(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第5536962号
(P5536962)
(45)発行日　平成26年7月2日(2014.7.2)　(24)登録日　平成26年5月9日(2014.5.9)

(51)Int.Cl.　F I
H04L 12/28 (2006.01)　H04L 12/28 200

請求項の数 7 (全 15 頁)

(21)出願番号　特願2013-544236(P2013-544236)
(86)(22)出願日　平成24年11月8日(2012.11.8)
(86)国際出願番号　PCT/JP2012/079000
(87)国際公開番号　WO2013/073448
(87)国際公開日　平成25年5月23日(2013.5.23)
審査請求日　平成26年2月12日(2014.2.12)
(31)優先権主張番号　特願2011-250179(P2011-250179)
(32)優先日　平成23年11月15日(2011.11.15)
(33)優先権主張国　日本国(JP)

早期審査対象出願

(73)特許権者　503360115
独立行政法人科学技術振興機構
埼玉県川口市本町四丁目１番８号
(74)代理人　110000338
特許業務法人HARAKENZO WORLD PATENT & TRADEMARK
(72)発明者　河野　健二
日本国千葉県船橋市習志野２－１－５
(72)発明者　山田　浩史
日本国東京都国分寺市東恋ヶ窪３－１３－２２　エクセレンスオザキ２０５

審査官　大石　博見

最終頁に続く

(54)【発明の名称】パケットデータ抽出装置、パケットデータ抽出装置の制御方法、制御プログラム、コンピュータ読み取り可能な記録媒体

(57)【特許請求の範囲】
【請求項１】
通信途中のパケットから目的とするデータを抽出するパケットデータ抽出装置であって、
通信途中のパケットを一時的に記憶する一時記憶部に格納されている対象パケットのペイロードを参照して、当該対象パケットのペイロードに含まれるメッセージのプロシージャ名を確認するプロシージャ名確認手段と、
上記プロシージャ名確認手段によって確認されたプロシージャ名にあらかじめ対応付けられた、目的データの格納位置を示すデータ位置情報に従って、該データ位置情報によって特定される目的パケットのペイロードから目的データを取得する目的データ取得手段と、を備えることを特徴とするパケットデータ抽出装置。
【請求項２】
対象パケットがメッセージの先頭部分を格納しているパケットであるか否かを確認するパケット確認手段をさらに備え、
上記プロシージャ名確認手段は、上記パケット確認手段によって、対象パケットがメッセージの先頭部分を格納していることが確認された場合のみ、当該対象パケットのペイロードに含まれるメッセージのプロシージャ名を確認するものであることを特徴とする請求項１に記載のパケットデータ抽出装置。
【請求項３】
上記目的データ取得手段は、上記データ位置情報によって指定された位置のデータのみ

を取得することを特徴とする請求項１または２に記載のパケットデータ抽出装置。
【請求項４】
ＶＭＭに設けられることを特徴とする請求項１または２に記載のパケットデータ抽出装置。
【請求項５】
通信途中のパケットから目的とするデータを抽出するパケットデータ抽出装置の制御方法であって、
通信途中のパケットを一時的に記憶する一時記憶部に格納されている対象パケットのペイロードを参照して、当該対象パケットのペイロードに含まれるメッセージのプロシージャ名を確認するプロシージャ名確認ステップと、
上記プロシージャ名確認ステップにて確認されたプロシージャ名にあらかじめ対応付けられた、目的データの格納位置を示すデータ位置情報に従って、該データ位置情報によって特定される目的パケットのペイロードから目的データを取得する目的データ取得ステップと、を含むことを特徴とするパケットデータ抽出装置の制御方法。
【請求項６】
請求項１または２に記載のパケットデータ抽出装置の上記各手段としてコンピュータを機能させるための制御プログラム。
【請求項７】
請求項６に記載の制御プログラムを記録したコンピュータ読み取り可能な記録媒体。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明は、通信途中のパケットから目的とするデータを抽出するパケットデータ抽出装置、パケットデータ抽出装置の制御方法、制御プログラム、コンピュータ読み取り可能な記録媒体に関する。
【背景技術】
【０００２】
例えば、ファイルメタデータ改竄型ルートキット（rootkit）は、コンピュータウィルスの一形態であり、感染するとサービスの停止や情報漏洩に繋がり、サービスの品質に深刻な被害を与える。ファイルメタデータ改竄型ルートキットは、オペレーティングシステムカーネル内のデータを変更するため、一般的なアンチウィルスソフトでの検出が極めて困難である。このルートキットを検出するためには、ネットワークシステムを構築して、仮想マシンモニタでやり取りされるパケットを監視することが極めて有効である。この例のように、従来、通信中のパケットから情報を抽出することが行われている。
【０００３】
ここで、図６を参照して、パケットを用いた通常のメッセージ送受信の概略について説明する。
【０００４】
図６に示すように、メッセージの送信の場合、まず、送信側のアプリケーション（図中左側のApplication）が、メッセージを作成し（Ｓ５１）、ＯＳ（operating system）（図中左側のＯＳ）にメッセージ送信をリクエストする（Ｓ５２）。そして、ＯＳが、取得したメッセージをパケットに分割し、ヘッダを付けて（Ｓ５３）、受信側へ送信する（Ｓ５４）。
【０００５】
また、メッセージの受信の場合、受信側のＯＳ（図中右側のＯＳ）が、パケットを受信し（Ｓ５４）、ヘッダを外すとともに、ヘッダに従ってメッセージを作成し（Ｓ５５）、作成したメッセージをアプリケーション（図中右側のApplication）へ送信する（Ｓ５６）。そして、アプリケーションが、ＯＳから取得したメッセージをメモリに格納する（Ｓ５７）。
【０００６】

このように、メッセージの送信側のＯＳは、ＮＩＣ（Network Interface Card）に送る際に、メッセージをパケットに変換する。このとき、ＯＳは、送信先でメッセージの復元する際に必要な情報、例えば、シーケンス番号（順番），ポート番号（接続の識別子）等を含むヘッダをパケットに付与する。一方、メッセージの受信側のＯＳは、パケットのヘッダを参照して、パケットからメッセージを構築する。

【０００７】

つづいて、図７および図８を参照して、パケットを用いて送受信されるメッセージからデータを抽出する従来手法について説明する。ここでは、ＶＭＭ（Virtual Machine Monitor）において、メッセージから目的とするデータを抽出する場合を例に説明する。

【０００８】

図７に示すように、送信側のＯＳ（図中左側のＯＳ）は、ＶＭＭが提供する仮想Network Interface Cardにパケットを送る。つまり、ＶＭＭは、ＯＳによって分割されたパケットを取得する。そして、ＶＭＭは、取得したパケットを受信側のＯＳ（図中右側のＯＳ）へ送信するとともに、取得したパケットからメッセージを再構成して、得たい情報である目的データを得る。

【０００９】

具体的には、図８に示すように、ＶＭＭは、取得したパケットのヘッダ情報を確認し（Ｓ６１）、ヘッダ以外（ペイロード）をコピーし（Ｓ６２）、その後、パケットを送信する（Ｓ６３）。これと同時に、ＶＭＭは、コピーしたデータからメッセージを構築し（Ｓ６４）、メッセージから目的データを抽出する（Ｓ６５）。

【００１０】

このように、ＶＭＭは、パケットのペイロードのコピーを生成して、パケットのヘッダ情報を基に配置する。すなわち、ＶＭＭは、パケットに分割されていたメッセージを再構成した後で、目的データを抽出する。

【先行技術文献】

【非特許文献】

【００１１】

【非特許文献１】TCP Reassembler for Layer7-aware Network Intrusion Detection/Prevention Systems, Miyuki Hanaoka, Makoto Shimamura, and Kenji Kono, IEICE Trans. on Information and Systems, Vol.E90-D, No.12, pp.2019-2032, Dec. 2007

【発明の概要】

【発明が解決しようとする課題】

【００１２】

しかし、上記の従来手法によれば、ＶＭＭは、パケットから必要な情報を抽出する際に、パケットのペイロードのコピーを生成し、当該コピーをパケットのヘッダ情報を基に配置して、メッセージとして構成した後で、当該メッセージから得たい情報を抽出していた。すなわち、ペイロードのコピーを生成し、メッセージとして構成する必要があった。具体的には、図７の例では、ＶＭＭは、目的データ“Ｅ”を取得するために、順次取得した３つのパケットのペイロードのコピーを作成した上で、メッセージに再構成していた。

【００１３】

このように、従来手法によれば、ペイロードのコピーを作成するために、処理に時間がかかるとともに、メッセージの再構成は、ＯＳでも行われる処理であるため、オーバヘッドとなっていた。すなわち、パケットから必要な情報を抽出する際に、大きなオーバヘッドが発生してしまい、稼働しているサービスの品質への影響が大きかった。

【００１４】

本発明は、上記の問題点に鑑みてなされたものであり、その目的は、パケットから必要なデータを効率よく抽出することができるパケットデータ抽出装置、パケットデータ抽出装置の制御方法、制御プログラム、コンピュータ読み取り可能な記録媒体を実現することにある。

【課題を解決するための手段】

【００１５】
上記課題を解決するために、本発明の一態様に係るパケットデータ抽出装置は、通信途中のパケットから目的とするデータを抽出するパケットデータ抽出装置であって、通信途中のパケットを一時的に記憶する一時記憶部に格納されている対象パケットのペイロードを参照して、当該対象パケットのペイロードに含まれるメッセージのプロシージャ名を確認するプロシージャ名確認手段と、上記プロシージャ名確認手段によって確認されたプロシージャ名にあらかじめ対応付けられた、目的データの格納位置を示すデータ位置情報に従って、該データ位置情報によって特定される目的パケットのペイロードから目的データを取得する目的データ取得手段と、を備える。
【００１６】
また、上記課題を解決するために、本発明の一態様に係るパケットデータ抽出装置の制御方法は、通信途中のパケットから目的とするデータを抽出するパケットデータ抽出装置の制御方法であって、通信途中のパケットを一時的に記憶する一時記憶部に格納されている対象パケットのペイロードを参照して、当該対象パケットのペイロードに含まれるメッセージのプロシージャ名を確認するプロシージャ名確認ステップと、上記プロシージャ名確認ステップにて確認されたプロシージャ名にあらかじめ対応付けられた、目的データの格納位置を示すデータ位置情報に従って、該データ位置情報によって特定される目的パケットのペイロードから目的データを取得する目的データ取得ステップと、を含む。
【発明の効果】
【００１７】
それゆえ、本発明の一態様に係るパケットデータ抽出装置およびパケットデータ抽出装置の制御方法によれば、ネットワークを流れる通信パケットすべてをメッセージに変換するのではなく、変換するパケットを適宜取捨選択するため、効率良くファイルのメタデータを取得できるという効果を奏する。また、従来のようにメッセージをコピーしないので、処理が早いという効果を奏する。よって、オーバヘッドがなく、稼働しているサービスが被るオーバヘッドを抑えて、パケットから必要なデータを効率よく抽出できるという効果を奏する。
【図面の簡単な説明】
【００１８】
【図１】本発明の実施形態を示すものであり、パケットデータ抽出装置の構成の詳細を示す機能ブロック図である。
【図２】図１に示したパケットデータ抽出装置の動作を示す模式図である。
【図３】図１に示したパケットデータ抽出装置の処理の流れを示すフローチャートである。
【図４】図１に示したパケットデータ抽出装置を適用した、ファイルメタデータ改竄型ルートキット検知システムの概略を示すブロック図である。
【図５】図４に示したファイルメタデータ改竄型ルートキット検知システムで用いるデータ位置情報の例を示す説明図である。
【図６】従来技術を示すものであり、パケットを用いたメッセージ送受信の概略を示す説明図である。
【図７】従来技術を示すものであり、パケットを用いて送受信されるメッセージからデータを抽出する従来手法を示す模式図である。
【図８】図７に示した従来手法の処理の流れを示すフローチャートである。
【発明を実施するための形態】
【００１９】
以下、本発明の一実施形態について、詳細に説明する。図１～図５に基づいて、本実施形態に係るパケットデータ抽出装置１０について説明すれば以下のとおりである。
【００２０】
（１．装置構成）
図１を参照して、パケットデータ抽出装置１０の構成について説明する。図１は、パケ

ットデータ抽出装置１０の構成の詳細を示す機能ブロック図である。

【００２１】

パケットデータ抽出装置１０は、通信途中のパケットを一時的に記憶するバッファ（一時記憶部）２０に格納されているパケットから、目的とするデータ（目的データ）を抽出する装置である。本実施の形態では、パケットデータ抽出装置１０は、ＶＭＭ（仮想マシンモニタ）１００に、その一部として組み込まれているものとする。なお、パケットデータ抽出装置１０は、パケットの送信側の装置に、パケットを送信するアプリケーションの下位層として設けられてよいし、パケットの受信側の装置に、パケットを受信するアプリケーションの下位層として設けられてよい。また、パケットデータ抽出装置１０は、パケットの送信側の装置とパケットの受信側の装置とを繋ぐネットワーク上に設けられてよい。

【００２２】

本実施の形態では、サーバークライアント間通信に適用した場合を例に説明する。図２において（図１も同様）、図中左側のApplicationがサーバ、図中右側のApplicationがクライアントである。クライアントはサーバへリクエストメッセージを送信し、サーバはクライアントからのリクエストメッセージに呼応して、データメッセージをクライアントへ送信する。なお、本実施の形態では、クライアントからサーバへのリクエストを記述したメッセージを「ＣｔｏＳメッセージ」、「ＣｔｏＳメッセージ」を分割したパケットを「ＳｔｏＣパケット」と記載する。また、リクエストメッセージに呼応してサーバから送信されるデータメッセージを「ＳｔｏＣメッセージ」、「ＳｔｏＣメッセージ」を分割したパケットを「ＳｔｏＣパケット」と記載する。

【００２３】

詳細には、図１に示すように、パケットデータ抽出装置１０は、パケット確認部（パケット確認手段）１１、プロシージャ名確認部（プロシージャ名確認手段）１２、目的データ取得部（目的データ取得手段）１３、位置情報記憶部１４を備えて構成されている。

【００２４】

パケット確認部１１は、対象パケットＰのヘッダを確認することにより、対象パケットＰがメッセージの先頭部分を格納しているパケットであるか否かを確認する。すなわち、パケット確認部１１は、メッセージの先頭部分を格納している先頭パケットＰｈを検出する。特に、パケット確認部１１は、ＣｔｏＳメッセージが分割された対象パケットＰ（ＣｔｏＳパケット）を検出し、検出した対象パケットＰのヘッダを確認する。そして、対象パケットＰがＣｔｏＳメッセージの先頭部分を格納しているパケットであるか否かを確認する。

【００２５】

また、パケット確認部１１は、プロシージャ名確認部１２によって確認されたプロシージャ名にあらかじめ対応付けられた、目的データの格納位置を示すデータ位置情報１４ｂ（後述）に従って、上記ＣｔｏＳメッセージに呼応するＳｔｏＣメッセージが分割された対象パケットＰ（ＳｔｏＣパケット）を検出する。そして、検出した対象パケットＰのヘッダを確認することにより、データ位置情報１４ｂによって特定される目的パケットＰｔを検出する。すなわち、パケット確認部１１は、目的データを含む目的パケットＰｔを検出する。

【００２６】

プロシージャ名確認部１２は、メッセージのプロシージャ名の格納位置を示すプロシージャ名位置情報１４ａ（後述）に従って、バッファ２０に格納されている対象パケットＰ（ＣｔｏＳパケット）のペイロードを参照する。そして、当該対象パケットＰのペイロードに含まれるＣｔｏＳメッセージのプロシージャ名を確認する。なお、本実施の形態では、プロシージャ名を利用するが、メッセージのプロシージャが識別可能であれば、各プロシージャにユニークに割り当てられたＩＤ番号等の他の情報を利用してもよい。

【００２７】

特に、本実施の形態では、パケット確認部１１によって、対象パケットＰ（ＣｔｏＳパ

ケット）がＣｔｏＳメッセージの先頭部分を格納していることが確認された場合のみ、プロシージャ名確認部１２は、当該対象パケットＰ（ＣｔｏＳパケット）のペイロードに含まれるＣｔｏＳメッセージのプロシージャ名を確認するものとする。通常、ＣｔｏＳメッセージのプロシージャ名は、当該ＣｔｏＳメッセージの先頭部分に存在する。そのため、当該ＣｔｏＳメッセージを分割した複数のＣｔｏＳパケットのうちの先頭のＣｔｏＳパケットに含まれることになる。よって、ＣｔｏＳメッセージにプロシージャ名を含むＣｔｏＳパケットを検出するためには、ＣｔｏＳパケットのヘッダを参照して、ＣｔｏＳメッセージの先頭部分を含む先頭パケットＰｈであるか否かを判定すればよい。それゆえ、ヘッダを参照して、ＣｔｏＳメッセージの先頭部分を含まないＣｔｏＳパケットであれば、それ以後の処理を省略できる。よって、プロシージャ名確認部１２は、ＣｔｏＳパケットのヘッダを参照するだけで、ＣｔｏＳメッセージのプロシージャ名を含むＣｔｏＳパケットを検出できるため、効率がよい。

【００２８】

目的データ取得部１３は、ＣｔｏＳメッセージに呼応するＳｔｏＣメッセージが分割されたＳｔｏＣパケットにおいて、プロシージャ名確認部１２によって確認されたプロシージャ名にあらかじめ対応付けられた、目的データの格納位置を示すデータ位置情報１４ｂ（後述）に従って、該データ位置情報１４ｂによって特定される目的パケットＰｔのペイロードから目的データを取得する。特に、本実施の形態では、目的データ取得部１３は、データ位置情報１４ｂによって指定された位置のデータのみを取得するものとする。また、目的データ取得部１３は、目的パケットＰｔのペイロードから目的データを取得する際、ペイロードの先頭からメッセージ変換を行うが、目的データを取得した時点でメッセージ変換を終了する。

【００２９】

位置情報記憶部１４は、プロシージャ名位置情報１４ａおよびデータ位置情報１４ｂをあらかじめ記憶している。

【００３０】

プロシージャ名位置情報１４ａは、ＣｔｏＳメッセージのプロシージャ名の当該ＣｔｏＳメッセージにおける格納位置を示す。なお、本実施の形態では、プロシージャ名のＣｔｏＳメッセージにおける格納位置は、メッセージのプロトコルに応じて決められており、同じプロトコルではプロシージャ間で共通であるものとする。

【００３１】

データ位置情報１４ｂは、プロシージャのプロシージャ名と、当該プロシージャのＳｔｏＣメッセージから取得するデータ（目的データ）のデータ名および当該ＳｔｏＣメッセージにおける格納位置とが対応付けられた情報である。すなわち、データ位置情報１４ｂを参照することで、プロシージャ名に対応付けられた目的データのデータ名と、当該目的データのＳｔｏＣメッセージにおける格納位置と、が分かる。

【００３２】

（２．位置情報）

さらに、図２に示す具体例に沿って、パケットデータ抽出装置１０において、位置情報記憶部１４にあらかじめ与えておく２種類の位置情報（プロシージャ名位置情報１４ａ、データ位置情報１４ｂ）について説明する。図２は、パケットデータ抽出装置１０の動作を示す模式図である。なお、ここでは、メッセージのプロトコルがＮＦＳ（Network File System）プロトコルであり、ファイルメタデータを抽出する場合を例に説明する。また、図２には、クライアントが送信したＣｔｏＳメッセージに呼応するＳｔｏＣメッセージが分割されたＳｔｏＣパケットを、パケットデータ抽出装置１０が処理する過程を示している。なお、クライアントが送信したＣｔｏＳメッセージを処理する過程は、図２では省略されているが、図２の記載と後述する動作の説明から理解できる。

【００３３】

上述のように、パケットデータ抽出装置１０には、プロシージャ名位置情報１４ａおよびデータ位置情報１４ｂがあらかじめ位置情報記憶部１４に記憶されている。

【００３４】
プロシージャ名位置情報１４ａは、プロシージャ名の位置を示す情報（例えば、「メッセージの先頭byte目から」）である。すなわち、メッセージのどの位置にプロシージャ名があるかを示している。具体的には、図２に示すように、「プロシージャ名の位置：80byte目」のように記述できる。
【００３５】
ここで、ＮＦＳプロトコルでは、メッセージ内のプロシージャ名が含まれている位置が決まっている。よって、プロシージャ名確認部１２は、プロシージャ名位置情報１４ａに従ってパケットのペイロードを参照すれば、プロシージャ名を抽出できる。すなわち、プロシージャ名を抽出するために、パケットをすべてスキャンする必要はない。
【００３６】
次に、データ位置情報１４ｂは、プロシージャ名と、取得するデータ名およびその位置とを対応付けた情報である。すなわち、プロシージャ名毎に、どのデータをどの位置から抽出できるかを示している。具体的には、図２の例では、「GETATTRには、Inodeが118byte目に、File Sizeが150byte目に格納されている」等の内容の情報が得られる。
【００３７】
ここで、ＮＦＳプロトコルでは、各プロシージャのメッセージに含まれるデータおよびその位置が決まっている。つまり、プロシージャによって、メッセージに含まれるデータおよびその位置が異なっている。そこで、目的データ取得部１３は、データ位置情報１４ｂに従ってペイロードを参照すれば、データを抽出できる。すなわち、目的とするデータを抽出するために、ペイロードからメッセージを構築する必要はない。
【００３８】
また、パケットのヘッダにはパケットサイズおよびシーケンス番号が含まれている。そのため、各パケットのパケットサイズおよびシーケンス番号から、当該パケットのペイロードのデータがパケットに分割される前のメッセージのどの部分に該当するかが分かる。このことを利用して、パケット確認部１１は目的データを含む目的パケットＰｔを検出する。
【００３９】
また、ＮＦＳプロトコルのメッセージからファイルメタデータを抽出する場合、ファイルメタデータを含むプロシージャのみをデータ位置情報１４ｂに登録しておくことで、不必要なプロシージャのパケットを処理しないようにできる。なお、ＮＦＳプロトコルでは、全２２種類のプロシージャ中、１５種類がファイルメタデータを含むプロシージャであり、そのすべてが１個のパケットによって送信可能な長さのメッセージを有する。すなわち、これら１５種のプロシージャを対象とする場合、常に先頭パケットＰｈにファイルメタデータ（目的データ）が含まれることになる。
【００４０】
（３．動作）
次に、図３を参照して、パケットデータ抽出装置１０の処理の流れを説明する。図３はパケットデータ抽出装置１０の処理の流れを示すフローチャートである。
【００４１】
バッファ２０にはＶＭＭ１００を通過する通信途中のパケットが順次、一時的に格納される。なお、バッファ２０に格納されており、パケットデータ抽出装置１０が処理中のパケットを対象パケットＰと記す。
【００４２】
パケットデータ抽出装置１０は、バッファ２０に格納されている対象パケットＰを順次検出しながら、以下の処理を行う。
【００４３】
まず、パケット確認部１１が、バッファ２０に格納されている対象パケットＰ（ＣｔｏＳパケット）のヘッダを順次確認することにより、ＣｔｏＳメッセージの先頭部分を含むＣｔｏＳパケットの先頭パケットＰｈを検出する（Ｓ１１；先頭パケット確認ステップ）

。
【００４４】
次に、プロシージャ名確認部１２が、プロシージャ名位置情報１４ａに従って、ステップ１１にて検出されたＣｔｏＳメッセージの先頭パケットＰｈのペイロードからメッセージのプロシージャ名を確認する（Ｓ１２；プロシージャ名確認ステップ）。
【００４５】
次に、目的データ取得部１３が、ステップＳ１２にて確認されたプロシージャ名にあらかじめ対応付けられたデータ位置情報１４ｂを、位置情報記憶部１４から取得する（Ｓ１３；データ位置情報取得ステップ）。
【００４６】
次に、ステップＳ１３にて取得されたデータ位置情報１４ｂに従って、パケット確認部１１が、上記ＣｔｏＳメッセージに呼応するＳｔｏＣメッセージが分割されたＳｔｏＣパケットを検出する。そして、パケット確認部１１は、検出したＳｔｏＣパケットから、データ位置情報１４ｂによって特定される位置（目的パケットＰｔの位置）のデータを含む目的パケットＰｔを検出する（Ｓ１４；目的パケット確認ステップ）。
【００４７】
このとき、データ位置情報１４ｂが示す位置がＳｔｏＣメッセージの先頭パケットＰｈのペイロードに含まれるメッセージ内にあれば、ＳｔｏＣメッセージの先頭パケットＰｈが目的パケットＰｔとなる。また、図２に示すように、データ位置情報１４ｂが示す位置が３番目のＳｔｏＣパケットのペイロードに含まれるＳｔｏＣメッセージ内にあれば、３番目のＳｔｏＣパケットが目的パケットＰｔとなる。この場合、パケット確認部１１が、２番目、３番目のＳｔｏＣパケットのヘッダを順次確認して、３番目のＳｔｏＣパケットをパケットＰｔとして検出する。
【００４８】
最後に、目的データ取得部１３が、ステップＳ１４にて検出された目的パケットＰｔのペイロードから目的データを取得する（Ｓ１５；目的データ取得ステップ）。
【００４９】
（４．まとめ）
以上のように、パケットデータ抽出装置１０によれば、仮想マシンモニタと呼ばれるソフトウェアレイヤにおいて、パケットレベルで、ネットワークファイルシステムプロトコルからファイルのメタデータを効率的に抽出することができる。すなわち、パケットデータ抽出装置１０は、ネットワークを流れる通信パケットすべてをメッセージに変換するのではなく、変換するパケットを適宜取捨選択することで、効率良くファイルのメタデータを取得することができる。詳細には、パケットデータ抽出装置１０は、従来のようにＳｔｏＣメッセージをコピーしないので、処理が早い。また、ＳｔｏＣメッセージを構築しないので、オーバヘッドがなく、稼働しているサービスが被るオーバヘッドを抑えることができる。
【００５０】
なお、本実施の形態では、ＮＦＳプロトコルを例に説明したが、これに限定されない。すなわち、適用可能なプロトコルとしては、ＮＦＳプロトコルのように、プロシージャを送信してファイルを操作するプロトコルであればよい。また、プロシージャ毎に、目的とするデータのメッセージにおける位置があらかじめ規定されていればよい。具体例としては、ＮＦＳプロトコルの他、ＦＴＰ（File Transfer Protocol）プロトコル、ＨＴＴＰ（Hyper Text Transfer Protocol）プロトコルが挙げられる。
【００５１】
（５．適用例）
以下、上記パケットデータ抽出装置１０の適用例について説明する。
【００５２】
（５．１．ルートキット検知）
ファイルメタデータ改竄型ルートキットは、コンピュータウィルスの一形態であり、感

染するとサービスの停止や情報漏洩に繋がり、サービスの品質に深刻な被害を与える。ファイルメタデータ改竄型ルートキットは、オペレーティングシステムカーネル内のデータを変更するため、一般的なアンチウィルスソフトでの検出が極めて困難である。なお、VM Watcherは、すでに知られたファイルメタデータ改竄型ルートキットを検知するが、未知のそれには無力である。Strider Ghostbusterは、未知のファイルメタデータ改竄型ルートキットを検知可能であるが、稼働しているサービスの品質への影響は大きい。

【００５３】

ファイルメタデータ改竄型ルートキットによるサービスの停止や情報漏洩は、サービス提供者側にとっては深刻であるため、システムがそれに感染しているかを監視することが必要である。そして、ファイルメタデータ改竄型ルートキットを検出するために、ネットワークシステムを構築して、仮想マシンモニタでやり取りされるパケットを監視することが極めて有効である。ただし、稼働しているサービスが被るオーバヘッドをできるだけ抑えることも必要である。

【００５４】

そこで、ファイルメタデータ改竄型ルートキットの検知機構に、上述したパケットデータ抽出装置１０を組み込むことで、仮想マシン上で動作するアプリケーションやオペレーティングシステムが被るオーバヘッドを抑えつつ、ルートキットの監視を実現することができる。

【００５５】

図４は、パケットデータ抽出装置１０を適用した、ファイルメタデータ改竄型ルートキット検知システムの概略を示すブロック図である。また、図５は、ファイルメタデータ改竄型ルートキット検知システムで用いるデータ位置情報の例を示す説明図である。図５には、上述したＮＦＳプロトコルのプロシージャのうち、ファイルメタデータを含む１５種類のプロシージャが挙げられている。

【００５６】

図４に示すＶＭＭには、受信したファイルのメタデータをパケットから抽出するために、パケットデータ抽出装置１０が組み込まれている。そして、アプリケーションは、パケットデータ抽出装置１０がパケットから抽出したファイルメタデータと、ＯＳが取得したファイルメタデータとの内容を比較する。そして、それら２つのファイルメタデータが一致していなければ、ＯＳがファイルメタデータ改竄型ルートキットに感染していると判断できる。

【００５７】

なお、詳細には、ＶＭ viewが、ファイルシステム関連のシステムコールの引数・返り値（stat(),fstat(),getdent()等）を取得する。一方、ＶＭＭ View（パケットデータ抽出装置１０）が、ネットワークファイルシステムＮＦＳのメッセージからファイルメタデータを取得する。そして、ＶＭ内のviewとＶＭＭ内のviewとを比較して、一致していなければ、ルートキットに感染していると判断する。

【００５８】

このように、パケットデータ抽出装置１０を、ファイルメタデータ改竄型ルートキットの検知機構に組み込むことで、仮想マシン上で動作するアプリケーションやオペレーティングシステムが被るオーバヘッドを抑えつつ、ルートキットの監視を実現することができる。よって、ルートキットが行うサービス停止や情報漏洩といった被害を防止することに大きく貢献することができる。

【００５９】

（５．２．ファイルアクセスモニタ）

パケットデータ抽出装置１０は、ＶＭＭ１００において、ＮＦＳプロトコルを監視するため、ＯＳやアプリケーションを変更することなく、ファイルのアクセス数やアクセスパターンを取得することができる。よって、ファイル配置の最適化や冗長化に有効である。具体的には、近いタイミングにアクセスされるファイル群を同じディレクトリに配置してディスクアクセスを削減したり、アクセス頻度の高いファイルは複製を作成して負荷を分

散するなどの措置が可能となる。

【００６０】

（５．３．ファイルアクセス制御）

パケットデータ抽出装置１０は、ＯＳやアプリケーションの設定を変更することなくファイルへのアクセス制御が可能である。よって、ファイル改竄を防止することができる。すなわち、たとえＯＳがウィルスに犯されたとしても、ＶＭＭは犯されていないので、アクセス禁止に設定しておいたファイルにはアクセスできない。

【００６１】

（６．補足）

最後に、パケットデータ抽出装置１０の各ブロック、特にパケット確認部１１、プロシージャ名確認部１２、データ取得部１３は、ハードウェアロジックによって構成してもよいし、次のようにＣＰＵを用いてソフトウェアによって実現してもよい。

【００６２】

後者の場合、パケットデータ抽出装置１０は、各機能を実現するプログラムの命令を実行するＣＰＵ（central processing unit）、上記プログラムを格納したＲＯＭ（read only memory）、上記プログラムを展開するＲＡＭ（random access memory）、上記プログラムおよび各種データを格納するメモリ等の記憶装置（記録媒体）などを備えている。そして、本発明の目的は、上述した機能を実現するソフトウェアであるパケットデータ抽出装置１０の制御プログラムのプログラムコード（実行形式プログラム、中間コードプログラム、ソースプログラム）をコンピュータで読み取り可能に記録した記録媒体を、上記パケットデータ抽出装置１０に供給し、そのコンピュータ（またはＣＰＵやＭＰＵ）が記録媒体に記録されているプログラムコードを読み出し実行することによっても、達成可能である。

【００６３】

上記記録媒体としては、例えば、磁気テープやカセットテープ等のテープ系、フロッピー（登録商標）ディスク／ハードディスク等の磁気ディスクやＣＤ－ＲＯＭ／ＭＯ／ＭＤ／ＤＶＤ／ＣＤ－Ｒ等の光ディスクを含むディスク系、ＩＣカード（メモリカードを含む）／光カード等のカード系、あるいはマスクＲＯＭ／ＥＰＲＯＭ／ＥＥＰＲＯＭ（登録商標）／フラッシュＲＯＭ等の半導体メモリ系などを用いることができる。

【００６４】

また、パケットデータ抽出装置１０を通信ネットワークと接続可能に構成し、上記プログラムコードを通信ネットワークを介して供給してもよい。この通信ネットワークとしては、特に限定されず、例えば、インターネット、イントラネット、エキストラネット、ＬＡＮ、ＩＳＤＮ、ＶＡＮ、ＣＡＴＶ通信網、仮想専用網（virtual private network）、電話回線網、移動体通信網、衛星通信網等が利用可能である。また、通信ネットワークを構成する伝送媒体としては、特に限定されず、例えば、ＩＥＥＥ１３９４、ＵＳＢ、電力線搬送、ケーブルＴＶ回線、電話線、ＡＤＳＬ回線等の有線でも、ＩｒＤＡやリモコンのような赤外線、Ｂｌｕｅｔｏｏｔｈ（登録商標）、８０２．１１無線、ＨＤＲ、携帯電話網、衛星回線、地上波デジタル網等の無線でも利用可能である。なお、本発明は、上記プログラムコードが電子的な伝送で具現化された、搬送波に埋め込まれたコンピュータデータ信号の形態でも実現され得る。

【００６５】

本発明は上述した実施形態に限定されるものではなく、請求項に示した範囲で種々の変更が可能であり、実施形態に開示された技術的手段を適宜組み合わせて得られる実施形態についても本発明の技術的範囲に含まれる。

【００６６】

以上より、本発明に係るパケットデータ抽出装置は、通信途中のパケットから目的とするデータを抽出するパケットデータ抽出装置であって、通信途中のパケットを一時的に記憶する一時記憶部に格納されている対象パケットのペイロードを参照して、当該対象パケットのペイロードに含まれるメッセージのプロシージャ名を確認するプロシージャ名確認

手段と、上記プロシージャ名確認手段によって確認されたプロシージャ名にあらかじめ対応付けられた、目的データの格納位置を示すデータ位置情報に従って、該データ位置情報によって特定される目的パケットのペイロードから目的データを取得する目的データ取得手段と、を備えることを特徴としている。

【００６７】

また、本発明に係るパケットデータ抽出装置の制御方法は、通信途中のパケットから目的とするデータを抽出するパケットデータ抽出装置の制御方法であって、通信途中のパケットを一時的に記憶する一時記憶部に格納されている対象パケットのペイロードを参照して、当該対象パケットのペイロードに含まれるメッセージのプロシージャ名を確認するプロシージャ名確認ステップと、上記プロシージャ名確認ステップにて確認されたプロシージャ名にあらかじめ対応付けられた、目的データの格納位置を示すデータ位置情報に従って、該データ位置情報によって特定される目的パケットのペイロードから目的データを取得する目的データ取得ステップと、を含むことを特徴としている。

【００６８】

上記の構成によれば、対象パケットのペイロードに含まれるメッセージのプロシージャ名を確認し、該プロシージャ名にあらかじめ対応付けられた、目的データの格納位置を示すデータ位置情報に従って、該データ位置情報によって特定される目的パケットのペイロードから目的データを取得する。

【００６９】

したがって、データ位置情報を参照することによって、プロシージャ名が分かれば、目的データを格納した目的パケットを特定できるため、プロシージャ名を確認した後は、目的パケットの検出と目的パケットのペイロードから目的データの取得を行えばよい。

【００７０】

このように、ネットワークを流れる通信パケットすべてをメッセージに変換するのではなく、変換するパケットを適宜取捨選択するため、効率良くファイルのメタデータを取得できるという効果を奏する。また、従来のようにメッセージをコピーしないので、処理が早いという効果を奏する。よって、オーバヘッドがなく、稼働しているサービスが被るオーバヘッドを抑えて、パケットから必要なデータを効率よく抽出できるという効果を奏する。

【００７１】

さらに、本発明に係るパケットデータ抽出装置は、対象パケットがメッセージの先頭部分を格納しているパケットであるか否かを確認するパケット確認手段をさらに備え、上記プロシージャ名確認手段は、上記パケット確認手段によって、対象パケットがメッセージの先頭部分を格納していることが確認された場合のみ、当該対象パケットのペイロードに含まれるメッセージのプロシージャ名を確認するものであることを特徴としている。

【００７２】

上記の構成によれば、さらに、対象パケットがメッセージの先頭部分を格納していることが確認された場合のみ、当該対象パケットのペイロードに含まれるメッセージのプロシージャ名を確認する。

【００７３】

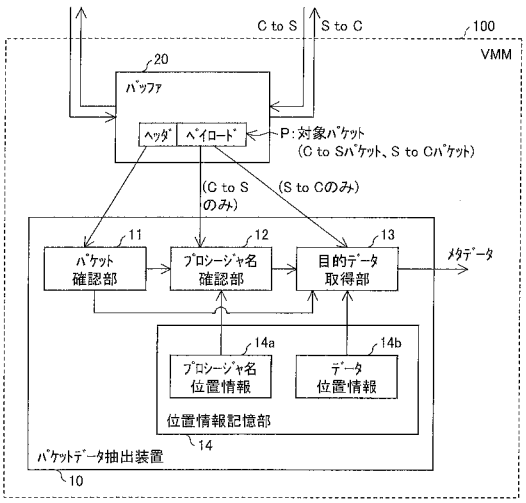
通常、メッセージのプロシージャ名は、当該メッセージの先頭部分に存在するため、当該メッセージを分割した複数のパケットのうちの先頭のパケットに含まれることになる。そこで、メッセージにプロシージャ名を含むパケットを検出するためには、パケットのヘッダを参照して、メッセージの先頭部分を含む先頭パケットであるか否かを判定すればよい。そして、メッセージの先頭部分を含まないパケットであれば、それ以後の処理を省略できる。よって、メッセージのプロシージャ名を含むパケットを効率良く検出できるという効果を奏する。

【００７４】

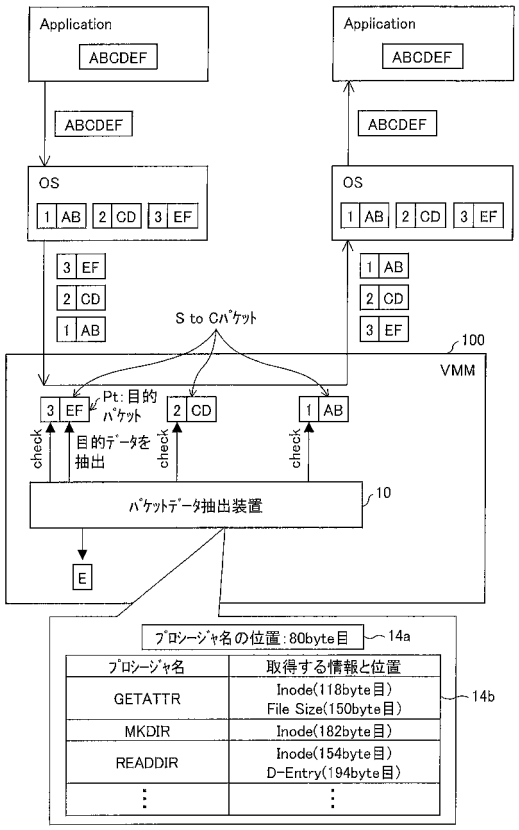
さらに、本発明に係るパケットデータ抽出装置は、上記目的データ取得手段は、上記データ位置情報によって指定された位置のデータのみを取得することを特徴としている。

【００７５】
上記の構成によれば、さらに、データ位置情報には目的データの位置が示されているため、データ位置情報によって指定された位置のデータを取得すれば、目的データが取得できる。よって、データ位置情報によって指定された位置のデータのみを取得することで、パケットから必要なデータを効率よく抽出できるという効果を奏する。
【００７６】
さらに、本発明に係るパケットデータ抽出装置は、ＶＭＭに設けられることを特徴としている。
【００７７】
上記の構成によれば、さらに、ＶＭＭに設けることによって、ＯＳやアプリケーションを変更することなく、ファイルメタデータを取得して、プロトコルを監視することができる。よって、種々の応用が可能となる。
【００７８】
なお、上記パケットデータ抽出装置は、コンピュータによって実現してもよく、この場合には、コンピュータを上記各手段として動作させることにより上記のパケットデータ抽出装置をコンピュータにて実現させる制御プログラム、およびそれを記録したコンピュータ読み取り可能な記録媒体も、本発明の範疇に入る。
【産業上の利用可能性】
【００７９】
本発明のパケットデータ抽出装置は、稼働しているサービスが被るオーバヘッドをできるだけ抑えて、パケットから必要なデータを抽出することができるため、ファイルメタデータ改竄型ルートキット検知、ファイルアクセスモニタ、ファイルアクセス制御等に利用することができる。
【符号の説明】
【００８０】
１０　パケットデータ抽出装置（パケットデータ抽出装置）
１１　パケット確認部（パケット確認手段）
１２　プロシージャ名確認部（プロシージャ名確認手段）
１３　目的データ取得部（目的データ取得手段）
１４ｂ　データ位置情報
２０　バッファ（一時記憶部）
１００　ＶＭＭ
Ｐ　対象パケット
Ｓ１１　先頭パケット確認ステップ
Ｓ１２　プロシージャ名確認ステップ
Ｓ１３　データ位置情報取得ステップ
Ｓ１４　目的パケット確認ステップ
Ｓ１５　目的データ取得ステップ

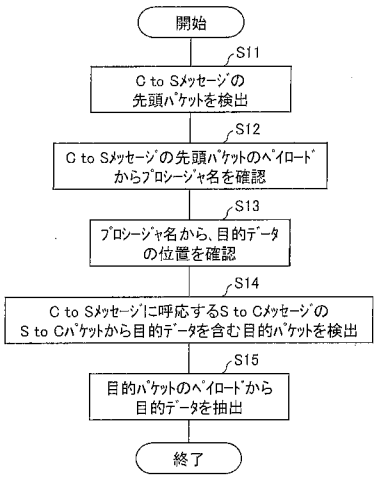
【図１】



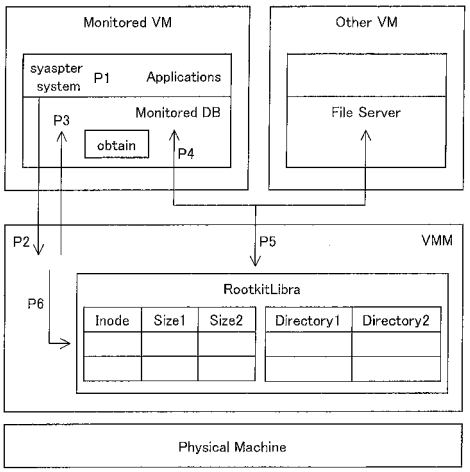
【図２】



【図３】



【図４】

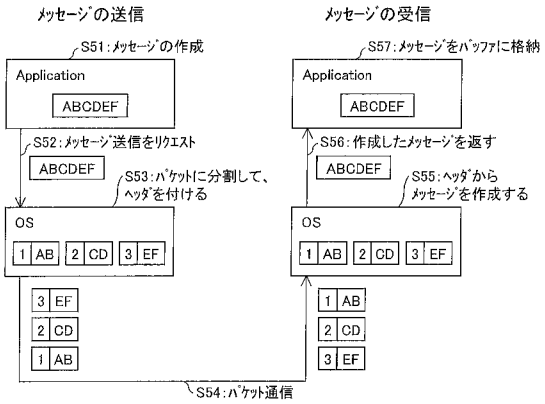


P1.system call
P2.fault to VMM
P3.VMM emulate instruction & give control to OS
P4.OS requests and gets filesystem Info. from the server
P5.RootkitLibra gets Trusted View
P6.RootkitLibra gets Untrusted View

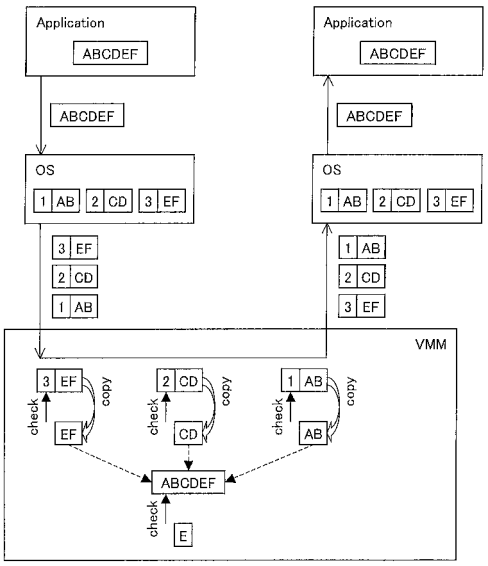
【図５】

ﾌﾟﾛｼｰｼﾞｬ名	取得する情報
GETATTR	Inode と File Size
SETATTR	Inode と File Size
LOOKUP	Inode と File Size
ACCESS	Inode と File Size
READLINK	Inode と File Size
READ	Inode と File Size
WRITE	Inode と File Size
CREAT	Inode
MKDIR	Inode
SYMLINK	Inode と File Size
REMOVE	Inode
RMDIR	Inode
READDIR	Inode と Directory Entry
READDIRPLUS	Inode と Directory Entry
COMMIT	Inode と File Size

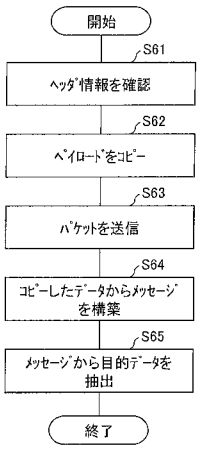
【図６】



【図７】



【図８】



フロントページの続き

(56)参考文献　特開２０１１－７０５４９（ＪＰ，Ａ）
特開２００９－４９９７２（ＪＰ，Ａ）

(58)調査した分野(Int.Cl.，ＤＢ名)
Ｈ０４Ｌ　１２／２８