(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第3961415号
(P3961415)
(45)発行日　平成19年8月22日(2007.8.22)　(24)登録日　平成19年5月25日(2007.5.25)

(51)Int.Cl.		F I		
H04L 29/14	(2006.01)	H04L	13/00	313
H04L 12/56	(2006.01)	H04L	12/56	400Z
H04L 29/06	(2006.01)	H04L	13/00	305Z

請求項の数 8　(全 12 頁)

(21)出願番号　特願2002-363894 (P2002-363894)
(22)出願日　平成14年12月16日 (2002.12.16)
(65)公開番号　特開2004-200773 (P2004-200773A)
(43)公開日　平成16年7月15日 (2004.7.15)
審査請求日　平成17年4月14日 (2005.4.14)

(73)特許権者　392026693
株式会社エヌ・ティ・ティ・ドコモ
東京都千代田区永田町二丁目１１番１号
(74)代理人　100066980
弁理士　森　哲也
(74)代理人　100075579
弁理士　内藤　嘉昭
(74)代理人　100103850
弁理士　崔　秀▲てつ▼
(72)発明者　石川　太朗
東京都千代田区永田町二丁目１１番１号
株式会社エヌ・ティ・ティ・ドコモ内
(72)発明者　稲村　浩
東京都千代田区永田町二丁目１１番１号
株式会社エヌ・ティ・ティ・ドコモ内
最終頁に続く

(54)【発明の名称】プロトコル不具合自動検出方法、及び、プロトコル不具合自動検出装置

(57)【特許請求の範囲】
【請求項１】
所定通信プロトコルに従って少なくとも１以上の送受信制御処理を行なう送受信端末間の通信において発生する、該送受信制御処理の不具合を検出する不具合検出方法であって、
前記送受信端末間における通信の間に送受されるパケットを取得することにより、前記通信プロトコルに従った送受信制御の結果に対応すべき該パケットの送受信状態に関する状態情報を算出する算出ステップと、
前記算出ステップにおいて算出された状態情報と、前記少なくとも１以上の送受信制御処理のそれぞれの不具合を特徴付ける不具合情報と、を比較する比較ステップと、
を有し、前記比較ステップにおける比較結果に基づいて前記不具合の発生している送受信制御処理を検出することを特徴とするプロトコル不具合自動検出方法。
【請求項２】
前記通信プロトコルに従って前記送受信端末において送受されるパケットに基づいて行われるべき送受信制御処理を特定し、該特定された送受信制御処理が正常に行われた処理結果に対応すべき正常情報を推定する推定ステップを、更に、含み、
前記不具合情報は、前記不具合がある場合における、前記算出ステップにおいて算出される状態情報と前記正常情報との関係を規定することを特徴とする請求項１に記載のプロトコル不具合自動検出方法。
【請求項３】
前記不具合情報は、前記状態情報と、前記送受信制御処理の不具合についてあらかじめ確

認されている固定値と、の関係を規定することを特徴とする請求項１又は２に記載のプロトコル不具合自動検出方法。
【請求項４】
前記算出ステップにおいては、前記パケットの取得のたびに、前記状態情報を更新し、
前記比較ステップにおいては、前記算出ステップにおいて更新される最新の状態情報と、前記不具合情報と、を比較することを特徴とする請求項１～３のいずれか１項に記載のプロトコル不具合自動検出方法。
【請求項５】
前記状態情報は、送受信パケット数の合計値、パケットサイズの最小値又は最大値、又は、送信されたパケットに対する応答パケットを受信するまでのラウンドトリップタイム等の情報であることを特徴とする請求項１～４のいずれか１項に記載のプロトコル不具合自動検出方法。
【請求項６】
所定通信プロトコルに従って少なくとも１以上の送受信制御処理を行なう送受信端末間の通信において発生する、該送受信制御処理の不具合を検出する不具合検出装置であって、
前記送受信端末間における通信の間に送受されるパケットを取得するパケット取得手段と、
前記パケット取得手段により取得したパケットに基づいて、前記通信プロトコルに従った送受信制御の結果に対応すべき該パケットの送受信状態に関する状態情報を算出する算出手段と、
前記算出手段により算出された状態情報と、あらかじめ蓄積される、前記少なくとも１以上の送受信制御処理のそれぞれの不具合を特徴付ける不具合情報と、を比較する比較手段と、
を有し、前記比較手段による比較結果に基づいて前記不具合の発生している送受信制御処理を検出することを特徴とするプロトコル不具合自動検出装置。
【請求項７】
前記パケット取得手段により取得されるパケットに基づいて、前記通信プロトコルに従って、前記送受信端末において該取得したパケットに対して行われるべき送受信制御処理を特定し、該特定された送受信制御処理が正常に行われた処理結果に対応すべき正常情報を推定する推定手段を、更に、含み、
前記不具合情報は、前記不具合がある場合における、前記算出手段により算出される状態情報と前記正常情報との関係を規定することを特徴とする請求項６に記載のプロトコル不具合自動検出装置。
【請求項８】
前記パケット取得手段により取得したパケットのヘッダ情報に基づいて、必要なパケットのみを選択して前記算出手段に転送するためのパケットフィルタ手段を、更に、有することを特徴とする請求項６又は７に記載のプロトコル不具合自動検出装置。
【発明の詳細な説明】
【０００１】
【発明の属する技術分野】
本発明はプロトコル不具合自動検出方法、及び、プロトコル不具合自動検出装置に関し、特に新規の通信装置の導入時の不具合検出等に用いることができるプロトコル不具合自動検出方法、及び、プロトコル不具合自動検出装置に関する。
【０００２】
【従来の技術】
近年インターネットアクセスが急速に増大しつつあり、ＴＣＰ／ＩＰ（Ｔｒａｎｓｍｉｓｓｉｏｎ Ｃｏｎｔｒｏｌ Ｐｒｏｔｏｃｏｌ／Ｉｎｔｅｒｎｅｔ Ｐｒｏｔｏｃｏｌ）プロトコルを実装する様々なコンピュータ、通信デバイスが開発されている。またＴＣＰ／ＩＰプロトコルを用いる新たなアプリケーションが開発され、アプリケーションの種類が増大している。ＴＣＰ／ＩＰプロトコルを実装するコンピュータ、通信デバイスの種類

が増えることでＴＣＰ／ＩＰプロトコル実装の不具合の種類が増える可能性がある。また、従来アプリケーションでは問題のなかったＴＣＰ／ＩＰ実装において、新たなアプリケーションに利用することにより、潜在していた不具合が誘発される可能性がある。すなわち、ここでいうＴＣＰ／ＩＰプロトコル実装の不具合としては、通信デバイス等がＴＣＰ／ＩＰプロトコルの仕様に従った動作を行わない場合やアプリケーションにおける利用上の不備等の実装上の不備に起因してＴＣＰ／ＩＰプロトコルにおいて予定する通信処理が行なわれない不具合が生じる場合のみでなく、新たなアプリケーション等への利用により従来の利用においては想定し得なかったＴＣＰ／ＩＰプロトコル自体の不備・欠陥により予定する通信処理が機能しない場合があげられる。

【０００３】

このようなＴＣＰ／ＩＰプロトコル実装の不具合を判断するための従来の手法では、例えば、ｔｃｐｄｕｍｐ（非特許文献１参照）に代表されるプロトコルアナライザを用いる。図６にプロトコルアナライザ８の機能ブロック図を示す。プロトコルアナライザ８は、主にネットワークに流れるパケットを収集する機能を有するものである（ネットワークインターフェース８ａ、パケット受信部８ｂ）。パケット中のプロトコルヘッダを各情報区切り毎に、認識可能なテキストデータ等に翻訳し（パケット翻訳部８ｄ）、画面出力させる（画面出力部８ｅ)ことにより、パケットの内容を把握してプロトコルの不具合判断が可能になる。

【０００４】

さらに、ＴＣＰに限ると、ｔｃｐｔｒａｃｅ（非特許文献２参照）に代表される解析ツールが提案され、ｔｃｐｄｕｍｐ等の標準的なプロトコルアナライザにより収集したパケットの保存データから、転送データ量、再送データ量、スループット、ラウンドトリップタイムなどの統計情報を得ることができる。これらの統計情報を、画面出力等することによりプロトコルの不具合の判断材料として用いることができる。図７に解析ツール９の機能ブロック図を示す。

【０００５】

【非特許文献１】

ＲＦＣ２３９８　Ｓｏｍｅ　Ｔｅｓｔｉｎｇ　Ｔｏｏｌｓ　ｆｏｒ　ＴＣＰ　Ｉｍｐｌｅｍｅｎｔｏｒｓ、［ｏｎｌｉｎｅ］、［平成１４年１２月１１日検索］、インターネット<ＵＲＬ：ｈｔｔｐ:／／ｗｗｗ．ｔｃｐｄｕｍｐ．ｏｒｇ／>

【非特許文献２】

ＲＦＣ２３９８　Ｓｏｍｅ　Ｔｅｓｔｉｎｇ　Ｔｏｏｌｓ　ｆｏｒ　ＴＣＰ　Ｉｍｐｌｅｍｅｎｔｏｒｓ、［ｏｎｌｉｎｅ］、［平成１４年１２月１１日検索］、インターネット<ＵＲＬ：ｈｔｔｐ:／／ｗｗｗ．ｔｃｐｔｒａｃｅ．ｏｒｇ／>

【０００６】

【発明が解決しようとする課題】

しかしながら、上述のプロトコルアナライザから得た翻訳出力や解析ツールで得た統計情報を用いたプロトコル不具合検出には不十分である。

すなわち、プロトコルアナライザは、同時に収集される複数のコネクションにおいて送受されるパケットそれぞれのヘッダの翻訳のみを行う。このため、プロトコル不具合検出時には、翻訳情報から、それぞれのパケットをコネクションごとに対応付け、更にどのパケットがプロトコル固有のシーケンスのどの部分に相当するものなのかの対応付けを行うなど、煩雑な作業を行わなければならない。

【０００７】

解析ツールについても、プロトコルアナライザの保存データを加工し、コネクション毎に伝送データ量、再送データ量、スループット等の統計値や、シーケンス図等のグラフを提供するが、ある程度の異常の発生については確認することができても、どのような処理の不具合によって異常が発生しているのかについての判断までも与えるものではない。このため、解析ツールが示す結果によってどのような処理の不具合であるのか等の原因特定を行うためには、解析ツールによって事後的に示される結果に基づいて、不具合が発生した

と思われる時間周辺のパケットを特定し、プロトコル固有のシーケンスに沿ってパケットの構成に異常がないかを調べるなどして処理の異常を特定する必要がある。更に、通信状態等に応じてプロトコルに従った処理の内容が変化することなども考慮する必要があり、原因を特定するには、専門的な知識と煩雑な作業を要することとなる。
本発明の目的は、上述の課題に鑑みてなされたものであり、専門的な知識と煩雑な作業なしにプロトコルの不具合を検出可能なプロトコル不具合自動検出方法、及び、プロトコル不具合自動検出装置を提供することにある。
【０００８】
【課題を解決するための手段】
本発明の請求項１によるプロトコル不具合自動検出方法は、所定通信プロトコルに従って少なくとも１以上の送受信制御処理を行なう送受信端末間の通信において発生する、該送受信制御処理の不具合を検出する不具合検出方法であって、前記送受信端末間における通信の間に送受されるパケットを取得することにより、前記通信プロトコルに従った送受信制御の結果に対応すべき該パケットの送受信状態に関する状態情報を算出する算出ステップと、
前記算出ステップにおいて算出された状態情報と、前記少なくとも１以上の送受信制御処理のそれぞれの不具合を特徴付ける不具合情報と、を比較する比較ステップと、
を有し、前記比較ステップにおける比較結果に基づいて前記不具合の発生している送受信制御処理を検出することを特徴とする。
【０００９】
本発明の請求項２によるプロトコル不具合自動検出方法は、請求項１において、前記通信プロトコルに従って前記送受信端末において送受されるパケットに基づいて行われるべき送受信制御処理を特定し、該特定された送受信制御処理が正常に行われた処理結果に対応すべき正常情報を推定する推定ステップを、更に、含み、
前記不具合情報は、前記不具合がある場合における、前記算出ステップにおいて算出される状態情報と前記正常情報との関係を規定することを特徴とする。
【００１０】
本発明の請求項３によるプロトコル不具合自動検出方法は、請求項１又は２おいて、前記不具合情報は、前記状態情報と、前記送受信制御処理の不具合についてあらかじめ確認されている固定値と、の関係を規定することを特徴とする。
本発明の請求項４によるプロトコル不具合自動検出方法は、請求項１～３のいずれか１項おいて、前記算出ステップにおいては、前記パケットの取得のたびに、前記状態情報を更新し、
前記比較ステップにおいては、前記算出ステップにおいて更新される最新の状態情報と、前記不具合情報と、を比較することを特徴とする。
【００１１】
本発明の請求項５によるプロトコル不具合自動検出方法は、請求項１～４のいずれか１項おいて、前記状態情報は、送受信パケット数の合計値、パケットサイズの最小値又は最大値、又は、送信されたパケットに対する応答パケットを受信するまでのラウンドトリップタイム等の情報であることを特徴とする。
本発明の請求項６によるプロトコル不具合自動検出装置は、所定通信プロトコルに従って少なくとも１以上の送受信制御処理を行なう送受信端末間の通信において発生する、該送受信制御処理の不具合を検出する不具合検出装置であって、前記送受信端末間における通信の間に送受されるパケットを取得するパケット取得手段と、
前記パケット取得手段により取得したパケットに基づいて、前記通信プロトコルに従った送受信制御の結果に対応すべき該パケットの送受信状態に関する状態情報を算出する算出手段と、
前記算出手段により算出された状態情報と、あらかじめ蓄積される、前記少なくとも１以上の送受信制御処理のそれぞれの不具合を特徴付ける不具合情報と、を比較する比較手段と、

を有し、前記比較手段による比較結果に基づいて前記不具合の発生している送受信制御処理を検出することを特徴とする。

【００１２】

本発明の請求項７によるプロトコル不具合自動検出装置は、請求項６において、前記パケット取得手段により取得されるパケットに基づいて、前記通信プロトコルに従って、前記送受信端末において該取得したパケットに対して行われるべき送受信制御処理を特定し、該特定された送受信制御処理が正常に行われた処理結果に対応すべき正常情報を推定する推定手段を、更に、含み、

前記不具合情報は、前記不具合がある場合における、前記算出手段により算出される状態情報と前記正常情報との関係を規定することを特徴とする。

【００１３】

本発明の請求項８によるプロトコル不具合自動検出装置は、請求項６又は７において、前記パケット取得手段により取得したパケットのヘッダ情報に基づいて、必要なパケットのみを選択して前記算出手段に転送するためのパケットフィルタ手段を、更に、有することを特徴とする。

【００１４】

【発明の実施の形態】

次に、図面を参照して本発明の実施の形態について説明する。なお、以下の説明において参照する各図においては、他の図と同等の部分が同一符号によって示されている。

（プロトコル不具合自動検出装置の構成）

図１には、本実施の形態におけるプロトコル不具合自動検出装置の構成を説明するブロック図が示されている。

【００１５】

ネットワークインターフェイス１ａは、外部ネットワークと通信を行う機能を有する。

パケット受信部１ｂは、ネットワークインターフェイス１ａに到着したパケットを受信し、保存が必要な場合はデータ保存のためにパケット保存・読込部１ｋにパケットを転送し、保存が必要でない場合は、パケットフィルタ・解析部１ｃにする機能を有する。

【００１６】

パケット保存・読込部１ｋは、パケットの保存が必要な場合、パケット受信部１ｂで受信したパケットの保存を行い、保存済みのパケットデータの不具合解析を行う場合には、保存済みのパケットデータを、パケット受信部１ｂを介してパケットフィルタ・解析部１ｃに転送する機能を有する。

パケットフィルタ・解析部１ｃは、パケット受信部１ｂから受け取ったパケットのヘッダ情報を解析し、必要な種類のパケット以外を廃棄する機能と、必要な種類のパケットのヘッダ情報及びペイロード情報を不具合比較判定部及びコネクション情報算出部１ｄに転送する機能を有する。

【００１７】

コネクション情報算出部１ｄは、パケットフィルタ・解析部１ｃを介してパケットのヘッダ情報及びペイロード情報を受け取り、ＴＣＰコネクション情報を作成し、コネクション情報保存部１ｅに保存する機能を有する。

ＴＣＰコネクション情報は、本実施の形態においては、ＴＣＰコネクションが設定されている間に送受されるパケットを取得することにより、ＴＣＰ／ＩＰプロトコルに従った送受信制御の結果に対応すべきパケットの送受信状態に関する状態情報である。本実施の形態においては、パケットのヘッダ情報、ペイロード情報やパケットの送受信事象の発生といったパケットそのものを解析することにより取得し、当該コネクションに該当するパケットのヘッダ情報及びペイロード情報を受け取る度に更新される。ＴＣＰコネクション情報に含まれる情報としては、送信パケット数、再送パケット数、ＳＡＣＫブロック数の各種合計値、最小パケットサイズ、再送間隔の最大値等のスループット、ラウンドトリップタイム等の各種評価値等がある。

【００１８】

コネクション情報保存部１ｅは、コネクション情報算出部１ｄにて作成されため情報を保存する機能を有する。
正常情報推定部１ｆは、コネクション情報算出部１ｄを介してパケットのヘッダ情報及びペイロード情報を受け取り、当該パケットのヘッダ情報等に基づいて、当該パケットの送信元又は送信先の送受信端末において行われるべき送受信制御処理を特定し、当該特定された送受信制御処理が正常に行われた場合におけるその処理結果に対応すべき正常情報を推定する。正常情報としては、例えば、ＴＣＰコネクションを確立し、ＴＣＰ／ＩＰプロトコルに従った送受信制御を行う送受信端末において、当該制御を実施するＴＣＰにおいて用いられるｃｗｎｄ、ｓｓｔｈｒｅｓｈ、ｓｒｔｔ、ｒｔｔｖａｒ等の内部変数やＴＣＰの状態遷移ダイヤグラムの状態推定値などがある。これらの推定された正常情報は、正常情報保存部１ｇに保存される。

【００１９】
正常情報保存部１ｇは、正常情報推定部１ｆにおいて推定された正常情報を保存する機能を有する。
不具合比較判定部１ｈは、パケットフィルタ・解析部１ｃの解析結果と、正常情報保存部１ｇに保存される正常情報と、不具合情報保存部１ｉに保存される不具合情報と、コネクション情報保存部１ｅに保存されるＴＣＰコネクション情報と、を比較することにより不具合の発生している処理を検出する。この比較判定の結果は、判定結果出力部１ｊに転送される。

【００２０】
不具合情報保存部１ｉは、これまで知られているプロトコルにおける少なくとも１以上の処理のそれぞれの不具合を特徴付ける情報を保存する機能を有する。不具合を特徴付けるデータの具体例としては、ＴＣＰコネクション情報に関する条件式、パケットヘッダ情報に関する条件式、又は、それらの組み合わせが挙げられる。ＴＣＰコネクション情報に関する条件式としては、例えば、後述するように、ＴＣＰコネクション情報に保持される値と推定された正常情報の値、あるいは、ＴＣＰコネクション情報が異常時にとる固定値・算出値、との大小関係を規定する式などが挙げられる。また、パケットヘッダ情報に関する条件式は、例えば不正なパケットの構成を規定するなど、パケットそのものの構成の不具合に関する条件式である。

【００２１】
判定結果出力部１ｊは、不具合比較判定部１ｈで行った判定結果を出力する機能を有する。
（プロトコル不具合自動検出装置及びプロトコル不具合自動検出方法の実施例）以下、図１のプロトコル不具合自動検出装置を用いた不具合検出の一例として、ＴＣＰによるパケットの送受信制御の１つである輻輳制御のためのＦａｓｔ　Ｒｅｔｒａｎｓｍｉｔ／Ｆａｓｔ　Ｒｅｃｏｖｅｒｙ　アルゴリスム（ＲＦＣ２５８１　ＴＣＰ　Ｃｏｎｇｅｓｔｉｏｎ　Ｃｏｎｔｒｏｌ）を行わない不具合を検出する方法を説明する。

【００２２】
図２には、本実施の形態にかかかる上述のプロトコル不具合自動検出装置が用いられるネットワークの全体構成を説明する図が示されている。
サーバ２１は、ルータ２３、インターネットＩ、ルータ２４を介してクライアント２２と通信している。サーバ２１はＦａｓｔ　Ｒｅｔｒａｎｓｍｉｔ／Ｆａｓｔ　Ｒｅｃｏｖｅｒｙアルゴリズム正しく行わない不具合がある。

【００２３】
プロトコル不具合自動検出装置１は、サーバ２１と同じイーサネット（登録商標）セグメントに接続されているため、サーバ２１が送受信するパケットをすべて受信することができる。
コネクション情報算出部１ｄは、統計値ｓｎｄ＿ｕｍａ、ｓｎｄ＿ｍａｘを扱う。ｓｎｄ＿ｕｍａは、これまでにサーバ２１に応答確認されたデータセグメントの値、ｓｎｄ＿ｍａｘは、サーバ２１から送信されたデータセグメントのシーケンス番号の最大値を示す統

計値である。ゆえに、（ｓｎｄ＿ｍａｘ　－　ｓｎｄ＿ｕｍａ）の統計値は、正常時には、サーバ２１における輻輳制御処理の結果に対応した値をとる。

【００２４】

正常情報推定部１ｆは、サーバ２１に送信されたパケットを受信する度に、Ｆａｓｔ　ｒｅｔｒａｎｓｍｉｔ／Ｆａｓｔ　Ｒｅｃｏｖｅｒｙアルゴリズムに基づいて、ｃｗｎｄ、ｓｓｔｈｒｅｓｈを推定し、それぞれ正常情報保存部１ｇのｓｎｄ＿ｃｗｎｄ、ｓｎｄ＿ｓｓｌｈｒｅｓｈに保存する。

正常情報保存部１ｇは、正常情報としてｓｎｄ＿ｃｗｎｄ、ｓｎｄ＿ｓｓｔｈｒｅｓｈを扱う。

【００２５】

不具合情報保存部１ｉはＦａｓｔ　Ｒｅｔｒａｎｓｍｉｔ／Ｆａｓｔ　Ｒｅｃｏｖｅｒｙアルゴリズムを行わない不具合を特徴づける条件式（１）を保有する。

（ｓｎｄ＿ｍａｘ　－　ｓｎｄ＿ｕｍａ）　＞　ｓｎｄ＿ｃｗｎｄ　・・・（１）

不具合比較判定部１ｈは、不具合情報保存部１ｉに保存されている条件式（１）が満たされると、正しくＦａｓｔ　ｒｅｔｒａｎｓｍｉｔ／Ｆａｓｔ　Ｒｅｃｏｖｅｒｙアルゴリズムが行われない不具合を観測した旨表示する。

【００２６】

正常情報保存部１ｇのｓｎｄ＿ｃｗｎｄが４３８００、ｓｎｄ＿ｓｓｔｈｒｅｓｈが６５５３５で、これらの推測値が正しく推測された状態において、図３に示すパケットを受信すると、ｓｎｄ＿ｍａｘ、ｓｎｄ＿ｕｍａ、（ｓｎｄ＿ｍａｘ　－　ｓｎｄ＿ｕｍａ）、及び、ｓｎｄ＿ｃｗｎｄは、それぞれ図４のように変化する。

【００２７】

Ｆａｓｔ　Ｒｅｔｒａｎｓｍｉｔ／Ｆａｓｔ　Ｒｅｃｏｖｅｒｙアルゴリズムを実装する正常なサーバ２１のＴＣＰでは、新たなＡＣＫを受信するたびに輻輳ウィンドウサイズを規定する内部変数ｃｗｎｄの値を増やし、３つの重複ＡＣＫ受信すると（輻輳によるパケットロスが確認されると）、ｃｗｎｄをこれまでの値の半分にし、さらに３セグメント分増やす。２つまでの重複ＡＣＫに対してはｃｗｎｄの更新は行わない。このような制御により、輻輳時におけるパケット転送量の制御を行っている。

【００２８】

図４に示される、時刻１５：３７：２１．６６７００７のＡＣＫ受信Ｐ１、及び、時刻１５：３７：２１．６９７００３のＡＣＫ受信Ｐ２は、重複ＡＣＫ受信であり、正常なＴＣＰでは、これらのＡＣＫ受信に対して、ｃｗｎｄの更新は行わない。

時刻１５：３７：２１．７２７００７のＡＣＫ受信Ｐ３は３つ目の重複ＡＣＫ受信でありこのＡＣＫ受信Ｐ３に対して、ｃｗｎｄをこれまでのこの半分にし、さらに３セグメント分増やす操作を行う。これ以後の重複ＡＣＫに対しては、重複ＡＣＫを受け取るたびに１セグメント分増やす。

【００２９】

この正常なアルゴリズムに従ったｃｗｎｄの推測値ｓｎｄ＿ｃｗｎｄは、ＡＣＫ受信Ｐ１及びＡＣＫ受信Ｐ２によっては変化せず、ＡＣＫ受信Ｐ３によって、これまでの値の半分＋３セグメント分（４６７２０／２＋１４６０×３＝２７７４０）に更新され、以後重複ＡＣＫ受信Ｐ４、Ｐ５、Ｐ６、、、のたびに、２９２００、３０６６０、３２１２０、、、と１セグメント分増やす。

【００３０】

これに対し、（ｓｎｄ＿ｍａｘ　－　ｓｎｄ＿ｕｍａ）は２つまでの重複ＡＣＫ受信Ｐ１、Ｐ２に連動して増加し（値４５２６０、４６７２０）、３つ目の重複ＡＣＫ受信Ｐ３に対しても値が小さくならずに（値４６７２０）、以後の重複ＡＣＫ受信Ｐ４、Ｐ５、Ｐ６、、、に連動して値が増加（値４８１８０、４９６４０、５１１００、、、）している。

【００３１】

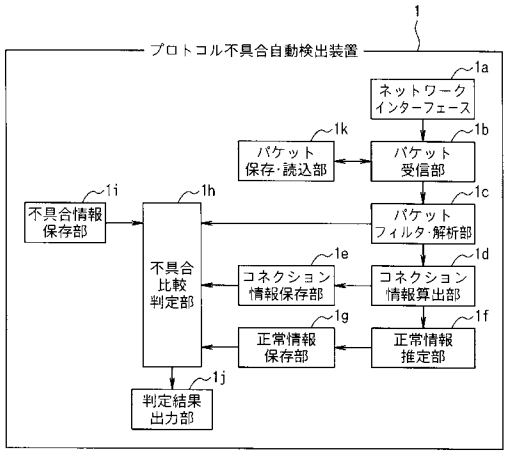
そのため、１５:３７:２１．７２７００７のＡＣＫ受信Ｐ３以降、条件式（１）が成立し、判定結果出力部１ｊに正しくＦａｓｔ　ｒｅｔｒａｎｓｍｉｔ／Ｆａｓｔ　Ｒｅｃｏｖ

ｅｒｙアルゴリズムが行わない不具合を観測した旨が表示される。
この例において実現されているプロトコル不具合自動検出方法のフローチャートが図５に示されている。以下、図２をも参照しながら説明する。
【００３２】
同図のステップＳ１０１においては、サーバ２１及びクライアント２２間で送受されるパケットがプロトコル不具合自動検出装置１において取得されることにより、ＴＣＰに従った送受信制御の結果に対応すべきパケットの送受信状態に関する状態情報、本例においては、ｓｎｄ＿ｍａｘ及びｓｎｄ＿ｕｍａが算出される。
【００３３】
同図のステップＳ１０２においては、ＴＣＰに従ってサーバ２１において送受されるパケットに基づいて行われるべき輻輳制御処理が正常に行われた処理結果に対応すべきｓｎｄ＿ｃｗｎｄがプロトコル不具合自動検出装置１において推定される。
同図のステップＳ１０３においては、ステップＳ１０１において算出された状態情報（ｓｎｄ＿ｍａｘ、ｓｎｄ＿ｕｍａ）と、輻輳制御処理の不具合を特徴付ける不具合情報との比較により不具合の検出が行われる。ここで不具合情報は、ステップＳ１０１で算出された状態情報と、ステップＳ１０２で推定された正常情報との関係が規定されるので、上記条件式（１）を満たすか否かの比較が行われる。
【００３４】
上記ステップＳ１０１～ステップＳ１０３の処理が、サーバ２１及びクライアント２２間における通信の間にわたって、繰り返される。
（プロトコルの不具合の検出の具体例）
また、上述の実施例のほかにも、プロトコルの不具合の検出として以下のような例が挙げられる。
【００３５】
例えば、コネクションで送受されるパケットの取得時刻の算出により、次のような不具合を検出することができる。例として、クライアントのＴＣＰからサーバへのＨＴＴＰ（ＨｙｐｅｒＴｅｘｔ　Ｔｒａｎｓｆｅｒ　Ｐｒｏｔｏｃｏｌ）接続処理においては、ＴＣＰコネクションの確立後２秒経てから、ＨＴＴＰ　ＧＥＴ要求パケットが送信されるという不具合がある。通常においては、ＨＴＴＰＧＥＴは、ＴＣＰコネクションの確立直後に送信される。
【００３６】
この場合には、例えば、以下のように不具合を検出する。
まず、例えば上述のプロトコル不具合自動検出装置のコネクション情報算出部において、ａｃｔｉｖｅ　ｏｐｅｎによるコネクション確立を検出し、そのコネクション確立時刻を記録する。すなわち、クライアントからサーバへＳＹＮパケット、これに応答するサーバからクライアントへのＡＣＫ＋ＳＹＮパケット、再びクライアントからサーバへのＡＣＫパケットの３つのパケットがやり取りされるのを検出する。このパケット取得のたびに取得・更新の必要な情報については、コネクション情報算出部にあらかじめ登録しておく。
【００３７】
次に、クライアントからサーバへの最初のパケットの送信を検出し、その最初のデータ送信時刻を記録する。
次に、上述の不具合比較判定部において、例えば、以下の条件式（２）を規定する不具合情報に基づいて、比較判定を行うことにより、ＨＴＴＰ接続処理の不具合を検出することができる。この比較判定は、例えば、最初のデータ送信時刻等の取得・更新を契機として、あるいは、パケット取得の度に行う。
（最初のデータ送信時刻－コネクション確立時刻）＞２秒　・・・（２）
また、次のような不具合を検出することもできる。例えば、ＴＣＰの再送処理について、タイムアウトによる再送の初期値が６０秒となる不具合がある。通常においては、再送処理を行なう通信装置においてラウンドトリップタイムを計測しており、計測したラウンドトリップタイムを基に、逐次タイムアウトの値を計算するため、その結果としてタイムア

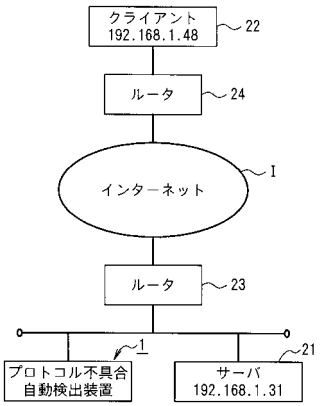
ウトの時刻が６０秒に設定されることはほとんどない。
【００３８】
この場合には、例えば、以下のように再送処理の不具合を検出する。
コネクション情報算出部において、送信データセグメントを取得すると、その送信時刻を記録する。
次に、コネクション情報算出部において、タイムアウトにより再送された送信パケットを取得したら、その再送時刻を記録する。
【００３９】
次に、不具合比較判定部において、不具合情報に基づいて、送信時刻から再送時刻を引き算することにより算出された再送時間が６０秒である場合には不具合と判定する。
更に、例えば、コネクションで送受されるパケットの構成に関する不具合情報により、次のような不具合を検出することができる。例として、ＴＣＰヘッダのオプションフィールドのパディングが必要以上に存在するという不具合（オプションフィールドに設定されるべき情報が設定されていないなど）がある。通常は、オプションデータが設定されるため、ＴＣＰオプションフィールドのパディングは、３バイト以下である。
【００４０】
この場合には、例えば、以下のように不具合を検出する。
まず、コネクション情報算出部又はパケットフィルタ・解析部にて、ＴＣＰヘッダのオプションフィールドを解釈し、オプションデータが占めるバイト数を求める。
次に、不具合比較判定部において、オプションフィールドの総バイト数からオプションデータが占めるバイト数を減算することにより算出される使用されない領域のバイト数が４バイト以上である場合には、不具合と判定し、警告を表示する。
【００４１】
【発明の効果】
以上説明詳細に説明したように、本発明にかかる請求項１に記載のプロトコル不具合自動検出方法、及び、本発明にかかる請求項６に記載のプロトコル不具合自動検出装置によれば、あらかじめ定められたプロトコルにしたがって行われる少なくとも１以上の処理の不具合を特徴付けるデータと、当該不具合データに対応する実際の通信状態と、の比較により、煩雑な処理を行なうことなく不具合が発生している処理を特定することができる。
【００４２】
本発明にかかる請求項２に記載のプロトコル不具合自動検出方法、及び、本発明にかかる請求項７に記載のプロトコル不具合自動検出装置によれば、プロトコルによりあらかじめ定められる正常処理の推定値と、当該推定値に対応する実際のパケットの送受信状態と、を比較するため、より正確に、かつ、汎用的にプロトコルの不具合を検出することができる。また、入力には実際のパケットを用いるので、通信状態に応じて制御内容が変化するような処理の不具合についても検出することができる。
【図面の簡単な説明】
【図１】本実施の形態におけるプロトコル不具合自動検出装置の構成を説明するブロック図である。
【図２】本実施の形態にかかかるプロトコル不具合自動検出装置が用いられるネットワークの全体構成を説明する図である。
【図３】プロトコル不具合自動検出装置において取得するパケットを説明する図である。
【図４】プロトコル自動検出装置におけるパケットの取得に基づいて、プロトコル不具合自動検出装置で算出あるいは推定される情報を説明する図である。
【図５】本実施の形態にかかるプロトコル不具合自動検出方法を説明するフローチャートである。
【図６】従来のプロトコルアナライザの機能ブロック図である。
【図７】従来の解析ツールの機能ブロック図である。
【符号の説明】
１　　プロトコル不具合自動検出装置

１ａ	ネットワークインターフェイス
１ｂ	パケット受信部
１ｃ	解析部
１ｄ	コネクション情報算出部
１ｅ	コネクション情報保存部
１ｆ	正常情報推定部
１ｇ	正常情報保存部
１ｋ	読込部
１ｊ	判定結果出力部
１ｉ	不具合情報保存部
１ｈ	不具合比較判定部
８	プロトコルアナライザ
８ａ	ネットワークインターフェース
８ｂ	パケット受信部
８ｄ	パケット翻訳部
８ｅ	画面出力部
９	解析ツール
２１	サーバ
２２	クライアント
２３、２４	ルータ
Ｉ	インターネット

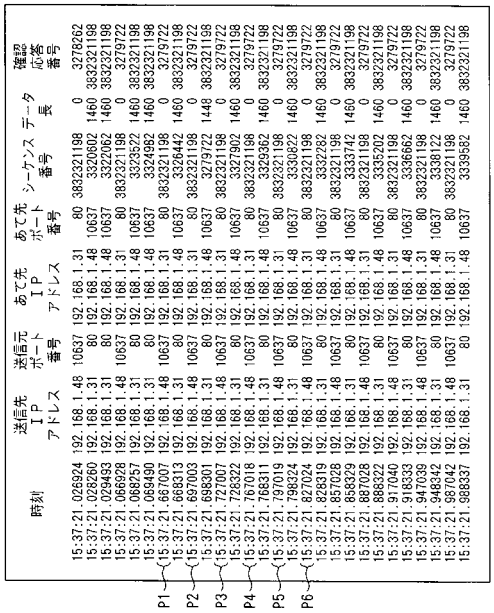
【図１】



【図２】

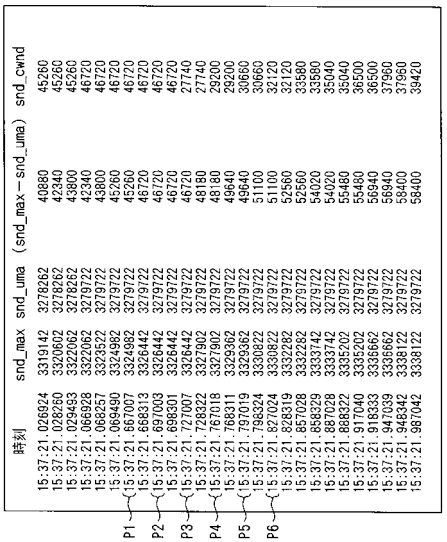


【図３】



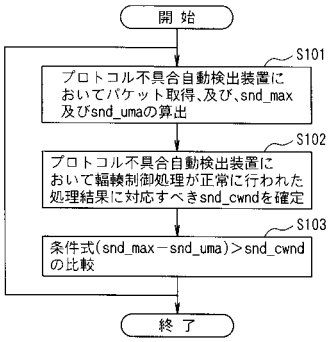
	時刻	送信先IPアドレス	送信元ポート番号	あて先IPアドレス	あて先ポート番号	シーケンス番号	データ長	確認応答番号
P1	15:37:21.026924	192.168.1.48	10637	192.168.1.31	80	3832321198	0	3278262
	15:37:21.028260	192.168.1.31	80	192.168.1.48	10637	3320602	1460	3832321198
	15:37:21.029493	192.168.1.48	10637	192.168.1.31	80	3322062	1460	3832321198
	15:37:21.066928	192.168.1.48	10637	192.168.1.31	80	3832321198	0	3279722
P2	15:37:21.068257	192.168.1.31	80	192.168.1.48	10637	3323522	1460	3832321198
	15:37:21.069490	192.168.1.48	10637	192.168.1.31	80	3832321198	0	3279722
P3	15:37:21.667007	192.168.1.31	80	192.168.1.48	10637	3324982	1460	3832321198
	15:37:21.668313	192.168.1.48	10637	192.168.1.31	80	3326442	1460	3832321198
P4	15:37:21.697003	192.168.1.48	10637	192.168.1.31	80	3832321198	0	3279722
	15:37:21.698301	192.168.1.31	80	192.168.1.48	10637	3279722	1448	3832321198
P5	15:37:21.727007	192.168.1.48	10637	192.168.1.31	80	3832321198	0	3279722
	15:37:21.728322	192.168.1.31	80	192.168.1.48	10637	3327902	1460	3832321198
P6	15:37:21.767018	192.168.1.48	10637	192.168.1.31	80	3832321198	0	3279722
	15:37:21.768311	192.168.1.31	80	192.168.1.48	10637	3329362	1460	3832321198
	15:37:21.797019	192.168.1.48	10637	192.168.1.31	80	3832321198	0	3279722
	15:37:21.798324	192.168.1.48	10637	192.168.1.31	80	3330822	1460	3832321198
	15:37:21.827024	192.168.1.31	80	192.168.1.48	10637	3332282	1460	3832321198
	15:37:21.828319	192.168.1.48	10637	192.168.1.31	80	3832321198	0	3279722
	15:37:21.857028	192.168.1.31	80	192.168.1.48	10637	3333742	1460	3832321198
	15:37:21.858329	192.168.1.48	10637	192.168.1.31	80	3832321198	0	3279722
	15:37:21.887028	192.168.1.31	80	192.168.1.48	10637	3335202	1460	3832321198
	15:37:21.888322	192.168.1.48	10637	192.168.1.31	80	3832321198	0	3279722
	15:37:21.917040	192.168.1.48	10637	192.168.1.31	80	3832321198	0	3279722
	15:37:21.918333	192.168.1.31	80	192.168.1.48	10637	3336662	1460	3832321198
	15:37:21.947039	192.168.1.48	10637	192.168.1.31	80	3338122	1460	3832321198
	15:37:21.948342	192.168.1.31	80	192.168.1.48	10637	3339582	1460	3832321198
	15:37:21.987042	192.168.1.48	10637	192.168.1.31	80	3832321198	0	3279722
	15:37:21.988337	192.168.1.31	80	192.168.1.48	10637	3339582	1460	3832321198

【図４】

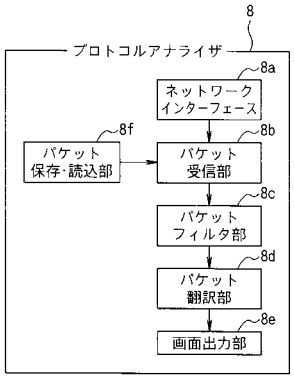


	時刻	snd_max	snd_uma	(snd_max−snd_uma)	snd_cwnd
P1	15:37:21.026924	3319142	3278262	40880	45260
	15:37:21.028260	3320602	3278262	42340	45260
P2	15:37:21.029493	3322062	3278262	43800	45260
	15:37:21.066928	3322062	3279722	42340	46720
P3	15:37:21.068257	3323522	3279722	43800	46720
	15:37:21.069490	3324982	3279722	45260	46720
P4	15:37:21.667007	3324982	3279722	45260	46720
	15:37:21.668313	3326442	3279722	46720	46720
P5	15:37:21.697003	3326442	3279722	46720	46720
	15:37:21.698301	3326442	3279722	46720	46720
P6	15:37:21.727007	3326442	3279722	46720	46720
	15:37:21.728322	3327902	3279722	48180	27740
	15:37:21.767018	3327902	3279722	48180	27740
	15:37:21.768311	3329362	3279722	49640	29200
	15:37:21.797019	3329362	3279722	49640	29200
	15:37:21.798324	3330822	3279722	51100	30660
	15:37:21.827024	3330822	3279722	51100	30660
	15:37:21.828319	3332282	3279722	52560	32120
	15:37:21.857028	3332282	3279722	52560	32120
	15:37:21.858329	3333742	3279722	54020	33580
	15:37:21.887028	3333742	3279722	54020	33580
	15:37:21.888322	3335202	3279722	55480	35040
	15:37:21.917040	3335202	3279722	55480	35040
	15:37:21.918333	3336662	3279722	56940	36500
	15:37:21.947039	3336662	3279722	56940	37960
	15:37:21.948342	3338122	3279722	58400	37960
	15:37:21.987042	3338122	3279722	58400	39420

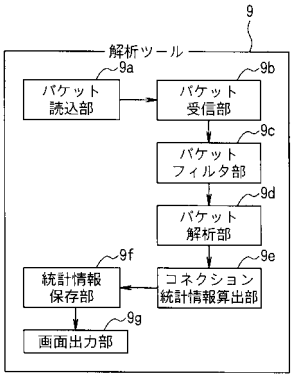
【図５】



【図６】



【図７】



フロントページの続き

(72)発明者　三宅　基治
東京都千代田区永田町二丁目１１番１号　株式会社エヌ・ティ・ティ・ドコモ内
(72)発明者　横田　和久
東京都千代田区永田町二丁目１１番１号　株式会社エヌ・ティ・ティ・ドコモ内
(72)発明者　高橋　修
東京都千代田区永田町二丁目１１番１号　株式会社エヌ・ティ・ティ・ドコモ内

審査官　安藤　一道

(56)参考文献　特開平８－３３１１４６（ＪＰ，Ａ）
特開２００２－１６４８９０（ＪＰ，Ａ）
特開平８－２３７３３４（ＪＰ，Ａ）
特開２００１－３１３６４０（ＪＰ，Ａ）

(58)調査した分野(Int.Cl.，ＤＢ名)
H04L 29/14
H04L 12/56
H04L 29/06