(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号

特許第5526747号
(P5526747)

(45)発行日　平成26年6月18日(2014.6.18)　(24)登録日　平成26年4月25日(2014.4.25)

(51)Int.Cl.			F I		
H04L	9/20	(2006.01)	H04L	9/00	653
H04L	9/36	(2006.01)	H04L	9/00	685
H04L	9/08	(2006.01)	H04L	9/00	601C

請求項の数 24　(全 32 頁)

(21)出願番号　特願2009-276294 (P2009-276294)
(22)出願日　平成21年12月4日 (2009.12.4)
(65)公開番号　特開2011-120051 (P2011-120051A)
(43)公開日　平成23年6月16日 (2011.6.16)
審査請求日　平成24年11月20日 (2012.11.20)

(73)特許権者　000005821
パナソニック株式会社
大阪府門真市大字門真１００６番地
(74)代理人　100104732
弁理士　徳田　佳昭
(74)代理人　100120156
弁理士　藤井　兼太郎
(74)代理人　100137202
弁理士　寺内　伊久郎
(72)発明者　松尾　正克
福岡県福岡市博多区美野島４丁目１番６２号　パナソニックコミュニケーションズ株式会社内

審査官　青木　重徳

最終頁に続く

(54)【発明の名称】復号化装置、暗号化装置、復号化方法、暗号化方法、および通信システム

(57)【特許請求の範囲】
【請求項１】
パケットに含まれる暗号文データを復号化する復号化装置であって、
暗号処理を実行する暗号化装置から送信されるパケットを受信する受信手段と、
前記パケットに対応しかつ前記パケットを識別するためのパケット情報を、共通鍵を用いて暗号化し、暗号化されたパケット情報を生成する鍵生成手段と、
前記鍵生成手段により生成される暗号化されたパケット情報をシードとして、乱数を生成する乱数生成手段と、
前記乱数生成手段により生成される乱数に基づいて、前記受信手段により受信されたパケットに含まれる暗号文データを復号化する復号化手段とを有することを特徴とする復号化装置。
【請求項２】
前記復号化手段は、前記乱数生成手段により生成された乱数と、前記パケットに含まれる暗号文データとを、ＸＯＲ演算することを特徴とする請求項１記載の復号化装置。
【請求項３】
前記鍵生成手段は、少なくとも１以上のパケットおきにシードを生成することを特徴とする請求項２記載の復号化装置。
【請求項４】
前記暗号化装置から、第１パケット、・・・、第Ｍパケット、・・・、第Ｎパケット（１≦Ｍ＜Ｎ、Ｎは２以上の整数）を当該復号化装置に順次、送信され、

前記復号化装置は、さらに、前記第１パケットから前記第Ｍ－１パケットまでの通信データ数の総和に応じて、前記乱数生成手段が生成した乱数から乱数列を設定する乱数列設定手段を有し、

前記復号化手段は、前記乱数列設定手段が設定した乱数列と、前記第Ｎパケットに含まれる暗号文データとを、ＸＯＲ演算することを特徴とする請求項２記載の復号化装置。

【請求項５】

前記第Ｍパケットには、前記第１パケットから前記第Ｍ－１パケットまでの通信データ数の総和を示す情報が含まれており、前記乱数列設定手段は、前記第Ｍパケットが示す情報を用いて乱数列を設定することを特徴とする請求項４記載の復号化装置。

【請求項６】

前記鍵生成手段は、逆関数を有する暗号方式に基づいてパケット情報を暗号化することにより、前記暗号化されたパケット情報を生成することを特徴とする請求項１記載の復号化装置。

【請求項７】

前記鍵生成手段は、前記パケット情報をブロック毎に暗号化することにより、前記暗号化されたパケット情報を生成することを特徴とする請求項６記載の復号化装置。

【請求項８】

前記パケット情報は、前記パケットを識別自在なカウンタデータであることを特徴とする請求項１記載の復号化装置。

【請求項９】

前記カウンタデータは、前記パケットの通し番号であることを特徴とする請求項８記載の復号化装置。

【請求項１０】

前記パケットは、コネクションレス型のプロトコルに基づくことを特徴とする請求項１記載の復号化装置。

【請求項１１】

前記コネクションレス型のプロトコルケットは、ＵＤＰ（Ｕｓｅｒ　Ｄａｔａｇｒａｍ　Ｐｒｏｔｏｃｏｌ）を含むことを特徴とする請求項１０記載の復号化装置。

【請求項１２】

平文データを暗号化し、暗号文データを復号化装置に送信する暗号化装置であって、

パケットに対応しかつ前記パケットを識別するためのパケット情報を、共通鍵を用いて暗号化し、暗号化されたパケット情報を生成する鍵生成手段と、

前記鍵生成手段により生成される暗号化されたパケット情報をシードとして、乱数を生成する乱数生成手段と、

前記乱数生成手段により生成される乱数に基づいて平文データを暗号化し、暗号文データを生成する暗号化手段と、

前記暗号文データと前記パケット情報とを含むパケットを、前記復号化装置に送信する送信手段とを有することを特徴とする暗号化装置。

【請求項１３】

前記暗号化手段は、前記乱数生成手段により生成された乱数と、前記平文データとを、ＸＯＲ演算することを特徴とする請求項１２記載の暗号化装置。

【請求項１４】

前記鍵生成手段は、少なくとも１以上のパケットおきにシードを生成することを特徴とする請求項１３記載の暗号化装置。

【請求項１５】

前記送信手段は、第１パケット、・・・、第Ｍパケット、・・・、第Ｎパケット（１≦Ｍ＜Ｎ、Ｎは２以上の整数）を順次、前記復号化装置に送信し、

前記第Ｍパケットには、前記第１パケットから前記第Ｍ－１パケットまでの通信データ数の総和を示す情報が含まれていることを特徴とする請求項１３記載の暗号化装置。

【請求項１６】

前記鍵生成手段は、逆関数を有する暗号方式に基づいてパケット情報を暗号化することにより、前記暗号化されたパケット情報を生成することを特徴とする請求項１２記載の暗号化装置。
【請求項１７】
前記鍵生成手段は、前記パケット情報をブロック毎に暗号化することにより、前記暗号化されたパケット情報を生成することを特徴とする請求項１６記載の暗号化装置。
【請求項１８】
前記パケット情報は、前記パケットを識別自在なカウンタデータであることを特徴とする請求項１２記載の暗号化装置。
【請求項１９】
前記カウンタデータは、前記パケットの通し番号であることを特徴とする請求項１８記載の暗号化装置。
【請求項２０】
前記パケットは、コネクションレス型のプロトコルに基づくことを特徴とする請求項１２記載の暗号化装置。
【請求項２１】
前記コネクションレス型のプロトコルケットは、ＵＤＰ（Ｕｓｅｒ　Ｄａｔａｇｒａｍ　Ｐｒｏｔｏｃｏｌ）を含むことを特徴とする請求項２０記載の暗号化装置。
【請求項２２】
パケットに含まれる暗号文データを復号化する復号化方法であって、
　暗号処理を実行する暗号化装置から送信されるパケットを受信する受信ステップと、
　前記パケットに対応しかつ前記パケットを識別するためのパケット情報を、共通鍵を用いて暗号化し、暗号化されたパケット情報を生成する鍵生成ステップと、
　前記鍵生成ステップで生成される暗号化されたパケット情報をシードとして、乱数を生成する乱数生成ステップと、
　前記乱数生成ステップで生成される乱数に基づいて、前記受信ステップで受信されたパケットに含まれる暗号文データを復号化する復号化ステップとを有することを特徴とする復号化方法。
【請求項２３】
平文データを暗号化し、暗号文データを復号化装置に送信する暗号化方法であって、
　パケットに対応しかつ前記パケットを識別するためのパケット情報を、共通鍵を用いて暗号化し、暗号化されたパケット情報を生成する鍵生成ステップと、
　前記鍵生成ステップで生成される暗号化されたパケット情報をシードとして、乱数を生成する乱数生成ステップと、
　前記乱数生成ステップで生成される乱数に基づいて平文データを暗号化し、暗号文データを生成する暗号化ステップと、
　前記暗号文データと前記パケット情報とを含むパケットを、前記復号化装置に送信する送信ステップとを有することを特徴とする暗号化方法。
【請求項２４】
通信システムであって、
　前記通信システムは、
　平文データを暗号化し、パケットに含まれる暗号文データを通信回線に送信する暗号化装置と、
　前記暗号化装置から通信回線を介してパケットを受信し、前記パケットに含まれる暗号文データを復号化する復号化装置とを備え、
　前記暗号化装置は、
　前記パケットに対応しかつ前記パケットを識別するためのパケット情報を、共通鍵を用いて暗号化し、暗号化されたパケット情報を生成する第１の鍵生成手段と、
　前記第１の鍵生成手段により生成される暗号化されたパケット情報をシードとして、乱数を生成する第１の乱数生成ステップと、

前記第１の乱数生成手段により生成される乱数に基づいて平文データを暗号化し、暗号文データを生成する暗号化手段と、
前記暗号文データと前記パケット情報とを含むパケットを、前記復号化装置に送信する送信手段とを有し、
前記復号化装置は、
前記暗号化装置から送信されるパケットを受信する受信手段と、
前記パケット情報を、前記共通鍵を用いて暗号化し、前記暗号化されたパケット情報を生成する第２の鍵生成手段と、
前記第２の鍵生成手段により生成される暗号化されたパケット情報をシードとして、乱数を生成する第２の乱数生成手段と、
前記第２の乱数生成手段により生成される乱数に基づいて、前記受信手段により受信されたパケットに含まれる暗号文データを復号化する復号化手段とを有することを特徴とする通信システム。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明は、パケットロスやパケットの到着順が逆になることが起こりえる通信方式であっても、送信側と受信側の間の同期ずれを抑制することの出来る、復号化装置、暗号化装置、復号化方法、暗号化方法、および通信システムに関するものである。
【背景技術】
【０００２】
近年、各種通信において、インターネットを利用するケースが増大している。インターネットが普及した当初は、非リアルタイム通信であるＥ－ｍａｉｌやＷｅｂが主流であった。しかし、インターネット技術の進歩に伴い、最近では、ＴＶや、電話や、監視カメラといった音声・画像系のリアルタイム通信が、インターネットでも多く利用されるようになった。
【０００３】
インターネットは、ＱｏＳ（Ｑｕａｌｉｔｙ ｏｆ Ｓｅｒｖｉｃｅ）を考慮していないので、リアルタイム通信には適していない。従って、インターネット経由の通信でリアルタイム性を向上するには、通信速度を向上することが望ましい。このような状況下、セキュリティ確保をしつつ、通信速度を向上させるために、ストリーミング暗号などの暗号方式が開示されている（例えば、特許文献１参照）。
【０００４】
一方、利用ユーザ数や転送データ容量の急激な増大（例えば画像通信におけるハイビジョン化など）もあり、インターネットの通信速度向上は、必ずしもリアルタイム性向上にはつながっていない。このような背景から、リアルタイム通信では、通信速度の遅いＴＣＰ（Ｔｒａｎｓｍｉｓｓｉｏｎ Ｃｏｎｔｒｏｌ Ｐｒｏｔｏｃｏｌ）通信ではなく、より速いＵＤＰ（Ｕｓｅｒ Ｄａｔａｇｒａｍ Ｐｒｏｔｏｃｏｌ）通信が採用されることが多い。
【０００５】
しかし、ＵＤＰ通信ではパケットロスやパケット到着順が逆になる場合があるので、共通鍵暗号化方式に特別の工夫を施さなくてはならない。例えば、ＳＳＬ暗号通信（ＴＣＰ通信）の共通鍵暗号化方式で最もよく利用されているストリーミング暗号のＡｒｃｆｏｕｒ暗号やブッロク暗号のＡＥＳ ＣＢＣモードでは、パケットロスやパケット到着順が逆になるようなことが起こると、同期ずれが起こり、それ以降、受信側で正しく復号できなくなる不都合がある。
【先行技術文献】
【特許文献】
【０００６】
【特許文献１】特開２００７－３３６４９号公報

【発明の概要】
【発明が解決しようとする課題】
【０００７】
解決しようとする問題点は、パケットロスやパケット到着順が逆になることが起こり得る通信方式であっても、同期ずれなく暗号・復号処理を行うことの出来る暗号方式を提供することにある。
【課題を解決するための手段】
【０００８】
本発明の復号化装置は、パケットに含まれる暗号文データを復号化する復号化装置であって、暗号処理を実行する暗号化装置から送信されるパケットを受信する受信手段と、暗号処理に用いられる鍵と同じ鍵を生成する鍵生成手段と、鍵生成手段により生成される鍵に基づいて、受信手段により受信されたパケットに含まれる暗号文データを復号化する復号化手段とを有し、受信手段により受信されたパケットには、鍵の生成に用いられかつパケットを識別するためのパケット情報が含まれており、鍵生成手段は、パケット情報を用いて鍵を生成する構成を備えている。
【０００９】
本発明の暗号化装置は、平文データを暗号化し、暗号文データを復号化装置に送信する暗号化装置であって、パケットに対応しかつパケットを識別するためのパケット情報を用いて、鍵を生成する鍵生成手段と、鍵生成手段により生成される鍵に基づいて平文データを暗号化し、暗号文データを生成する暗号化手段と、暗号文データとパケット情報とを含むパケットを、復号化装置に送信する送信手段とを有する構成を備えている。
【発明の効果】
【００１０】
本発明の復号化装置は、鍵の生成に用いられたパケット情報と、この鍵を用いて暗号化した平文データを、同一のパケットでまとめて受信するので、鍵の生成に必要な情報を、暗号文データの復号化に使用することが出来る。これにより、パケットが送信される途中で、パケットがロストしたり、パケットの順番が入れ替わったりした場合であっても、暗号化装置と復号化装置の間の同期ずれを抑制することが出来る。
【００１１】
本発明の暗号化装置は、暗号化装置は、鍵の生成に用いられたパケット情報と、この鍵を用いて暗号化した平文データを、同一のパケットでまとめて復号化装置に送信するので、パケットが送信される途中で、パケットがロストしたり、パケットの順番が入れ替わったりした場合であっても、復号化装置は、暗号文データの復号化に際して、当該復号化処理の鍵生成に必要な情報を受信することが出来る。
【図面の簡単な説明】
【００１２】
【図１】実施例１における通信システムの機能ブロック図
【図２】実施例１における通信システムの動作を示すタイミングチャート
【図３】実施例１における送信装置の動作の説明図
【図４】実施例１における送信装置の動作を示すフローチャート
【図５】実施例１におけるカウンタデータの実装する方法の説明図
【図６】実施例１におけるカウンタデータ算出方法のフローチャート
【図７】実施例１における受信装置の動作を示すフローチャート
【図８】実施例１における処理速度の測定結果を示す図
【図９】実施例１における、ＭＡＣチェックの説明図
【図１０】実施例１におけるＭＡＣチェックの説明図
【図１１】実施例２における通信システムの動作を示すタイミングチャート
【図１２】実施例２における送信装置の動作を示すフローチャート
【図１３】実施例２における受信装置の動作を示すフローチャート
【図１４】実施例３における監視システムの全体図

【図１５】実施例３における監視システムの機能ブロック図
【図１６】実施例４における通信システムの動作を示すタイミングチャート
【図１７】実施例５における通信システムの動作を示すタイミングチャート
【発明を実施するための形態】
【００１３】
本発明の復号化装置は、パケットに含まれる暗号文データを復号化する復号化装置であって、暗号処理を実行する暗号化装置から送信されるパケットを受信する受信手段と、暗号処理に用いられる鍵と同じ鍵を生成する鍵生成手段と、鍵生成手段により生成される鍵に基づいて、受信手段により受信されたパケットに含まれる暗号文データを復号化する復号化手段とを有し、受信手段により受信されたパケットには、鍵の生成に用いられかつパケットを識別するためのパケット情報が含まれており、鍵生成手段は、パケット情報を用いて鍵を生成する構成を備えている。これにより、復号化装置は、鍵の生成に用いられたパケット情報と、この鍵を用いて暗号化した平文データを、同一のパケットでまとめて受信するので、鍵の生成に必要な情報を、暗号文データの復号化に使用することが出来る。これにより、パケットが送信される途中で、パケットがロストしたり、パケットの順番が入れ替わったりした場合であっても、暗号化装置と復号化装置の間の同期ずれを抑制することが出来る。
【００１４】
さらに、復号化装置は、さらに、鍵生成手段により生成される鍵をシードとして、乱数を生成する乱数生成手段を有し、復号化手段は、乱数生成手段により生成された乱数と、パケットに含まれる暗号文データとを、ＸＯＲ演算する構成を備えている。これにより、暗号処理に処理速度の速い乱数生成処理を利用するので、復号化処理の高速化を図ることが出来る。
【００１５】
さらに、鍵生成手段は、少なくとも１以上のパケットおきにシードを生成する構成を備えている。これにより、パケットあたりのシード生成の処理回数が少なくなるので、復号化処理をさらに高速化させることが出来る。
【００１６】
さらに、暗号化装置から、第１パケット、・・・、第Ｍパケット、・・・、第Ｎパケット（１≦Ｍ＜Ｎ、Ｎは２以上の整数）を当該復号化装置に順次、送信され、復号化装置は、さらに、受信手段により受信された第Ｍパケットが示す通信データ数の総和に応じて、乱数生成手段が生成した乱数から乱数列を設定する乱数列設定手段を有し、復号化手段は、乱数列設定手段が設定した乱数列と、第Ｎパケットに含まれる暗号文データとを、ＸＯＲ演算する構成を備えている。これにより、通信データ数の総和によって、第Ｎパケットの復号化に必要な乱数列を設定することが出来る。これにより、パケットが送信される途中で、パケットがロストしたり、パケットの順番が入れ替わったりした場合であっても、復号化装置は、必要な分だけ乱数を生成して、同期ずれの抑制を向上させることが出来る。
【００１７】
さらに、第Ｍパケットには、第１パケットから第Ｍ－１パケットまでの通信データ数の総和を示す情報が含まれており、乱数列設定手段は、第Ｍパケットが示す情報を用いて乱数列を設定する構成を備えている。
【００１８】
さらに、鍵生成手段は、逆関数を有する暗号方式に基づいてパケット情報を暗号化することにより、鍵を生成する構成を備えている。これにより、パケット情報から鍵が一意的に定まるので、暗号文データに最適な鍵を用いて復号することが出来る。
【００１９】
さらに、鍵生成手段は、パケット情報をブロック毎に暗号化することにより、鍵を生成する構成を備えている。
【００２０】

さらに、パケット情報は、パケットを識別自在なカウンタデータである構成を備えている。これにより、パケットのカウンタデータを利用するので、簡易な構成を実現することが出来る。

【００２１】

さらに、カウンタデータは、パケットの通し番号である構成を備えている。これにより、同じカウンタデータが出現しないので、セキュリティレベルを向上させることが出来る。

【００２２】

さらに、パケットは、コネクションレス型のプロトコルに基づく構成を備えている。これにより、コネクションレス型のため、パケットが送信される途中で、パケットがロストしたり、パケットの順番が入れ替わったりした場合があり得るが、このような場合であっても、暗号化装置と復号化装置の間の同期ずれを抑制することが出来る。

【００２３】

さらに、コネクションレス型のプロトコルケットは、ＵＤＰ（Ｕｓｅｒ　Ｄａｔａｇｒａｍ　Ｐｒｏｔｏｃｏｌ）を含む構成を備えている。これにより、ＵＤＰでは、パケットが送信される途中で、パケットがロストしたり、パケットの順番が入れ替わったりした場合があり得るが、このような場合であっても、暗号化装置と復号化装置の間の同期ずれを抑制することが出来る。

【００２４】

本発明の復号化装置は、平文データを暗号化し、暗号文データを復号化装置に送信する暗号化装置であって、パケットに対応しかつパケットを識別するためのパケット情報を用いて、鍵を生成する鍵生成手段と、鍵生成手段により生成される鍵に基づいて平文データを暗号化し、暗号文データを生成する暗号化手段と、暗号文データとパケット情報とを含むパケットを、復号化装置に送信する送信手段とを有する構成を備えている。これにより、暗号化装置は、鍵の生成に用いられたパケット情報と、この鍵を用いて暗号化した平文データを、同一のパケットでまとめて復号化装置に送信するので、パケットが送信される途中で、パケットがロストしたり、パケットの順番が入れ替わったりした場合であっても、復号化装置は、暗号文データの復号化に際して、当該復号化処理の鍵生成に必要な情報を受信することが出来る。

【００２５】

さらに、暗号化装置は、さらに、鍵生成手段により生成される鍵をシードとして、乱数を生成する乱数生成手段を有し、暗号化手段は、乱数生成手段により生成された乱数と、パケットに含まれる暗号文データとを、ＸＯＲ演算する構成を備えている。これにより、暗号処理に処理速度の速い乱数生成処理を利用するので、暗号化処理の高速化を図ることが出来る。

【００２６】

さらに、鍵生成手段は、少なくとも１以上のパケットおきにシードを生成する構成を備えている。これにより、パケットあたりのシード生成の処理回数が少なくなるので、暗号化処理をさらに高速化させることが出来る。

【００２７】

さらに、送信手段は、第１パケット、・・・、第Ｍパケット、・・・、第Ｎパケット（１≦Ｍ＜Ｎ、Ｎは２以上の整数）を順次、復号化装置に送信し、第Ｍパケットには、第１パケットから第Ｍ－１パケットまでの通信データ数の総和を示す情報が含まれている構成を備えている。これにより、暗号化装置は、第Ｎパケットの復号化に必要な情報を、復号化装置に通知することが出来る。

【００２８】

さらに、鍵生成手段は、逆関数を有する暗号方式に基づいてパケット情報を暗号化することにより、鍵を生成する構成を備えている。これにより、パケット情報から鍵が一意的に定まるので、暗号文データに最適な鍵を用いて復号することが出来る。

【００２９】

さらに、鍵生成手段は、パケット情報をブロック毎に暗号化することにより、鍵を生成する構成を備えている。

【００３０】

さらに、パケット情報は、パケットを識別自在なカウンタデータである構成を備えている。これにより、パケットのカウンタデータを利用するので、簡易な構成を実現することが出来る。

【００３１】

さらに、カウンタデータは、パケットの通し番号である構成を備えている。これにより、同じカウンタデータが出現しないので、セキュリティレベルを向上させることが出来る。

【００３２】

さらに、パケットは、コネクションレス型のプロトコルに基づく構成を備えている。これにより、コネクションレス型のため、パケットが送信される途中で、パケットがロストしたり、パケットの順番が入れ替わったりした場合があり得るが、このような場合であっても、暗号化装置と復号化装置の間の同期ずれを抑制することが出来る。

【００３３】

さらに、コネクションレス型のプロトコルケットは、ＵＤＰ（Ｕｓｅｒ　Ｄａｔａｇｒａｍ　Ｐｒｏｔｏｃｏｌ）を含む構成を備えている。これにより、ＵＤＰでは、パケットが送信される途中で、パケットがロストしたり、パケットの順番が入れ替わったりした場合があり得るが、このような場合であっても、暗号化装置と復号化装置の間の同期ずれを抑制することが出来る。

【００３４】

本発明の復号化方法は、パケットに含まれる暗号文データを復号化する復号化方法であって、暗号処理を実行する暗号化装置から送信されるパケットを受信する受信ステップと、暗号処理に用いられる鍵と同じ鍵を生成する鍵生成ステップと、鍵生成ステップで生成される鍵に基づいて、受信ステップで受信されたパケットに含まれる暗号文データを復号化する復号化ステップとを有し、受信ステップで受信されたパケットには、鍵の生成に用いられかつパケットを識別するためのパケット情報が含まれており、鍵生成ステップで、パケット情報を用いて鍵を生成する構成を備えている。これにより、復号化方法は、鍵の生成に用いられたパケット情報と、この鍵を用いて暗号化した平文データを、同一のパケットでまとめて受信するので、鍵の生成に必要な情報を、暗号文データの復号化に使用することが出来る。これにより、パケットが送信される途中で、パケットがロストしたり、パケットの順番が入れ替わったりした場合であっても、暗号化装置と復号化装置の間の同期ずれを抑制することが出来る。

【００３５】

本発明の暗号化方法は、平文データを暗号化し、暗号文データを復号化装置に送信する暗号化方法であって、パケットに対応しかつパケットを識別するためのパケット情報を用いて、鍵を生成する鍵生成ステップと、鍵生成ステップで生成される鍵に基づいて平文データを暗号化し、暗号文データを生成する暗号化ステップと、暗号文データとパケット情報とを含むパケットを、復号化装置に送信する送信ステップとを有する構成を備えている。これにより、暗号化方法は、鍵の生成に用いられたパケット情報と、この鍵を用いて暗号化した平文データを、同一のパケットでまとめて復号化装置に送信するので、パケットが送信される途中で、パケットがロストしたり、パケットの順番が入れ替わったりした場合であっても、復号化装置は、暗号文データの復号化に際して、当該復号化処理の鍵生成に必要な情報を受信することが出来る。

【００３６】

本発明の通信方法は、平文データを暗号化し、パケットに含まれる暗号文データを通信回線に送信する暗号化装置と、暗号化装置から通信回線を介してパケットを受信し、パケットに含まれる暗号文データを復号化する復号化装置とを備え、暗号化装置は、パケットに対応するパケット情報を用いて、鍵を生成する第１の鍵生成手段と、第１の鍵生成手段

により生成される鍵と同じ鍵に基づいて平文データを暗号化し、暗号文データを生成する暗号化手段と、暗号文データとパケット情報とを含むパケットを、復号化装置に送信する送信手段とを有し、復号化装置は、暗号化装置から送信されるパケットを受信する受信手段と、受信手段により受信されたパケットに含まれかつパケットを識別するためのパケット情報を用いて、鍵を生成する第２の鍵生成手段と、第２の鍵生成手段により生成される鍵に基づいて、受信手段により受信されたパケットに含まれる暗号文データを復号化する復号化手段とを有することを特徴とする構成を備えている。これにより、暗号化装置は、鍵の生成に用いられたパケット情報と、この鍵を用いて暗号化した平文データを、同一のパケットでまとめて復号化装置に送信するので、復号化装置は、暗号文データの復号化に際して、当該復号化処理の鍵生成に必要な情報を受信することが出来る。復号化装置は、鍵の生成に用いられたパケット情報と、この鍵を用いて暗号化した平文データを、同一のパケットでまとめて受信するので、鍵の生成に必要な情報を、暗号文データの復号化に使用することが出来る。これにより、パケットが送信される途中で、パケットがロストしたり、パケットの順番が入れ替わったりした場合であっても、暗号化装置と復号化装置の間の同期ずれを抑制することが出来る。

【実施例１】

【００３７】

通信システムは、図１に示すように、送信装置１０Ａおよび受信装置５０Ａを有している。送信装置１０Ａおよび受信装置５０Ａは、それぞれ暗号化装置および復号化装置の一例である。送信装置１０Ａおよび受信装置５０Ａは、インターネットなどの通信経路で接続自在である。なお、通信経路は、有線、無線いずれでも良い。また、本実施例では、通信プロトコルとして、ＵＤＰ（Ｕｓｅｒ Ｄａｔａｇｒａｍ Ｐｒｏｔｏｃｏｌ）などのコネクションレス型のプロトコルを使用している。なお、ＴＣＰ（Ｔｒａｎｓｍｉｓｓｉｏｎ Ｃｏｎｔｒｏｌ Ｔｒａｎｓｍｉｓｓｉｏｎ）などのコネクション型のプロトコルを使用することも可能である。

【００３８】

送信装置１０Ａは、通信データ生成部１１、暗号器１２、鍵交換部１３、ＣＴＲ記憶部１５、乱数生成器１６、ＸＯＲ処理部１７、データ合成部１９、カウンタアップ部２０、ＵＤＰデータ送受信部２１、およびネットワーク制御部２２を有している。送信装置１０Ａは、図示しないＣＰＵやＡＳＩＣなどの集積回路を有しており、図１に示す機能ブロックは、例えばＣＰＵによって実現される。

【００３９】

受信装置５０Ａは、データ分解部５１、暗号器５２、鍵交換部５３、乱数生成器５５、乱数生成器５５、ＸＯＲ処理部５６、通信データ解釈部５７、ＵＤＰデータ送受信部６０、およびネットワーク制御部５９を有している。受信装置５０Ａは、送信装置１０Ａと同様に、図示しないＣＰＵやＡＳＩＣなどの集積回路を有しており、図１に示す機能ブロックは、例えばＣＰＵによって実現される。

【００４０】

乱数生成器１６は、共通鍵を用いて乱数を生成し、ＸＯＲ処理部１７は、平文データをビット毎に暗号化する。すなわち、乱数生成器１６およびＸＯＲ処理部１７は、ストリーミング暗号を実行する機能ブロックを構成する。乱数生成器５５は、乱数生成器１６と同様に、共通鍵を用いて乱数を生成し、ＸＯＲ処理部５６は、暗号文データをビット毎に復号化する。すなわち、乱数生成器５５およびＸＯＲ処理部５６は、ストリーミング暗号を実行する機能ブロックを構成する。

【００４１】

ストリーミング暗号は、共通鍵暗号方式の一種で、平文をビット単位、若しくはバイト単位で逐次、暗号化する暗号方式である。ストリーミング暗号は、共通鍵暗号化方式の一種なので、暗号側（乱数生成器１６およびＸＯＲ処理部１７）と復号側（乱数生成器５５およびＸＯＲ処理部５６）は、等価な計算手段である。ここでは同一の計算手段を用いている。ストリーミング暗号としては、例えばＡｒｃｆｏｕｒがあるが、これに限られるも

のではない。例えば、ＣＢＣ（Ｃｉｐｈｅｒ　Ｂｌｏｃｋ　Ｃｈａｉｎｉｎｇ）モードなどのように、ブロック暗号をストリーミング暗号として利用する方法でも良い。なお、Ａｒｃｆｏｕｒのような専用のストリーミング暗号は、ブロック暗号の代表格であるＡＥＳ（Ａｄｖａｎｃｅｄ　Ｅｎｃｒｙｐｔｉｏｎ　Ｓｔａｎｄａｒｄ）暗号よりも、ソフト処理では処理速度が速いことから、高速なＵＤＰ通信に適している。先に説明したように、リアルタイム性を要求される音声・画像通信では、音飛びや映像飛びを防ぐために、Ａｒｃｆｏｕｒのような専用のストリーミング暗号が望ましい。本実施例では、ストリーミング暗号として、Ａｒｃｆｏｕｒを利用している。

【００４２】

暗号器１２は、カウンタモードが適用可能なブロック暗号器である。本実施例では、ＡＥＳ暗号を利用している。ブロック暗号は、共通鍵暗号方式の一種で、平文をブロック単位で処理する暗号方式である。ブロック単位は、固定長であっても、可変長であってもよい。ブロック暗号としては、例えば、ＡＥＳや３ＤＥＳ（Ｄａｔａ　Ｅｎｃｒｙｐｔｉｏｎ　Ｓｔａｎｄａｒｄ）がある。このように、ブロック暗号は共通鍵暗号化方式の一種なので、暗号器１２と暗号器５２と同様、等価な計算手段である。ここでは同一の計算手段を用いている。従って、暗号器１２、５２に入力される、共通鍵の鍵ＫＹ１は、同じ値である。また、ブロック暗号は、逆関数を有する暗号方式の一種であって、逆関数を有する暗号方式とは、異なる平文を同じ暗号文に変換しない方式をいう。

【００４３】

カウンタモードは、カウンタデータを暗号化し、その数値を乱数として使用する処理である。また、カウンタデータは、パケットを識別自在な番号である。ここでは、カウンタデータは、パケットの通し番号である。通し番号の場合、同じカウンタデータが出現しないので、セキュリティレベルを向上させることが出来る。

【００４４】

カウンタデータは、パケット情報の一例である。パケット情報とは、送信装置が、第１パケットＰＫ１、・・・、第Ｎ－１パケットＰＫｎ－１、第ＮパケットＰＫｎ（Ｎ、ｎは２以上）を、受信装置に送信する場合に、パケットＰＫ１、・・・、ＰＫｎ－１、ＰＫｎのそれぞれに対応する情報であって、パケットＰＫ１、・・・、ＰＫｎ－１、ＰＫｎのそれぞれを識別自在な情報である。従って、パケット情報は、カウンタデータに限る必要はない。例えば、ＵＤＰパケット内にカウンタデータとして利用可能なものがあれば、それをカウンタデータとして代用しても構わない。例えば、音声データの通信を行っている際に、音声データのデータフォーマットに通し番号があれば、これをカウンタデータとして扱っても良い。

【００４５】

以下、送信装置１０Ａの動作を説明する。はじめに、送信装置１０Ａが、鍵ＫＹ１を鍵交換部１３で生成し、これを暗号器１２にセットする。暗号器１２は、この鍵ＫＹ１を公開鍵暗号化方式の公開鍵として用いて暗号化を行い、この暗号化された鍵をＵＤＰデータ送受信部２１に送る。ＵＤＰデータ送受信部２１は、ネットワーク制御部２２を介して、この暗号化された鍵ＫＹ１を受信装置５０Ａに送信する。受信装置５０Ａでは、ＵＤＰデータ送受信部６０が、ネットワーク制御部５９を介して、暗号化された鍵ＫＹ１を受信する。

【００４６】

鍵交換部５３は、この暗号化された鍵を、ＵＤＰデータ送受信部６０から受け取る。鍵交換部５３は、公開鍵と対となる公開鍵暗号化方式の秘密鍵を用いて、この暗号化された鍵ＫＹ１を復号化する。鍵交換部５３は、送信装置１０Ａが生成した鍵ＫＹ１を取得し、この鍵ＫＹ１を暗号器５２にセットする。ここでは、単純な公開鍵暗号化方式を用いて鍵の受け渡しを行っているが、実際は、ＳＳＬ（Ｓｅｃｕｒｅ　Ｓｏｃｋｅｔｓ　Ｌａｙｅｒ）通信のように、暗号化においては、攻撃を受けないために、いくつもの注意事項がある。しかし、本発明の説明を簡単にするために、この鍵の交換は、このような簡単な説明に留める。また、ＵＤＰパケットで鍵の交換を行っているが、これはＳＳＬ通信のように

ＴＣＰ通信で行っても良い。あるいは、手作業で共通鍵をセットしても良い。共通鍵の交換方法は、この他どのような手段を用いても良い。

【００４７】

鍵ＫＹ１をセットすると、通信データ生成部１１は、平文データの一例である通信データＣＤを生成し、ＸＯＲ処理部１７に送る。これと同時に、通信データ生成部１１は、暗号器１２に、暗号処理開始を通知する。なお、この暗号処理開始の通知は、ＸＯＲ処理部１７が行っても良い。

【００４８】

暗号器１２は、カウンタデータＣＴＲを記憶しているＣＴＲ記憶部１５から、現在のカウンタデータＣＴＲを読み出す。暗号器１２は、このカウンタデータＣＴＲを暗号化して、暗号カウンタデータＥ（ＣＴＲ）を生成する。暗号器１２は、この暗号カウンタデータＥ（ＣＴＲ）を、乱数生成器１６のシードＳＤとして、つまりストリーミング暗号の共通鍵として設定する。これにより、乱数生成器１６は、暗号学的に安全な乱数（つまり予測不可能な乱数）を発生させることができる。なお、カウンタデータＣＴＲの初期値は、どんな値であっても良い。

【００４９】

次に、暗号器１２は、乱数生成器１６に乱数の生成を要求する。この要求は、通信データ生成部１１、またはＸＯＲ処理部１７が行ってもよい。乱数生成器１６は、通信データＣＤのデータ長と同じ、あるいは大きい乱数ＲＮを生成し、乱数ＲＮをＸＯＲ処理部１７に送る。

【００５０】

ＸＯＲ処理部１７は、乱数ＲＮ１を受けて、通信データＣＤと乱数ＲＮのＸＯＲ（排他的論理和）を計算（つまり、通信データＣＤを暗号化）し、暗号通信データＥＣＤを生成する。以下の説明では、排他的論理輪を単に「ＸＯＲ演算」と称す。ＸＯＲ処理部１７は、暗号通信データＥＣＤを、データ合成部１９に送る。ここでは、通信データＣＤと乱数ＲＮのＸＯＲ演算を一度に計算しているが、逐次行っても良い。

【００５１】

データ合成部１９は、ＣＴＲ記憶部１５から読み出したカウンタデータＣＴＲを暗号通信データＥＣＤに付加して、カウンタデータＣＴＲ付きの暗号通信データＥＣＤを生成する。そして、データ合成部１９は、これをＵＤＰデータ送受信部２１に送る。

【００５２】

また、データ合成部１９は、カウンタアップ部２０に、カウンタデータＣＴＲの更新を要求する。カウンタアップ部２０は、ＣＴＲ記憶部１５からカウンタデータＣＴＲを読み出し、この値を更新する。最も簡単な更新方法は、現在のカウンタデータＣＴＲに１を加えたデータを次のカウンタデータＣＴＲにする方法である。現在のカウンタデータのハッシュ値を次のカウンタデータＣＴＲにするなど、この更新はどのようなものであっても良い。

【００５３】

ＵＤＰデータ送受信部２１は、カウンタデータＣＴＲ付きの暗号通信データＥＣＤにＵＤＰヘッダを付加して、ネットワーク制御部２２を介して、カウンタデータＣＴＲ付きの暗号通信データＥＣＤを、つまりＵＤＰパケットを受信装置５０Ａに送信する。

【００５４】

ついで、受信装置５０Ａの動作を説明する。ＵＤＰデータ送受信部６０は、送信装置１０Ａが送信したＵＤＰパケットを、ネットワーク制御部５９を介して受信する。ＵＤＰデータ送受信部６０は、ＵＤＰパケットのＵＤＰヘッダを削除し、カウンタデータＣＴＲ付きの暗号通信データＥＣＤを、データ分解部５１に送信する。

【００５５】

データ分解部５１は、カウンタデータＣＴＲ付きの暗号通信データＥＣＤから、カウンタデータＣＴＲを読み取り、カウンタデータＣＴＲを暗号器１２に送る。またデータ分解部５１は、カウンタデータＣＴＲ付きの暗号通信データＥＣＤから暗号通信データＥＣＤ

を読み取り、ＸＯＲ処理部５６に送る。

【００５６】

一方、暗号器５２は、読み取ったカウンタデータＣＴＲを暗号化して、暗号カウンタデータＥ（ＣＴＲ）を生成する。そして暗号器５２は、この暗号カウンタデータＥ（ＣＴＲ）を、乱数生成器５５のシードＳＤとして、つまりストリーミング暗号の共通鍵として設定する。このように、乱数生成器１６、５５に入力するシードＳＤを、異なる乱数生成器で生成させるのではなく、カウンタデータを利用した同一の暗号方式で生成させている。

【００５７】

暗号器５２は、乱数生成器５５に乱数の生成を要求する。この要求は、データ分解部５１、またはＸＯＲ処理部５６が行ってもよい。乱数生成器５５は、暗号通信データＥＣＤのデータ長と同じ、あるいは大きい乱数ＲＮを発生する。そして乱数生成器５５は、乱数ＲＮ１をＸＯＲ処理部５６に送る。

【００５８】

ＸＯＲ処理部５６は、乱数ＲＮを受けて、暗号通信データＥＣＤと乱数ＲＮのＸＯＲ演算を行う。ある値Ｘに対して、同じ値ＹでＸＯＲ演算を２回行うと元の値Ｘに戻る。つまり、暗号・復号で、同じ乱数を２回用いて、ＸＯＲ演算を行えば、正しく復号できる。従って、ＸＯＲ処理部５６は、暗号通信データＥＣＤを通信データＣＤに復号化する。ＸＯＲ処理部５６は、通信データＣＤを、通信データ解釈部５７に送り、通信データ解釈部５７は、送信装置１０Ａが送信した通信データＣＤの内容の解釈を行う。

【００５９】

ＵＤＰパケットＰＫに付加されたカウンタデータＣＴＲは、暗号化されていないので、第三者はカウンタデータを知り得る。しかし、暗号カウンタデータＥ（ＣＴＲ）は、暗号化されたカウンタデータＣＴＲなので、第三者には復号化できない。このことは、この方法で発生させたシードＳＤもまた第三者には予想されないということと等価である。つまり、ブロック暗号の共通鍵さえ秘密にできれば、カウンタデータＣＴＲは公開されていても、暗号化カウンタデータＥ（ＣＴＲ）、つまりシードＳＤを第三者に知られることはない。

【００６０】

なお、カウンタデータＣＴＲは、ブロック暗号で暗号化するので、暗号化カウンタデータＥ（ＣＴＲ）のデータサイズは、ブロック暗号のブロック長になる。乱数生成器に必要なシードが、このブロック長サイズより小さければ、暗号化されたカウンタデータの一部分を利用するか、若しくはこの暗号化されたカウンタデータに何らかの計算処理を施し、シードのデータサイズに縮小すれば良い。もし逆に、乱数生成器に必要なシードが、このブロック長より大きければ、この暗号化されたカウンタデータに何らかの計算処理を施し、シードのデータサイズに拡張すれば良い。このような方法は種々あり、どのような方法を使っても良い。

【００６１】

図２に沿って、パケットロスした場合の同期ずれ抑制について説明する。図２では、第１パケットＰＫ１、・・・、第Ｎ－１パケットＰＫｎ－１、第ＮパケットＰＫｎの送受信を図示している（Ｎ、ｎは３以上）。図２では、カウンタデータＣＴＲ１、・・・、ＣＴＲｎ－１、ＣＴＲｎは１から始まるケースで説明しており、ＣＴＲ１、ＣＴＲ２、ＣＴＲ３・・・はそれぞれ、１、２、３、・・・である。暗号器１２、５２で暗号化したものは、それぞれＥ（ＣＴＲ１）、・・・、Ｅ（ＣＴＲｎ－１）、Ｅ（ＣＴＲｎ）である。暗号器１２、５２が、カウンタデータＣＴＲを暗号化することで、パケット毎に、シードＳＤを設定している。

【００６２】

ＵＤＰ通信は、ＴＣＰ通信とは異なり、コネクションレス通信なので、通信経路上でパケットロスしたり、パケットの到着順が逆になったり、することが起こり得る。例えば、通信経路上のルータの処理が混雑した場合などで、ＵＤＰパケットのロストが発生しやすい。

【００６３】
まず、送信装置１０Ａが、パケットＰＫ１を受信装置５０Ａに送信する。パケットＰＫ１には、カウンタデータＣＴＲ１が含まれている。受信装置５０Ａは、受信したパケットＰＫ１からカウンタデータＣＴＲ１を読み出し、暗号器５２がこれを暗号化して、暗号カウンタデータＥ（ＣＴＲ１）を取得する。乱数生成器５５は、暗号カウンタデータＥ（ＣＴＲ１）をシードＳＤ１として乱数ＲＮ１を生成し、ＸＯＲ処理部５６は、生成された乱数ＲＮ１を用いて、パケットＰＫ１に含まれる暗号通信データを復号化する。
【００６４】
この処理を繰り返し、送信装置１０Ａは、パケット毎に、シードＳＤを設定して、パケットＰＫ１、ＰＫ２、ＰＫ３、・・・を順次、受信装置５０Ａに送信していく。こうして、送信装置１０Ａが、乱数ＲＮｎ－１を発生させ、暗号カウンタデータＥ（ＣＴＲｎ－１）を付与したパケットＰＫｎ－１を、受信装置５０Ａに送信した場合に、パケットＰＫｎ－１が、通信経路上のどこかでロストしたとする。この場合、受信装置５０Ａは、ロストしたパケットＰＫｎ－１を受信しない。
【００６５】
この状態で、送信装置１０Ａが、乱数ＲＮｎを発生させ、暗号カウンタデータＥ（ＣＴＲｎ）を付与したパケットＰＫｎを、受信装置５０Ａに送信する。パケットＰＫｎはロストすることなく、受信装置５０Ａに受信されたとする。シードはパケット毎に設定されているので、パケットＰＫｎには、暗号通信データと、その暗号通信データを復号するための情報としてカウンタデーＣＴＲｎとが含まれている。
【００６６】
暗号器５２は、受信したパケットＰＫｎから、カウンタデータＣＴＲｎを暗号化して暗号カウンタデータＥ（ＣＴＲｎ）を生成し、乱数生成器５５のシードＳＤｎとして入力する。乱数生成器５５は、パケットＰＫｎのデータ部のデータサイズ分の乱数ＲＮｎを発生する。受信装置５０Ａは、パケットＰＫｎ－１を受信していないが、パケットＰＫｎに含まれる暗号通信データを、パケットＰＫｎからカウンタデータＣＴＲｎが読み出せるので、カウンタデータＣＴＲｎを暗号化することで、暗号文データを復号化することが出来る。これにより、パケットＰＫｎの暗号と復号では問題なく同期が取れる。これは、通常のストリーミング暗号とは異なり、パケット毎に、カウンタデータを元にシードを生成し、これを元に乱数を生成しているからである。
【００６７】
このように、送信装置は、鍵の生成に用いられたパケット情報と、この鍵を用いて暗号化した平文データを、同一のパケットでまとめて復号化装置に送信するので、受信装置は、暗号文データの復号化に際して、復号化処理の鍵生成に必要な情報を受信することが出来る。これにより、パケットが送信される途中で、パケットロスしたり、パケットの順番が入れ替わったりした場合であっても、送信側と受信側の間の同期ずれを抑制することが出来る。特に、パケットロスや到着順が入れ替わることが起き得る、ＵＤＰ通信において、良好に、暗号化、および復号化処理を行える。また、ストリーミング暗号による暗号・復号とは異なり、かなり先のカウンタデータを付与された不正なパケットを送信されても、サービス不能状態に陥ることを抑制することが出来る。
【００６８】
一般に、ブロック暗号は、ストリーミング暗号に比較して処理速度が遅いが、本実施例では、ブロック暗号をストリーミング暗号のシード生成に利用しているので、平文を直接ブロック暗号で暗号化する場合に比べて、パケットあたりのブロック暗号の処理回数を少なくすることが出来る。これにより、トータルの処理速度が速くすることが出来る。
【００６９】
ここで、本実施例（ＸＯＲ処理部と乱数生成器にストリーミング暗号を適用し、暗号器にブロック暗号を適用する場合）と、単なるブロック暗号の処理速度との対比について、具体的に説明する。リアルタイム性を要求される音声・画像通信のＵＤＰ通信では、１２８バイトや２５６バイト程度のＵＤＰ通信を行う。ＡＥＳ暗号のブロック暗号方式で暗号

・復号を行うとすると、１パケットが１２８バイトで、ブロック暗号の処理単位が１６バイトの場合、１パケット処理するのに、８回の暗号・復号処理が必要となる。しかしこの方法を用いれば、１パケット処理するのに、１回の１２８ビットのＡＥＳ暗号は必要となるが、その後は、１２８バイトのＡｒｃｆｏｕｒで暗号・復号処理を行える。もしも、Ａｒｃｆｏｕｒが、１２８ビットのＡＥＳ暗号より処理速度で倍のスピードがあれば、１パケットが１２８バイトの場合、約５回の１２８ビットのＡＥＳ暗号処理をしたのと同じ速度になる。つまり約１．６倍の速度で処理できることになる。実際は、Ａｒｃｆｏｕｒのシードのセット時間に余計な処理時間がかかるので、これよりも速度は落ちるが、１パケットのデータサイズが増加すれば、この処理速度はさらに向上する。

【００７０】

このように、リアルタイム性を要求される音声・画像通信のＵＤＰ通信において、高速な暗号・復号処理を行うことが可能となり、暗号通信と高速通信とを両立できるようになる。なお、暗号器１２、５２の暗号・復号処理には、ブロック暗号ではなく、ストリーミング暗号を利用することが可能である。ストリーミング暗号として、Ａｒｃｆｏｕｒのような、ソフト処理ではブロック暗号より、処理速度が速いストリーミング暗号を用いれば、高速な暗号・復号処理が可能となる。

【００７１】

図３では、送信装置１０Ａの主要要素の詳細を説明している。主要要素は、図３上方に示す破線枠内の、暗号器１２、乱数生成器１６、およびＸＯＲ処理部１７である。図３下方は、この主要要素の詳細を示している。ここでは、カウンタデータＣＴＲ１の処理のみを示している。

【００７２】

暗号器１２により暗号化されたカウンタデータＣＴＲ、つまり暗号カウンタデータＥ（ＣＴＲ）が、乱数生成器１６に入力されると、乱数ＲＮ１を生成する。乱数ＲＮ１は、乱数列ＲＮ１１、ＲＮ１２、ＲＮ１３、・・・から構成される。一方、平文データは、平文ＰＴ１、ＰＴ２、ＰＴ３、・・・から構成される。平文ＰＴ１、ＰＴ２、ＰＴ３のデータ長は、乱数列ＲＮ１１、ＲＮ１２、ＲＮ１３、・・・の乱数長とそれぞれ一致している。従って、ＸＯＲ処理部１７は、平文ＰＴ１と乱数列ＲＮ１１、平文ＰＴ２と乱数列ＲＮ１２、平文ＰＴ３と乱数列ＲＮ１３、・・を順次、ＸＯＲ演算する。これらの計算結果が暗号文データとなる。この時点で、「暗号化されたＵＤＰパケット」を生成したことになる。なお、ＵＤＰパケットを暗号化するというのは、ＵＤＰパケット全体を暗号化するという意味ではない。通常は、ＵＤＰパケットのＵＤＰデータ領域の一部、またはすべてを暗号化する。これ以降の説明も同じである。

【００７３】

送信装置１０Ａの動作を、図４のフローチャートに沿って再度説明する。Ｓ１０１では、カウンタデータＣＴＲの初期値を設定する。カウンタデータＣＴＲは第三者に知られても問題ないので、暗号化しない。カウンタデータＣＴＲの初期値は０でもその他の値でも良い。実際の実装では、送信装置１０Ａ、および受信装置５０Ａの間で、所定のＩＶ（イニシャルベクタ）値を交換しておき、これをカウンタデータと混合するのが、セキュリティレベルが向上し、望ましい。

【００７４】

Ｓ１０２では、カウンタデータＣＴＲを暗号器１２で暗号し、暗号カウンタデータＥ（ＣＴＲ）を求める。例えば、暗号器１２がＡＥＳ暗号を実行する場合は、ＡＥＳの暗号化関数であるＡＥＳ＿Ｅｎｃｒｙｐｔに、カウンタデータＣＴＲを代入する。

【００７５】

Ｓ１０３では、暗号カウンタデータＥ（ＣＴＲ）を乱数生成器１６のシードとして入力する。例えば、乱数生成器１６が、ストリーミング暗号としてＡｒｃｆｏｕｒを実行する場合は、Ａｒｃｆｏｕｒの初期化関数であるＡｒｃｆｏｕｒ＿Ｉｎｉｔに、この暗号カウンタデータＥ（ＣＴＲ）を代入する。

【００７６】

Ｓ１０４では、１パケット分の平文データと、乱数生成器１６が発生した１パケット分のデータサイズの乱数とをＸＯＲ演算して、１パケット分の暗号文データを生成する。例えば、乱数生成器、つまりストリーミング暗号がＡｒｃｆｏｕｒであれば、Ａｒｃｆｏｕｒの暗号化関数であるＡｒｃｆｏｕｒ＿Ｅｎｃｒｙｐｔに、平文データを代入する。

【００７７】

Ｓ１０５では、カウンタデータＣＴＲをＵＤＰパケットＰＫに付加し、暗号文データをＵＤＰパケットとして受信装置５０Ａに送信する。

【００７８】

Ｓ１０６では、送信データの準備ができていない場合は（Ｓ１０６のＮｏ）、送信データの準備ができるまで待つ。なお、これ以上の送信データがなければ、ここで終了しても良い。

【００７９】

送信データの準備ができた場合は（Ｓ１０６のＹｅｓ）、Ｓ１０７で、カウンタデータＣＴＲを更新する。以下、Ｓ１０２～Ｓ１０７を繰り返す。ここでは、カウンタデータＣＴＲを１ずつアップさせているが、次に利用するカウンタデータＣＴＲが、以前利用したカウンタデータＣＴＲと同じものにならないように工夫を行えば、その方法はどのようなものであっても良い。カウンタデータＣＴＲのハッシュ値を、新たなカウンタデータＣＴＲとすることも可能である。

【００８０】

ＵＤＰパケットＰＫは、図５に示すように、ヘッダ領域と、ＵＤＰデータ領域ＵＤＦを有している。ヘッダ領域は、ＭＡＣヘッダＨＤ１、ＩＰヘッダＨＤ２、およびＵＤＰヘッダＨＤ３を含む。ＵＤＰデータ領域ＵＤＦは、カウンタデータＣＴＲ、および暗号ＵＤＰデータＥＵＤを含む。ＵＤＰデータ領域ＵＤＦには、実データの他、改ざんチェックやデータ誤り検知のために、ＭＡＣやＣＲＣが付加されていても良い。

【００８１】

カウンタデータＣＴＲのデータ長を固定にしておけば、受信装置５０Ａは、暗号ＵＤＰデータＥＵＤがどこからスタートするかは簡単に分かる。またカウンタデータＣＴＲにデータ長を付与すれば、カウンタデータＣＴＲが可変長であっても、受信装置５０Ａはこのデータ長を読み取ることで、カウンタデータＣＴＲのサイズを求めることが可能なので、同じく、暗号ＵＤＰデータＥＵＤがどこからスタートするかが簡単に分かる。

【００８２】

カウンタデータＣＴＲは、図５に示すように、ＵＤＰデータ領域ＵＤＦの先頭にセットされている。カウンタデータＣＴＲは、ＵＤＰパケットＰＫに関連付けられていれば、カウンタデータＣＴＲを付加する箇所は、ＵＤＰデータ領域ＵＤＦである必要はなく、ＵＤＰパケットＰＫのどこにセットしても良い。

【００８３】

また、ＵＤＰパケットに付与されているいずれかのヘッダ（例えば音声データであれば、この音声パケットのヘッダなど）より、カウンタデータＣＴＲが導くことが出来れば、カウンタデータＣＴＲをＵＤＰパケットＰＫに付加する必要はない。例えば、カウンタデータＣＴＲを、既存のＥｔｈｅｒｎｅｔ（登録商標）ヘッダ、ＩＰヘッダ、ＵＤＰヘッダの中から導くことの出来る値にすることも可能である。

【００８４】

また、カウンタデータＣＴＲを導く元データをＵＤＰパケットＰＫに付加しても良い。例えば、図６に示すように、Ｓ３０１で、元データＯＤをハッシュ計算ルーチンで処理する、つまり元データＯＤのハッシュ値を計算することで、カウンタデータＣＴＲを生成する。

【００８５】

受信装置５０Ａの動作を、図７のフローチャートに沿って再度説明する。はじめに、Ｓ２０１で、図４の要領で生成されたＵＤＰパケットＰＫを受信する。

【００８６】

Ｓ２０２で、データ分解部５１が、ＵＤＰパケットＰＫに付加されたカウンタデータＣＴＲを読み取る。送信装置１０Ａおよび受信装置５０Ａ間で、所定のＩＶ（イニシャルベクタ）値を交換しておき、これをカウンタデータと混合することになっていれば、ここで、カウンタデータＣＴＲとＩＶの混合を行い、新たなカウンタデータＣＴＲを計算する

Ｓ２０３で、暗号器５２が、カウンタデータＣＴＲを暗号化し、暗号カウンタデータＥ（ＣＴＲ）を求める。例えば、送信側の暗号器１２がＡＥＳ暗号であれば、ＡＥＳの暗号化関数であるＡＥＳ＿Ｅｎｃｒｙｐｔに、カウンタデータＣＴＲを、暗号器５２に代入する。

【００８７】

Ｓ２０４で、暗号器５２は、暗号カウンタデータＥ（ＣＴＲ）をシードＳＤとして、乱数生成器５５に入力する。例えば、送信側の乱数生成器１６が実行するストリーミング暗号がＡｒｃｆｏｕｒであれば、Ａｒｃｆｏｕｒの初期化関数であるＡｒｃｆｏｕｒ＿ｉｎｉｔに、暗号カウンタデータＥ（ＣＴＲ）を、乱数生成器５５に代入する。

【００８８】

乱数発生器５５は、暗号カウンタデータＥ（ＣＴＲ）を用いて乱数を発生し、Ｓ２０５で、１パケット分の暗号文データと、発生した１パケット分のデータサイズの乱数をＸＯＲ演算して、１パケット分の平文データを生成する。例えば、乱数生成器５５が実行するストリーミング暗号がＡｒｃｆｏｕｒであれば、Ａｒｃｆｏｕｒの暗号化関数であるＡｒｃｆｏｕｒ＿Ｅｎｃｒｙｐｔに、暗号文データを代入する。送信装置１０ＡからまたＵＤＰパケットが送信されてきたら、Ｓ２０１～Ｓ２０６を繰り返す。

【００８９】

ついで、本実施例の効果を、図８に沿って説明する。図８は、ＵＤＰの１パケットのデータサイズが２５６バイトの場合における、ブロック暗号のＣＴＲモード、実施例１、ストリーミング暗号でのそれぞれの処理速度を示している。

【００９０】

図８左方は、ＣＴＲモードの処理速度：約１５Ｍｂｐｓを示している。ＣＴＲモードとは、２５６バイトすべてを１２８ビットのＡＥＳカウンタモードで暗号処理する場合である。１２８ビットのＡＥＳの１ブロックのデータサイズは１６バイトなので、２５６バイトのブロック数は、２５６バイト　／　１６バイト　＝　１６ブロックである。従って、１６回のＡＥＳ暗号（＋２５６バイト分の排他的論理和の計算）が必要となる。ＵＤＰの１パケットに対して１回しかカウンタデータをＡＥＳ暗号せず、この暗号化されたカウンタデータを２５６バイトすべての排他的論理和の計算に用いるとすることも可能であるが、これでは、同じ平文が同じ暗号文となるため、著しくセキュリティレベルを下げてしまう。

【００９１】

図８右方は、ストリーミング暗号の処理速度：３１Ｍｂｐｓである。ここでは、ストリーミング暗号として、Ａｒｃｆｏｕｒを使用している。

【００９２】

図８中央は、本実施例の処理速度：約２７Ｍｂｐｓを示している。ここでは、暗号器１２、５２にＡＥＳカウンタモードを適用し、乱数生成器１６、５５およびＸＯＲ処理部１７、５６にＡｒｃｆｏｕｒ暗号を適用した場合である。

【００９３】

本実施例では、ＵＤＰ１パケットに１回だけカウンタデータをＡＥＳ暗号化し、この暗号化されたカウンタデータをＡｒｃｆｏｕｒ暗号の共通鍵（シード）として、２５６バイトすべてをＡｒｃｆｏｕｒ暗号で暗号化する。通常のＡｒｃｆｏｕｒ暗号に比べると、１回のＡＥＳ暗号と、共通鍵のセット時間が余計な処理時間となる。通常、ソフト処理では、ＡｒｃｆｏｕｒはＡＥＳ暗号の半分以下の時間で暗号・復号処理を行えるので、本方式をＡＥＳ暗号処理で換算すると、１回のＡＥＳ暗号（カウンタデータの暗号化）＋１６回のＡＥＳ暗号／２（図８左方に示したＣＴＲモードの場合の半分）＋α（共通鍵のセット時間）により、おおよそ、９～１０回程度のＡＥＳ暗号・復号処理となる。測定環境は、

ＣＰＵ：ＭＩＰＳ　３２ビット、２００ＭＨｚである。測定方法としては、ＵＤＰヘッダ作成時間＋暗号処理時間を測定した。また、１パケットあたりの平分データサイズは１０２４バイト、データ中身は“０×００～０×ＦＦ”を４回セット、１０Ｍバイト分繰り返した時間を計測し、ｂｐｓを算出、ＣＴＲモードには高速なＡＥＳを利用、ストリーミング暗号にはＡｒｃｆｏｕｒを利用した。以上の測定結果から、ブロック暗号を基準にすれば、本実施例の処理速度は、ストリーミング暗号の処理速度：３１Ｍｂｐｓに接近している。

【００９４】

このように、本実施例では、ブロック暗号で暗号化したカウンタデータを、通信データの暗号・復号処理に直接用いるのではなく、暗号化したカウンタデータを、通信データを暗号化する共通鍵暗号の共通鍵として用いている。なお、図１のストリーミング暗号を実行する要素（乱数生成器１６、５５およびＸＯＲ処理部１７、５６）は、Ａｒｃｆｏｕｒのようなストリーミング暗号に限られるものではなく、ＡＥＳや３ＤＥＳのようなブロック暗号であっても良い。なお、全体の処理速度を上げる点で、暗号・復号を実行する要素（乱数生成器１６、５５およびＸＯＲ処理部１７、５６）の暗号方式は、鍵を生成する要素（暗号器１２、５２）の暗号方式より処理速度が速いことが望ましい。

【００９５】

なお、悪意のある攻撃者が送信装置１０Ａのふりをして、意図的に、かなり先のカウンタデータが付与された不正なＵＤＰパケットを送信するかもしれないので、受信装置５０Ａは、ＭＡＣチェックを行うようにして、これが正しい場合のみ、最大のカウンタデータを更新するようにしてもよい。一般に、正規のＵＤＰ通信であれば、極端なパケットロスがない限り、カウンタデータが極端に飛ぶことはない。従って、極端なカウンタデータの飛びがあれば、それはかなりパケットロスが発生していることになるので、受信装置５０Ａは飛んだカウンタデータをいつまでも受信できるものとして、受信装置５０Ａに記憶しておく必要はない。

【００９６】

この仕組みを図９、および図１０を用いて、詳細を説明する。図９は、ＭＡＣチェックを行う際のＵＤＰパケットＰＫのフォーマットの一例を示している。送信装置１０Ａは、ＵＤＰデータ領域ＵＤＦに、カウンタデータＣＴＲ、実際のＵＤＰデータ（平文ＵＤＰデータＰＵＤ）、ＭＡＣデータＭＤをセットしておく。ＸＯＲ処理部１７が、乱数ＲＮ１と平文ＵＤＰデータＰＵＤのＸＯＲ演算を行い、暗号ＵＤＰデータＥＵＤを生成する。送信装置１０Ａは、ＵＤＰデータ領域ＵＤＦに、カウンタデータＣＴＲ、暗号ＵＤＰデータＥＵＤ、ＭＡＣデータＭＤをセットして、ＵＤＰパケットＰＫを受信装置５０Ａに送る。受信装置５０Ａでは、ＵＤＰパケットＰＫを受信し、ＸＯＲ処理部５６で、この乱数ＲＮ１と暗号ＵＤＰデータＥＵＤとのＸＯＲ演算を行い、平文ＵＤＰデータＰＵＤを取得する。

【００９７】

図１０は、図９のＵＤＰパケットＰＫのフォーマットで通信している際の、ＭＡＣチェック方法の一例を示すものである。送信装置１０Ａは、平文ＵＤＰデータＰＵＤ、つまり通信データを用意する。ＭＡＣ計算器２３は、平文ＵＤＰデータＰＵＤからＭＡＣデータＭＤ１を計算する。ＭＡＣ計算手法には、ＭＤ５、ＳＨＡ１、ＳＨＡ２など、様々な手法があるが、送信装置１０Ａ・受信装置５０Ａで同じ計算手法を用いるものとする。

【００９８】

図１０では、カウンタデータＣＴＲは通し番号で、カウンタデータＣＴＲは１ずつカウントアップするものとする。送信装置１０Ａは、送信するＵＤＰパケットＰＫに付与するカウンタデータＣＴＲ１を用意する。実際は、暗号器１２が、現在のカウンタデータＣＴＲ１を、ＣＴＲ記憶部１５より読み出す。暗号器１２、および乱数生成器１６は、図２と同様の暗号を行い、平文ＵＤＰデータＰＵＤから暗号ＵＤＰデータＥＵＤを生成する。暗号化が終了したら、次回のパケット送信に備えて、カウンタデータを１アップする。実際は、暗号器１２がカウンタアップ部２０に指示して、カウンタデータＣＴＲのアップを行わせ、このアップしたカウンタデータをＣＴＲ記憶部１５に保存させる。

【００９９】
送信装置１０Ａは、カウンタデータＣＴＲと暗号ＵＤＰデータＥＵＤとＭＡＣデータＭＤ１を受信装置５０Ａに送信する。受信装置５０Ａは、これらのデータを受信する。受信装置５０Ａでは、現在までに取得したカウンタデータＣＴＲの最大値ＣＴＲｍａｘをＣＴＲ記憶部６１に記憶している。また、ＣＴＲ記憶部６１は、最大値未満でまだ受信していないカウンタデータＣＴＲｎｙも記憶している。
【０１００】
受信装置５０Ａは、受信したカウンタデータＣＴＲ１をカウンタチェック部６２でチェックする。具体的には、次のような方法でチェックする。まず、ＣＴＲ記憶部６１から、２つのカウンタデータＣＴＲｍａｘ、ＣＴＲｎｙを読み出す。受信したカウンタデータＣＴＲ１が、カウンタデータＣＴＲｍａｘより大きいか、またはカウンタデータＣＴＲｎｙに該当するか否かをチェックする。受信したカウンタデータＣＴＲ１が、カウンタデータＣＴＲｍａｘより小さく、かつ、カウンタデータＣＴＲｎｙに該当しない場合は、このＵＤＰパケットＰＫを、カウンタチェック部６２が破棄する。これによりリトライ攻撃が防御できるようになる。
【０１０１】
一方、受信したカウンタデータＣＴＲ１が、カウンタデータＣＴＲｍａｘより大きい、または、カウンタデータＣＴＲｎｙに該当する場合は、ＵＤＰパケットＰＫは破棄されない。暗号器５２および乱数生成器５５は、図２と同様の復号を行い、暗号ＵＤＰデータＥＵＤから平文ＵＤＰデータＰＵＤを取得する。
【０１０２】
ＭＡＣ計算器６３は、平文ＵＤＰデータＰＵＤからＭＡＣデータＭＤ２を計算する。ＭＡＣデータ比較器６５は、この計算で求めたＭＡＣデータＭＤ２と、受信したＵＤＰパケットに付与されていたＭＡＣデータＭＤ１とを比較する。一致しない場合は、何もしないか、またはＭＡＣデータ比較器６５が、このＵＤＰパケットを破棄する。一致した場合は、ＭＡＣデータ比較器６５は、カウンタアップ部６６に指示して、カウンタデータのアップを行わせ、このアップしたカウンタデータＣＴＲをＣＴＲ記憶部６１に保存させる。なお、受信したカウンタデータＣＴＲが現在の最大値より大きい場合は、このカウンタデータＣＴＲを新たな最大値として記憶する。またその際、受信したカウンタデータＣＴＲが現在の最大値より２以上飛んでいた場合には、この飛んだカウンタデータＣＴＲはまだ未受信のカウンタデータＣＴＲとして記憶する。受信したカウンタデータＣＴＲが未受信のカウンタデータＣＴＲであった場合には、その未受信で記憶しているカウンタデータＣＴＲを消去する。
【実施例２】
【０１０３】
図１１に沿って実施例２における、通信システムの動作を説明する。実施例２の通信システムは、図１に示す実施例１の通信システムと同一である。後述する動作方法が異なる。
【０１０４】
図１１の動作方法は、図２の動作方法と比較して、２つの相違点がある。１つ目の相違点は、乱数生成器が、一定のパケット毎にシードを変更する点である。例えば、カウンタデータが所定値を超えた場合に、若しくは受信側へ送信したパケットの全ての通信データ数が所定値を超えた場合に、シードを変更する。
【０１０５】
２つ目の相違点は、送信装置が、所定パケットから、いずれかのパケットまでの送信した、全ての通信データ数を、受信装置が把握できるようにする。例えば、パケットＰＫｎ＋ｘやパケットＰＫｎ＋ｘ＋１などに、パケットＰＫｎからの通信データ数の総和（つまり、パケットＰＫｎを起点として、パケットＰＫｎ＋ｘやパケットＰＫｎ＋ｘ＋１までの全パケットのデータサイズの総和）をＵＤＰパケットのＵＤＰデータ領域に記載する。なお、ｘは、ｘ＝１、２、・・・である。

【０１０６】
また、実施例２におけるストリーミング暗号は、通信データに依存しない乱数を用いたものである。乱数生成器が、通信データとは独立に存在するストリーミング暗号で、例えば、Ａｒｃｆｏｕｒなどである。ＣＢＣモードなど、通信データに依存するストリーミング暗号（ブロック暗号の操作モードを利用したストリーミング暗号）は該当しない。
【０１０７】
受信装置５０Ｂは、この通信データ数を取得することで、各ＵＤＰパケット（ＵＤＰデータ部）が送信される前に、送信装置１０Ｂがどれくらいのデータサイズの通信データを送信したかを知ることができる。これにより、各ＵＤＰパケットの復号処理を行う前に、乱数生成器に、この通信データ数分の乱数を発生させれば、たとえ、いくつかのＵＤＰパケットがロストしても、同期をとることができる。
【０１０８】
図１２に沿って、実施例２における送信装置の動作について説明する。はじめに、Ｓ１１１で、カウンタデータＣＴＲの初期値を設定する。カウンタデータＣＴＲは第三者に知られても問題ないので、初期値は０でもその他の値でも良い。実際の実装では、送信装置１０Ｂと受信装置５０Ｂの間で所定のＩＶ（イニシャルベクタ）値を交換しておき、これをカウンタデータＣＴＲと混合するのが、セキュリティレベルが向上し、望ましい。
【０１０９】
Ｓ１１２で、カウンタデータＣＴＲが一定の値Ｎを超えたか否かを調べる。具体的には、ＣＴＲ　Ｍｏｄ　Ｎ＝０の方程式が成り立つか否かを判断する。ＣＴＲ　Ｍｏｄ　Ｎ≠０であれば（Ｓ１１２のＮｏ）、すなわち、ＮをカウンタデータＣＴＲで割った剰余が０でない場合、Ｓ１１６に進む。例えば、Ｎが５の場合、カウンタデータＣＴＲが、１、２、３、４、６、７、８・・・の場合に、Ｓ１１６に進む。一方、ＣＴＲ　Ｍｏｄ　Ｎ＝０であれば（Ｓ１１２のＹｅｓ）、すなわち、ＮをカウンタデータＣＴＲで割った剰余が０である場合、Ｓ１１３で、通信データ数ＴＮを０に設定する。
【０１１０】
Ｓ１１４で、カウンタデータＣＴＲを暗号器１２で暗号し、暗号カウンタデータＥ（ＣＴＲ）を求める。例えば、ＡＥＳの暗号化関数であるＡＥＳ＿Ｅｎｃｒｙｐｔに、カウンタデータＣＴＲを代入する。
【０１１１】
なお、カウンタデータＣＴＲが一定の値Ｎを超える場合に限らず（つまり、複数のカウンタデータを超えた場合に限らず）、一定のカウンタデータのみを超えた場合に、本発明を適用することも可能です。または、ある決められたステップ数を越えるたびに実行してもよい。例えば、ステップ数を１０とすれば、カウンタデータが１０、２０、３０、・・・というように、１０刻みのタイミングとすることができる。
【０１１２】
Ｓ１１５で、暗号カウンタデータＥ（ＣＴＲ）を乱数生成器１６のシードＳＤとして入力する。例えば、Ａｒｃｆｏｕｒの初期化関数であるＡｒｃｆｏｕｒ＿Ｉｎｉｔに、この暗号カウンタデータＥ（ＣＴＲ）を代入する。
【０１１３】
Ｓ１１６で、パケット分の平文データと、乱数生成器１６が発生した１パケット分のデータサイズの乱数とをＸＯＲ演算して、１パケット分の暗号文データを生成する。例えば、Ａｒｃｆｏｕｒの暗号化関数であるＡｒｃｆｏｕｒ＿Ｅｎｃｒｙｐｔに、平文データを代入する。
【０１１４】
Ｓ１１７で、カウンタデータＣＴＲと通信データ数ＴＮ（ｎ）をＵＤＰパケットＰＫに付加し、このＵＤＰパケットを受信装置５０Ｂに送信する。
【０１１５】
Ｓ１１８で、通信データ数ＴＮ（ｎ）を更新する。前の通信データ数ＴＮ（ｎ）に、このＵＤＰパケットＰＫのデータサイズを加算し、通信データ数ＴＮ（ｎ＋１）を求め、通

信データ数ＴＮ（ｎ＋１）を所定のメモリ（図示せず）に記憶する。

【０１１６】

Ｓ１１９で、送信データの準備ができていない場合は（Ｓ１１９のＮｏ）、送信データの準備ができるまで待つ。なお、これ以上の送信データがなければ、ここで終了しても良い。一方、送信データの準備ができた場合は（Ｓ１１９のＹｅｓ）、Ｓ１２０で、カウンタデータＣＴＲを更新する。これ以降、Ｓ１１２からＳ１２０を繰り返す。

【０１１７】

ついで、図１３に沿って、実施例２における受信装置の動作を説明する。受信装置５０Ｂでは、図示しないメモリが、次の通信データ数ＲＤを記憶している。次の通信データ数とは、送信装置が所定のパケットから任意のパケットまでを順次送信し、かつパケットロスやパケットの到着順の入れ違いがない場合に、受信装置が受信すべく、所定のパケットから任意のパケットまでの通信データ数の総和をいう。次の通信データ数ＲＤは、初期値は０である。はじめに、Ｓ２１１では、図１２の要領で生成されたＵＤＰパケットＰＫを受信する。

【０１１８】

Ｓ２１２で、ＵＤＰパケットＰＫに付加されたカウンタデータＣＴＲと通信データ数ＴＤ（ｎ）を読み取る。

【０１１９】

Ｓ２１３で、カウンタデータＣＴＲが一定の値Ｎを超えたか否かを調べる。具体的には、ＣＴＲ　Ｍｏｄ　Ｎ＝０の方程式が成り立つか否かを判断する。この処理は、図１２のＳ１１２と同様である。

【０１２０】

ＣＴＲ　Ｍｏｄ　Ｎ≠０であれば（Ｓ２１３のＮｏ）、Ｓ２１７に進む。ＣＴＲ　Ｍｏｄ　Ｎ＝０であれば（Ｓ２１３のＹｅｓ）、Ｓ２１４で、次の通信データ数ＲＤを０として所定のメモリ（図示せず）に記憶する。なお、図示していないが、ここで、読み取った通信データ数ＴＤ（ｎ）が０以外の値であれば、エラーとしてこのＵＤＰパケットＰＫを破棄する。

【０１２１】

Ｓ２１５で、カウンタデータＣＴＲを暗号器５２で暗号し、暗号カウンタデータＥ（ＣＴＲ）を求める。例えば、ＡＥＳの暗号化関数であるＡＥＳ＿Ｅｎｃｒｙｐｔに、カウンタデータＣＴＲを代入する。

【０１２２】

Ｓ２１６で、暗号カウンタデータＥ（ＣＴＲ）を乱数生成器５５のシードＳＤとして入力する。例えば、Ａｒｃｆｏｕｒの初期化関数であるＡｒｃｆｏｕｒ＿Ｉｎｉｔに、この暗号カウンタデータＥ（ＣＴＲ）を代入する。

【０１２３】

次にＳ２１７で、読み取った通信データ数ＴＤ（ｎ）と次の通信データ数ＲＤ（ｎ）を比較する。読み取った通信データ数ＴＤ（ｎ）が、次の通信データ数ＲＤ（ｎ）より小さければ（Ｓ２１７のＹｅｓ）、パケットの到着順が逆になったため、受信したＵＤＰパケットＰＫが、処理済みのパケットよりカウンタデータの値が若い未処理のパケットがあるということなので、Ｓ２１４に進む。なお、処理済みのパケットよりカウンタデータの値が若い未処理のパケットがなければ（Ｓ２１７のＮｏ）、読み取った通信データ数ＴＤ（ｎ）＜次の通信データ数ＲＤ（ｎ）となるのは不正なので、このＵＤＰパケットＰＫを破棄する。

【０１２４】

Ｓ２１８で、通信データ数ＴＤ（ｎ）と次の通信データ数ＲＤ（ｎ）が等しければ（Ｓ２１８のＹｅｓ）、Ｓ２２０に進む。通信データ数ＴＤ（ｎ）と次の通信データ数ＲＤ（ｎ）が等しくない場合（Ｓ２１８のＮｏ）、パケットロス、若しくはパケット到着順が入れ替わっているので、Ｓ２１９で、乱数列を飛ばす処理を行う。具体的には、通信データ数ＴＤ（ｎ）から次の通信データ数ＲＤ（ｎ）を差し引いた通信データ数と同一の乱数列

分、乱数を発生させる。この乱数列は直ちに利用せず、後で、これに対応したＵＤＰパケットＰＫが受信した場合に備えて、この乱数列を所定のメモリ（図示せず）に記憶しておく。これにより、次に再び乱数を発生させなくて済み、Ｓ２１７でＳ２１４に戻る処理を行わなくて済むようになる。このように、通信データ数ＴＤ（ｎ）に応じて、パケットに対応した乱数列を設定する。この設定処理は、図示しないＣＰＵが実行する。

【０１２５】

Ｓ２２０で、１パケット分の暗号文データと、乱数生成器５２が発生した１パケット分のデータサイズの乱数とをＸＯＲ演算して、１パケット分の平文データを生成する。例えば、Ａｒｃｆｏｕｒの暗号化関数であるＡｒｃｆｏｕｒ＿Ｅｎｃｒｙｐｔに、暗号文データを代入する。

【０１２６】

Ｓ２２１で、次の通信データ数ＲＤ（ｎ）を更新し、所定のメモリ（図示せず）に記憶する。具体的には、通信データ数ＲＤ（ｎ）に１パケット分のデータサイズを加算した値を、次の通信データ数ＲＤ（ｎ＋１）として、その値を所定メモリに記憶する。送信装置１０ＢからまたＵＤＰパケットが送信されてきたら、Ｓ２１１～Ｓ２２１を繰り返す。

【０１２７】

このように、受信装置５０Ｂは、パケットＰＫｎ＋ｘ＋１の前に送信された通信データ数を知ることで、パケットＰＫｎ＋ｘがロストしたか、または順番が前後しているかを知ることができる。そして、パケットＰＫｎ＋ｘ＋１を復号化するために、パケットＰＫｎ＋ｘ＋１の前に送信された通信データ数から、パケットＰＫｎ＋ｘ－１の前に送信された通信データ数を差し引いた通信データ数を計算して、この差の分、乱数生成器５５に乱数を生成させる。なお、１パケットのデータ部のデータサイズが固定されている場合には、送信装置１０Ｂは送信した通信データ数をＵＤＰのデータ部に記載する必要はない。受信装置５０ＢはＵＤＰパケットからカウンタデータを読み取ることで、送信装置１０Ｂが送信した通信データ数を知ることができる。

【０１２８】

こうすれば、パケットＰＫｎ＋ｘ＋１に用いられる乱数は、送信装置１０Ｂが暗号に利用した乱数ＲＮｎ＋ｘ＋１と同じものとなり、同期が取れるので、正しく復号化できる。

【０１２９】

しかも、ストリーミング暗号による暗号・復号処理とは異なり、悪意のある攻撃者が、送信装置１０Ｂのふりをして、かなり先のカウンタデータが付与されたＵＤＰパケットを受信装置５０Ｂに送ったとしても、図２に比べて、対応に多少時間を要するだけでサービス不能状態に陥ることがない。かなり先のカウンタデータが付与されたＵＤＰパケットが送られたとしても、受信装置５０ＢはＮ、Ｎ＋１、Ｎ＋２、・・・を基準に乱数列を生成するだけなので、短い時間で対応できる。なお、パケットの暗号・復号処理のほとんどを、Ａｒｃｆｏｕｒなど処理の速いストリーミング暗号で行えるので、通信データ数を付加することによる処理時間のロスはあるものの、図２よりさらに高速に暗号・復号処理を行うことが可能となる。

【０１３０】

なお、ここでは、送信装置１０Ｂがシードを設定後に送信した通信データ数を、受信装置５０Ｂが把握できるようにするために、送信装置１０ＢがＵＤＰパケットに通信データ数の情報を付加する、つまり通信データ数をＵＤＰパケットに記載するとして説明したが、送信装置１０Ｂと受信装置５０Ｂで送受信するＵＤＰパケットのデータ部のデータサイズを固定にするなど、１回のＵＤＰパケットで送信する前記通信データ数を事前に取り決めておけば、通信データ数をＵＤＰパケットに記載する必要はなく、さらに高速なＵＤＰ通信が実現できる。

【０１３１】

このように、実施例２を用いれば、リアルタイム性を要求される音声・画像通信のＵＤＰ通信で、さらに高速な暗号・復号処理を行うことが可能となり、暗号通信と高速通信とを両立できるようになる。

【０１３２】
このように、パケットロスやパケットの到着順に入れ替わりがなければ、毎回、シードＳＤを入力することも、乱数を余計に発生させることもなくなるので、高速な復号処理が行えるようになる。
【０１３３】
また、ＵＤＰ通信で、高速な暗号通信を行うことが可能となるだけでなく、ＡＥＳカウンタモードなど、処理速度の遅い暗号処理を減らせるので、より高速な暗号・復号処理が可能となる。
【実施例３】
【０１３４】
図１４は、実施例３における監視システムを示す構成図である。図１４において、通信システムの一例である、監視システム１を示す。監視システム１は、ＰＣ（パーソナルコンピュータ）５０Ｃと、ネットワークカメラ１０Ｃを有している。ＰＣ５０Ｃとネットワークカメラ１０Ｃは、ネットワークケーブル２０５で接続されている。ＰＣ５０Ｃは、受信装置の一例であり、映像を受信する。モニタ２０３は、受信した映像を表示する。ＰＣ５０Ｃは、ネットワーク制御部５９と鍵設定部５３を、外部から接続自在に構成されている。ネットワーク制御部５９は、ネットワークケーブルが接続自在であり、ネットワーク通信を制御する。鍵設定部５３は、ＵＳＢ（Ｕｎｉｖｅｒｓａｌ　Ｓｅｒｉａｌ　Ｂｕｓ）インタフェースなど、外部メモリと鍵データの受け渡しを行い、鍵の設定を行う。
【０１３５】
ネットワークカメラ１０Ｃは、送信装置の一例であり、監視を行う。ネットワークカメラ１０Ｃは、カメラ部１０１を有しており、ネットワーク制御部２２と鍵設定部１３を、外部から接続自在に構成されている。カメラ部１０１は、映像データを撮影し、映像データの生成を行う。ネットワーク制御部２２は、ネットワークケーブルが接続自在であり、ネットワーク通信を制御する。鍵設定部１３は、鍵設定部５３と同様、ＵＳＢインタフェースなど、外部メモリと鍵データの受け渡しを行い、鍵の設定を行う。ネットワークケーブル２０５は、例えばイーサネット（登録商標）ケーブルやシリアルケーブルなど、各種のケーブルが適用可能である。なお、無線通信の場合は、ケーブルは不要である。
【０１３６】
図１５は、図１４の監視システムの機能ブロック図である。図１と共通する要素は、同一の符号を付している。映像通信を開始する前に、事前設定を行う。暗号器１２と暗号器５２は、カウンタモードを実行するブロック暗号器であり、両者で同じ鍵ＫＹ１を設定する。
【０１３７】
はじめに、設定者が、ＵＳＢメモリ内に鍵ＫＹ１を用意する。設定者は、このＵＳＢメモリを、ＵＳＢインタフェース機能を持つ鍵設定部７０に挿入する。鍵設定部７０は、ＵＳＢメモリから鍵ＫＹ１を読み込み、これを暗号器１２にセットする。
【０１３８】
また設定者は、このＵＳＢメモリを、ＵＳＢインタフェース機能を持つ鍵設定部７１にも挿入する。鍵設定部７１は、ＵＳＢメモリから鍵ＫＹ１を読み込み、これを暗号器５２にセットする。ここでは、設定者はＵＳＢメモリを鍵設定部７０に先に挿入したが、鍵設定部７１に先に挿入しても構わない。また、ＵＳＢメモリを利用して鍵の設定を行っているが、図１と同様に、公開鍵暗号方式を用いて、設定者を介さずに自動に設定させることも可能である。
【０１３９】
次に、カメラ部１０１が、映像撮影を行い、映像データ生成部７２が、映像データＩＤを生成し、この映像データＩＤをＸＯＲ処理部１７に送る。これと同時にカメラ部１０１は暗号器１２に、暗号処理開始を通知する。なお、この暗号処理開始の通知は、ＸＯＲ処理部１７が行っても良い。
【０１４０】

暗号器１２は、ＣＴＲ記憶部１５から現在のカウンタデータＣＴＲを読み出し、このカウンタデータＣＴＲを暗号化して、暗号カウンタデータＥ（ＣＴＲ）を生成する。そして、この暗号カウンタデータＥ（ＣＴＲ）を、乱数生成器１６のシードＳＤとして設定する。なお、カウンタデータの初期値はどんな値であっても良い。

【０１４１】

次に暗号器１２は、乱数生成器１６に乱数の生成を要求する。この要求は、カメラ部１０１、またはＸＯＲ処理部１７が行ってよい。乱数生成器１６は、映像データＩＤのデータ長と同じ、あるいは大きい乱数を発生する。そして、乱数ＲＮをＸＯＲ処理部１７に送る。

【０１４２】

ＸＯＲ処理部１７は、乱数ＲＮを受けて、映像データＩＤと乱数のＸＯＲ演算を行い、暗号映像データＥＩＤを生成し、この暗号通信データＥＩＤを、データ合成部１９に送る。ここでは、一度に映像データＩＤと乱数のＸＯＲ演算を行っているが、逐次行っても良い。

【０１４３】

データ合成部１９は、ＣＴＲ記憶部１５から、現在のカウンタデータＣＴＲを読み出し、このカウンタデータＣＴＲを暗号映像データＥＩＤに付加する。カウンタデータＣＴＲ付きの暗号映像データＥＩＤを生成し、これをＵＤＰデータ送受信部２１に送る。

【０１４４】

また、データ合成部１９は、カウンタアップ部２０に、カウンタデータＣＴＲの更新を要求する。カウンタアップ部２０は、カウンタデータＣＴＲをＣＴＲ記憶部１５から読み出し、この値を更新する。最も簡単な更新方法は、現在のカウンタデータに１を加えたデータを次のカウンタデータにする方法であるが、現在のカウンタデータのハッシュ値を次のカウンタデータにするなど、この更新はどのようなものであっても良い。但し、同じカウンタデータが出現しない方法を採用するのが望ましい。

【０１４５】

ＵＤＰデータ送受信部２１は、カウンタデータＣＴＲ付きの暗号映像データＥＩＤにＵＤＰヘッダを付加して、ネットワーク制御部２２を介して、カウンタデータＣＴＲ付きの暗号映像データＥＩＤを、ＰＣ５０Ｃに送信する。

【０１４６】

ＰＣ５０Ｃでは、ＵＤＰデータ送受信部６０が、ネットワーク制御部５９を介して、カウンタデータＣＴＲ付きの暗号映像データＥＩＤを含むＵＤＰパケットを受信する。

【０１４７】

データ分解部５１は、カウンタデータＣＴＲ付きの暗号映像データＥＩＤを、ＵＤＰデータ送受信部６０から受け取る。このカウンタデータＣＴＲ付きの暗号映像データＥＩＤから、カウンタデータＣＴＲを抜き出し、このカウンタデータＣＴＲを暗号器５２に送る。また、カウンタデータＣＴＲ付きの暗号映像データＥＩＤから、暗号映像データＥＩＤを抜き出し、この暗号映像データＥＩＤをＸＯＲ処理部５６に送る。

【０１４８】

暗号器５２は、カウンタデータＣＴＲを暗号化して暗号カウンタデータＥ（ＣＴＲ）を生成する。そして、この暗号カウンタデータＥ（ＣＴＲ）を、乱数生成器５５のシードＳＤとして設定する。

【０１４９】

また、暗号器５２は、乱数生成器５５に乱数ＲＮの生成を要求する。この要求は、データ分解部５１、またはＸＯＲ処理部５６が行うものとすることもできる。乱数生成器５５は、暗号映像データＥＩＤのデータ長と同じ、あるいは大きい乱数ＲＮを発生する。そして、乱数ＲＮをＸＯＲ処理部５６に送る。

【０１５０】

ＸＯＲ処理部５６は、乱数ＲＮを受けて、暗号映像データＥＩＤと乱数ＲＮのＸＯＲ演算を行い、映像データＩＤを取得する。そして、この映像データＩＤを、映像データ生成

部７３に送る。映像データ生成部７３は、ネットワークカメラ１０Ｃが送信した映像データＩＤをモニタ２０３に表示させる。

【０１５１】

このように、本実施例の監視システムは、実施例１と同様に、同期ずれを抑制することが出来るので、リアルタイム通信が要求される映像通信で、ＵＤＰ通信が適用された場合であっても、セキュリティを確保しながら、映像通信を円滑に行うことが出来る。

【実施例４】

【０１５２】

図１６は、実施例４におけるＵＤＰ通信に利用できるように、ストリーミング暗号の利用方法を改良した手段を説明するブロック図である。本実施例では、通信データ数を用いて同期ずれを抑制する点で、実施例２と同様であるが、シード生成を行う暗号器１２、５２を備えていない。

【０１５３】

図１６の暗号化・復号化手段の特徴は、パケットＰＫｎ－１やパケットＰＫｎなどに、送信装置１０Ｄがそれ以前に送信した通信データ数の情報を記載するところである。この通信データ数の情報は、送信装置１０Ｄが、ＵＤＰパケットのデータ部に記載し、これを受信装置５０Ｄが読み取るようになっている。

【０１５４】

受信装置５０Ｄは、この通信データ数を取得することで、各ＵＤＰパケット（ＵＤＰデータ部）が送信される前に、送信装置１０Ｄがどれくらいのデータサイズの通信データを送信したかを知ることができる。これにより、各ＵＤＰパケットの復号化の前に、この通信データ数分の乱数を発生させれば、たとえ、いくつかのＵＤＰパケットがロストしても、同期をとることができるようになる。

【０１５５】

図１６では、受信装置５０Ｄは、パケットＰＫｎの前に送信された通信データ数を知ることで、パケットＰＫｎ－１がロストしたか、または順番が前後しているかを知ることができる。そして、パケットＰＫｎを復号化するために、パケットＰＫｎの前に送信された通信データ数から、パケットＰＫｎ－２の前に送信された通信データ数を差し引いた、通信データ数を計算して、この差分、乱数生成器５５に乱数を生成させる。なお、パケットＰＫｎ－１は後で到着するかもしれないので、この際に発生させた乱数は、受信装置５０Ｄに記憶しておき、パケットＰＫｎ－１が到着したら利用する。

【０１５６】

こうすれば、パケットＰＫｎに用いられる乱数は、送信装置１０Ｄが暗号に利用した乱数ＲＮｎと同じになり、同期が取れるので、正しく復号化できる。

【実施例５】

【０１５７】

図１７は、ＵＤＰ通信に利用できるように、ストリーミング暗号の利用方法を図２からさらに改良した手段を説明するブロック図である。本実施例は、実施例４の構成に、第２乱数生成器１６２、５５２を付加したものである。なお、第１乱数生成器１６１、５５１、実施例４の乱数生成器１６、５５と同様である。

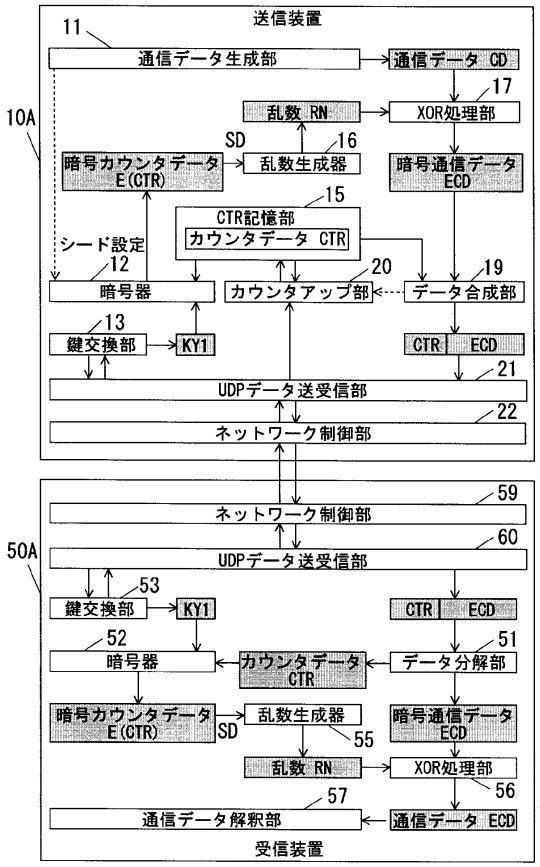
【０１５８】

図１７の暗号化・復号化手段の特徴は、シードＳＤ１を、送信装置１０Ｅ、受信装置５０Ｅで交換するのではなく、それぞれ第２乱数生成器１６２、および第２乱数生成器５５２によって生成するものとしたところである。従って、シードＳＤ２は事前に交換されて設定されていなければならない。また、パケットＰＫｎ＋ｘやパケットＰＫｎ＋ｘ＋１に記載されている通信データ数は、図１６とは異なり、１番目のパケットからの通信データ数ではなく、パケットＰＫｎからの通信データ数、つまりパケットＰＫｎを起点として、パケットＰＫｎ＋ｘやパケットＰＫｎ＋ｘ＋１を受け取る前までに、送信装置１０Ｅが送信した通信データのデータサイズとなっている。これにより、通信データ数を小さくできるので、パケットロスがあった場合に、転送速度を上げることができるようになっている

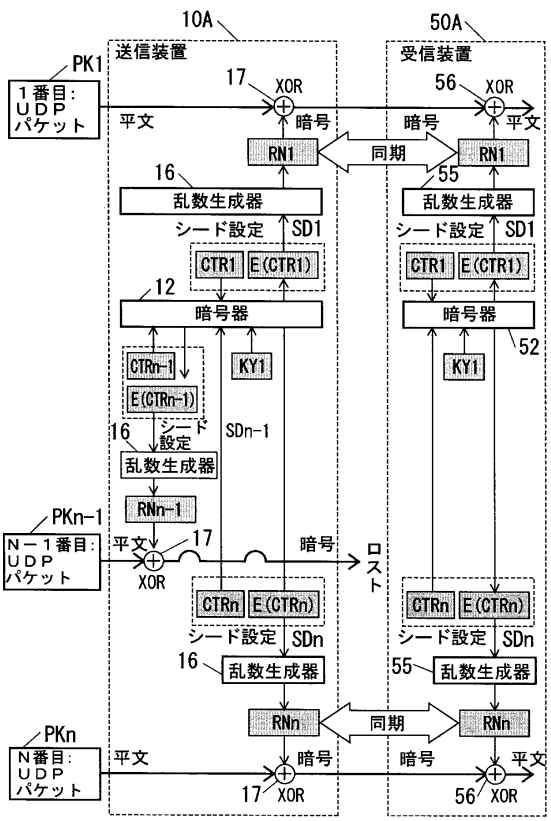
。
【０１５９】
なお、第２乱数生成器１６２、および第２乱数生成器５５２は、「Ｎ，２Ｎ，３Ｎ，・・・」というようにカウンタデータがＮステップするたびに、シード（乱数）を発生させるものとする。従って、受信装置５０ＥはカウンタデータをＵＤＰパケットから読み出せるものとする。受信装置５０ＥがＵＤＰパケットからカウンタデータを読み出せるようにするためには、送信装置１０ＥがカウンタデータをＵＤＰパケットに付与するか、若しくはＵＤＰの各ヘッダからカウンタデータを読み出すものとすれば良い。
【０１６０】
このような工夫を施せば、図１６で説明したような、かなり先のカウンタデータが付与された不正なＵＤＰパケット、あるいはかなり先の通信データ数が付与されたＵＤＰパケットを送信されても、Ｎの値を大きくしておけば、乱数発生時間を低下させることができる。具体的には、受信装置５０Ｅは、カウンタデータをＮで割った商の数の乱数を、第２乱数生成器５５２に発生させ、この最後の乱数のシードＳＤ１を第１乱数生成器５５１に設定し、カウンタデータをＮで割った余りの数の乱数を、第１乱数生成器５５１に発生させることになるので、乱数発生時間を短縮できる。
【０１６１】
ここでは、乱数発生器を２段にしているが、３段以上で構成し、さらに効率の良い乱数発生方法を得ることも可能である。
【０１６２】
但し、図１６で説明したような不正なＵＤＰパケットを送信されると、第２乱数生成器１６２が発生した乱数の中に、利用しなかった乱数が出てくる。この乱数は取っておかないと、送信装置１０Ｅが正規のＵＤＰパケットを送信してきた場合に復号化できなくなる。しかしどれくらいのデータサイズ、覚えておけば良いかを決められないので問題となる。
【０１６３】
そこで、第２乱数生成器５５２で、シードＳＤ１を発生させる場合に、第２乱数生成器５５２の内部情報を受信装置５０Ｅに記憶しておけば良い。そしてＭＡＣを利用して、ＵＤＰパケットが正しいか否かを判定し、正しくないと判定された場合は、受信装置５０Ｅに記憶しておいた第２乱数生成器５５２の内部情報を、第２乱数生成器５５２に戻せば良い。
【産業上の利用可能性】
【０１６４】
本発明に係る復号化装置、暗号化装置、復号化方法、暗号化方法、および通信システムは、パケットのロスや到着順の入れ替わりがあった場合であっても、送信側と受信側の間の同期ずれを抑制することの出来るので、セキュリティと円滑な通信を要求される通信方式に有用である。
【符号の説明】
【０１６５】
１　監視システム
１０Ａ、１０Ｂ、１０Ｄ、１０Ｅ　送信装置
１０Ｃ　ネットワークカメラ
１１　通信データ生成部
１２　暗号器
１３　鍵交換部
１５　ＣＴＲ記憶部
１６　乱数生成器
１７　ＸＯＲ処理部
１９　データ合成部
２０　カウンタアップ部

２１　ＵＤＰデータ送受信部
２２　ネットワーク制御部
２３　ＭＡＣ計算器
５０Ａ、５０Ｂ、５０Ｄ、５０Ｅ　受信装置
５０Ｃ　ＰＣ
５１　データ分解部
５２　暗号器
５３　鍵交換部
５５　乱数生成器
５６　ＸＯＲ処理部
５７　通信データ解釈部
５９　ネットワーク制御部
６０　ＵＤＰデータ送受信部
６１　ＣＴＲ記憶部
６２　カウンタチェック部
６３　ＭＡＣ計算器
６５　ＭＡＣデータ比較器
６６　カウンタアップ部
７０、７１　鍵生成部
７２、７３　映像データ生成部
１０１　カメラ部
１６１　第１乱数生成器
１６２　第２乱数生成器
２０３　モニタ
２０５　ネットワークケーブル
５５１　第１乱数生成器
５５２　第２乱数生成器
ＣＤ　通信データ
ＣＰ　暗号文
ＣＴＲ　カウンタデータ
ＥＣＤ　暗号通信データ
Ｅ（ＣＴＲ）　暗号カウンタデータ
ＥＩＤ　暗号映像データ
ＥＵＤ　暗号ＵＤＰデータ
ＨＤ１　ＭＡＣヘッダ
ＨＤ２　ＩＰヘッダ
ＨＤ３　ＵＤＰヘッダ
ＩＤ　映像データ
ＫＹ１　鍵
ＭＤ　ＭＡＣデータ
ＯＤ　元データ
ＰＫ　ＵＤＰパケット
ＰＴ　平文
ＰＵＤ　平文ＵＤＰデータ
ＲＮ　乱数
ＳＤ　シード
ＵＤＦ　ＵＤＰデータ領域

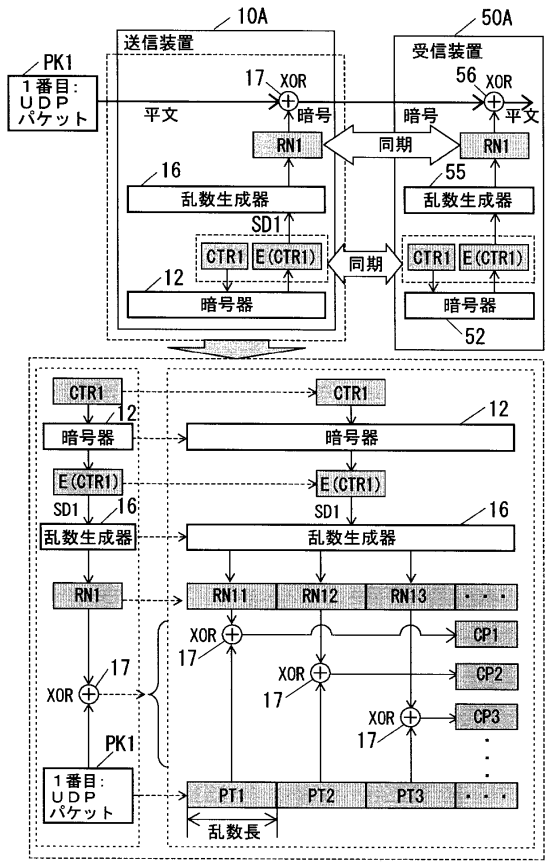
【図１】


送信装置
11
通信データ生成部
通信データ CD
乱数 RN
17
XOR処理部
10A
SD
16
暗号カウンタデータ E(CTR)
乱数生成器
暗号通信データ ECD
15
CTR記憶部
カウンタデータ CTR
シード設定
12
暗号器
20
カウンタアップ部
19
データ合成部
13
鍵交換部
KY1
CTR ECD
21
UDPデータ送受信部
22
ネットワーク制御部
59
ネットワーク制御部
60
UDPデータ送受信部
50A
53
鍵交換部
KY1
CTR ECD
52
暗号器
カウンタデータ CTR
51
データ分解部
暗号カウンタデータ E(CTR)
SD
乱数生成器
55
暗号通信データ ECD
乱数 RN
XOR処理部
56
57
通信データ解釈部
通信データ ECD
受信装置

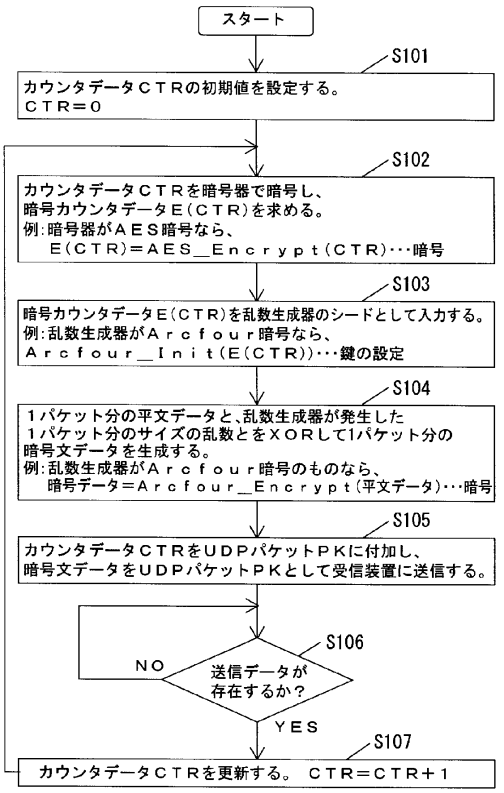
【図２】


10A
送信装置
50A
受信装置
PK1
1番目: UDP パケット
17
XOR
56
XOR
平文
暗号
暗号
平文
RN1
同期
RN1
16
乱数生成器
55
乱数生成器
シード設定
SD1
シード設定
SD1
CTR1
E(CTR1)
CTR1
E(CTR1)
12
暗号器
暗号器
52
KY1
KY1
CTRn-1
E(CTRn-1)
16
シード設定
SDn-1
乱数生成器
RNn-1
PKn-1
N－1番目: UDP パケット
平文
17
暗号
ロスト
XOR
CTRn
E(CTRn)
CTRn
E(CTRn)
シード設定
SDn
シード設定
SDn
16
乱数生成器
55
乱数生成器
RNn
同期
RNn
PKn
N番目: UDP パケット
平文
暗号
暗号
平文
17
XOR
56
XOR

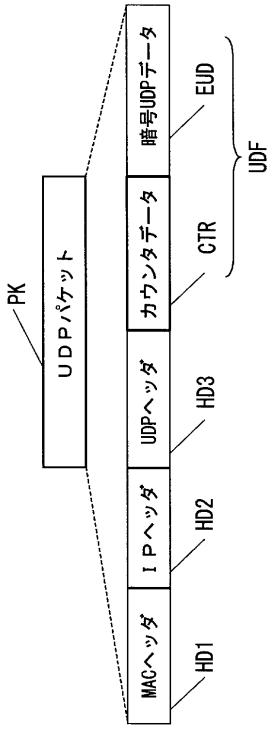
【図３】


10A
送信装置
50A
受信装置
PK1
1番目: UDP パケット
17
XOR
56
XOR
平文
暗号
暗号
平文
RN1
同期
RN1
16
乱数生成器
55
乱数生成器
SD1
CTR1
E(CTR1)
同期
CTR1
E(CTR1)
12
暗号器
暗号器
52
CTR1
CTR1
12
暗号器
暗号器
E(CTR1)
E(CTR1)
SD1
16
SD1
16
乱数生成器
乱数生成器
RN1
RN11
RN12
RN13
・・・
XOR
17
CP1
XOR
17
CP2
XOR
17
CP3
PK1
1番目: UDP パケット
PT1
PT2
PT3
・・・
乱数長

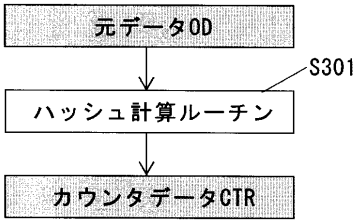
【図４】


スタート
S101
カウンタデータＣＴＲの初期値を設定する。
ＣＴＲ＝０
S102
カウンタデータＣＴＲを暗号器で暗号し、
暗号カウンタデータＥ(ＣＴＲ)を求める。
例:暗号器がＡＥＳ暗号なら、
Ｅ(ＣＴＲ)＝ＡＥＳ＿Ｅｎｃｒｙｐｔ(ＣＴＲ)･･･暗号
S103
暗号カウンタデータＥ(ＣＴＲ)を乱数生成器のシードとして入力する。
例:乱数生成器がＡｒｃｆｏｕｒ暗号なら、
Ａｒｃｆｏｕｒ＿Ｉｎｉｔ(Ｅ(ＣＴＲ))･･･鍵の設定
S104
１パケット分の平文データと、乱数生成器が発生した
１パケット分のサイズの乱数とをＸＯＲして1パケット分の
暗号文データを生成する。
例:乱数生成器がＡｒｃｆｏｕｒ暗号のものなら、
暗号データ＝Ａｒｃｆｏｕｒ＿Ｅｎｃｒｙｐｔ(平文データ)･･･暗号
S105
カウンタデータＣＴＲをＵＤＰパケットＰＫに付加し、
暗号文データをＵＤＰパケットＰＫとして受信装置に送信する。
S106
送信データが存在するか？
NO
YES
S107
カウンタデータＣＴＲを更新する。 ＣＴＲ＝ＣＴＲ＋１

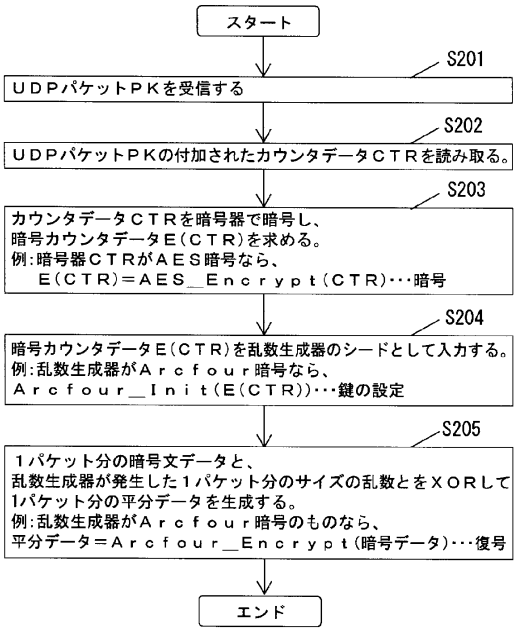
【図５】


UDPパケット
PK
暗号UDPデータ
カウンタデータ
UDPヘッダ
IPヘッダ
MACヘッダ
EUD
CTR
UDF
HD3
HD2
HD1

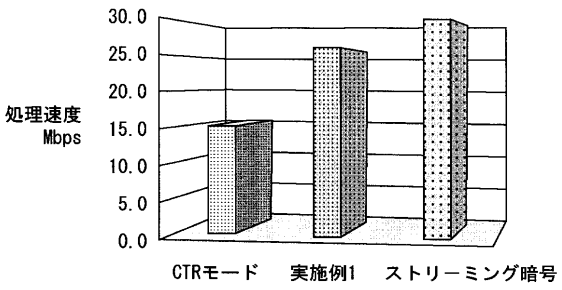
【図６】


元データOD
S301
ハッシュ計算ルーチン
カウンタデータCTR

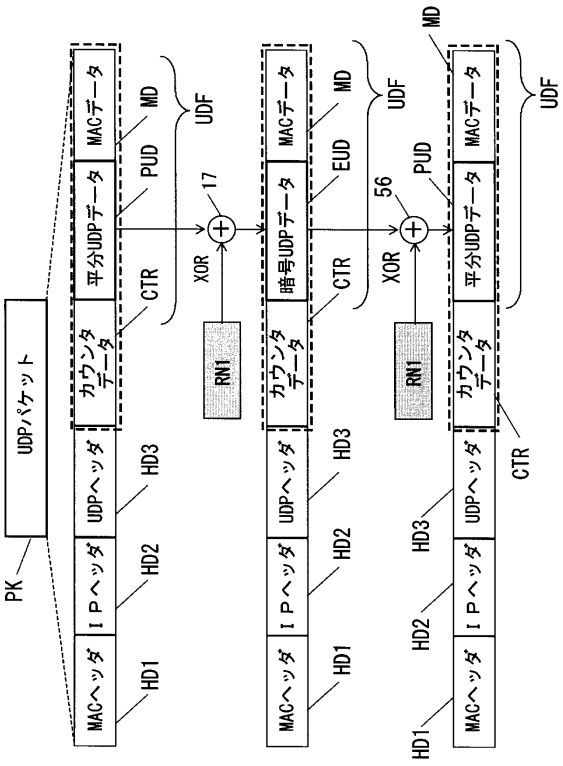
【図７】


スタート
S201
UDPパケットPKを受信する
S202
UDPパケットPKの付加されたカウンタデータCTRを読み取る。
S203
カウンタデータCTRを暗号器で暗号し、
暗号カウンタデータE(CTR)を求める。
例:暗号器CTRがAES暗号なら、
E(CTR)=AES_Encrypt(CTR)･･･暗号
S204
暗号カウンタデータE(CTR)を乱数生成器のシードとして入力する。
例:乱数生成器がArcfour暗号なら、
Arcfour_Init(E(CTR))･･･鍵の設定
S205
1パケット分の暗号文データと、
乱数生成器が発生した1パケット分のサイズの乱数とをXORして
1パケット分の平分データを生成する。
例:乱数生成器がArcfour暗号のものなら、
平分データ=Arcfour_Encrypt(暗号データ)･･･復号
エンド

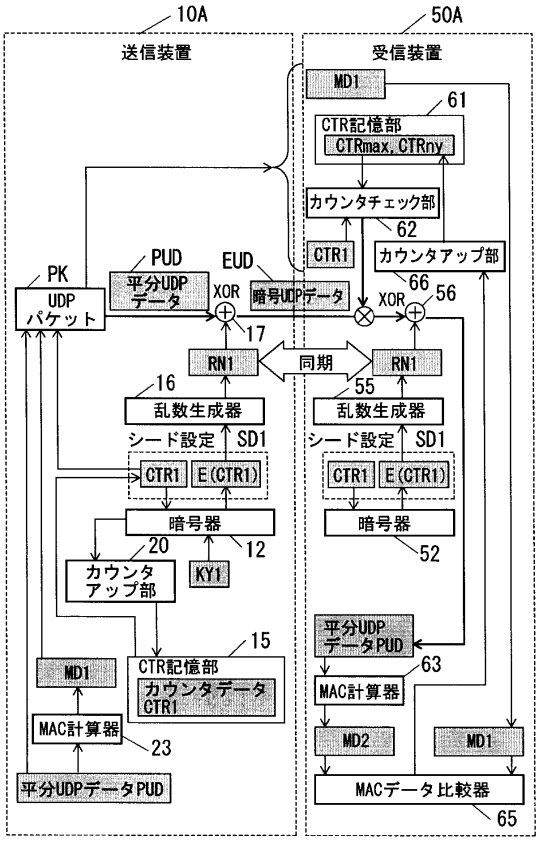
【図８】


処理速度
Mbps
30.0
25.0
20.0
15.0
10.0
5.0
0.0
CTRモード
実施例1
ストリーミング暗号

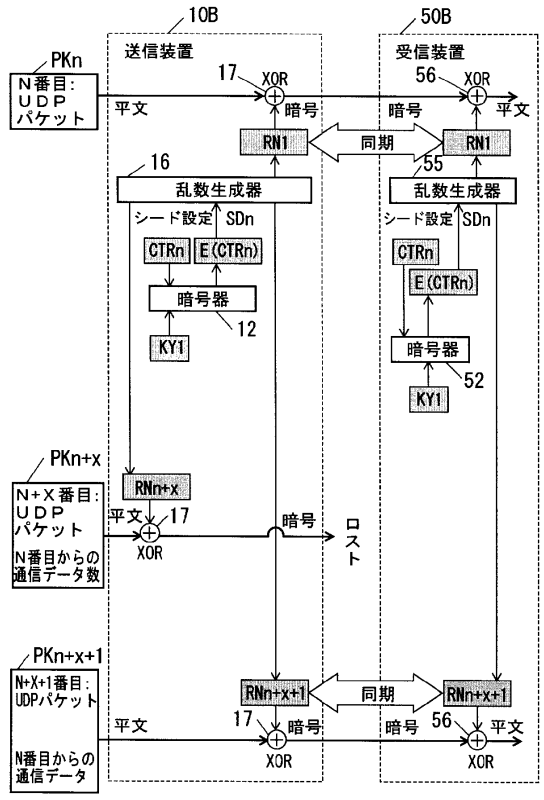
【図９】



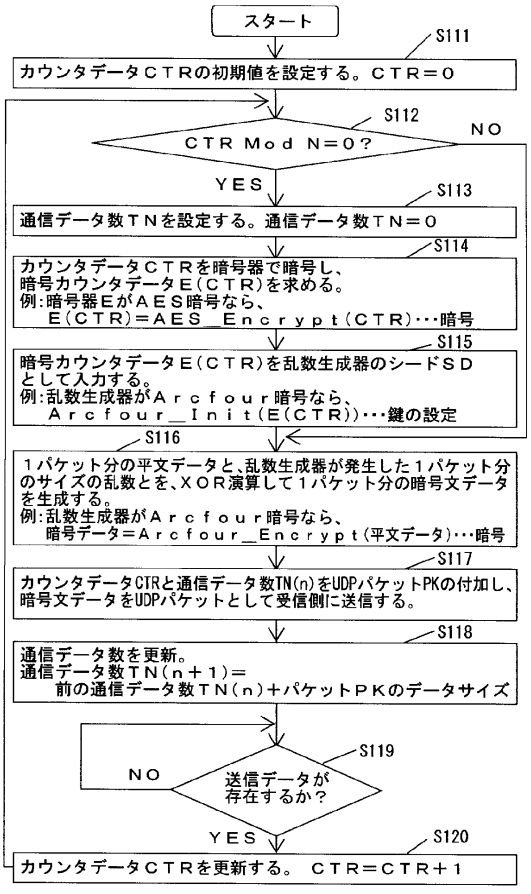
【図１０】



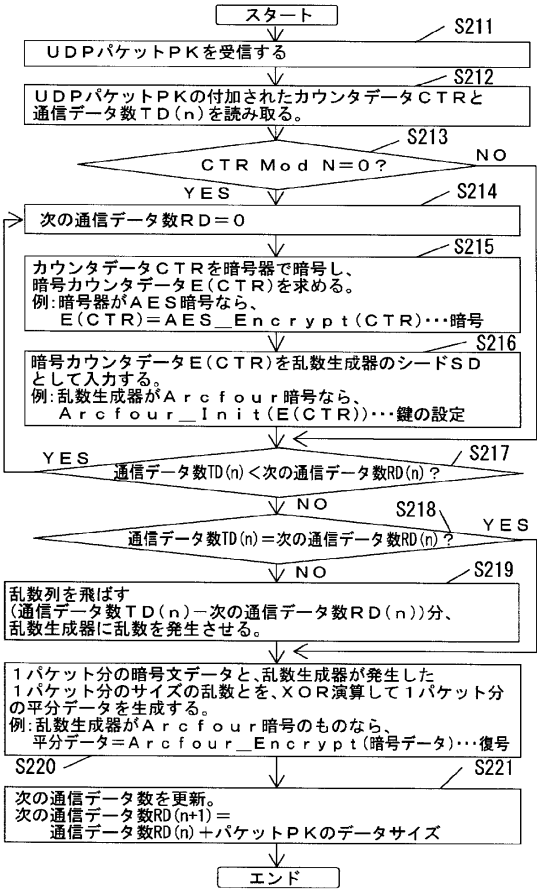
【図１１】



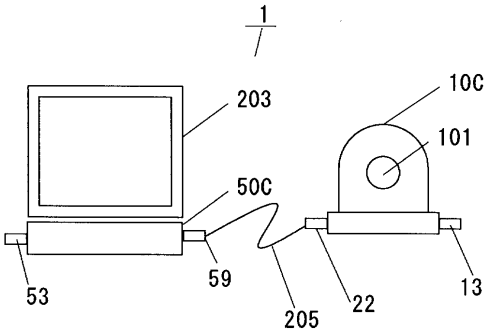
【図１２】



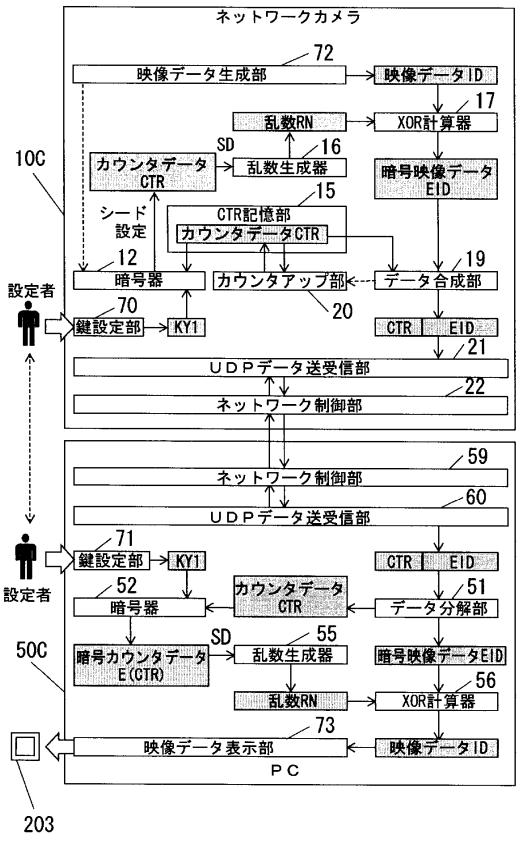
【図１３】



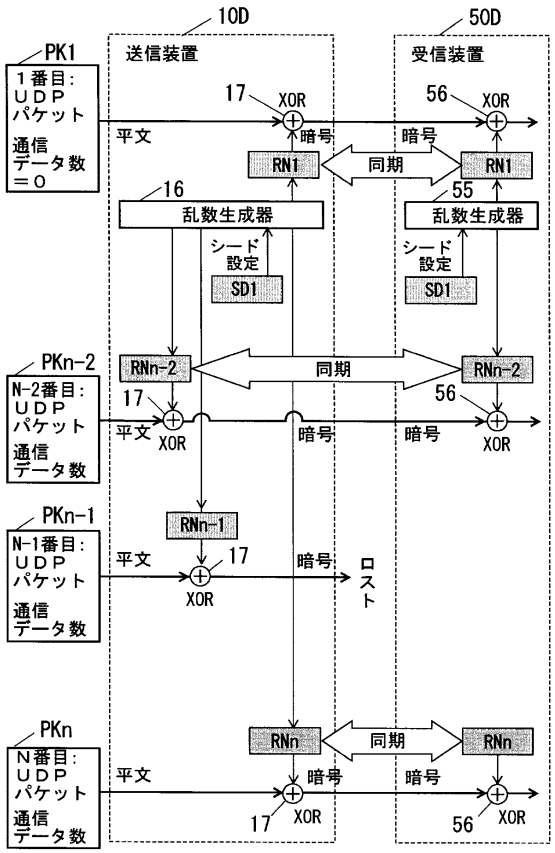
【図１４】



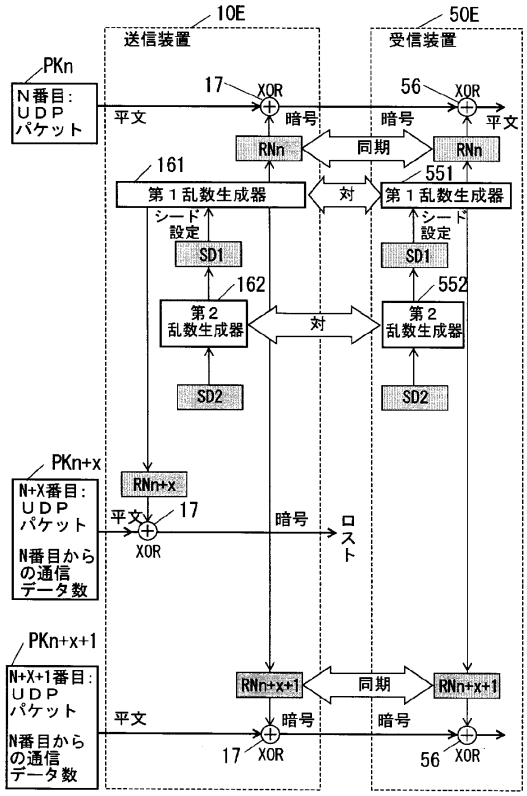
【図１５】



【図１６】



【図１７】


10E
送信装置
50E
受信装置
PKn
N番目:
UDP
パケット
17
XOR
56
XOR
平文
暗号
暗号
平文
RNn
同期
RNn
161
第１乱数生成器
対
551
第１乱数生成器
シード
設定
SD1
SD1
シード
設定
162
第２
乱数生成器
対
552
第２
乱数生成器
SD2
SD2
PKn+x
N+X番目:
UDP
パケット
N番目から
の通信
データ数
RNn+x
平文
17
XOR
暗号
ロスト
PKn+x+1
N+X+1番目:
UDP
パケット
N番目から
の通信
データ数
RNn+x+1
同期
RNn+x+1
平文
暗号
暗号
17
XOR
56
XOR

フロントページの続き

(56)参考文献 特開平０８－３３５０４０（ＪＰ，Ａ）
特開平１０－３０１４９２（ＪＰ，Ａ）
特開２００９－０８１５６４（ＪＰ，Ａ）
特開２００８－０６０８１７（ＪＰ，Ａ）

(58)調査した分野(Int.Cl.，ＤＢ名)
Ｈ０４Ｌ　　９／２０
Ｈ０４Ｌ　　９／０８
Ｈ０４Ｌ　　９／３６