側で K A パケットを除去する。言い換えると、受信側において、受信した K A アプリケーションが K A パケットを除去するのは必須要件ではない。

10

【 0 0 4 4 】

図 6 は、接続方針に基づいた接続管理方法を示す。6 0 0 において、ネーティブ・アプリケーション間で N A T 接続を開く。6 0 2 において、N A T タイムアウト期間に基づいて、関連する K A アプリケーションによってリフレッシュ値を選択する。6 0 4 において、接続に付随する方針を入手し処理する。6 0 6 において、送出側で選択したリフレッシュ値に基づいて K A アプリケーションを用いて、K A パケットを自動的に接続に挿入する。6 0 8 において、方針に応じて、K A パケットを用いて接続を動作させる。言い換えると、方針は、帯域内または O O B パケット・トラフィックが終了したか否かには関わらず、接続が所定の時間期間開いたままにして、その時間が過ぎ去ったときに接続を遮断することを示すことができる。

20

【 0 0 4 5 】

図 7 は、タイプ・トランスポート・プロトコルに基づいて N A T 接続不活性を管理する方法を示す。7 0 0 において、ネーティブ・アプリケーション間で N A T 接続を開く。7 0 2 において、ネーティブ・アプリケーションの活動に基づいて、対応する K A アプリケーションの 1 つ以上を起動する。7 0 4 において、K A アプリケーションの 1 つ以上が、N A T タイムアウト期間に基づいて、リフレッシュ期間を選択する。7 0 6 において、K A アプリケーションは、スタック・テーブル活動に基づいて、対応するネーティブ・アプリケーション接続を監視する。7 0 8 において、T C P パケットの直前の伝達に基づいて、K A アプリケーションの各々から接続に K A パケットを自動的に挿入する。あるいは、7 1 0 において、U D P パケットの直前の伝達に基づいて、K A アプリケーションの一方または双方から K A パケットを自動的に接続に挿入する。T C P は接続指向エンド・ツー・エンド・トランスポート・プロトコルであるので、K A アプリケーションの双方が K A パケットを挿入し、K A パケットを除去するように動作することが望ましい。U D P は無接続トランスポート・プロトコルであるので、殆どの場合、K A アプリケーションの内 1 つだけが K A パケットを N A T 接続に挿入するように動作すればよい。

30

【 0 0 4 6 】

図 8 は、接続管理のために K A パケットを発生し利用する方法を示す。8 0 0 において、1 つのネーティブ・アプリケーションによって別のネーティブ・アプリケーションへの N A T 接続を、N A T デバイスを通じて開く。8 0 2 において、K A アプリケーションを起動した後、K A アプリケーションの 1 つ以上によってリフレッシュ期間を利用する。8 0 4 において、発見プロセスを開始して、ネーティブ・アプリケーションのネットワーク 5 - タプルを発見する。これは、T C P / I P テーブルによって行うことができる。8 0 6 において、5 - タプル情報を用いて、解く知恵の接続に合わせて K A パケットを組み立てる。8 0 8 において、リフレッシュ期間に基づいて、接続に K A パケットを挿入して N A T タイムアウト・タイマをリセットする。8 1 0 において、N A T デバイスによって処理された K A パケットを、ネーティブ・アプリケーションに到達する前に除去する。

40

50

【 0 0 4 7 】

本願で用いる場合、「コンポーネント」および「システム」という用語は、コンピューター関連エンティティを指すことを意図しており、ハードウェア、ハードウェアとソフトウェアとの組み合わせ、ソフトウェア、または実行ソフトウェア(software in execution)のいずれでもよい。例えば、コンポーネントは、プロセッサ上で走るプロセス、プロセッサ、ハード・ディスク・ドライブ、(光および/または磁気記憶媒体の)多数の記憶装置、オブジェクト、エクゼキュタブル、実行スレッド、プログラム、および/またはコンピューターとすることができるが、これらに限定されるのではない。例示として、サーバー上で走るアプリケーションおよびそのサーバー双方共、コンピューター・コンポーネントであることができる。1つ以上のコンポーネントは、プロセスおよび/または実行スレッド内部に位置することができ、コンポーネントは1つのコンピューターに集中させる(localize)ことおよび/または2つ以上のコンピューター間で分散することもできる。

10

【 0 0 4 8 】

これより図9を参照すると、開示したK Aアーキテクチャを実行するように動作可能なコンピューティングシステム900のブロック図が示されている。その種々の態様について追加のコンテキスト(context)を提示するために、図9および以下の論述は、実施形態の種々の態様を実施することができる、適したコンピューティングシステム900の端的な総合的な説明を行うことを意図している。以上ではローカル・コンピューターおよび/またはリモート・コンピューター上で走るコンピューター実行可能命令という一般的なコンテキストにおいて実施形態を説明したが、本アーキテクチャは、他のプログラム・モジュールとの組み合わせでも、および/またはハードウェアとソフトウェアとの組み合わせでも実施できることは、当業者には認められよう。

20

【 0 0 4 9 】

一般に、プログラム・モジュールは、ルーチン、プログラム、コンポーネント、データ構造等を含み、特定のタスクを実行するか、または特定の抽象的データ・タイプを実現する。更に、当業者には、本発明の方法は他のコンピューター・システム構成でも実用化可能であることが認められよう。他のコンピューター・システム構成には、単一プロセッサまたはマルチプロセッサ・コンピューター・システム、ミニコンピューター、メインフレーム・コンピューター、ならびにパーソナル・コンピューター、ハンド・ヘルドコンピューティングデバイス、マイクロプロセッサ主体消費者用電子機器および/またはプログラマブル消費者用電子機器等が含まれ、これらの各々は1つ以上の関連するデバイスと動作的に通信することができる。

30

【 0 0 5 0 】

また、図示した実施形態の態様は、通信ネットワークを通じてリンクされているリモート処理デバイスがタスクを実行する分散型コンピューティング環境においても実用可能である。分散型コンピューティング環境では、プログラム・モジュールは、ローカルおよびリモート・コンピューター記憶媒体双方に配することができる。

【 0 0 5 1 】

コンピューターは、通例、種々のコンピューター読み取り可能媒体を含む。コンピューター読み取り可能媒体は、コンピューターがアクセス可能な入手可能な媒体であればいずれでも可能であり、揮発性および不揮発性媒体、リムーバブルおよび非リムーバブル媒体を含む。限定ではない一例をあげると、コンピューター読み取り可能媒体は、コンピューター記憶媒体および通信媒体を含むことができる。コンピューター記憶媒体は、コンピューター読み取り可能命令、データ構造、プログラム・モジュール、またはその他のデータというような情報の格納のために、あらゆる方法または技術で実施される、揮発性および不揮発性の双方、リムーバブルおよび非リムーバブル媒体を含む。コンピューター記憶媒体は、RAM、ROM、EEPROM、フラッシュ・メモリーまたはその他のメモリー技術、CD-ROM、デジタル・バーサタイル・ディスク(DVD)またはその他の光ディスク・ストレージ、磁気カセット、磁気テープ、磁気ディスク・ストレージまたは

40

50

その他の磁気記憶デバイス、あるいは所望の情報を格納するために用いることができしかもコンピューターがアクセス可能なその他のいずれの媒体も含むが、これらに限定されるのではない。

【0052】

再度図9を参照すると、本発明の種々の態様を実施するためのシステム環境例900は、コンピューター902を含む。コンピューター902は、演算装置904、システム・メモリー906、およびシステム・バス908を含むことができる。システム・バス908は、システム・メモリー906を含むがこれに限定されない種々のシステム構成要素を演算装置904に結合する。演算装置904は、市販されている種々のプロセッサのいずれでもよい。また、二重マイクロプロセッサやその他のマルチプロセッサ・アーキテクチャも演算装置904として用いることもできる。

10

【0053】

システム・バス908は、数種類のバス構造のいずれでもよく、更に、(メモリー・コントローラを有するまたは有さない)メモリー・バス、周辺バス、ならびに種々の従来のバス・アーキテクチャのいずれかを用いるローカル・バスを含む。システム・メモリー906は、リード・オンリ・メモリー(RAM)910およびランダム・アクセス・メモリー(RAM)912を含む。ROM、EPROM、EEPROM、のような不揮発性メモリー910に、基本入出力システム(BIOS)が格納されており、BIOSは、起動中のように、コンピューター902内のエレメント間におけるデータ転送を補助する基本的なルーチンを含む。また、RAM912は、データをキャッシュするスタティックRAMのような、高速RAMも含むことができる。

20

【0054】

また、コンピューター902は、更に、ハード・ディスク・ドライブ(HDD)914(例えば、EIDE SATE)も含み、内部ハード・ディスク・ドライブ914は適したシャーシ(図示せず)において外部の使用に合わせて構成することができる。更に、磁気フロッピー・ディスク・ドライブ916(例えば、リムーバブル・ディスク918からの読み取りおよびこれへの書き込みを行なう)、および光ディスク・ドライブ920(CD-ROMディスク922を読み取る、あるいはDVDのようなその他の光学媒体からの読み取りおよびこれへの書き込みを行なう)も含む。ハード・ディスク・ドライブ914、磁気ディスク・ドライブ916、および光ディスク・ドライブ920は、ハードウェア・ディスク・ドライブ・インターフェース924、磁気ディスク・ドライブ・インターフェース926、および光ドライブ・インターフェース928によって、それぞれ、システム・バス908に接続されている。外部ドライブ用インターフェース924は、ユニバーサル・シリアル・バス(USB)およびIEEE1394インターフェース技術の少なくとも一方または双方を含む。

30

【0055】

前述のドライブおよびそれらと関連のあるコンピューター読み取り可能媒体は、データ、データ構造、コンピューター実行可能命令等の不揮発性格納を行う。コンピューター902のために、ドライブおよび媒体は、あらゆるデータの適したデジタル・フォーマットでの格納に対処する。以上のコンピューター読み取り可能媒体の説明は、HDD、リムーバブル磁気ディスク、およびCDまたはDVDのようなリムーバブル光媒体に言及したが、zipドライブ、磁気カセット、フラッシュ・メモリー・カード、カートリッジ等のような、コンピューターによる読み取りが可能な他の種類の媒体も、本動作環境例において用いることができ、更に、このような媒体はいずれも、開示した方法を実行するためのコンピューター実行可能命令を収容することができることは、当業者には認められてしかるべきである。

40

【0056】

多数のプログラム・モジュールを前述のドライブおよびRAM912に格納することができ、オペレーティング・システム930、1つ以上のアプリケーション・プログラム932、その他のプログラム・モジュール934、およびプログラム・データ936を含

50

む。オペレーティング・システム、アプリケーション、モジュール、および/またはデータの全部または一部は、RAM 912にキャッシュすることもできる。尚、本アーキテクチャは、種々の市販のオペレーティング・システムまたはオペレーティング・システムの組み合わせでも実現できることは認められてしかるべきである。アプリケーション 932および/またはモジュール 934は、ネイティブ・アプリケーション、K Aアプリケーション、および/または既に述べたK Aコンポーネントを含むことができる。

【0057】

ユーザは、1つ以上のユーザ入力デバイス、例えば、キーボード 938およびマウス 940のようなポインティング・デバイスによって、コマンドおよび情報をコンピューター 902に入力することができる。他の入力デバイス(図示せず)には、マイクロフォン、IRリモート・コントロール、ジョイスティック、ゲーム・パッド、スタイラス・ペン、タッチ・スクリーン等を含むことができる。これらおよびその他の入力デバイスは、多くの場合、システム・バス 908に結合されている入力デバイス・インターフェース 942を通じて、演算装置 904に接続されているが、パラレル・ポート、IEEE 1394シリアル・ポート、ゲーム・ポート、USBポート、IRインターフェース等のようなその他のインターフェースによって接続することも可能である。

10

【0058】

モニター 944またはその他の形式の表示装置も、ビデオ・アダプタ 946のようなインターフェースを介して、システム・バス 908に接続されている。モニター 944に加えて、コンピューターは、スピーカーおよびプリンター等のような、その他の周辺出力装置(図示せず)も含むのが通例である。

20

【0059】

尚、コンピューター 902は、1つ以上のリモート・コンピューター 960への論理接続を用いて、リモート・コンピューター 948のような1つ以上のリモート・コンピューターへの有線および/またはワイヤレス接続を通じて、ネットワーク環境において動作することも可能である。リモート・コンピューター 948は、ワークステーション、サーバー・コンピューター、ルータ、パーソナル・コンピューター、携帯コンピューター、マイクロプロセッサ主体の娯楽機器、ピア・デバイス、またはその他の共通コンピューター・ノードとすればよく、通例、コンピューター 902に関して先に説明したエレメントの多くまたは全てを含むが、簡略化の目的上、メモリー/記憶装置 950のみを示す。図示する論理接続は、ローカル・エリア・ネットワーク(LAN) 952および/またはそれよりも大きなネットワーク、例えば、ワイド・エリア・ネットワーク(WAN) 954への有線/ワイヤレス接続を含む。このようなLANおよびWANネットワーク環境は、事務所や会社では珍しくなく、イントラネットのような企業全体に及ぶコンピューター・ネットワークが容易にでき、その全ては、大域通信ネットワーク、例えば、インターネットに接続することができる。

30

【0060】

LANネットワーク環境で用いる場合、例えば、コンピューター 902は、有線および/またはワイヤレス通信ネットワーク・インターフェースあるいはアダプタ 956を介してローカル・ネットワーク 952に接続する。アダプタ 956は、LAN 952への有線またはワイヤレス通信を容易にすることができ、LAN 952は、ワイヤレス・アダプタ 956と通信するために配置したワイヤレス・アクセス・ポイントも含むことができる。

40

【0061】

WANネットワーク環境で用いる場合、コンピューター 902は、通例、モデム 958を含むことができ、あるいはWAN 954上にある通信サーバーに接続し、あるいはインターネットによってというように、WAN 954上で通信を確立するその他の手段を有する。モデム 958は、コンピューター 902に対して内部または外部でも可能であり、シリアル・ポート・インターフェース 942を介してシステム・バス 908に接続する。ネットワーク環境では、コンピューター 902に関して図示したプログラム・モジュール

50

ルまたはその一部を、リモート・メモリー/記憶装置 950 に格納することができる。尚、図示のネットワーク接続は一例であり、コンピューター間で通信リンクを確立する他の手段も使用可能であることは認められよう。

【0062】

コンピューター 902 は、ワイヤレス通信内に動作的に配置されているあらゆるワイヤレス・デバイスまたはエンティティ、例えば、プリンター、スキャナー、デスクトップおよび/または携帯コンピューター、携帯データ・アシスタント、通信衛星、ワイヤレスで検出可能なタグと関連付けられた機器または場所（例えば、キオスク、新聞売店、休憩室）の任意のもの、そして電話機と通信するように動作可能である。これには、少なくとも Wi-Fi および Bluetooth（商標）ワイヤレス技術が含まれる。つまり、通信は、従来のネットワークのように、既定の構造とすることができ、あるいは単に少なくとも 2 つのデバイス間におけるその場限りの通信とすることもできる。

10

【0063】

これより図 10 を参照すると、OOB KA 処理を採用することができるコンピューティング環境例 1000 のモードブロック図が示されている。システム 1000 は、1 つ以上のクライアント 1002 を含む。クライアント 1002 は、ハードウェアおよび/またはソフトウェア（例えば、スレッド、プロセス、コンピューティングデバイス）とすることができる。クライアント 1002 は、例えば、クッキーおよび/または関連する文脈情報を収容することができる。

【0064】

また、システム 1000 は、1 つ以上のサーバー 1004 も含む。サーバー 1004 も、ハードウェアおよび/またはソフトウェア（例えば、スレッド、プロセス、コンピューティングデバイス）とすることができる。サーバー 1004 は、例えば、前述のアーキテクチャを採用することによって、変換を実行するスレッドを収容することができる。クライアント 1002 とサーバー 1004 との間で可能な 1 つの通信は、2 つ以上のコンピューター・プロセス間で送信するように構成したデータ・パケットの形態とすることができる。データ・パケットは、例えば、クッキーおよび/または関連する文脈情報を含むことができる。システム 1000 は、クライアント 1002 とサーバー 1004 との間における通信をしやすくするために用いることができる通信フレームワーク 1006（例えば、インターネットのような大域通信ネットワーク）を含む。

20

30

【0065】

通信は、有線（光ファイバを含む）および/またはワイヤレス技術によって促進することができる。クライアント 1002 は、クライアント 1002 に対してローカルな情報（例えば、クッキーおよび/または関連する文脈情報）を格納するために用いることができる 1 つ以上のクライアント・データ・ストア 1008 に動作的に接続されている。同様に、サーバー 1004 も、サーバー 1004 に対してローカルな情報を格納するために用いることができる 1 つ以上のサーバー・データ・ストア 1010 に動作的に接続されている。

【0066】

クライアント 1002 およびサーバー 1004 は、双方共、NAT ルータ、ゲートウェイ等のような、ネットワーク・インターフェース・デバイス（図示せず）を監視する KA アプリケーションを含むことができる。前述のように、クライアント 1002 は、ローカル KA アプリケーションを用いることによって、接続状態管理をクライアントの一方または双方の内部で行うことができるように、ピア・ツー・ピア式に相互接続することができる。

40

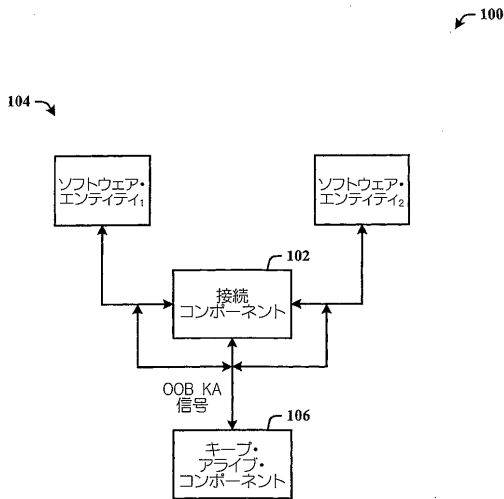
【0067】

以上説明したことは、開示したアーキテクチャの例を含む。勿論、実施形態を記載する目的のためにコンポーネントおよび方法の想起し得る全ての組み合わせを記載することは可能ではないが、実施形態の更に別の多くの組み合わせおよび変形も可能であることは、当業者には認められよう。したがって、本主題は、添付した特許請求の範囲の主旨および

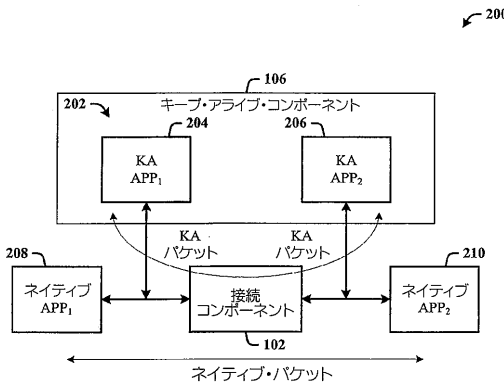
50

範囲に該当するような変更、修正、および変形を全て包含することとする。更に、「含む」(include)という用語が詳細な説明または特許請求の範囲のいずれかにおいて用いられる限りでは、特許請求の範囲において「備えている」(comprising)が移行句(transitional word)として用いられているときに解釈する際の「備えている」と同様に、このような用語は、包含的であることとする。

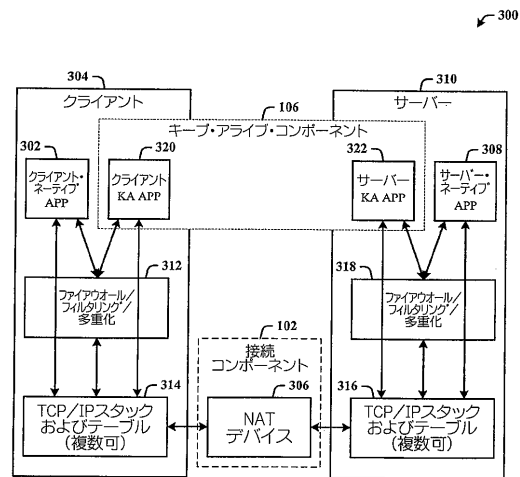
【図1】



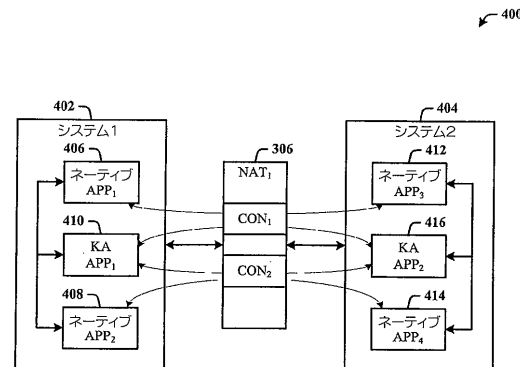
【図2】



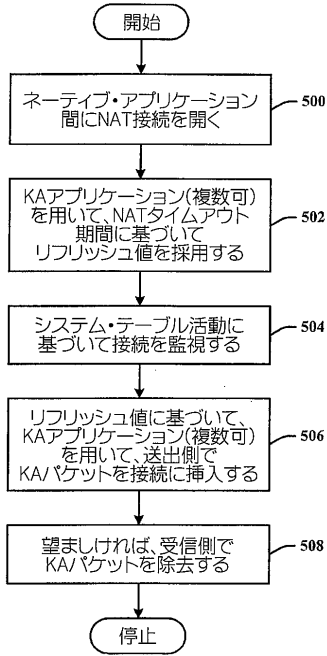
【図3】



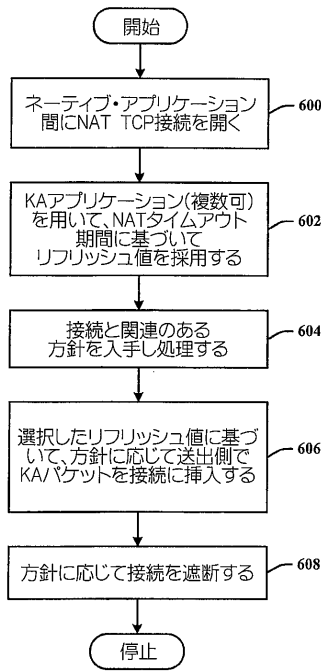
【図4】



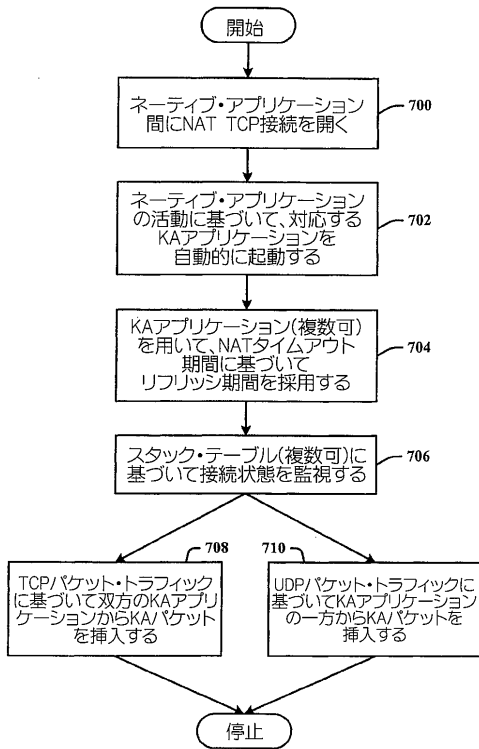
【図5】



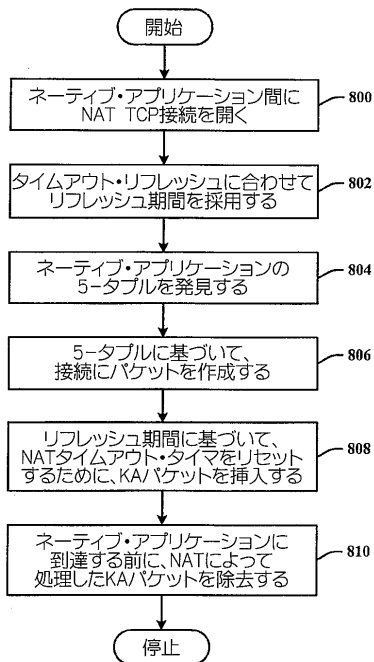
【図6】



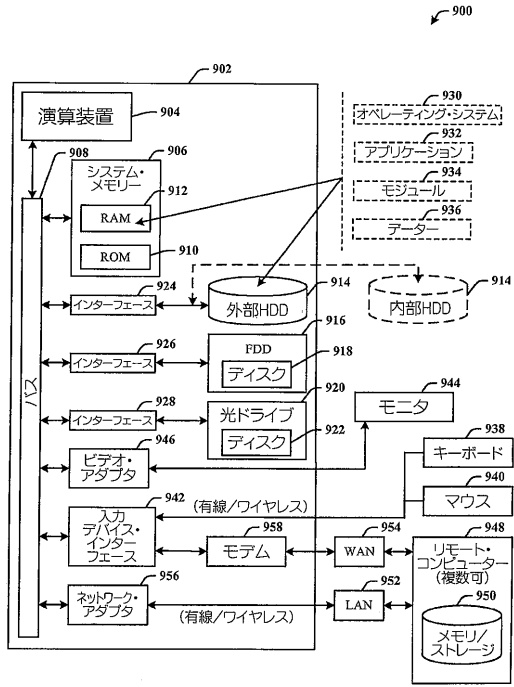
【図7】



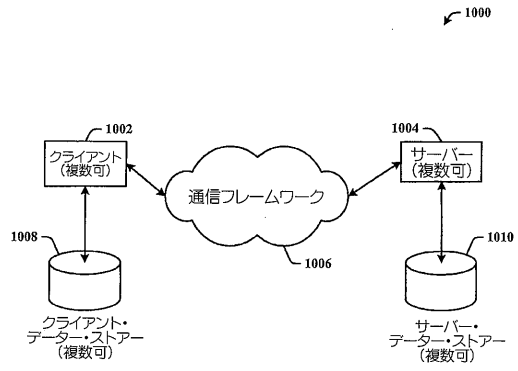
【図8】



【図9】



【図10】



フロントページの続き

(74)代理人 100153028

弁理士 上田 忠

(72)発明者 ヘルゾグ, シャイ

アメリカ合衆国ワシントン州 9 8 0 5 2 , レッドモンド, ワン・マイクロソフト・ウェイ, マイクロソフト コーポレーション, インターナショナル・パテント

(72)発明者 ハグマン, マリー

アメリカ合衆国ワシントン州 9 8 0 5 2 , レッドモンド, ワン・マイクロソフト・ウェイ, マイクロソフト コーポレーション, インターナショナル・パテント

審査官 永井 啓司

(56)参考文献 国際公開第 0 3 / 0 9 6 6 5 3 (W O , A 1)

特開 2 0 0 6 - 2 0 3 5 7 5 (J P , A)

特開 2 0 0 4 - 1 2 0 5 4 7 (J P , A)

米国特許第 6 9 7 6 0 7 1 (U S , B 1)

(58)調査した分野(Int.Cl. , D B 名)

H04L 12/00-12/26、12/50-12/66