(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第5692244号
(P5692244)
(45)発行日　平成27年4月1日(2015.4.1)　(24)登録日　平成27年2月13日(2015.2.13)

(51)Int.Cl.　F I
G06F 21/55 (2013.01)　G06F 21/55 320

請求項の数 11 (全 34 頁)

(21)出願番号　特願2012-555602(P2012-555602)
(86)(22)出願日　平成23年1月31日(2011.1.31)
(86)国際出願番号　PCT/JP2011/051965
(87)国際公開番号　WO2012/104978
(87)国際公開日　平成24年8月9日(2012.8.9)
審査請求日　平成25年8月6日(2013.8.6)

(73)特許権者　000005223
富士通株式会社
神奈川県川崎市中原区上小田中４丁目１番１号
(74)代理人　100104190
弁理士　酒井　昭徳
(72)発明者　伊豆　哲也
神奈川県川崎市中原区上小田中４丁目１番１号　富士通株式会社内
(72)発明者　武仲　正彦
神奈川県川崎市中原区上小田中４丁目１番１号　富士通株式会社内
(72)発明者　古川　和快
神奈川県川崎市中原区上小田中４丁目１番１号　富士通株式会社内
最終頁に続く

(54)【発明の名称】通信方法、ノード、およびネットワークシステム

(57)【特許請求の範囲】
【請求項１】
複数のノードを有するアドホックネットワーク内のノードが、
前記アドホックネットワーク内の前記ノードの近隣ノードの送信元アドレスと前記近隣ノードからの第１のパケット送信回数とを含む第１のパケットを受信する第１の受信工程と、
前記第１の受信工程によって受信された第１のパケットの中から、前記第１のパケット送信回数を抽出する第１の抽出工程と、
前記第１の受信工程によって前記第１のパケットが受信された後、前記近隣ノードの送信元アドレスと前記近隣ノードからの第２のパケット送信回数とを含む第２のパケットを受信する第２の受信工程と、
前記第２の受信工程によって受信された第２のパケットの中から、前記第２のパケット送信回数を抽出する第２の抽出工程と、
前記第１の抽出工程によって抽出された前記第１のパケット送信回数と、前記第２の抽出工程によって抽出された前記第２のパケット送信回数と、に基づいて、前記第２のパケットが不正パケットか否かを判定する判定工程と、
前記判定工程によって不正パケットと判定された場合、前記第２のパケットを廃棄する廃棄工程と、
前記ノードの記憶装置に保持された前記近隣ノードからのパケット送信回数の要求パケットを、前記近隣ノードから受信する第３の受信工程と、

前記第３の受信工程によって前記要求パケットを受信したことに応じて、前記近隣ノードからのパケット送信回数を含む応答パケットを作成する作成工程と、
前記作成工程によって作成された応答パケットを、前記近隣ノードに送信する送信工程と、
を実行することを特徴とする通信方法。
【請求項２】
複数のノードを有するアドホックネットワーク内のノードが、
前記アドホックネットワーク内の前記ノードの近隣ノードへのパケット送信イベントを検出する検出工程と、
前記検出工程によって前記パケット送信イベントが検出された場合、前記ノードの記憶装置に保持された前記ノードからのパケット送信回数を、前記パケット送信回数に１を加算した値に更新する更新工程と、
前記更新工程によって更新されたパケット送信回数を含むパケットを前記近隣ノードに送信する送信工程と、
前記ノードの記憶装置に保持された前記ノードからのパケット送信回数の消去を検知する検知工程と、
前記検知工程によって消去が検知された場合に、前記近隣ノードの記憶装置に保持された前記ノードからのパケット送信回数の要求パケットを、前記近隣ノードに通知する通知工程と、
前記通知工程によって前記要求パケットが通知された結果、前記近隣ノードから送信されてくる前記ノードからのパケット送信回数を含む応答パケットを受信する受信工程と、
前記受信工程によって受信された応答パケットの中から、前記ノードからのパケット送信回数を抽出する抽出工程と、
前記抽出工程によって抽出された前記ノードからのパケット送信回数を前記ノードの記憶装置に格納する格納工程と、
を実行することを特徴とする通信方法。
【請求項３】
前記アドホックネットワーク内の各ノードが有する共通鍵を用いて、前記要求パケットを暗号化する暗号化工程と、
前記近隣ノードによって前記共通鍵を用いて暗号化された前記応答パケットから、前記共通鍵を用いて前記応答パケットを復号する復号工程と、を実行し、
前記通知工程は、
前記暗号化工程によって暗号化された前記要求パケットを、前記近隣ノードに通知し、
前記受信工程は、
前記近隣ノードから送信されてくる前記共通鍵を用いて暗号化された前記応答パケットを受信し、
前記復号工程は、
前記受信工程によって受信された暗号化された前記応答パケットから、前記応答パケットを復号し、
前記抽出工程は、
前記復号工程によって復号された前記応答パケットの中から、前記ノードからのパケット送信回数を抽出することを特徴とする請求項２に記載の通信方法。
【請求項４】
前記通知工程は、
前記ノード固有の第１の識別情報を含む暗号化された前記要求パケットを前記近隣ノードに通知し、
前記受信工程は、
第２の識別情報を含む前記共通鍵を用いて暗号化された前記応答パケットを前記近隣ノードから受信し、
前記抽出工程は、

前記受信工程によって受信された暗号化された前記応答パケットから前記復号工程によって復号された前記応答パケットの中から、前記第２の識別情報を抽出し、
前記格納工程は、
前記第１の識別情報と前記抽出工程によって抽出された第２の識別情報とが一致した場合に、前記抽出工程によって抽出された前記ノードからのパケット送信回数を前記ノードの記憶装置に格納することを特徴とする請求項２または３に記載の通信方法。
【請求項５】
前記第１の識別情報となる乱数を生成する生成工程を実行することを特徴とする請求項４に記載の通信方法。
【請求項６】
前記検知工程によって消去が検知された後に、前記アドホックネットワーク内の各ノードが有する第１の共通鍵を用いて暗号化された前記近隣ノードが有する第２の共通鍵を、前記近隣ノードから受信する鍵受信工程と、
前記鍵受信工程によって受信された暗号化された前記第２の共通鍵から、前記第１の共通鍵を用いて前記第２の共通鍵を復号する鍵復号工程と、
前記鍵復号工程によって復号された前記第２の共通鍵を用いて、前記要求パケットを暗号化する暗号化工程と、
前記近隣ノードによって前記第２の共通鍵を用いて暗号化された前記応答パケットから、前記鍵復号工程によって復号された前記第２の共通鍵を用いて前記応答パケットを復号する復号工程と、を実行し、
前記通知工程は、
前記暗号化工程によって暗号化された前記要求パケットを、前記近隣ノードに通知し、
前記受信工程は、
前記近隣ノードから送信されてくる前記第２の共通鍵を用いて暗号化された前記応答パケットを受信し、
前記復号工程は、
前記受信工程によって受信された暗号化された前記応答パケットから、前記応答パケットを復号し、
前記抽出工程は、
前記復号工程によって復号された前記応答パケットの中から、前記ノードからのパケット送信回数を抽出することを特徴とする請求項２に記載の通信方法。
【請求項７】
前記通知工程は、
前記ノード固有の第１の識別情報を含む暗号化された前記要求パケットを前記近隣ノードに通知し、
前記受信工程は、
第２の識別情報を含む前記第２の共通鍵を用いて暗号化された前記応答パケットを前記近隣ノードから受信し、
前記抽出工程は、
前記受信工程によって受信された暗号化された前記応答パケットから前記復号工程によって復号された前記応答パケットの中から、前記第２の識別情報を抽出し、
前記格納工程は、
前記第１の識別情報と前記抽出工程によって抽出された第２の識別情報とが一致した場合に、前記抽出工程によって抽出された前記ノードからのパケット送信回数を前記ノードの記憶装置に格納することを特徴とする請求項２または６に記載の通信方法。
【請求項８】
前記第１の識別情報となる乱数を生成する生成工程を実行することを特徴とする請求項７に記載の通信方法。
【請求項９】
複数のノードを有するアドホックネットワーク内のノードであって、

前記アドホックネットワーク内の前記ノードの近隣ノードの送信元アドレスと前記近隣ノードからの第１のパケット送信回数とを含む第１のパケットを受信する第１の受信手段と、
前記第１の受信手段によって受信された第１のパケットの中から、前記第１のパケット送信回数を抽出する第１の抽出手段と、
前記第１の受信手段によって前記第１のパケットが受信された後、前記近隣ノードの送信元アドレスと前記近隣ノードからの第２のパケット送信回数とを含む第２のパケットを受信する第２の受信手段と、
前記第２の受信手段によって受信された第２のパケットの中から、前記第２のパケット送信回数を抽出する第２の抽出手段と、
前記第１の抽出手段によって抽出された前記第１のパケット送信回数と、前記第２の抽出手段によって抽出された前記第２のパケット送信回数と、に基づいて、前記第２のパケットが不正パケットか否かを判定する判定手段と、
前記判定手段によって不正パケットと判定された場合、前記第２のパケットを廃棄する廃棄手段と、
前記ノードの記憶装置に保持された前記近隣ノードからのパケット送信回数の要求パケットを、前記近隣ノードから受信する第３の受信手段と、
前記第３の受信手段によって前記要求パケットを受信したことに応じて、前記近隣ノードからのパケット送信回数を含む応答パケットを作成する作成手段と、
前記作成手段によって作成された応答パケットを、前記近隣ノードに送信する送信手段と、
を備えることを特徴とするノード。
【請求項１０】
複数のノードを有するアドホックネットワーク内のノードであって、
前記アドホックネットワーク内の前記ノードの近隣ノードへのパケット送信イベントを検出する検出手段と、
前記検出手段によって前記パケット送信イベントが検出された場合、前記ノードの記憶装置に保持された前記ノードからのパケット送信回数を、前記パケット送信回数に１を加算した値に更新する更新手段と、
前記更新手段によって更新されたパケット送信回数を含むパケットを前記近隣ノードに送信する送信手段と、
前記ノードの記憶装置に保持された前記ノードからのパケット送信回数の消去を検知する検知手段と、
前記検知手段によって消去が検知された場合に、前記近隣ノードの記憶装置に保持された前記ノードからのパケット送信回数の要求パケットを、前記近隣ノードに通知する通知手段と、
前記通知手段によって前記要求パケットが通知された結果、前記近隣ノードから送信されてくる前記ノードからのパケット送信回数を含む応答パケットを受信する受信手段と、
前記受信手段によって受信された応答パケットの中から、前記ノードからのパケット送信回数を抽出する抽出手段と、
前記抽出手段によって抽出された前記ノードからのパケット送信回数を前記ノードの記憶装置に格納する格納手段と、
を備えることを特徴とするノード。
【請求項１１】
複数のノードから構成されるアドホックネットワーク内のノードと、前記アドホックネットワーク内の前記ノードの近隣ノードと、を含むネットワークシステムにおいて、
前記ノードが、
前記アドホックネットワーク内の前記ノードの近隣ノードへのパケット送信イベントを検出する検出手段と、
前記検出手段によって前記パケット送信イベントが検出された場合、前記ノードの記憶

装置に保持された前記ノードからの第１のパケット送信回数を、前記第１のパケット送信回数に１を加算した値に更新する更新手段と、
前記更新手段によって更新された第１のパケット送信回数を含む第１のパケットを前記近隣ノードに送信する第１の送信手段と、
前記ノードの記憶装置に保持された前記ノードからのパケット送信回数の消去を検知する検知手段と、
前記検知手段によって消去が検知された場合に、前記近隣ノードの記憶装置に保持された前記ノードからのパケット送信回数の要求パケットを、前記近隣ノードに通知する通知手段と、
前記通知手段によって前記要求パケットが通知された結果、前記近隣ノードから送信されてくる前記ノードからのパケット送信回数を含む応答パケットを受信する第１の受信手段と、
前記第１の受信手段によって受信された応答パケットの中から、前記ノードからのパケット送信回数を抽出する第１の抽出手段と、
前記第１の抽出手段によって抽出された前記ノードからのパケット送信回数を前記ノードの記憶装置に格納する格納手段と、を備え、
前記近隣ノードが、
前記第１の送信手段によって前記ノードから送信された前記第１のパケットを受信する第２の受信手段と、
前記第２の受信手段によって受信された第１のパケットの中から、前記第１のパケット送信回数を抽出する第２の抽出手段と、
前記第２の受信手段によって前記第１のパケットが受信された後、前記ノードの送信元アドレスと前記ノードからの第２のパケット送信回数とを含む第２のパケットを受信する第３の受信手段と、
前記第３の受信手段によって受信された第２のパケットの中から、前記第２のパケット送信回数を抽出する第３の抽出手段と、
前記第２の抽出手段によって抽出された前記第１のパケット送信回数と、前記第３の抽出手段によって抽出された前記第２のパケット送信回数と、に基づいて、前記第２のパケットが不正パケットか否かを判定する判定手段と、
前記判定手段によって不正パケットと判定された場合、前記第２のパケットを廃棄する廃棄手段と、
前記通知手段によって通知された前記近隣ノードの記憶装置に保持された前記ノードからのパケット送信回数の要求パケットを、前記ノードから受信する第３の受信手段と、
前記第３の受信手段によって前記要求パケットを受信したことに応じて、前記ノードからのパケット送信回数を含む応答パケットを作成する作成手段と、
前記作成手段によって作成された応答パケットを、前記ノードに送信する第２の送信手段と、
を備えることを特徴とするネットワークシステム。

【発明の詳細な説明】

【技術分野】

【０００１】

本発明は、通信をおこなう通信方法、ノード、およびネットワークシステムに関する。

【背景技術】

【０００２】

アドホックネットワークは、無線通信でリンクする自己構成型のネットワークの一種である。アドホックネットワークは複数のノードにより構成される。また、アドホックネットワーク内の各ノードは、マルチホップ通信によりパケットの送受信をおこなう。マルチホップ通信は、互いの通信圏内に存在しないノード同士が、各ノードの通信圏内に存在する別のノードを介して通信をおこなう技術である。

【０００３】

アドホックネットワークを利用した技術として、各家庭の電力メータに無線通信可能なノードを組み込んで、作業員が現地に出向くことなく、アドホックネットワーク経由でメータ検針などの業務をおこなうシステムがある。各家庭の電力の使用量などの個人情報を扱うアドホックネットワークでは、秘匿性、改ざん防止などの観点からセキュアな通信をおこなうことが要求される。そこで、従来のシステムでは、アドホックネットワーク内のノード間で送受信されるパケットを暗号化することで、セキュアな通信の確保がおこなわれている（例えば、下記特許文献１～４参照。）。

【０００４】

パケットが暗号化されている場合であっても、アドホックネットワークでは、過去にアドホックネットワーク内に送信された正規パケットが攻撃者によりキャプチャされうる。そして、攻撃者は、キャプチャした正規パケットをアドホックネットワーク内に再送することで、ネットワークを輻輳させる攻撃（再送攻撃）をすることができる。そこで、アドホックネットワークは、ネットワークとしての通信品質を確保するために、再送攻撃に備える必要がある。

【０００５】

従来のシステムでは、パケットを送信するノードは、パケットの送信時刻をパケット内に格納するようにしていた。そして、パケットを受信したノードは、自ノードの時刻と、受信したパケット内の送信時刻とを比較し、両者が大きくかけ離れている場合には、再送攻撃がおこなわれたと見なし、そのパケットを廃棄するという技術があった。

【先行技術文献】

【特許文献】

【０００６】

【特許文献１】特開２００３－３４８０７２号公報

【特許文献２】特開２０１０－９８５９７号公報

【特許文献３】特開２００７－８８７９９号公報

【特許文献４】特開２００９－８１８５４号公報

【発明の概要】

【発明が解決しようとする課題】

【０００７】

しかしながら、上述した従来技術では、アドホックネットワーク内の全ノードの時刻が同期されている必要がある。例えば、他のネットワークとの中継機器であるゲートウェイまたはアドホックネットワーク内の特定のノードが、定期的に時刻同期用のパケットをブロードキャストすることで、アドホックネットワーク内の全ノードの時刻同期を実現する。

【０００８】

そのため、メータ検針をおこなうアドホックネットワークでは、各メータの検針データパケット以外に、時刻同期用のブロードキャストパケットが送受信されることとなり、アドホックネットワークの通信負荷の増大を招くという問題があった。

【０００９】

本発明は、上述した従来技術による問題点を解消するため、再送攻撃を簡単かつ効率的に検出することにより、通信負荷の低減を図ることができる通信方法、ノード、およびネットワークシステムを提供することを目的とする。

【課題を解決するための手段】

【００１０】

上述した課題を解決し、目的を達成するため、本発明の一観点によれば、複数のノードを有するアドホックネットワーク内のノードが、アドホックネットワーク内の近隣ノードの送信元アドレスと近隣ノードからの第１のパケット送信回数とを含む第１のパケットを受信し、受信された第１のパケットの中から、第１のパケット送信回数を抽出し、第１のパケットが受信された後、近隣ノードの送信元アドレスと近隣ノードからの第２のパケット送信回数とを含む第２のパケットを受信し、受信された第２のパケットの中から、第２

のパケット送信回数を抽出し、抽出した第１のパケット送信回数と第２のパケット送信回数とに基づいて、第２のパケットが不正パケットか否かを判定し、不正パケットと判定された場合、第２のパケットを廃棄する通信方法およびノードが提案される。

【００１１】

上述した課題を解決し、目的を達成するため、本発明の一観点によれば、複数のノードを有するアドホックネットワーク内のノードが、アドホックネットワーク内の近隣ノードへのパケット送信イベントを検出した場合に、記憶装置に保持された自ノードからのパケット送信回数を、パケット送信回数に１を加算した値に更新し、更新されたパケット送信回数を含むパケットを近隣ノードに送信する通信方法およびノードが提案される。

【００１２】

上述した課題を解決し、目的を達成するため、本発明の一観点によれば、複数のノードから構成されるアドホックネットワーク内のノードが、アドホックネットワーク内の近隣ノードへのパケット送信イベントを検出し、パケット送信イベントが検出された場合、記憶装置に保持された自ノードからの第１のパケット送信回数を、第１のパケット送信回数に１を加算した値に更新し、更新された第１のパケット送信回数を含む第１のパケットを近隣ノードに送信し、近隣ノードが、第１のパケットを受信し、受信された第１のパケットの中から、第１のパケット送信回数を抽出し、第１のパケットが受信された後、第１のパケットと同一の送信元アドレスと第２のパケット送信回数とを含む第２のパケットを受信し、受信された第２のパケットの中から、第２のパケット送信回数を抽出し、第１のパケット送信回数と、第２のパケット送信回数と、に基づいて、第２のパケットが不正パケットか否かを判定し、不正パケットと判定された場合、第２のパケットを廃棄するネットワークシステムが提案される。

【発明の効果】

【００１３】

本発明の一観点によれば、再送攻撃を簡単かつ効率的に検出することにより、通信負荷の低減を図ることができるという効果を奏する。

【図面の簡単な説明】

【００１４】

【図１】図１は、実施の形態にかかるノードによる受信したパケットの正当性判定の具体例を示す説明図である。

【図２】図２は、実施の形態にかかるネットワークシステムの一実施例を示す説明図である。

【図３】図３は、実施の形態にかかるノードＮのハードウェア構成例を示すブロック図である。

【図４】図４は、図１に示したノードＤＢ１１０の記憶内容の一例を示す説明図である。

【図５】図５は、ノードＮの受信側としての機能的構成を示す機能ブロック図である。

【図６】図６は、ノードＮの送信側としての機能的構成を示す機能ブロック図である。

【図７】図７は、ネットワークシステム２００におけるアクセス鍵ＡＫを用いた暗号化通信を示す説明図である。

【図８】図８は、ネットワークシステム２００におけるアクセス鍵ＡＫを用いた暗号化通信の詳細を示す説明図である。

【図９】図９は、ネットワークシステム２００におけるアクセス鍵ＡＫの更新例を示す説明図である。

【図１０】図１０は、ネットワークシステム２００への新規ノードの導入例を示す説明図である。

【図１１】図１１は、暗号化通信におけるパケット送信処理の詳細を示すフローチャートである。

【図１２】図１２は、暗号化通信におけるパケット受信処理の詳細を示すフローチャートである。

【図１３】図１３は、ノードの送信カウンタ値が消去された場合の動作例１を示す説明図

である。
【図１４】図１４は、図１３に示したノードの送信カウンタ値が消去された場合の動作例１のシーケンス図である。
【図１５】図１５は、図１３および図１４に示した例における送信カウンタ値更新処理の詳細を示すフローチャートである。
【図１６】図１６は、ノードの送信カウンタ値が消去された場合の動作例２を示す説明図である。
【図１７】図１７は、図１６に示したノードの送信カウンタ値が消去された場合の動作例２のシーケンス図である。
【図１８】図１８は、図１６および図１７に示した例における送信カウンタ値更新処理の詳細を示すフローチャートである。
【図１９】図１９は、送信カウンタ値ＤＢ１９００の記憶内容の一例を示す説明図である。
【図２０】図２０は、実施例２にかかるネットワークシステム２００におけるアクセス鍵ＡＫを用いた暗号化通信を示す説明図である。
【発明を実施するための形態】
【００１５】
以下に添付図面を参照して、この発明にかかる通信方法、ノード、およびネットワークシステムの実施の形態を詳細に説明する。本実施の形態では、各ノードは、時刻情報ではなく、自ノードからのパケット送信回数を示すカウンタ値を用いる。各ノードが時刻情報を用いる場合、絶対的な基準時刻との同期をネットワークを介しておこなう必要があり、ネットワークに通信負荷をかける。一方、各ノードがカウンタ値を用いる場合、カウンタ値は自ノード内の処理のみで更新が可能な相対的な値であるから、ネットワークへの通信負荷を削減できる。
【００１６】
（実施の形態にかかるノードによる受信したパケットの正当性判定の具体例）
図１は、実施の形態にかかるノードによる受信したパケットの正当性判定の具体例を示す説明図である。図１において、アドホックネットワークＡは複数のノードＮ（ここでは、ノードＮ１およびノードＮ２）から構成される。ノードＮ１とノードＮ２は互いの通信圏内に存在する近隣ノードであり、パケットＰを暗号化した暗号化パケットＥＰの送受信をおこなっている。ここで、各ノードＮは、受信した暗号化パケットＥＰの正当性判定をおこない、再送攻撃による不正パケットと判定された場合、暗号化パケットＥＰを廃棄することにより、通信の品質を確保する。
【００１７】
各ノードＮは、今までに自ノードが暗号化パケットＥＰを送信した回数を示す送信カウンタ値を保持している。この送信カウンタ値は、各ノードＮが暗号化パケットＥＰを送信する際にインクリメントされる。そして、各ノードＮは、インクリメントした値を送信する暗号化パケットＥＰに含ませておく。また、各ノードＮは、自ノードが受信したパケットに含まれていた送信カウンタ値を受信カウンタ値として、受信したパケットの送信元ノードと関連付けて、ノードデータベース（以下、「ノードＤＢ（ＤａｔａＢａｓｅ）１１０」という）に保持している。
【００１８】
そのため、各ノードＮは、送信カウンタ値とノードＤＢ１１０に保持した受信カウンタ値とに基づいて、受信した暗号化パケットＥＰの正当性判定をおこなうことができる。具体的には、ノードＮが、ある近隣ノードからの正規の暗号化パケットＥＰを受信し続けている限り、今回受信した暗号化パケットＥＰ内の送信カウンタ値は、前回受信した暗号化パケットＥＰ内の送信カウンタ値（すなわち、受信カウンタ値）より大きくなる。
【００１９】
一方で、再送攻撃では過去に送信された暗号化パケットＥＰをそのまま送信しているため、ノードＮが今回受信した暗号化パケットＥＰ内の送信カウンタ値は、前回受信した暗

号化パケットＥＰ内の送信カウンタ値（すなわち、受信カウンタ値）以下となる。よって、ノードＮは、送信カウンタ値と受信カウンタ値を比較することにより、送信攻撃による不正パケットを判定することができる。

【００２０】

例えば、図１を用いて、ノードＮ２を暗号化パケットＥＰの送信側のノードＮとし、ノードＮ１を暗号化パケットＥＰの受信側のノードＮとする例を挙げて、受信した暗号化パケットＥＰの正当性判定の具体例について説明する。

【００２１】

ここで、各ノードＮは、自ノードへ送信されるパケットＰの暗号化に使用される共通鍵（以下、「アクセス鍵」という）を、他ノードに送信している。アクセス鍵を受信した各ノードＮは、受信したアクセス鍵を送信元ノードに関連付けて保持しておく。そして、各ノードＮは、該送信元ノードを宛先とするパケットＰを送信する際には、関連付けられたアクセス鍵を用いてパケットＰを暗号化して暗号化パケットＥＰにして送信する。送信元ノードに関連付けたアクセス鍵は、受信カウンタ値とともにノードＤＢ１１０に記憶されてもよい。

【００２２】

具体的には、ノードＮ１は、ノードＮ１へ送信するパケットＰの暗号化に用いるアクセス鍵ＡＫ１をノードＮ２に送信する。ノードＮ２は、ノードＮ１から受信したアクセス鍵ＡＫ１を保持し、ノードＮ１にパケットＰを送信する際には、アクセス鍵ＡＫ１を用いてパケットＰを暗号化して暗号化パケットＥＰにして送信する。一方、暗号化パケットＥＰを受信したノードＮ１は、アクセス鍵ＡＫ１を用いて暗号化パケットＥＰを復号する。

【００２３】

図１の（Ａ）に示すように、（１）ノードＮ２は、１１回目に暗号化パケットＥＰを送信する際に、自ノード内の記憶装置に記憶されている送信カウンタ値を「１０」から「１１」へと更新する。そして、（２）ノードＮ２は、送信するパケットＰに更新した送信カウンタ値「１１」を含めて暗号化した暗号化パケットＥＰをノードＮ１に送信する。

【００２４】

（３）暗号化パケットＥＰを受信したノードＮ１は、暗号化パケットＥＰを復号し、復号したパケットＰから抽出した送信カウンタ値「１１」と、自ノード内のノードＤＢ１１０に保持された受信カウンタ値「１０」とを比較する。この場合、ノードＮ１は、過去にノードＮ２が１０回目に送信した暗号化パケットＥＰを受け取ったことがあり、今回受信した暗号化パケットＥＰはノードＮ２が１１回目に送信したパケットであることが分かる。

【００２５】

そのため、受信した暗号化パケットＥＰが正規の暗号化パケットＥＰであると判定できる。また、ノードＮ１は、自ノード内のノードＤＢ１１０に、送信カウンタ値「１１」を、ノードＮ２に関連付けたあらたな受信カウンタ値として保持する。

【００２６】

（４）ノードＮ２は、１２回目に暗号化パケットＥＰを送信する際に、自ノード内の記憶装置に記憶されている送信カウンタ値を「１１」から「１２」へと更新する。そして、（５）ノードＮ２は、送信するパケットＰに更新した送信カウンタ値「１２」を含めて暗号化した暗号化パケットＥＰをノードＮ１に送信する。

【００２７】

（６）暗号化パケットＥＰを受信したノードＮ１は、暗号化パケットＥＰを復号し、復号したパケットＰから抽出した送信カウンタ値「１２」と、自ノード内のノードＤＢ１１０に保持された受信カウンタ値「１１」とを比較する。この場合、ノードＮ１は、過去にノードＮ２が１１回目に送信した暗号化パケットＥＰを受け取ったことがあり、今回受信した暗号化パケットＥＰはノードＮ２が１２回目に送信したパケットであることが分かる。

【００２８】

そのため、受信した暗号化パケットＥＰが正規の暗号化パケットＥＰであると判定できる。また、ノードＮ１は、自ノード内のノードＤＢ１１０に、送信カウンタ値「１２」を、ノードＮ２に関連付けたあらたな受信カウンタ値として保持する。

【００２９】

１３回目以降に暗号化パケットＥＰを送信する際にも、ノードＮ２は、（４）および（５）と同様にして、送信カウンタ値を更新し、更新したカウンタ値を含む暗号化パケットＥＰを送信する。また、暗号化パケットＥＰを受信したノードＮ１は（６）と同様にして、受信した暗号化パケットＥＰの正当性判定をおこない、ノードＤＢ１１０を更新する。このようにして、ノードＮ２から１５回暗号化パケットＥＰが送信されたとする。

【００３０】

ここで、図１の（Ｂ）に示すように、再送攻撃をおこなう攻撃者のノードＮｃがアドホックネットワークＡに接続した場合を例に挙げる。（７）ノードＮ２は、１６回目に暗号化パケットＥＰを送信する際に、自ノード内の記憶装置に記憶されている送信カウンタ値を「１５」から「１６」へと更新する。そして、（８）ノードＮ２は、送信するパケットＰに更新した送信カウンタ値「１６」を含めて暗号化した暗号化パケットＥＰをノードＮ１に送信する。

【００３１】

（９）暗号化パケットＥＰを受信したノードＮ１は、暗号化パケットＥＰを復号し、復号したパケットから抽出した送信カウンタ値「１６」と、自ノード内のノードＤＢ１１０に保持された受信カウンタ値「１５」とを比較する。この場合、ノードＮ１は、過去にノードＮ２が１５回目に送信した暗号化パケットＥＰを受け取ったことがあり、今回受信した暗号化パケットＥＰはノードＮ２が１６回目に送信した暗号化パケットＥＰであることが分かる。そのため、ノードＮ１は、受信した暗号化パケットＥＰが正規の暗号化パケットＥＰであると判定できる。また、ノードＮ１は、自ノード内のノードＤＢ１１０に、送信カウンタ値「１６」を、ノードＮ２に関連付けたあらたな受信カウンタ値として保持する。

【００３２】

一方、（１０）攻撃者のノードＮｃによって、ノードＮ２が送信した暗号化パケットＥＰがアドホックネットワーク上からキャプチャされ、再送攻撃のために保持されてしまう。そして、攻撃者のノードＮｃによって、ノードＮ１に対して再送攻撃がおこなわれる。

【００３３】

具体的には、図１の（Ｃ）に示すように、（１１）攻撃者のノードＮｃは、（１０）で保持した暗号化パケットＥＰをノードＮ１に再送する。ここで、（１２）ノードＮ１は、受信した暗号化パケットＥＰを復号し、復号したパケットＰから抽出した送信カウンタ値「１６」と、自ノード内のノードＤＢ１１０に保持された受信カウンタ値「１６」とを比較する。

【００３４】

この場合、ノードＮ１は、過去にノードＮ２が１６回目に送信した暗号化パケットＥＰを受け取ったことがあるにもかかわらず、今回受信した暗号化パケットＥＰもノードＮ２が１６回目に送信した暗号化パケットＥＰであることが分かる。そのため、ノードＮ１は、受信した暗号化パケットＥＰが不正パケットであると判定できる。そして、ノードＮ１は、不正パケットを廃棄する。

【００３５】

これにより、ノードＮ１は、ノードＮ２から送信された暗号化パケットＥＰは廃棄せず、攻撃者のノードＮｃから送信された不正パケットのみを廃棄することができるため、再送攻撃を防ぐことができ、通信の安全性を担保することができる。また、ノードＮ１は、不正パケットを廃棄するため、不正パケットの内容に基づいて処理をおこなう必要がなく、ノードＮ１の処理負担を軽減できる。

【００３６】

また、時刻同期を必要とせず不正パケットを判定することができるため、時刻同期用の

パケットによる通信負荷が生じないようにできる。さらに、時刻同期を必要としないため、時刻同期用のパケットを送信するゲートウェイがアドホックネットワークＡ内にない場合でも不正パケットを判定できる。

【００３７】

（ネットワークシステムの一実施例）

図２は、実施の形態にかかるネットワークシステムの一実施例を示す説明図である。図２において、ネットワークシステム２００は、複数のノードＮ（ノードＮ１～Ｎｍ）を含む構成である。ここでｍは、アドホックネットワークＡ内のノード数である。ネットワークシステム２００において、各ノードＮは、アドホックネットワークＡを介して接続されている。

【００３８】

各ノードＮは、所定の通信圏内の他ノード（以下、「近隣ノード」という）とマルチホップ通信をおこなう無線通信装置である。アドホックネットワークＡ内の各ノードＮは、アドホックネットワークＡにおける共通の鍵（以下、「固定鍵ＦＫ」という)を保持している。従って、各ノードＮは、近隣ノードとの固定鍵ＦＫを用いた暗号化通信が可能である。

【００３９】

また、各ノードＮは、自ノードへ送信されるパケットＰの暗号化に使用されるアクセス鍵ＡＫを生成する機能を持つ（以下では、ノードＮ１～Ｎｍのそれぞれが生成したアクセス鍵ＡＫをＡＫ１～ＡＫｍとする）。各ノードＮは、アクセス鍵ＡＫの生成機能によってアクセス鍵ＡＫを生成すると、自ノードのノード番号とアクセス鍵ＡＫとを固定鍵ＦＫによって暗号化し、近隣ノードにユニホップ通信する。ユニホップ通信とは、ノードＮから近隣ノードのみに暗号化パケットＥＰを送信することをいい、近隣ノードは暗号化パケットＥＰを中継せず他ノードに再送しない。

【００４０】

ノード番号とアクセス鍵ＡＫとを含む暗号化パケットＥＰを受信したノードは、固定鍵ＦＫによって該ノード番号と該アクセス鍵ＡＫを復号し、自ノード内のデータベース（例えば、上述したノードＤＢ１１０）に保持する。すなわち、各ノードＮ１～Ｎｍは、近隣ノードのアクセス鍵ＡＫを有するが、近隣ノード以外のノードのアクセス鍵ＡＫは有していない。ただし、ノード番号は、ヘッダなどの暗号化されない部分に含まれる。

【００４１】

各ノードＮが近隣ノードに対してアクセス鍵ＡＫを用いた暗号化通信をする場合、自ノード内のデータベースに保持されている該近隣ノードのノード番号に関連付けられたアクセス鍵ＡＫを用いて暗号化した暗号化パケットＥＰを、宛先である近隣ノードに送信する。暗号化パケットＥＰを受信した近隣ノードは、自ノードのアクセス鍵ＡＫによって暗号化パケットＥＰを復号する。

【００４２】

各ノードＮは、一定の間隔でアクセス鍵ＡＫを生成して、生成したアクセス鍵ＡＫを近隣ノードに固定鍵ＦＫを用いて暗号化して送信する。このように一定の間隔でアクセス鍵を更新することで、各ノードＮは、再送攻撃の脅威を低減することができる。

【００４３】

ネットワークシステム２００は、例えば、各家庭の電力やガスの使用量を収集するシステムに適用することができる。具体的には、例えば、各家庭の電力メータやガスメータに各ノードＮ１～Ｎｍを組み込むことで、アドホックネットワークＡ内のノード間で各家庭の電力やガスの使用量を送受信する。なお、各家庭の電力やガスの使用量は、各ノードＮ１～Ｎｍが計測してもよく、また、各ノードＮ１～Ｎｍが電力メータやガスメータから取得してもよい。

【００４４】

そして、アドホックネットワークＡ内のゲートウェイが、アドホックネットワークＡ内のノードＮ１～Ｎｍから受信した各家庭の電力やガスの使用量を、ネットワークを介して

電力会社やガス会社のサーバに送信するようにしてもよい。これにより、作業員が現地に出向くことなく電力やガスの使用量を収集することができる。

【００４５】

また、ネットワークシステム２００では、アドホックネットワークＡの固定鍵ＦＫおよび各ノードＮのアクセス鍵ＡＫを用いてパケットＰを暗号化している。これにより、アドホックネットワークＡのセキュア通信（データ秘匿性、改ざん防止など）を確保できる。

【００４６】

（ノードＮのハードウェア構成例）

図３は、実施の形態にかかるノードＮのハードウェア構成例を示すブロック図である。図３において、ノードＮは、ＣＰＵ３０１と、ＲＡＭ３０２と、フラッシュメモリ３０３と、Ｉ／Ｆ３０４と、暗号化回路３０５と、を備えている。ＣＰＵ３０１～暗号化回路３０５は、バス３００によってそれぞれ接続されている。

【００４７】

ここで、ＣＰＵ３０１は、ノードＮの全体の制御を司る。ＲＡＭ３０２は、ＣＰＵ３０１のワークエリアとして使用される。また、書き換えが頻繁におこなわれる送信カウンタ値と受信カウンタ値とはＲＡＭ３０２に記憶されている。フラッシュメモリ３０３は、プログラムや暗号鍵（固定鍵ＦＫおよびアクセス鍵ＡＫ）などの鍵情報を記憶している。また、書き換えが頻繁におこなわれるアクセス鍵ＡＫはＲＡＭ３０２に記憶されてもよい。Ｉ／Ｆ３０４は、マルチホップ通信によりパケットを送受信する。

【００４８】

暗号化回路３０５は、データを暗号化する場合に暗号鍵によりデータを暗号化する回路である。また、暗号化回路３０５は、暗号化されたデータを復号鍵により復号する回路でもある。暗号化および復号をソフトウェア的に実行する場合は、暗号化回路３０５に相当するプログラムをフラッシュメモリ３０３に記憶させておくことで、暗号化回路３０５は不要となる。

【００４９】

（図１に示したノードＤＢ１１０の記憶内容）

図４は、図１に示したノードＤＢ１１０の記憶内容の一例を示す説明図である。図４に示すように、ノードＤＢ１１０は、ノード番号項目のそれぞれに対応付けて、アクセス鍵項目と、受信カウンタ値項目と、を有する。ノードＤＢ１１０は、ノードがパケットを受信するごとにレコードを構成する。

【００５０】

ノード番号項目には、アドホックネットワークＡ内の各ノードＮの識別子（Ｎ１～Ｎｍ）が記憶される。ここでは、簡単のため識別子をＮ１～Ｎｍとするが、識別子としては、ノードＮ固有のネットワークアドレスであるＭＡＣ（Ｍｅｄｉａ　Ａｃｃｅｓｓ　Ｃｏｎｔｒｏｌ）アドレスやＩＰ（Ｉｎｔｅｒｎｅｔ　Ｐｒｏｔｏｃｏｌ）アドレスを採用できる。

【００５１】

アクセス鍵項目には、各ノード番号のノードへ送信するパケットの暗号化に使用するアクセス鍵ＡＫが記憶される。アクセス鍵ＡＫは、例えば、１２８～２５６ビット程度のバイナリデータである。受信カウンタ値項目には、各ノード番号のノードから受信した暗号化パケットＥＰに含まれていた送信カウンタ値が記憶される。

【００５２】

（ノードＮの機能的構成例）

次に、図５および図６を用いて、ノードＮの機能的構成例について説明する。ここで、図５は、ノードＮの受信側としての機能的構成例を示している。また、図６は、ノードＮの送信側としての機能的構成例を示している。ここでは、便宜的に受信側としての機能と送信側としての機能を分けて説明しているが、各ノードＮは受信側としての機能および送信側としての機能の両方を有している。

【００５３】

図５は、ノードＮの受信側としての機能的構成を示す機能ブロック図である。図５に示すように、ノードＮは、第１の受信部５０１と、第１の抽出部５０２と、第２の受信部５０３と、第２の抽出部５０４と、判定部５０５と、廃棄部５０６と、データ処理部５０７と、を備える。

【００５４】

第１の受信部５０１は、アドホックネットワークＡ内のノードＮの近隣ノードの送信元アドレスと近隣ノードからの第１のパケット送信回数とを含む第１のパケットを受信する機能を有する。ここで、近隣ノードとは、ノードＮの所定の通信圏内にある他ノードである。送信元アドレスとは、上述したノードＮの識別子である。第１の送信回数とは、上述した送信カウンタ値である。第１のパケットとは、近隣ノードから送信されたパケットであり、例えば、上述した暗号化パケットＥＰである。

【００５５】

具体的には、例えば、第１の受信部５０１は、近隣ノードから、固定鍵ＦＫまたはアクセス鍵ＡＫを用いて暗号化された暗号化パケットＥＰを受信する。第１の受信部５０１は、具体的には、例えば、図３に示したフラッシュメモリ３０３に記憶されたプログラムをＣＰＵ３０１に実行させることにより、または、Ｉ／Ｆ３０４により、その機能を実現する。

【００５６】

第１の抽出部５０２は、第１の受信部５０１によって受信された第１のパケットの中から、第１のパケット送信回数を抽出する機能を有する。具体的には、例えば、第１の抽出部５０２は、第１の受信部５０１によって受信された暗号化パケットＥＰを、固定鍵ＦＫまたはアクセス鍵ＡＫを用いて復号し、復号したパケットＰに含まれる送信カウンタ値を抽出する。これにより、第１の抽出部５０２は、第１の受信部５０１によって受信された暗号化パケットＥＰが、近隣ノードが何回目に送信した暗号化パケットＥＰであるのかを把握できる。

【００５７】

第１の抽出部５０２は、具体的には、例えば、図３に示したフラッシュメモリ３０３に記憶されたプログラムをＣＰＵ３０１に実行させることにより、その機能を実現する。

【００５８】

第２の受信部５０３は、第１の受信部５０１によって第１のパケットが受信された後、近隣ノードの送信元アドレスと近隣ノードからの第２のパケット送信回数とを含む第２のパケットを受信する機能を有する。ここで、第２のパケットは、近隣ノードから送信された正規の暗号化パケットＥＰである場合がある。一方で、第２のパケットは、近隣ノードのノード番号を有するが、近隣ノードが過去に送信した暗号化パケットＥＰをキャプチャした攻撃者のノードＮｃから送信された暗号化パケットＥＰである場合がある。第２のパケット送信回数とは、第２のパケットに含まれる送信カウンタ値である。

【００５９】

具体的には、例えば、第２の受信部５０３は、固定鍵ＦＫまたはアクセス鍵ＡＫを用いて暗号化された暗号化パケットＥＰを受信する。第２の受信部５０３は、具体的には、例えば、図３に示したフラッシュメモリ３０３に記憶されたプログラムをＣＰＵ３０１に実行させることにより、または、Ｉ／Ｆ３０４により、その機能を実現する。

【００６０】

第２の抽出部５０４は、第２の受信部５０３によって受信された第２のパケットの中から、第２のパケット送信回数を抽出する機能を有する。具体的には、例えば、第２の抽出部５０４は、第２の受信部５０３によって受信された暗号化パケットＥＰを、固定鍵ＦＫまたはアクセス鍵ＡＫを用いて復号し、復号したパケットＰに含まれる送信カウンタ値を抽出する。これにより、第２の抽出部５０４は、第２の受信部５０３によって受信された暗号化パケットＥＰが、近隣ノードが何回目に送信した暗号化パケットＥＰであるのかを把握できる。

【００６１】

第２の抽出部５０４は、具体的には、例えば、図３に示したフラッシュメモリ３０３に記憶されたプログラムをＣＰＵ３０１に実行させることにより、その機能を実現する。

【００６２】

判定部５０５は、第１の抽出部５０２によって抽出された第１のパケット送信回数と、第２の抽出部５０４によって抽出された第２のパケット送信回数と、に基づいて、第２のパケットが不正パケットか否かを判定する機能を有する。

【００６３】

具体的には、例えば、判定部５０５は、今回受信した暗号化パケットＥＰ内の送信カウンタ値が、前回受信した暗号化パケットＥＰ内の送信カウンタ値より増加している場合に正規パケットと判定し、増加していない場合には不正パケットと判定する。

【００６４】

すなわち、近隣ノードからの正規の暗号化パケットＥＰを受信し続けている限り、今回受信した暗号化パケットＥＰ内の送信カウンタ値は、前回受信した暗号化パケットＥＰ内の送信カウンタ値より増加する。よって、判定部５０５は、第２のパケットを正規パケットと判定する。

【００６５】

一方で、再送攻撃では過去に送信された暗号化パケットＥＰをそのまま送信しているため、今回受信した暗号化パケットＥＰ内の送信カウンタ値は、前回受信した暗号化パケットＥＰ内の送信カウンタ値以下となる。よって、判定部５０５は、第２のパケットを送信攻撃による不正パケットと判定する。

【００６６】

これにより、判定部５０５は、第２の受信部５０３によって受信された暗号化パケットＥＰが、不正パケットであるか否かを判定できる。判定部５０５は、具体的には、例えば、図３に示したフラッシュメモリ３０３に記憶されたプログラムをＣＰＵ３０１に実行させることにより、その機能を実現する。

【００６７】

廃棄部５０６は、判定部５０５によって不正パケットと判定された場合、第２のパケットを廃棄する機能を有する。具体的には、例えば、廃棄部５０６は、第２のパケットの内容が処理要求であっても処理をおこなわず、また、第２のパケットのマルチホップ通信もおこなわずに、第２のパケットを廃棄する。

【００６８】

これにより、廃棄部５０６は、不正パケットが他ノードへさらに送信されることを防止でき、また、不正パケットの内容を実行することによるノードＮの処理負担を軽減できる。廃棄部５０６は、具体的には、例えば、図３に示したフラッシュメモリ３０３に記憶されたプログラムをＣＰＵ３０１に実行させることにより、その機能を実現する。

【００６９】

データ処理部５０７は、判定部５０５によって、不正パケットと判定されなかった暗号化パケットＥＰの内容に従って処理をおこなう機能を有する。具体的には、例えば、データ処理部５０７は、暗号化パケットＥＰをマルチホップ通信により他ノードへ送信したり、暗号化パケットＥＰの内容が処理要求である場合に処理をおこなったりする。これにより、正規パケットには適切な処理がおこなわれる。データ処理部５０７は、具体的には、例えば、図３に示したフラッシュメモリ３０３に記憶されたプログラムをＣＰＵ３０１に実行させることにより、または、Ｉ／Ｆ３０４により、その機能を実現する。

【００７０】

図６は、ノードＮの送信側としての機能的構成を示す機能ブロック図である。図６に示すように、ノードＮは、検出部６０１と、更新部６０２と、送信部６０３と、を備える。ノードＮは、検知部６０４と、通知部６０５と、受信部６０６と、抽出部６０７と、格納部６０８と、を備える。ノードＮは、暗号化部６０９と、復号部６１０と、生成部６１１と、鍵受信部６１２と、鍵復号部６１３を備える。

【００７１】

検出部６０１は、アドホックネットワーク内のノードの近隣ノードへのパケット送信イベントを検出する機能を有する。ここで、近隣ノードとは、ノードＮの所定の通信圏内にある他ノードである。送信イベントとは、近隣ノードへの暗号化パケットＥＰの送信のトリガとなるイベントであり、例えば、ノードＮのユーザが入力した送信命令であったり、ノードＮのフラッシュメモリ３０３に記憶されたプログラムによって自動的に発生した送信命令であったりする。

【００７２】

これにより、検出部６０１は、暗号化パケットＥＰの送信をおこなうトリガを検出できる。検出部６０１は、具体的には、例えば、図３に示したフラッシュメモリ３０３に記憶されたプログラムをＣＰＵ３０１に実行させることにより、その機能を実現する。

【００７３】

更新部６０２は、検出部６０１によってパケット送信イベントが検出された場合、ノードＮの記憶装置に保持されたノードＮからのパケット送信回数を、パケット送信回数に１を加算した値に更新する機能を有する。ここで、記憶装置とは、図３に示したＲＡＭ３０２である。パケット送信回数とは、上述した送信カウンタ値である。

【００７４】

具体的には、例えば、更新部６０２は、ＲＡＭ３０２に保持されている送信カウンタ値をインクリメントして更新する。これにより、更新部６０２は、後述する送信部６０３によって送信される暗号化パケットＥＰが何回目に送信される暗号化パケットＥＰであるのかを算出して、送信カウンタ値を更新することができる。更新部６０２は、具体的には、例えば、図３に示したフラッシュメモリ３０３に記憶されたプログラムをＣＰＵ３０１に実行させることにより、その機能を実現する。

【００７５】

送信部６０３は、更新部６０２によって更新されたパケット送信回数を含むパケットを近隣ノードに送信する機能を有する。具体的には、例えば、送信部６０３は、更新された送信カウンタ値を含む暗号化パケットＥＰを近隣ノードに送信する。これにより、送信部６０３は、近隣ノードに、自ノードが何回目に送信した暗号化パケットＥＰであるのかを通知することができる。送信部６０３は、具体的には、例えば、図３に示したフラッシュメモリ３０３に記憶されたプログラムをＣＰＵ３０１に実行させることにより、その機能を実現する。

【００７６】

検知部６０４は、ノードＮの記憶装置に保持されたノードＮからのパケット送信回数の消去を検知する機能を有する。具体的には、例えば、検知部６０４は、停電によって送信カウンタ値が消去されたことや、ノードＮのユーザによって送信カウンタ値が初期化されたことを検知する。これにより、検知部６０４は、送信する暗号化パケットＥＰに含めるべき送信カウンタ値が消去されたことを検知し、暗号化通信を一時中止することができる。検知部６０４は、具体的には、例えば、図３に示したフラッシュメモリ３０３に記憶されたプログラムをＣＰＵ３０１に実行させることにより、その機能を実現する。

【００７７】

通知部６０５は、検知部６０４によって消去が検知された場合に、近隣ノードの記憶装置に保持されたノードからのパケット送信回数の要求パケットを、近隣ノードに通知する機能を有する。ここで、近隣ノードの記憶装置に保持されたノードからのパケット送信回数とは、近隣ノードのノードＤＢ１１０に記憶されている受信カウンタ値である。

【００７８】

具体的には、例えば、通知部６０５は、アドホックネットワーク内のノードＮで予め受信カウンタ値の要求用のカウンタ値として定めておいた送信カウンタ値「０」を含む暗号化パケットＥＰを近隣ノードに送信する。これにより、通知部６０５は、自ノードが過去に送信した暗号化パケットＥＰに含まれていた送信カウンタ値を、近隣ノードに要求することができる。通知部６０５は、具体的には、例えば、図３に示したフラッシュメモリ３０３に記憶されたプログラムをＣＰＵ３０１に実行させることにより、または、Ｉ／Ｆ３

０４により、その機能を実現する。
【００７９】
受信部６０６は、通知部６０５によって要求パケットが通知された結果、近隣ノードから送信されてくるノードからのパケット送信回数を含む応答パケットを受信する機能を有する。具体的には、例えば、受信部６０６は、近隣ノードから、自ノードが過去に送信した暗号化パケットＥＰに含まれていた送信カウンタ値を含む暗号化パケットＥＰを受信する。受信部６０６は、具体的には、例えば、図３に示したフラッシュメモリ３０３に記憶されたプログラムをＣＰＵ３０１に実行させることにより、または、Ｉ／Ｆ３０４により、その機能を実現する。
【００８０】
抽出部６０７は、受信部６０６によって受信された応答パケットの中から、ノードからのパケット送信回数を抽出する機能を有する。具体的には、抽出部６０７は、応答パケットの中から近隣ノードのノードＤＢ１１０に記憶されていた受信カウンタ値を抽出する。
【００８１】
これにより、抽出部６０７は、自ノードの過去の送信カウンタ値である他ノードの受信カウンタ値を抽出できる。抽出部６０７は、具体的には、例えば、図３に示したフラッシュメモリ３０３に記憶されたプログラムをＣＰＵ３０１に実行させることにより、その機能を実現する。
【００８２】
格納部６０８は、抽出部６０７によって抽出されたノードからのパケット送信回数をノードの記憶装置に格納する機能を有する。具体的には、例えば、格納部６０８は、消去された送信カウンタ値の代わりに、要求パケットに含まれていた受信カウンタ値を、ＲＡＭ３０３に格納する。
【００８３】
これにより、格納部６０８は、送信する暗号化パケットＥＰに含めるべき送信カウンタ値をあらたに格納することができ、ノードＮは暗号化通信を再開することができる。格納部６０８は、具体的には、例えば、図３に示したフラッシュメモリ３０３に記憶されたプログラムをＣＰＵ３０１に実行させることにより、その機能を実現する。
【００８４】
暗号化部６０９は、アドホックネットワーク内の各ノードが有する共通鍵を用いて、要求パケットを暗号化する機能を有する。具体的には、例えば、暗号化部６０９は、送信するパケットを固定鍵ＦＫやアクセス鍵ＡＫを用いて暗号化する。これにより、暗号化部６０９は、セキュアな暗号化通信をおこなうことができる。暗号化部６０９は、具体的には、例えば、図３に示したフラッシュメモリ３０３に記憶されたプログラムをＣＰＵ３０１に実行させることにより、または、暗号化回路３０５により、その機能を実現する。
【００８５】
復号部６１０は、近隣ノードによって共通鍵を用いて暗号化された応答パケットから、共通鍵を用いて応答パケットを復号する機能を有する。具体的には、例えば、復号部６１０は、受信した暗号化パケットＥＰを、固定鍵ＦＫやアクセス鍵ＡＫを用いて復号する。これにより、復号部６１０は、暗号化パケットＥＰの内容を復号することができる。復号部６１０は、具体的には、例えば、図３に示したフラッシュメモリ３０３に記憶されたプログラムをＣＰＵ３０１に実行させることにより、または、暗号化回路３０５により、その機能を実現する。
【００８６】
生成部６１１は、第１の識別情報となる乱数を生成する機能を有する。具体的には、例えば、生成部６１１は、乱数表を用いて乱数を生成する。これにより、暗号化パケットＥＰに一時的な識別情報を含めることができるようになり、セキュアな暗号化通信をおこなうことができる。生成部６１１は、具体的には、例えば、図３に示したフラッシュメモリ３０３に記憶されたプログラムをＣＰＵ３０１に実行させることにより、その機能を実現する。

【００８７】
鍵受信部６１２は、検知部６０４によって消去が検知された後に、アドホックネットワーク内の各ノードＮが有する第１の共通鍵を用いて暗号化された近隣ノードが有する第２の共通鍵を、近隣ノードから受信する機能を有する。ここで、第１の共通鍵とは、上述した固定鍵ＦＫである。第２の共通鍵とは、上述したアクセス鍵ＡＫである。
【００８８】
具体的には、例えば、鍵受信部６１２は、固定鍵ＦＫによって暗号化されたアクセス鍵ＡＫを受信する。これにより、鍵受信部６１２は、近隣ノードからアクセス鍵ＡＫを受信し、セキュアな暗号化通信をおこなうことができる。鍵受信部６１２は、具体的には、例えば、図３に示したフラッシュメモリ３０３に記憶されたプログラムをＣＰＵ３０１に実行させることにより、または、Ｉ／Ｆ３０４により、その機能を実現する。
【００８９】
鍵復号部６１３は、鍵受信部６１２によって受信された暗号化された第２の共通鍵から、第１の共通鍵を用いて第２の共通鍵を復号する機能を有する。具体的には、例えば、鍵復号部６１３は、固定鍵ＦＫを用いて、アクセス鍵ＡＫを復号する。これにより、ノードＮは、近隣ノードへ、アクセス鍵ＡＫを用いたセキュアな暗号化通信をおこなうことができるようになる。鍵復号部６１３は、具体的には、例えば、図３に示したフラッシュメモリ３０３に記憶されたプログラムをＣＰＵ３０１に実行させることにより、または、暗号化回路３０５により、その機能を実現する。このように、ノードＮによれば、再送攻撃を簡単かつ効率的に検出することにより、通信負荷の低減を図ることができる。以下、上述したノードＮについての実施例について説明する。
【００９０】
（実施例１）
ここで、送信カウンタ値が、送信側のノードにおける暗号化パケットＥＰの送信回数の総和である場合について説明する。送信回数の総和を用いることで、ノードＮがアドホックネットワークＡ内を移動して、宛先となる近隣ノードが変わった場合でも、送信カウンタ値を更新する必要がない。このように、時刻情報ではなく、送信回数の総和を用いることで、再送攻撃を簡単かつ効率的に検出することができ、通信負荷の低減を図ることができる。
【００９１】
（アクセス鍵ＡＫを用いた暗号化通信の例）
まず、図７および図８を用いて、図２に示したネットワークシステム２００における実施例１にかかるアクセス鍵ＡＫを用いた暗号化通信の例について説明する。
【００９２】
図７は、ネットワークシステム２００におけるアクセス鍵ＡＫを用いた暗号化通信を示す説明図である。図７では、アドホックネットワークＡは、ノードＮ１、ノードＮ２およびノードＮ３で構成されているとする。
【００９３】
（１）ノードＮ２は、９回目に暗号化パケットＥＰを送信する際に、自ノード内の記憶装置に記憶されている送信カウンタ値を「８」から「９」へと更新する。そして、（２）ノードＮ２は、送信するパケットＰに更新した送信カウンタ値「９」を含めて暗号化した暗号化パケットＥＰをノードＮ１に送信する。
【００９４】
（３）暗号化パケットＥＰを受信したノードＮ１は、暗号化パケットＥＰを復号したパケットＰから、送信元のノード番号と送信カウンタ値とを抽出する。ノードＮ１は、復号したパケットＰから抽出した送信カウンタ値「９」と、自ノード内のノードＤＢ１１０に保持された受信カウンタ値「６」とを比較する。この場合、ノードＮ１は、過去にノードＮ２が６回目に送信した暗号化パケットＥＰを受け取ったことがあり、今回受信した暗号化パケットＥＰはノードＮ２が９回目に送信したパケットであることが分かる。
【００９５】

そのため、受信した暗号化パケットＥＰが正規の暗号化パケットＥＰであると判定できる。また、ノードＮ１は、自ノード内のノードＤＢ１１０に、送信カウンタ値「９」を、ノードＮ２に関連付けたあらたな受信カウンタ値として保持する。

【００９６】

（４）ノードＮ２は、１０回目に暗号化パケットＥＰを送信する際に、自ノード内の記憶装置に記憶されている送信カウンタ値を「９」から「１０」へと更新する。そして、（５）ノードＮ２は、送信するパケットＰに更新した送信カウンタ値「１０」を含めて暗号化した暗号化パケットＥＰをノードＮ３に送信する。

【００９７】

（６）暗号化パケットＥＰを受信したノードＮ３は、暗号化パケットＥＰを復号したパケットＰから、送信元のノード番号と送信カウンタ値とを抽出する。ノードＮ３は、復号したパケットＰから抽出した送信カウンタ値「１０」と、自ノード内のノードＤＢ１１０に保持された受信カウンタ値「８」とを比較する。この場合、ノードＮ３は、過去にノードＮ２が８回目に送信した暗号化パケットＥＰを受け取ったことがあり、今回受信した暗号化パケットＥＰはノードＮ２が１０回目に送信したパケットであることが分かる。

【００９８】

そのため、受信した暗号化パケットＥＰが正規の暗号化パケットＥＰであると判定できる。また、ノードＮ３は、自ノード内のノードＤＢ１１０に、送信カウンタ値「１０」を、ノードＮ２に関連付けたあらたな受信カウンタ値として保持する。次に、図８を用いて、ネットワークシステム２００におけるアクセス鍵による暗号化通信の詳細について説明する。

【００９９】

図８は、ネットワークシステム２００におけるアクセス鍵ＡＫを用いた暗号化通信の詳細を示す説明図である。図８では、アドホックネットワークＡ内のノードＮ１からノードＮ２への暗号化通信を説明する。ここでは、ノードＮ１のノード番号が「Ｎ１」、アクセス鍵が「１１１１」、カウンタが「１２３４」である場合を説明する。

【０１００】

図８において、（１）ノードＮ１は、ノードＮ１のユーザからのデータの入力を受けると、ノードＮ２との暗号化通信を開始する。ここでは、ユーザからのデータ入力をトリガにして暗号化通信を開始したが、ノードＮ１が自動的に発生させたデータ送信イベントをトリガとしてもよい。

【０１０１】

（２）ノードＮ１は、まず送信カウンタ値を１増加させる。このときノードＮ１の送信カウンタ値は「１２３５」となる。（３）次に、ノードＮ１は、カウンタ「１２３５」と送信したいデータを、宛先のノードＮ２のアクセス鍵ＡＫ２によって暗号化し、ノードＮ１のノード番号「Ｎ１」とともにユニホップ通信する。ただし、ノード番号「Ｎ１」の部分は、ヘッダなどの暗号化されないデータに含まれる。以下の説明においても、同様にノード番号はヘッダなどの暗号化されないデータに含まれる。また、暗号化されるデータに、さらにノード番号を含んでおいてもよい。

【０１０２】

ノードＮ１からの暗号化パケットＥＰを受信したノードＮ２は、ノード番号から暗号化パケットＥＰの送信者がノードＮ１であることを特定する。（４）次にノードＮ２は、保持するノードＤＢ１１０を参照し、ノードＮ２のアクセス鍵ＡＫ２を用いて、受信した暗号化パケットＥＰを復号する。

【０１０３】

（５）次にノードＮ２は、復号したパケットＰから送信カウンタ値を抽出し、ノードＤＢ１１０に保持している受信カウンタ値と比較する。そして、ノードＮ２は、送信カウンタ値「１２３５」が受信カウンタ値「１２３４」よりも大きいため、復号したパケットＰは正しく復号できたと見なして、ノードＤＢ１１０のノード番号「Ｎ１」の受信カウンタ値をパケットＰのカウンタ「１２３５」に更新する。

【０１０４】
一方、受信した送信カウンタ値が保持している受信カウンタ値より小さい場合または同一である場合には、受信した暗号化パケットＥＰは不正パケットと見なして廃棄する。なお、保持している受信カウンタ値との差が所定の閾値以内であれば、正しく復号できたと見なすこともできる。
【０１０５】
（アクセス鍵ＡＫの更新例）
次に、図２に示したネットワークシステム２００において、ノードＮがアクセス鍵ＡＫを更新する場合について説明する。
【０１０６】
図９は、ネットワークシステム２００におけるアクセス鍵ＡＫの更新例を示す説明図である。図９では、アドホックネットワークＡ内のノードＮ１からノードＮ２への暗号化通信を説明する。ここでは、ノードＮ１のノード番号が「Ｎ１」、アクセス鍵ＡＫが「１１１１」、受信カウンタ値が「１２３４」である場合を説明する。
【０１０７】
図９において、（１）ノードＮ１がアクセス鍵ＡＫを更新する場合、まず、ノードＮ１は新しいアクセス鍵ＡＫ「８８８８」を生成する。次に、ノードＮ１は、送信カウンタ値をインクリメントする。このとき、ノードＮ１の送信カウンタ値は「１３５８」となる。
【０１０８】
（２）そして、ノードＮ１は、送信カウンタ値「１３５８」と新しいアクセス鍵ＡＫ「８８８８」、アドホックネットワークＡの固定鍵ＦＫによって暗号化し、ノードＮ１のノード番号「Ｎ１」とともにユニホップ通信する。ただし、ノード番号「Ｎ１」の部分は、ヘッダなどの暗号化されないデータに含まれて送信される。
【０１０９】
ノードＮ１からの暗号化パケットＥＰを受信したノードＮ２は、ノード番号からその暗号化パケットＥＰの送信者がノードＮ１であることを特定する。（３）次にノードＮ２は、保持する固定鍵ＦＫを用いて復号し、復号したパケットＰから送信カウンタ値「１３５８」と新しいアクセス鍵ＡＫ「８８８８」を入手する。
【０１１０】
そしてノードＮ２は、送信カウンタ値「１３５８」と、ノードＤＢに保持している受信カウンタ値「１２３４」とを比較する。ここで、ノードＮ２は、送信カウンタ値「１３５８」が受信カウンタ値「１２３４」よりも大きいため、復号したパケットＰは正しく復号できたと見なす。（４）そして、ノードＮ２は、ノードＤＢ１１０のノード番号「Ｎ１」の受信カウンタ値を、受信したパケットの送信カウンタ値「１３５８」で更新し、アクセス鍵ＡＫ「１１１１」を新しいアクセス鍵ＡＫ「８８８８」で更新する。
【０１１１】
一方、受信した送信カウンタ値が保持している受信カウンタ値より小さい場合または同一である場合には、受信した暗号化パケットＥＰは不正パケットと見なして廃棄する。なお、保持している受信カウンタ値との差が所定の閾値以内であれば、正しく復号できたと見なすこともできる。
【０１１２】
（新規ノードの導入時におけるアクセス鍵ＡＫの登録例）
次に、図２に示したネットワークシステム２００への新規ノードの導入時における設定例について説明する。
【０１１３】
図１０は、ネットワークシステム２００への新規ノードの導入例を示す説明図である。図１０において、ネットワークシステム２００のアドホックネットワークＡ内に新規ノードが導入されたとする。図１０では、アドホックネットワークＡ内のノードＮ２に、新規ノードＮ１が追加される場合を示している。
【０１１４】

（１）新規ノードＮ１は、まずアクセス鍵ＡＫ「１１１１」を生成する。次に新規ノードＮ１は、送信カウンタ値をインクリメントして「０００１」とする。（２）そして、新規ノードＮ１は、送信カウンタ値「０００１」とアクセス鍵ＡＫ「１１１１」を固定鍵ＦＫによって暗号化し、ノード番号「Ｎ１」とともにユニホップ通信する。ただし、ノード番号「Ｎ１」の部分は、ヘッダなどの暗号化されないデータに含まれる。

【０１１５】

ノードＮ１からの暗号化パケットＥＰを受信したノードＮ２は、ノード番号からその暗号化パケットの送信者がノードＮ１であることを特定する。（３）次にノードＮ２は、保持する固定鍵ＦＫを用いて復号し、送信カウンタ値「０００１」とアクセス鍵ＡＫ「１１１１」を入手する。

【０１１６】

そして、ノードＮ２は、送信カウンタ値と、ノードＤＢ１１０に保持している受信カウンタ値と比較しようとするが、ノードＤＢ１１０にノード番号「Ｎ１」に対応する受信カウンタ値を保持していない。そのため、ノードＮ２はノードＮ１との通信が初めてであることを認識し、ノードＤＢ１１０にノード番号「Ｎ１」と、対応するアクセス鍵ＡＫとして「１１１１」を格納し、送信カウンタ値「０００１」を対応する受信カウンタ値として格納する。

【０１１７】

ここで、ノードＮ１は新規追加でなくてもよく、送信カウンタ値が「０００１」でない場合でも、ノードＮ２のノードＤＢ１１０にノード番号「Ｎ１」の情報が登録されていない場合には、同様な処理をおこなう。例えば、ノードＮ１が可搬である場合に、ノードＮ２の近隣にノードＮ１が移動した場合が挙げられる。

【０１１８】

（パケット送信処理）

次に、暗号化通信におけるパケット送信処理について説明する。

【０１１９】

図１１は、暗号化通信におけるパケット送信処理の詳細を示すフローチャートである。まず、ＣＰＵ３０１は、送信イベントが発生したか否かを判定する（ステップＳ１１０１）。送信イベントが発生していない場合（ステップＳ１１０１：Ｎｏ）、ＣＰＵ３０１は、ステップＳ１１０１に戻る。

【０１２０】

一方、送信イベントが発生した場合（ステップＳ１１０１：Ｙｅｓ）、ＣＰＵ３０１は、送信カウンタ値をインクリメントする（ステップＳ１１０２）。次に、ＣＰＵ３０１は、宛先のノードＮのアクセス鍵ＡＫがノードＤＢ１１０にあるか判定する（ステップＳ１１０３）。

【０１２１】

アクセス鍵ＡＫがある場合（ステップＳ１１０３：Ｙｅｓ）、ＣＰＵ３０１は、アクセス鍵ＡＫを用いて送信するパケットを暗号化し（ステップＳ１１０４）、宛先のノードＮに暗号化パケットＥＰを送信する（ステップＳ１１０６）。そして、パケット送信処理を終了する。

【０１２２】

一方、アクセス鍵ＡＫがない場合（ステップＳ１１０３：Ｎｏ）、ＣＰＵ３０１は、固定鍵ＦＫを用いて送信するパケットを暗号化し（ステップＳ１１０５）、宛先のノードＮに暗号化パケットＥＰを送信する（ステップＳ１１０６）。そして、パケット送信処理を終了する。

【０１２３】

（パケット受信処理）

次に、暗号化通信におけるパケット受信処理について説明する。

【０１２４】

図１２は、暗号化通信におけるパケット受信処理の詳細を示すフローチャートである。

まず、ＣＰＵ３０１は、暗号化パケットＥＰを受信したか否かを判定する（ステップＳ１２０１）。暗号化パケットＥＰを受信していない場合（ステップＳ１２０１：Ｎｏ）、ＣＰＵ３０１は、ステップＳ１２０１に戻る。

【０１２５】

一方、暗号化パケットＥＰを受信した場合（ステップＳ１２０１：Ｙｅｓ）、ＣＰＵ３０１は、自ノードのアクセス鍵ＡＫまたは固定鍵ＦＫを用いて、暗号化パケットＥＰを復号し（ステップＳ１２０２）、正常に復号できたか否かを判定する（ステップＳ１２０３）。正常に復号できない場合（ステップＳ１２０３：Ｎｏ）、ＣＰＵ３０１は、受信した暗号化パケットＥＰを不正パケットであるとして廃棄し（ステップＳ１２０６）、パケット受信処理を終了する。

【０１２６】

一方、正常に復号ができた場合（ステップＳ１２０３：Ｙｅｓ）、ＣＰＵ３０１は、復号したパケットＰ内の送信カウンタ値が、ノードＤＢ１１０内の受信カウンタ値より大きいか否かを判定する（ステップＳ１２０４）。受信カウンタ値以下の場合（ステップＳ１２０４：Ｎｏ）、ＣＰＵ３０１は、受信した暗号化パケットＥＰを不正パケットであるとして廃棄し（ステップＳ１２０６）、パケット受信処理を終了する。

【０１２７】

一方、受信カウンタ値より大きい場合（ステップＳ１２０４：Ｙｅｓ）、ＣＰＵ３０１は、受信した暗号化パケットＥＰは正規パケットであると判断し、ノードＤＢ１１０を更新して（ステップＳ１２０５）、パケット送信処理を終了する。ノードＤＢ１１０の更新とは、図８の（５）における更新、図９の（４）における更新、または図１０の（４）における更新である。

【０１２８】

（ノードの送信カウンタ値が消去された場合の処理）

次に、ノードＮの送信カウンタ値が消去された場合について説明する。例えば、ノードＮが停電により電源供給されずＲＡＭ３０２の記憶内容（送信カウンタ値と受信カウンタ値とアクセス鍵ＡＫ）が消去された場合である。また、ユーザ操作によってＲＡＭ３０２がリセットされた場合であってもよい。

【０１２９】

この場合、ノードＮは、自ノードが過去に送信した暗号化パケットＥＰに含めた送信カウンタ値が不明であるため、以後送信する暗号化パケットＥＰに含めるべき送信カウンタ値が分からず、アドホックネットワークＡへ復帰することができない。

【０１３０】

（ノードの送信カウンタ値が消去された場合の動作例１）

図１３は、ノードの送信カウンタ値が消去された場合の動作例１を示す説明図である。図１３において、（１）ノードＮ１に停電が起こったとする。そのため、ノードＮ１に保持されていた送信カウンタ値およびアクセス鍵ＡＫが消去されたとする。

【０１３１】

（２）ここで、ノードＮ１は、乱数「５８１９」を生成し、ノードＮ２のノードＤＢ１１０に保持されたノードＮ１の受信カウンタ値の要求を示す特別な送信カウンタ値「００００」と乱数「５８１９」を含めたパケットを作成し、固定鍵ＦＫを用いて暗号化する。そして、ノードＮ１は、ノードＮ２に対して、暗号化パケットＥＰをノード番号「Ｎ１」とともにユニホップ通信する。ただし、ノード番号「Ｎ１」の部分はヘッダなどの暗号化されない部分に含まれる。以下の説明においても、同様にノード番号は暗号化されない部分に含まれる。

【０１３２】

ノードＮ１からの暗号化パケットＥＰを受信したノードＮ２は、ノード番号からその暗号化パケットの送信者がノードＮ１であることを特定する。次にノードＮ２は、保持する固定鍵ＦＫを用いて復号し、送信カウンタ値「００００」と乱数「５８１９」を入手する。

【０１３３】
（３）送信カウンタ値「００００」は、受信カウンタ値の要求を示す特別な送信カウンタ値であるため、ノードＮ２は、ノードＤＢ１１０を参照し、受信カウンタ値「１２３４」と乱数「５８１９」を含めたパケットを作成し、固定鍵ＦＫを用いて暗号化する。（４）そして、ノードＮ２は、ノードＮ１に対して、暗号化パケットＥＰをノード番号「Ｎ２」とともにユニホップ通信する。
【０１３４】
（５）ノードＮ２からの暗号化パケットＥＰを受信したノードＮ１は、固定鍵ＦＫを用いて復号し、ノードＮ２における受信カウンタ値「１２３４」と乱数「５８１９」を入手する。ノードＮ１は、乱数が自ノードが送信した乱数と一致することを確認し、正規パケットであると判断された場合には、自ノードの送信カウンタ値を、ノードＮ２における受信カウンタ値「１２３４」で更新する。
【０１３５】
また、複数のノードＮに、要求パケットを送信した場合は、各ノードにおける受信カウンタ値のうち最も大きい値を、自ノードの送信カウンタ値とする。使用済みの乱数を記憶しておくことにより、要求パケットの再送攻撃を防止することができる。
【０１３６】
（ノードの送信カウンタ値が消去された場合の動作例１のシーケンス図）
図１４は、図１３に示したノードの送信カウンタ値が消去された場合の動作例１のシーケンス図である。図１４において、（１）送信カウンタ値が消去されたことを検知したノードＮ１は、（２）乱数「５８１９」を生成し、（３）受信カウンタ値の要求を示す特別な送信カウンタ値「００００」と乱数「５８１９」を固定鍵ＦＫで暗号化し、ノード番号「Ｎ１」とともに近隣ノードにユニホップ通信する。
【０１３７】
（４）暗号化パケットＥＰを受信したノードＮ２は、ノード番号から、そのパケットの送信者がノードＮ１であることを認識するとともに、暗号化パケットＥＰを固定鍵ＦＫを用いて復号し、送信カウンタ値「００００」と乱数「５８１９」を入手する。ノードＮ２はノードＤＢ１１０を参照し、ノードＮ１の受信カウンタ値を保持していない場合には、暗号化パケットＥＰを廃棄する。
【０１３８】
（５）ノードＮ２は、受信カウンタ値を保持している場合には、受信カウンタ値「１２３４」と乱数「５８１９」を固定鍵ＦＫによって暗号化し、ノードＮ１に送信する。ノードＮ２からの暗号化パケットＥＰを受信したノードＮ１は、暗号化パケットＥＰを固定鍵ＦＫを用いて復号し、受信カウンタ値「１２３４」と乱数「５８１９」を入手する。入手した乱数と、自ノードが送信した乱数が一致する場合には、ノードＮ２における受信カウンタ値をノードＮ１の送信カウンタ値として更新する。ノードＮ１は、一致しない場合には、受信した暗号化パケットＥＰを不正パケットとして廃棄する。ここで、乱数を使用するのは、固定鍵ＦＫを持たない攻撃者による不正パケットの送信を検出するためである。
【０１３９】
（送信カウンタ値更新処理）
図１５は、図１３および図１４に示した例における送信カウンタ値更新処理の詳細を示すフローチャートである。まず、ＣＰＵ３０１は、送信カウンタ値が消去されたか否かを検知する（ステップＳ１５０１）。送信カウンタ値が消去されていない場合（ステップＳ１５０１：Ｎｏ）、ＣＰＵ３０１は、ステップＳ１５０１に戻る。
【０１４０】
一方、送信カウンタ値が消去されている場合（ステップＳ１５０１：Ｙｅｓ）、ＣＰＵ３０１は、乱数を生成し、生成した乱数と受信カウンタ値の要求とを固定鍵ＦＫを用いて暗号化して暗号化パケットＥＰを作成する（ステップＳ１５０２）。次に、ＣＰＵ３０１は、作成した暗号化パケットＥＰを近隣ノードに送信する（ステップＳ１５０３）。
【０１４１】

そして、ＣＰＵ３０１は、近隣ノードからの応答を受信したか否かを判定する（ステップＳ１５０４）。応答を受信していない場合（ステップＳ１５０４：Ｎｏ）、ステップＳ１５０４に戻る。

【０１４２】

一方、応答を受信した場合（ステップＳ１５０４：Ｙｅｓ）、ＣＰＵ３０１は、固定鍵ＦＫを用いて応答された暗号化パケットＥＰを復号する（ステップＳ１５０５）。そしてＣＰＵ３０１は、復号したパケットＰ内の乱数がステップＳ１５０２で生成した乱数と一致するか否かを判定する（ステップＳ１５０６）。

【０１４３】

一致する場合（ステップＳ１５０６：Ｙｅｓ）、ＣＰＵ３０１は、パケットＰ内の受信カウンタ値により、自ノードのノードＤＢ１１０内の送信カウンタ値を更新して（ステップＳ１５０７）、送信カウンタ値更新処理を終了する。

【０１４４】

一致しない場合（ステップＳ１５０６：Ｎｏ）、ＣＰＵ３０１は、受信した暗号化パケットＥＰは不正パケットであるとして廃棄し（ステップＳ１５０８）、送信カウンタ値更新処理を終了する。

【０１４５】

これにより、ノードＮは、送信カウンタ値が消去された場合であっても、送信カウンタ値を更新することができ、アドホックネットワークＡに復帰できる。また、要求パケットは固定鍵ＦＫを用いて暗号化されているため、固定鍵ＦＫを有さない攻撃者による要求パケットの解析を防止できる。

【０１４６】

そして、ノードＮは、要求パケットに近隣ノードとの通信ごとに生成した乱数を含めることにより、正規の応答か否かを、通信ごとに判定する。すなわち、通信の都度、乱数が異なるため、前回の通信で近隣ノードが応答した暗号化パケットＥＰを、今回の要求パケットに対する応答と偽って攻撃者が再送攻撃に利用した場合に、ノードＮは、不正パケットと判定でき通信品質を確保できる。

【０１４７】

また、一度使用した乱数を記憶しておくようにすれば、今回の通信で近隣ノードが応答した暗号化パケットＥＰを、次回の通信より前に攻撃者が再送攻撃に利用した場合にも、ノードＮは、不正パケットと判定でき通信品質を確保できる。このように、時刻情報ではなく、送信回数の総和を用いることで、再送攻撃を簡単かつ効率的に検出することができ、通信負荷の低減を図ることができる。

【０１４８】

（ノードの送信カウンタ値が消去された場合の動作例２）

図１６は、ノードの送信カウンタ値が消去された場合の動作例２を示す説明図である。図１６において、（１）ノードＮ１に停電が起こったとする。そのため、ノードＮ１に保持されていた送信カウンタ値およびアクセス鍵ＡＫが消去されたとする。ここで、ノードＮ１は、近隣ノードであるＮ２からアクセス鍵ＡＫが送信されてくるまで待機する。

【０１４９】

（２）ノードＮ２は、一定時間ごとにアクセス鍵ＡＫを生成して、近隣ノードに送信する。ここでは、ノードＮ１は、ノードＮ２から送信されたアクセス鍵ＡＫ「６６６６」を受信したとする。

【０１５０】

（３）次に、ノードＮ１は、乱数「５８１９」を生成する。そして、ノードＮ１は、ノードＮ２のノードＤＢ１１０に保持されたノードＮ１の受信カウンタ値の要求を示す特別な送信カウンタ値「００００」と乱数「５８１９」を含めたパケットＰを作成し、アクセス鍵ＡＫ「６６６６」を用いて暗号化する。そして、ノードＮ１は、ノードＮ２に対して、暗号化パケットＥＰをノード番号「Ｎ１」とともにユニホップ通信する。ただし、ノード番号「Ｎ１」の部分は、ヘッダなどの暗号化されない部分に含まれる。以下の説明にお

いても、同様にノード番号は暗号化されない部分に含まれる。
【０１５１】
ノードＮ１からの暗号化パケットＥＰを受信したノードＮ２は、ノード番号からその暗号化パケットの送信者がノードＮ１であることを特定する。次にノードＮ２は、暗号化パケットＥＰをアクセス鍵ＡＫ「６６６６」を用いて復号し、送信カウンタ値「００００」と乱数「５８１９」を入手する。
【０１５２】
（４）送信カウンタ値「００００」は、受信カウンタ値の要求を示す特別な送信カウンタ値であるため、ノードＮ２は、ノードＤＢ１１０を参照し、受信カウンタ値「１２３４」と乱数「５８１９」を含めたパケットを作成し、固定鍵ＦＫを用いて暗号化する。（５）そして、ノードＮ２は、ノードＮ１に対して、暗号化パケットＥＰをノード番号「Ｎ２」とともにユニホップ通信する。
【０１５３】
（６）ノードＮ２からの暗号化パケットＥＰを受信したノードＮ１は、固定鍵ＦＫを用いて復号し、ノードＮ２における受信カウンタ値「１２３４」と乱数「５８１９」を入手する。ノードＮ１は、乱数が自ノードが送信した乱数と一致することを確認し、正規パケットであると判断された場合には、自ノードの送信カウンタ値を、ノードＮ２における受信カウンタ値「１２３４」で更新する。また、複数のノードＮに、要求パケットを送信した場合は、各ノードにおける受信カウンタ値のうち最も大きい値を、自ノードの送信カウンタ値とする。
【０１５４】
（ノードの送信カウンタ値が消去された場合の動作例２のシーケンス図）
図１７は、図１６に示したノードの送信カウンタ値が消去された場合の動作例２のシーケンス図である。図１７において、（１）送信カウンタ値が消去されたことを検知したノードＮ１は、近隣ノードであるＮ２からアクセス鍵ＡＫが送信されてくるまで待機する。（２）ノードＮ２は、一定時間ごとにアクセス鍵ＡＫを生成して、近隣ノードに送信する。ここでは、ノードＮ２は、近隣ノードにアクセス鍵ＡＫ「６６６６」を送信する。（３）そして、ノードＮ１は、ノードＮ２から送信されたアクセス鍵ＡＫ「６６６６」を受信する。
【０１５５】
（４）次に、ノードＮ１は、乱数「５８１９」を生成する。そして、ノードＮ１は、ノードＮ２のノードＤＢ１１０に保持されたノードＮ１の受信カウンタ値の要求を示す特別な送信カウンタ値「００００」と乱数「５８１９」を含めたパケットＰを作成し、アクセス鍵ＡＫ「６６６６」を用いて暗号化する。（５）そして、ノードＮ１は、ノードＮ２に対して、暗号化パケットＥＰをノード番号「Ｎ１」とともにユニホップ通信する。
【０１５６】
（６）暗号化パケットＥＰを受信したノードＮ２は、ノード番号から、そのパケットの送信者がノードＮ１であることを認識するとともに、暗号化パケットＥＰを固定鍵ＦＫを用いて復号し、送信カウンタ値「００００」と乱数「５８１９」を入手する。ノードＮ２はノードＤＢ１１０を参照し、ノードＮ１の受信カウンタ値を保持していない場合には、暗号化パケットＥＰを廃棄する。
【０１５７】
（７）ノードＮ２は、受信カウンタ値を保持している場合には、受信カウンタ値「１２３４」と乱数「５８１９」を固定鍵ＦＫによって暗号化し、ノードＮ１に送信する。ノードＮ２からの暗号化パケットＥＰを受信したノードＮ１は、暗号化パケットＥＰを固定鍵ＦＫを用いて復号し、受信カウンタ値「１２３４」と乱数「５８１９」を入手する。
【０１５８】
ノードＮ１は、入手した乱数と、自ノードが送信した乱数が一致する場合には、ノードＮ２における受信カウンタ値をノードＮ１の送信カウンタ値として更新する。ノードＮ１は、一致しない場合には、受信した暗号化パケットＥＰを不正パケットとして廃棄する。

ここで、乱数を使用するのは、固定鍵ＦＫを持たない攻撃者による不正パケットの送信を検出するためである。

【０１５９】

（送信カウンタ値更新処理）

図１８は、図１６および図１７に示した例における送信カウンタ値更新処理の詳細を示すフローチャートである。まず、ＣＰＵ３０１は、送信カウンタ値が消去されたか否かを検知する（ステップＳ１８０１）。送信カウンタ値が消去されていない場合（ステップＳ１８０１：Ｎｏ）、ＣＰＵ３０１は、ステップＳ１８０１に戻る。

【０１６０】

一方、送信カウンタ値が消去されている場合（ステップＳ１８０１：Ｙｅｓ）、ＣＰＵ３０１は、アクセス鍵ＡＫを受信したか否かを検知する（ステップＳ１８０２）。アクセス鍵ＡＫを受信していない場合（ステップＳ１８０２：Ｎｏ）、ＣＰＵ３０１は、ステップＳ１８０２に戻る。

【０１６１】

一方、アクセス鍵ＡＫを受信した場合（ステップＳ１８０２：Ｙｅｓ）、ＣＰＵ３０１は、乱数を生成し、生成した乱数と受信カウンタ値の要求とをアクセス鍵ＡＫを用いて暗号化して暗号化パケットＥＰを作成する（ステップＳ１８０３）。次に、ＣＰＵ３０１は、作成した暗号化パケットＥＰを近隣ノードに送信する（ステップＳ１８０４）。

【０１６２】

そして、ＣＰＵ３０１は、近隣ノードからの応答を受信したか否かを判定する（ステップＳ１８０５）。応答を受信していない場合（ステップＳ１８０５：Ｎｏ）、ステップＳ１８０５に戻る。

【０１６３】

一方、応答を受信した場合（ステップＳ１８０５：Ｙｅｓ）、ＣＰＵ３０１は、固定鍵ＦＫを用いて応答された暗号化パケットＥＰを復号する（ステップＳ１８０６）。そしてＣＰＵ３０１は、復号したパケットＰ内の乱数がステップＳ１８０３で生成した乱数と一致するか否かを判定する（ステップＳ１８０７）。

【０１６４】

一致する場合（ステップＳ１８０７：Ｙｅｓ）、ＣＰＵ３０１は、パケットＰ内の受信カウンタ値により、自ノードのノードＤＢ１１０内の送信カウンタ値を更新して（ステップＳ１８０８）、送信カウンタ値更新処理を終了する。

【０１６５】

一致しない場合（ステップＳ１８０７：Ｎｏ）、ＣＰＵ３０１は、受信した暗号化パケットＥＰは不正パケットであるとして廃棄し（ステップＳ１８０９）、送信カウンタ値更新処理を終了する。

【０１６６】

これにより、ノードＮは、送信カウンタ値が消去された場合であっても、送信カウンタ値を更新することができ、アドホックネットワークＡに復帰できる。また、近隣ノードから送信されたアクセス鍵ＡＫは固定鍵ＦＫを用いて暗号化されているため、固定鍵ＦＫを有さない攻撃者にはアクセス鍵ＡＫを取得されない。そして、アクセス鍵ＡＫを用いて要求パケットを暗号化しているため、攻撃者による要求パケットの解析を防止できる。

【０１６７】

そして、ノードＮは、要求パケットに近隣ノードとの通信ごとに生成した乱数を含めることにより、正規の応答か否かを、通信ごとに判定する。すなわち、通信の都度、乱数が異なるため、前回の通信で近隣ノードが応答した暗号化パケットＥＰを、今回の要求パケットに対する応答と偽って攻撃者が再送攻撃に利用した場合に、ノードＮは、不正パケットと判定でき通信品質を確保できる。また、一度使用した乱数を記憶しておくようにすれば、今回の通信で近隣ノードが応答した暗号化パケットＥＰを、次回の通信より前に攻撃者が再送攻撃に利用した場合にも、ノードＮは、不正パケットと判定でき通信品質を確保できる。

【０１６８】
さらに、アクセス鍵が一定時間で更新されるため、攻撃者は、暗号化パケットＥＰをキャプチャしても、一定時間経過後には再送攻撃をおこなうことができなくなる。このように、時刻情報ではなく、送信回数の総和を用いることで、再送攻撃を簡単かつ効率的に検出することができ、通信負荷の低減を図ることができる。
【０１６９】
（実施例２）
次に、実施例２について説明する。実施例１では、送信カウンタ値は、送信側のノードにおける暗号化パケットＥＰの送信回数の総和であったが、実施例２では、送信カウンタ値を宛先ノードごとに保持する例である。送信カウンタ値を宛先ノードごとに保持することで、宛先ノードが受信する暗号化パケットＥＰ内の送信カウンタ値は必ず１ずつ増加する。
【０１７０】
よって、攻撃者がキャプチャした暗号化パケットＥＰの送信カウンタ値を改ざんし、より大きい値に変更した場合でも、ノードＮは、より精度よく再送攻撃の判定ができる。また、ノードＮは、改ざんされた送信カウンタ値を受信カウンタ値として誤ってノードＤＢ１１０に更新することもない。このように、時刻情報ではなく、送信回数の総和を用いることで、再送攻撃を簡単かつ効率的に検出することができ、通信負荷の低減を図ることができる。
【０１７１】
（送信カウンタ値ＤＢ１９００の記憶内容）
図１９は、送信カウンタ値ＤＢ１９００の記憶内容の一例を示す説明図である。図１９に示すように、送信カウンタ値ＤＢ１９００は、ノード番号項目のそれぞれに対応付けて、送信カウンタ値項目を有する。ノードＤＢ１１０は、ノードＮが暗号化パケットＥＰを送信する際にレコードを構成する。
【０１７２】
ノード番号項目には、アドホックネットワークＡ内の各ノードＮの識別子（Ｎ１～Ｎｍ）が記憶される。送信カウンタ値項目には、各ノード番号のノードへ暗号化パケットＥＰを送信した回数が記憶される。
【０１７３】
（実施例２にかかる暗号化通信の例）
次に、図２０を用いて、実施例２にかかる暗号化通信の例について説明する。
【０１７４】
図２０は、実施例２にかかるネットワークシステム２００におけるアクセス鍵ＡＫを用いた暗号化通信を示す説明図である。図２０では、アドホックネットワークＡは、ノードＮ１、ノードＮ２およびノードＮ３で構成されているとする。
【０１７５】
（１）ノードＮ２は、ノードＮ１に６回目に暗号化パケットＥＰを送信する際に、自ノード内の記憶装置に記憶されているノードＮ１の送信カウンタ値を「５」から「６」へと更新する。そして、（２）ノードＮ２は、送信するパケットＰに更新した送信カウンタ値「６」を含めて暗号化した暗号化パケットＥＰをノードＮ１に送信する。
【０１７６】
（３）暗号化パケットＥＰを受信したノードＮ１は、暗号化パケットＥＰを復号したパケットＰから、送信元のノード番号と送信カウンタ値とを抽出する。ノードＮ１は、復号したパケットＰから抽出した送信カウンタ値「６」と、自ノード内のノードＤＢ１１０に保持されたノードＮ２の受信カウンタ値「５」とを比較する。この場合、ノードＮ１は、過去にノードＮ２がノードＮ１に５回目に送信した暗号化パケットＥＰを受け取ったことがあり、今回受信した暗号化パケットＥＰはノードＮ２がノードＮ１に６回目に送信したパケットであることが分かる。
【０１７７】

そのため、受信した暗号化パケットＥＰが正規の暗号化パケットＥＰであると判定できる。また、ノードＮ１は、自ノード内のノードＤＢ１１０に、送信カウンタ値「６」を、ノードＮ２に関連付けたあらたな受信カウンタ値として保持する。

【０１７８】

（４）ノードＮ２は、ノードＮ３に９回目に暗号化パケットＥＰを送信する際に、自ノード内の記憶装置に記憶されているノードＮ３の送信カウンタ値を「８」から「９」へと更新する。そして、（５）ノードＮ２は、送信するパケットＰに更新した送信カウンタ値「９」を含めて暗号化した暗号化パケットＥＰをノードＮ３に送信する。

【０１７９】

（６）暗号化パケットＥＰを受信したノードＮ３は、暗号化パケットＥＰを復号したパケットＰから、送信元のノード番号と送信カウンタ値とを抽出する。ノードＮ３は、復号したパケットＰから抽出した送信カウンタ値「９」と、自ノード内のノードＤＢ１１０に保持されたノードＮ２の受信カウンタ値「８」とを比較する。この場合、ノードＮ３は、過去にノードＮ２がノードＮ３に８回目に送信した暗号化パケットＥＰを受け取ったことがあり、今回受信した暗号化パケットＥＰはノードＮ２がノードＮ３に９回目に送信したパケットであることが分かる。

【０１８０】

そのため、受信した暗号化パケットＥＰが正規の暗号化パケットＥＰであると判定できる。また、ノードＮ３は、自ノード内のノードＤＢ１１０に、送信カウンタ値「９」を、ノードＮ２に関連付けたあらたな受信カウンタ値として保持する。このように、送信側のノードＮは、宛先ノードごとに送信回数をカウントし、送信カウンタ値を更新して暗号化パケットＥＰを送信する。これにより、宛先ノードごとに送信カウンタ値を管理することができる。

【０１８１】

実施例２において、暗号化通信におけるパケット送信処理とパケット受信処理と、送信カウンタ値更新処理と、については図１１、図１２、図１５、図１８に示した処理と同様のため説明を省略する。

【０１８２】

以上説明したように、本実施の形態にかかる通信方法、ノードおよびネットワークシステムによれば、暗号化パケットＥＰの送信側のノードＮが、送信カウンタ値を暗号化パケットＥＰに含めておく。受信側のノードＮは、前回受信した暗号化パケットＥＰの送信カウンタ値と、今回受信した暗号化パケットＥＰの送信カウンタ値とを比較することで、不正パケットを検出する。

【０１８３】

そのため、ノードＮは、近隣ノードから送信された正規の暗号化パケットＥＰは廃棄せず、攻撃者のノードＮｃから送信された不正パケットのみを廃棄することができるため、再送攻撃を防ぐことができ、通信の安全性を担保することができる。また、ノードＮは、不正パケットを廃棄するため、不正パケットの内容に基づいて処理をおこなう必要がなく、ノードＮの処理負担を軽減できる。

【０１８４】

また、時刻同期を必要とせず不正パケットを判定することができるため、時刻同期用のパケットによる通信負荷が生じないようにできる。さらに、時刻同期を必要としないため、時刻同期用のパケットを送信するゲートウェイがアドホックネットワークＡ内にない場合でも不正パケットを判定できる。

【０１８５】

また、送信カウンタ値が消去された場合でも、近隣ノードが保持する受信カウンタ値の要求パケットを送信することにより、早期のアドホックネットワークＡへの復帰ができる。また、この際、アクセス鍵ＡＫの受信を待ってアクセス鍵ＡＫを用いて要求パケットを暗号化することで、要求パケットの再送攻撃を防止することができる。また、要求パケットに、乱数を含めておくことにより、近隣ノードから応答された暗号化パケットＥＰと、

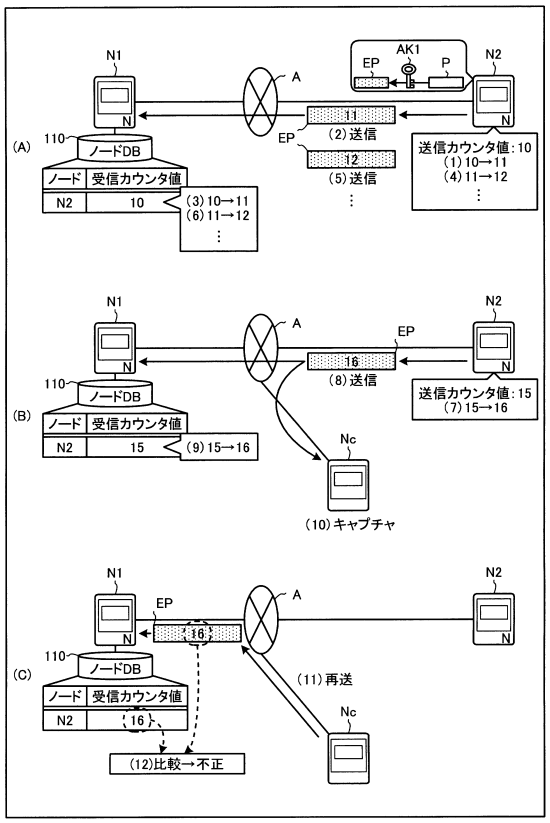
不正パケットとを判定することができる。

【符号の説明】

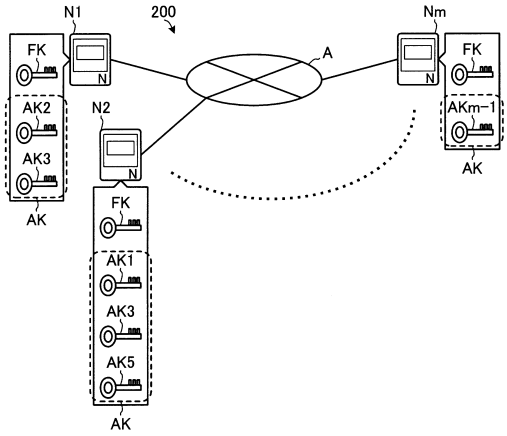
【０１８６】

Ａ　アドホックネットワーク
Ｎ　ノード
ＡＫ　アクセス鍵
ＦＫ　固定鍵
Ｐ　パケット
ＥＰ　暗号化パケット
５０１　第１の受信部
５０２　第１の抽出部
５０３　第２の受信部
５０４　第２の抽出部
５０５　判定部
５０６　廃棄部
５０７　データ処理部
６０１　検出部
６０２　更新部
６０３　送信部
６０４　検知部
６０５　通知部
６０６　受信部
６０７　抽出部
６０８　格納部
６０９　暗号化部
６１０　復号部
６１１　生成部
６１２　鍵受信部
６１３　鍵復号部

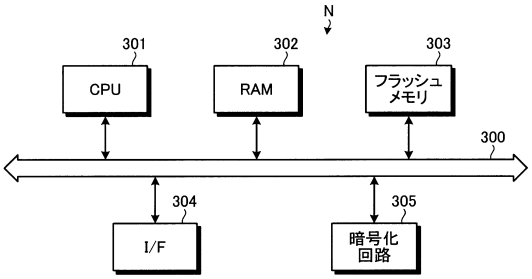
【図１】



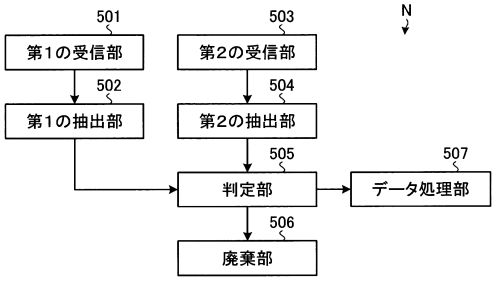
【図２】



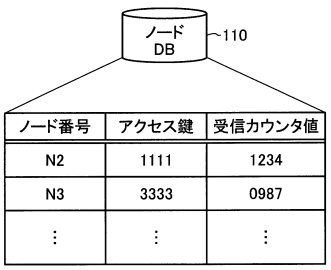
【図３】



【図５】

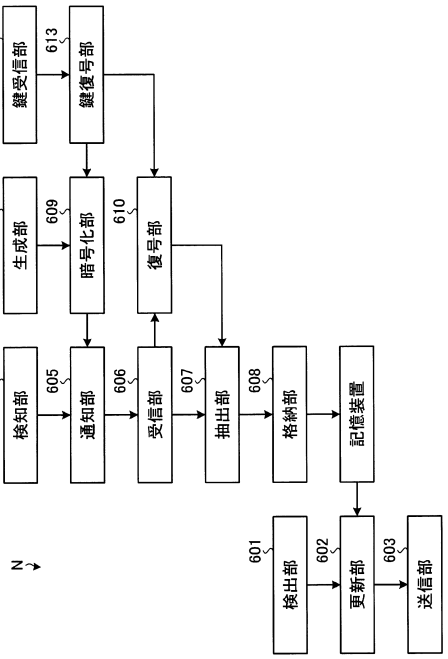


【図４】

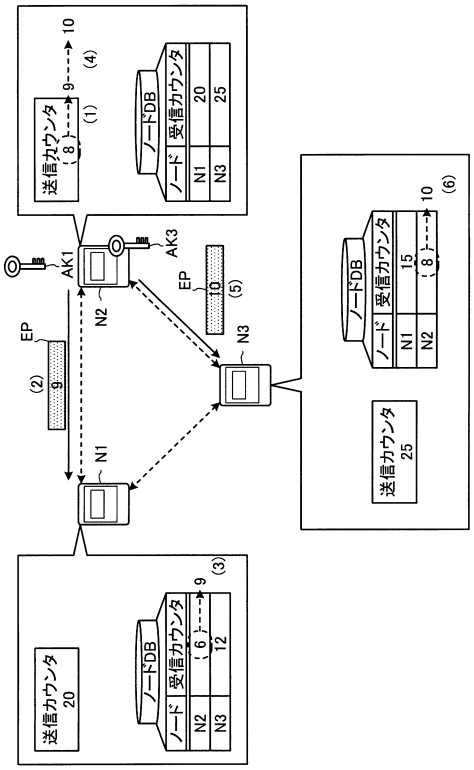


ノード番号	アクセス鍵	受信カウンタ値
N2	1111	1234
N3	3333	0987
⋮	⋮	⋮

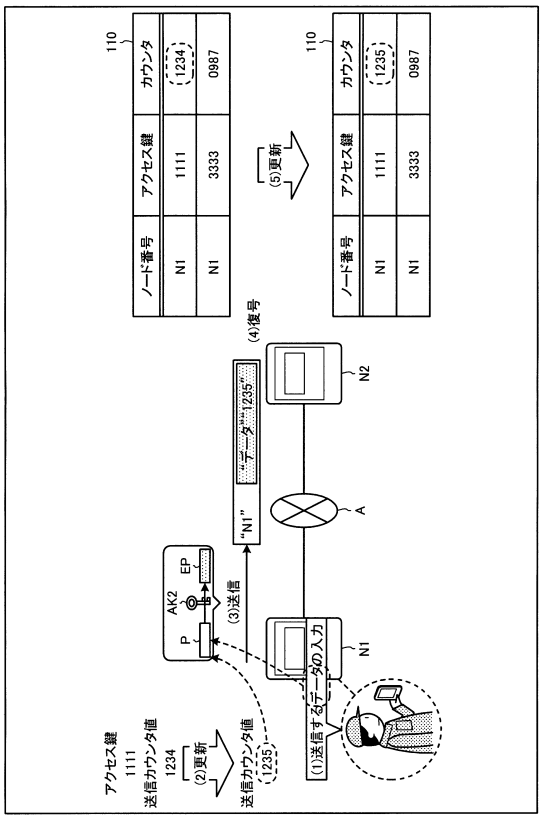
【図６】


612
鍵受信部
613
鍵復号部
611
生成部
609
暗号化部
610
復号部
604
検知部
605
通知部
606
受信部
607
抽出部
608
格納部
記憶装置
N
601
検出部
602
更新部
603
送信部

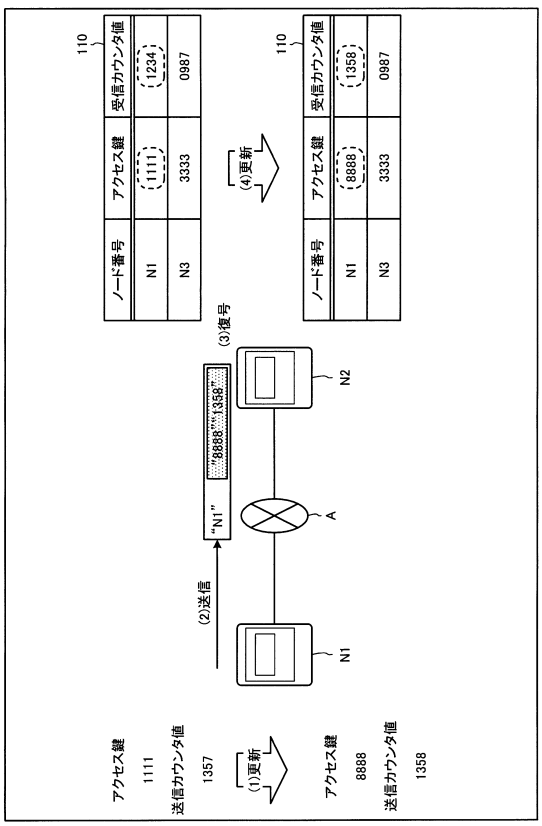
【図７】


送信カウンタ
8
9
10
(1)
(4)
ノードDB
ノード
受信カウンタ
N1
20
N3
25
AK1
AK3
N2
EP
10
(5)
N3
EP
(2)
9
N1
送信カウンタ
20
ノードDB
ノード
受信カウンタ
N2
6
12
N3
9
(3)
送信カウンタ
25
ノードDB
ノード
受信カウンタ
N1
15
N2
8
10
(6)

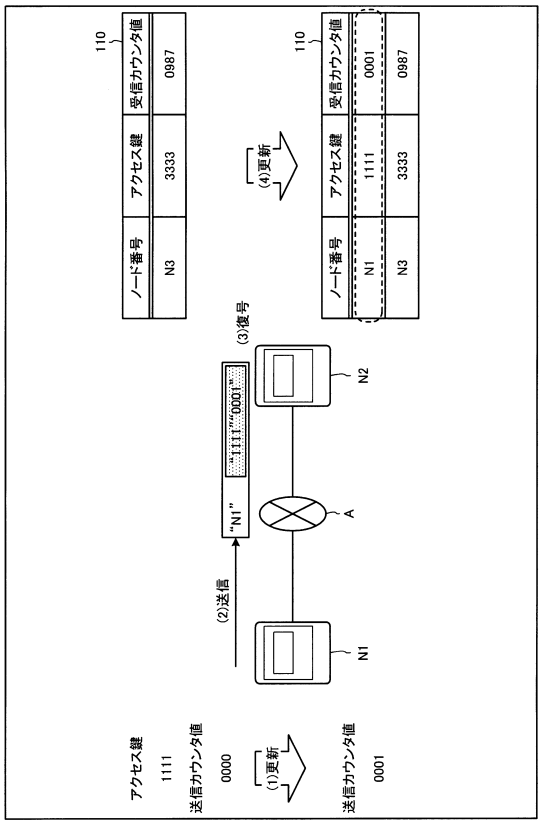
【図８】


110
ノード番号
アクセス鍵
カウンタ
N1
1111
1234
N1
3333
0987
(5)更新
110
ノード番号
アクセス鍵
カウンタ
N1
1111
1235
N1
3333
0987
(4)復号
N2
A
"N1"
"データ""1235"
AK2
EP
P
(3)送信
N1
(1)送信するデータのD入力
アクセス鍵
1111
送信カウンタ値
1234
(2)更新
送信カウンタ値
1235

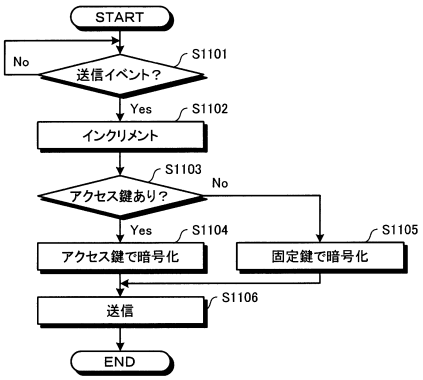
【図９】


110
ノード番号
アクセス鍵
受信カウンタ値
N1
1111
1234
N3
3333
0987
(4)更新
110
ノード番号
アクセス鍵
受信カウンタ値
N1
8888
1358
N3
3333
0987
(3)復号
N2
A
"N1"
"8888""1358"
(2)送信
N1
アクセス鍵
1111
送信カウンタ値
1357
(1)更新
アクセス鍵
8888
送信カウンタ値
1358

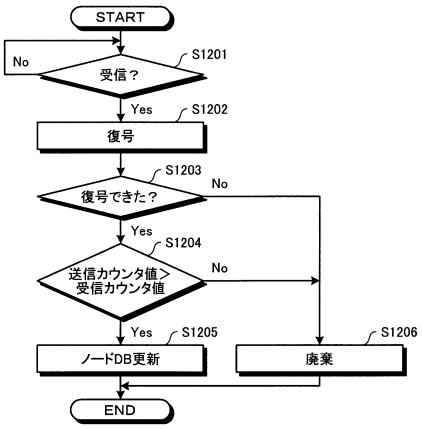
【図１０】



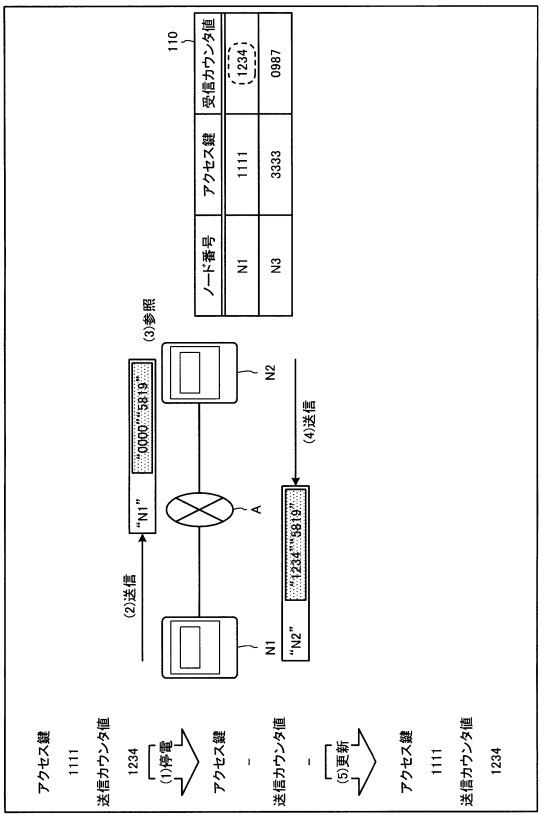
【図１１】



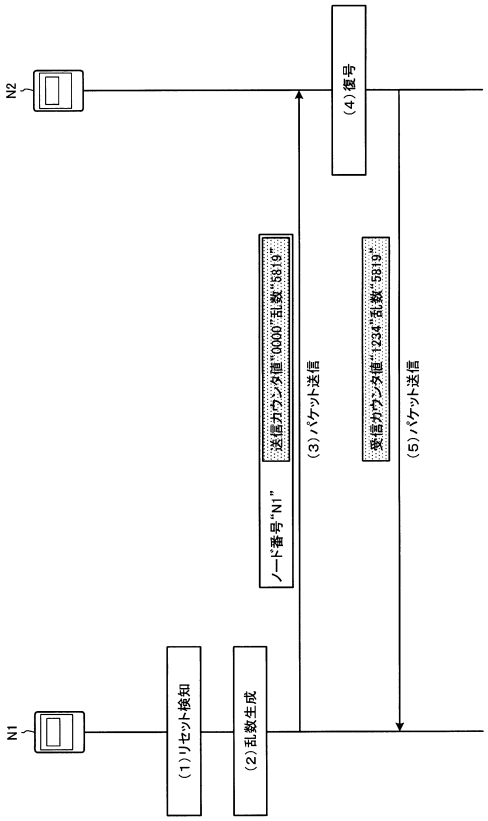
【図１２】



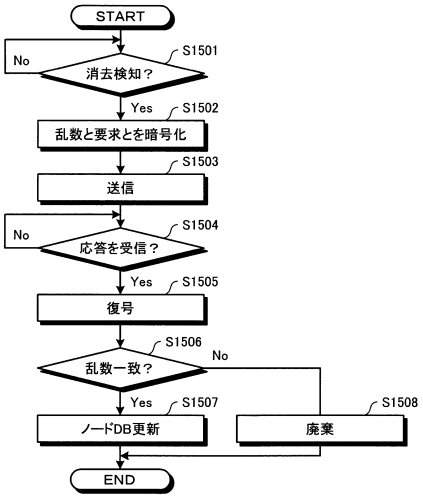
【図１３】



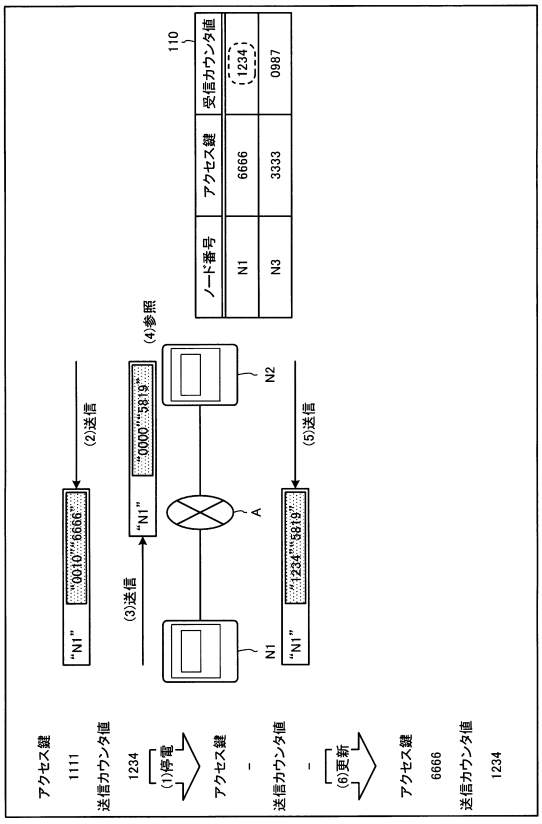
【図１４】



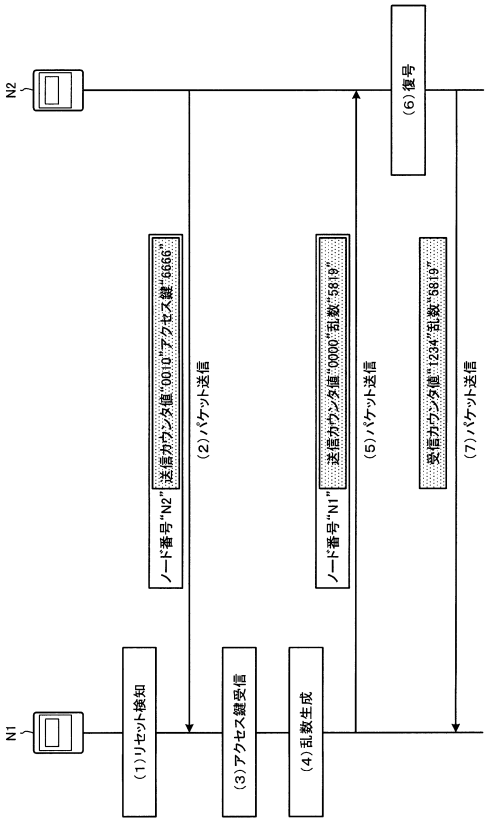
【図１５】



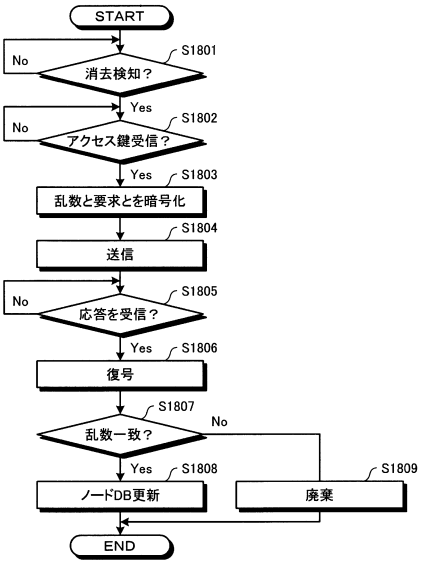
【図１６】



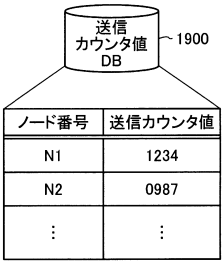
【図１７】



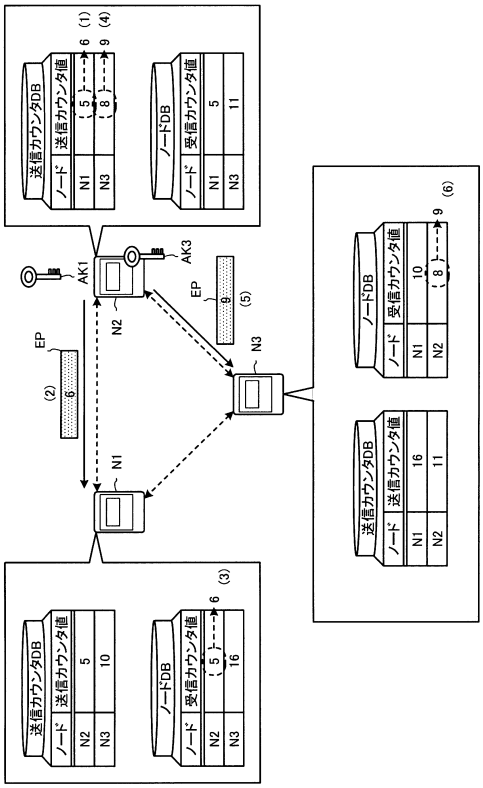
【図１８】


START
S1801
No
消去検知？
Yes
S1802
No
アクセス鍵受信？
Yes
S1803
乱数と要求とを暗号化
S1804
送信
S1805
No
応答を受信？
Yes
S1806
復号
S1807
乱数一致？
No
Yes
S1808
ノードDB更新
S1809
廃棄
END

【図１９】


送信
カウンタ値
DB
1900
ノード番号
送信カウンタ値
N1
1234
N2
0987

【図２０】


(1)
(4)
(2)
(3)
(5)
(6)
AK1
AK3
EP
N1
N2
N3
送信カウンタDB
ノードDB
ノード
送信カウンタ値
受信カウンタ値

フロントページの続き

(72)発明者　兒島　尚
神奈川県川崎市中原区上小田中４丁目１番１号　富士通株式会社内

審査官　打出　義尚

(56)参考文献　特開２００４－２４１８６５（ＪＰ，Ａ）
特開２００３－２８３４８９（ＪＰ，Ａ）
特開２００８－１０８２３０（ＪＰ，Ａ）
特開２０１１－０６６７０３（ＪＰ，Ａ）
特開２０１１－１６０２１０（ＪＰ，Ａ）
Chin-Tser Huang, et al., Convergence of IPsec in Presence of Resets, Proceedings of the 23rd International Conference on Distributed Computing Systems Workshops, 2003 (I, IEEE, ２００３年　5月１９日, Pages 22-27

(58)調査した分野(Int.Cl., ＤＢ名)
Ｇ０６Ｆ　２１／５５