MANETにおける汎用OSを用いたセキュリティメカニズムの応答性評価

宇野　美穂子[†]　　小口　正人[†]

近年大きく注目されている MANET は，端末が集まるだけで構築可能な自律分散型のネットワークである．MANET では，マルチホップルーティングプロトコルによりノードが通信を中継する通信経路が構築され，無線の電波範囲にとらわれない広範囲の通信を可能にしている．無線通信は有線と比べ傍受や改竄がされやすく，特に第三者が中継を行うマルチホップネットワークでは，暗号化によりデータを守ることが必須である．ただし利用環境やアプリケーションにより認証強度や応答時間の要求は異なる．そこで本研究では，セキュリティメカニズムのリアルタイム性に着目した．汎用 OS に IPsec 等の既存技術を取り入れて構築したマルチホップネットワークにおいて，OS のプリエンプション機能を有効にし，CPU に負荷を与えた場合と与えない場合の応答時間を測定することにより，マルチメディアデータのエンコーディング負荷による影響を評価した．

An Evaluation of Response Time of the Security Mechanism Using the Universal OS for MANET

Mihoko Uno[†] and Masato Oguchi[†]

Mobile Ad-hoc Network(MANET) is an autonomous distributed network which is constructed only with gathered nodes on the spot. In MANET, wide area communication on wireless networks is realized because a node can relay other node's packets by adopting a multi-hop routing protocol. In this case, the demand for the level of authentication and response time is different depending on the environment and applications. Thus, we focus on real-time control of the security mechanism. In this paper, we have established multi-hop communication environment and introduced methods to control secure connections by applying IPsec as an encryption scheme using the general-purpose operating systems with enabled preemption. Under such conditions, we have measured response time with or without burdened CPU. Inaddition, we have evaluated the influence by encording multimedia data as an application load and investigated on a security realization method for MANET.

1. はじめに

近年，無線通信技術の進歩に伴い，様々な形態の無線ネットワークが考えられるようになった．特に，既設のインターネットなどインフラネットワークとの接点を持たず，無線 LAN 機能を持つ端末のみでネットワークを構築することで，より自由な形でネットワークを構築できる MANET (Mobile Ad-hoc Network)[1)] が，大いに注目されている．MANET では，ノードがある程度広い範囲に分散している場合など目的のノードへ直接通信できない時に，途中ノードが通信を中継していくことでより広範囲の通信を可能にしている．このように構築されるネットワークを，一般にマルチホップネットワークと呼び，これを図 1 に示す． マルチホップネットワークでは，各端末が通信経路の制御等を行うルーティングプロトコルに従ってルーティングを行い，ネットワークを構築している．このためノードの参加や離脱によるネットワーク構成の変化を気にすることなく，無線の電波範囲にとらわれない通信が行えるため，ユビキタスネットワークを実現する技術として大いに期待されている．利用法としては，

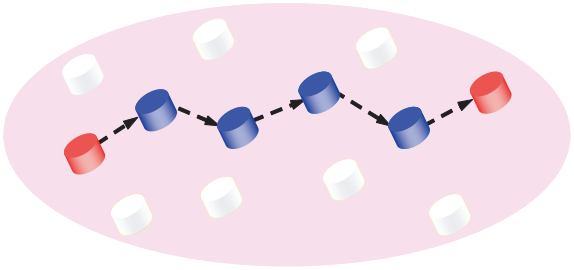

図 1　マルチホップネットワーク

† お茶の水女子大学
Ochanomizu University

例えば，災害時やイベント会場などでの通信やインフラが整備されておらずアクセスポイントの電波が届かない場所での通信，車車間通信などが挙げられる．

一般に無線通信は有線と比べ通信の傍受や改竄がされやすい環境であり，セキュリティを考慮することは必要不可欠である．特に MANET のような環境は不特定多数のノードが存在し，知らない相手が通信経路に加わるため，よりセキュリティ上の危険性が高くなる可能性がある．

MANET におけるセキュリティ技術は現在研究が進められているが，その提案の多くはアクセスポイントなどの固定インフラを介し，一時的にインターネットに接続できるような環境を前提としている．しかしマルチホップネットワークにおいては，固定インフラの存在しない状況で安全な通信が必要となる場合も考えられる．そのため，固定インフラの存在しないネットワークにおいても有効となり，かつネットワーク構成の変化にも柔軟に対応するセキュリティ対策が求められる．ただし，利用環境により認証強度や応答時間の要求は異なるという点を配慮する必要がある．ユーザやアプリケーションの制約によっては，応答時間が長過ぎるセキュリティメカニズムの利用は難しい．

そこで本研究では，セキュリティメカニズムのリアルタイム性に着目した．リアルタイム制御可能なシステムには組込み OS と汎用 OS があるが，前者は厳格なリアルタイム制御を実現しやすいものの，利用できるアプリケーションが限られるなど拡張性が乏しい．よって，プラットフォームとしては応用範囲の広い汎用 OS を用いることにした．評価としては，マルチメディアデータのストリーミング配信を行っているノードに対し，MANET 上でセキュアコネクション構築のリクエストを送り，認証が行われてセキュアコネクションが生成されるまでの応答時間を測定する．一般にストリーミング配信を行うノードは負荷が高いのが，それによりセキュリティメカニズムの応答性が悪くなっては利用が制約されてしまう．

本研究では，まずマルチホップ通信環境を構築してセキュリティメカニズムを実装し，プリエンプション機能を有効にするためにカーネル再構築を行った．そして，マルチメディアデータをエンコードすることによりアプリケーション負荷が高い状態とし，応答時間を測定して評価する．

本稿は以下のように構成される．まず第 2 章では研究背景としてマルチホップネットワークを実現するルーティングプロトコルについて記述する．第 3 章では無線 LAN におけるセキュリティ技術と IPsec，第 4 章ではリアルタイム制御について触れ，第 5 章では研究方針として既存研究を紹介し，ストリーミングについて述べる．そして，第 6 章ではアプリケーション負荷を与えた際の IPsec コネクション時間についての実験と結果を示す．最後に第 7 章でまとめる．

2. マルチホップルーティングプロトコル

マルチホップネットワークは，MANET に集まったノードに通信を中継する機能を持たせ，直接通信できないノード間のコネクションを実現するものである．そのためには MANET 内でマルチホップ通信の経路を探して選択するルーティングプロトコルが必要である．MANET はノードの移動や参加，離脱などネットワーク構成が動的に変化する環境であり，また低ビットレートで不安定な無線通信を用いるため，その性質を考慮したルーティングプロトコルが多く提案されている．

MANET におけるルーティングプロトコルは，パケットをルーティングする経路の取得方法により，プロアクティブ型とリアクティブ型の大きく 2 つに分けることができる．プロアクティブ型は，ネットワーク内のノード同士が定期的に経路情報を交換し，他ノードへの経路を常に把握する方法である．各ノードはパケットをフォワーディングする次ノードを，目的地のノード各々についてルーティングテーブルで保持するため，通信のリクエストが発生した場合即座に経路生成ができる．しかしネットワーク構成の変化に応じて頻繁に経路情報の交換をする必要があり，通信要求の発生パターンによっては無駄な処理が多く起こり得る．プロアクティブ型のルーティングプロトコルには，OLSR (Optimized Link State Routing)[2] や TBRPF (Topology Broadcast based on Reverse-Path Forwarding)[3] などがある．

一方リアクティブ型は，通信のリクエストが発生するごとに，送信ノードから受信ノードまでの経路を探索する方法である．経路探索によって経路が構築されてからパケットがルーティングされるため，通信のリクエストが発生してからの処理に遅延が生じるという問題点がある．リアクティブ型のルーティングプロトコルには，AODV (Ad-hoc On-demand Distance Vector)[4] や DSR (Dynamic Source Routing)[5] などがある．

3. ネットワークセキュリティ技術

3.1 無線 LAN におけるセキュリティ技術

ネットワークにおける通信プロトコルは階層構造

を持つことが一般的であるが，暗号化通信方式も各階層に様々な方式が存在し，要求される目的に応じ異なったセキュリティ技術を設定することが可能である．現在無線 LAN では，データリンク層で暗号化を行う WEP (Wired Equivalent Privacy) の使用が標準として規定されている．WEP は，端末とアクセスポイント間で事前に秘密鍵を共有するが，その鍵長の短さや暗号化アルゴリズム RC4 適用手法の安全性に対する不安等，脆弱性が指摘されている．その後，WEP の弱点を補強する暗号化プロトコル TKIP (Temporal Key Integrity Protocol) と，安全性の高いユーザ認証を行う規格 IEEE802.1X を併用した WPA (Wi-Fi Protected Access) が制定された．この WPA は，IEEE802.11i が普及するまでのサブセットの位置づけとなっている．IEEE802.11i は，TKIP や IEEE802.1X に加え，新たな暗号化アルゴリズムとして強固な AES (Advanced Encryption Standard) を採用しており，認証後，暗号通信に用いる秘密鍵の作成と交換を行っている．この認証に必要となる認証サーバ等は，アクセスポイントを介した固定インフラ上に置かれることが一般的である．

3.2 IPsec (IP Security)

3.2.1 IPsec 概要

IPsec はネットワーク層で暗号化と認証を行うことでセキュリティを保証する規格で，IP パケットを暗号化するため，上位のアプリケーションは透過的な通信を行うことができる．IP パケットそのものを暗号化することで通信の傍受を防ぎ，また IP パケットに改竄を検知する数値を組み込むことで，パケットが通信経路上で改竄されなかったことを保証する．暗号化には共通鍵暗号方式を用いており，暗号化アルゴリズムとしては DES (Data Encryption Standard) や 3DES が使用される．またセキュリティプロトコルとして，暗号化と認証機能を提供する ESP (Encapsulating Security Payload)，暗号化機能はないがより強力な認証機能を提供する AH (Authentication Header) のどちらかを選択することができる．

3.2.2 IKE (Internet Key Exchange)

IPsec では，生成されるコネクション SA (Security Association) の管理および生成や，ESP や AH で用いる鍵の交換などを行うプロトコルとして，IKE がサポートされている．IKE は IPsec 化すべきパケットが発生すると，セキュリティポリシに従って SA を自動的に生成する．この際，暗号化や認証に使用する鍵なども自動的に生成され，さらに確立した SA では一定期間ごとに使用された鍵が作り直される．IPsec で使用される鍵は共通暗号鍵であり，送信側と受信側で同じ鍵を持つ必要があるが，この共通暗号鍵を IKE が作成する際には公開鍵暗号技術を用いた Diffie-Hellman アルゴリズムが使用されており，アルゴリズムにしたがって発生した乱数を交換することで，その交換が盗聴されていたとしても盗聴者には知ることのできない共通鍵暗号鍵を作成して共有することができる．

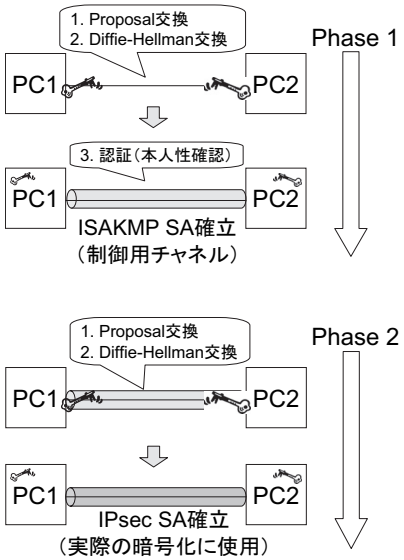

図 2 IPsec SA 確立までの IKE のプロセス

ここで，IKE が IPsec SA 確立までに行うプロセスを説明する [図 2]．IKE は，生成する SA のパラメータをネゴシエートして決定する Proposal 交換，生成する SA の秘密対称鍵を公開鍵暗号技術により安全に作成する Diffie-Hellman 交換，IKE 通信している相手が本物であることを確認する認証（本人性確認），以上 3 つの基本的な機能を通して SA の自動生成を行う．プロセスは大きく二つの段階に分けられ，Phase 1 で IKE の制御用チャネルである ISAKMP (Internet Security Association and Key Management Protocol) SA の確立を行い，その後 Phase 2 で暗号化経路 IPsec SA の確立を行う．

Phase 1 ではまず Proposal 交換を行い，続いて ISAKMP SA 用の共通暗号鍵を Diffie-Hellman 交換により作成する．そして，その鍵を用いた暗号化経路上で，IKE 相手の認証（本人性確認）が行われ，制御用チャネル ISAKMP SA が確立される．

次に Phase 2 では，IPsec SA を生成するための Proposal 交換を行い，実際に暗号化に用いる IPsec SA 用の共通暗号鍵を Diffie-Hellman 交換により作成

し，IPsec SA が確立される．この Phase 2 のやりとりは，Phase 1 で既に作られた ISAKMP SA を通して送られるので，暗号化された安全な経路上で通信を行うことができる．

4. リアルタイム制御

4.1 OS のリアルタイム性

OS におけるリアルタイムとは，あらかじめ定められた一定の時間内に処理が完了することを保証する能力のことである．その結果，アプリケーションのレベルでも常に一定の時間内に応答が返ることが期待できる．

リアルタイム制御可能なシステムとしては，組込み OS と汎用 OS がある．組込み OS とは、電子機器や産業用機械などに内臓される組込みシステムを制御するために用いられる OS のことである。それ故，高いリアルタイム性や少ないメモリで動作するコンパクトさ，信頼性などにおいては優れているが，拡張性には乏しい．それに対し，汎用 OS は様々なアプリケーションをサポートすることを主としており，組込み OS と比べると応答性という点では劣る．そこで本研究では，汎用 OS におけるリアルタイム性に焦点を当てることにした．MANET において汎用 OS でマルチメディアアプリケーションなどが動作している場合でも，システムが高い応答性を保つための実現手法を検討する．

4.2 プリエンプション

プリエンプションとは，カーネルがプロセスを実行中に自ら割り込み，サウンドや画面表示など優先度の高い他のプロセスを実行する機能である[6]．プロセススケジューラは，新しいプロセスが実行可能状態になるたびに，プリエンプトする必要があるか否かを判断する．この動作を図 3 に示す．プロセス A とプロセス B があり，プロセス B はプロセス A よりも優先度が高いとする．プロセス A の処理中に割り込みが発生すると再スケジューリングが行われる．プロセス B が実行可能状態になるとプロセス A の処理は中断され，プロセス B の処理に切り替えられる．そして，プロセス B の処理が終了した後に，再びプロセス A の処理を再開することになる[7]．

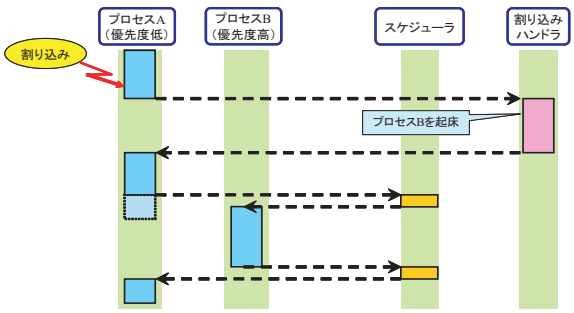

図 3 プリエンプティブな場合の割り込み処理

プリエンプションが可能なマルチタスク処理のことをプリエンプティブ・マルチタスク処理という．Linux kernel 2.6 はプリエンプティブ・マルチタスク処理をサポートした OS である．通常，カーネルモードでプロセスを実行している間はプロセススケジューラが起動されないようになっている．カーネルモードでのプリエンプションを可能にしたカーネルは，プリエンプティブ・カーネルと呼ばれる．kernel 2.6 では，一部の処理でプリエンプションが可能であり，カーネル・コンパイル時の設定でプリエンプティブ・カーネルを作ることができる[8]．

5. 研究方針

5.1 既存研究

既存研究では，マルチホップネットワークにおいてセキュアコネクションを自動構築し管理を行う手法を提案している[9]．セキュアコネクションの生成の際，目的ノードへの経路については事前に確定していることが望ましいため，マルチホップルーティングプロトコルには，プロアクティブ型である OLSR を使用している．暗号化技術には WPA や IEEE802.11i などがあるが，これらの規格は認証の機能も含んでおり，アクセスポイントなどの固定インフラを含むネットワークにおいて有効となる手法である．そこで，各コネクションごとに共通暗号鍵を作成することができ，それぞれ独立したセキュアコネクションを生成することができる IPsec を使用している．

我々は、これまでに上記の研究と同様に，OLSR によって管理されるマルチホップ環境を構築した上で，プロセスを実行中に割り込みが発生した際に優先度が高い他のプロセスを実行するプリエンプションの機能を取り入れ，送受信ノード間の通信を IPsec によって暗号化してセキュア通信経路を生成する際に要する時間を測定し，評価してきた[10]．具体的には，1 対 1 通信と 2 ホップのマルチホップ通信において，優先度を指定したループプログラムをバックグラウンドプロセスとして実行することでアプリケーションが CPU に負荷を与える方法を変えて実験を行い，その環境における IPsec コネクション構築時間を測定した．その結果，宛先ノードに負荷を与えたときには，どちらの場合もバックグラウンドプロセスであるループプログラ

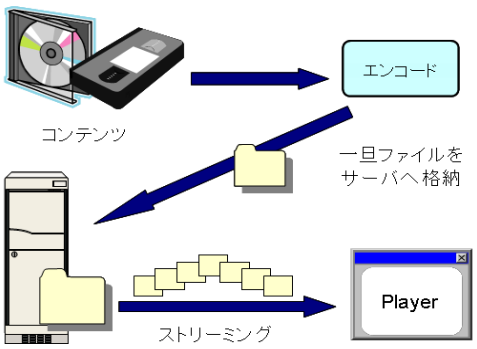

図 4 オンデマンド型配信

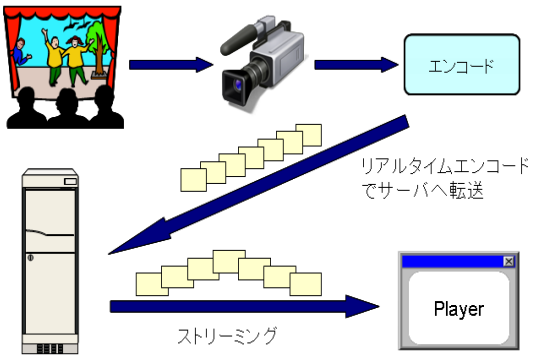

図 5 ライブ型配信

ムの優先度が高くなると，セキュアコネクション構築時間が長くなることが確認された．また，マルチホップ通信では中継ノードに負荷を与えた場合は，構築時間は負荷がないときと同程度であるという結果が得られた．

既存研究では仮想的なアプリケーションとしてループプログラムを用いたが，本研究では，バックグラウンドプロセスとしてマルチメディアデータ・エンコーダを使用し，ストリーミング配信などを想定したより現実的なアプリケーションを用いて実験を行うことにする．

5.2 ストリーミング

ストリーミングとは，ネットワークを通じて映像や音声などのマルチメディアデータを視聴する際にデータを受信しながら同時に再生を行なう方式である．コンテンツをダウンロードする時間を待つことなく，すぐに再生することができる．また，コンテンツが視聴者のハードディスクにはデータとして残らないので，不正コピーを防ぐことができる．

ストリーミングにはオンデマンド型配信とライブ型配信の２通りの配信方法がある．オンデマンド型配信では、あらかじめ動画ファイルを作成し、それをサーバにアップロードしておく．そして，ユーザが見たい時にサーバにアクセスし，見たい映像コンテンツを引き出して自由に再生する事ができる [図 4]．これに対し，ライブ型配信は撮った映像を順次ストリーミング配信用のデータに変換してリアルタイムに配信する [図 5]．

ストリーミングを行うには，映像や音声をインターネットを通じて配信できるフォーマットに変換する必要がある．この映像や音声を変換することをエンコードといい，変換するソフトウェアまたはハードウェアのことをエンコーダと呼ぶ．ライブ型配信では，エンコーダはキャプチャされた映像，音声をリアルタイムにエンコードしてサーバに送り出す[11]．

本研究では，MANET 上でストリーミング配信のエンコードを行っているノードに対し，セキュアコネクション構築のリクエストを送ってその応答性を評価する．

6. 負荷による応答時間への影響評価

6.1 実 験 概 要

本研究では，2 ホップのマルチホップ通信の場合について，CPU に負荷を与える方法を変えて実験を行い，IPsec コネクション構築時間を測定する．

実験環境として，IEEE802.11b 無線 LAN 機能を持つ 3 台の端末を用い，セキュアな通信を行う実験システムを構築した．各マシンのスペックを表 1 に示す．

表 1 各マシンのスペック

Node	OS	CPU	メインメモリ
A	Linux2.6.11	Intel PentiumM 1.73GHz	512MB
B	Linux2.6.11	Intel PentiumM 1.73GHz	512MB
C	Linux2.6.11	Intel PentiumM 1.3GHz	512MB

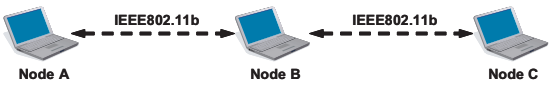

図 6 マルチホップ通信

無線 LAN クライアントには BUFFALO WLI-PCM-L11GP を用いた[12]．OLSR と IPsec の Linux における実装として，それぞれ olsrd[13] と openswan[14] を使用した．IPsec の設定では，暗号化アルゴリズムは 3DES，セキュリティプロトコルは暗号化を行う ESP，パケットのカプセル化モードはエンドツーエンドの通

信を行うトランスポートモードを選択している．この実験環境において，カーネルのプリエンプション機能を有効にすることにより，プリエンプティブカーネルを再構築した．

IPsec は複数のプロセスから成り立っており，デフォルトでの優先度は，IPsec の鍵交換をつかさどる IKE デーモンである PLUTO は 10，それ以外のプロセスは 0 と設定されている．優先度は数字が大きくなるにつれて低くなる．また，OLSR は 0 と設定されているが，両者ともこのデフォルトの状態で実験を行った．CPU に負荷を与えるアプリケーションには，REALNETWORK 社のエンコーダである Real Producer を使用し，音声データのエンコーディングを行った．以降では，このエンコード・プロセスをバックグラウンドプロセスと呼ぶことにする．バックグラウンドプロセスの優先度は，nice コマンドで-20(高) から 19(低) の範囲で指定した．

図 6 に示すように，ノード A を送信元ノード，ノード B を中継ノード，ノード C を宛先ノードとして 2 ホップ通信でのセキュアコネクション構築時間を測定した．OLSR を用いマルチホップネットワーク環境を構築する場合，実験の都合上，全ノードが直接無線通信できる範囲に存在してしまうため，送信ノードと受信ノードが直接通信を行わないよう互いのパケットを遮断させる必要がある．そこで，ノード A からノード B を中継しノード C に辿り着くようルーティングテーブルを設定し，ノード A ではノード C からのパケットを，ノード C ではノード A からのパケットを，iptables コマンドにより遮断することで，マルチホップ通信を行う環境を実現した．また，olsrd 実行時に IPsec 接続を行うと，IPsec 処理によりルーティングが変更されてしまい，その後 OLSR により正しいルーティングに戻らず通信ができなくなってしまうという問題が発生した．これは IPsec または olsrd の実装上の問題と考えられ，両端のノードを一時的にネットワークから落とし，復活させることで OLSR により正しいルーティングに再設定される．そのため，測定の都度，ネットワークを再起動し，ルーティングを確認するという作業を行った．

上記の実験環境において，中継ノード B に負荷を与えたときと宛先ノード C に負荷を与えたときの 2 つの場合について測定を行った．実験手順は，まず送信元ノード A と宛先ノード C で IPsec デーモンを起動し IPsec 接続が可能な状態にする．次に，中継ノード B または宛先ノード C でバックグラウンドプロセスを実行し，IPsec の接続開始から IPsec コネクションの確立までに要するレスポンスタイムを tcpdump コマンドで測定する．

6.2 測定結果

図 7 は中継ノード B に負荷を与えたとき，図 8 は宛先ノード C に負荷を与えたときの実験結果である．それぞれ負荷を掛けない場合，バックグラウンドプロセスの優先度を 0 に指定した場合，-10，-20 に指定した高優先度の場合，19 に指定した低優先度の場合の５つのケースについて測定した．グラフの横軸はバックグラウンドプロセスの優先度，縦軸はレスポンスタイム [sec] を表している．

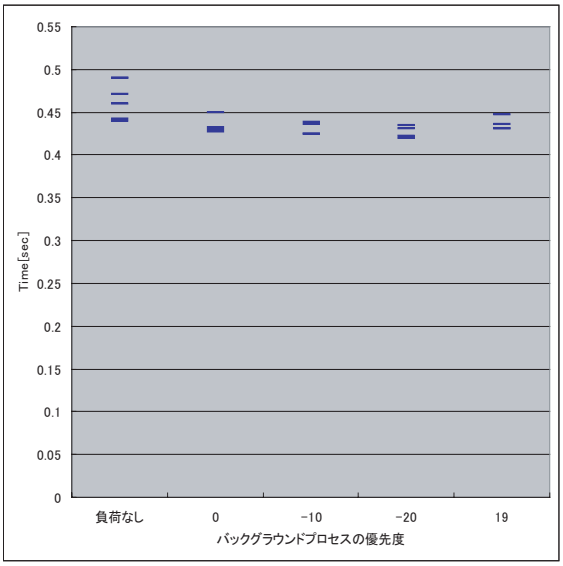

図 7 中継ノード B に負荷を与えた場合

図 7 から，中継ノードに負荷を与えても，セキュアコネクション構築時間には影響を及ぼさないということが結果が得られた．中継ノードでは IP 層におけるルーティングのみが行われており，その処理には負荷や優先度があまり影響しないためと推測される．一方，図 8 で示されるように宛先ノードに負荷を与えた場合は，バックグラウンドプロセスの優先度によってセキュアコネクション構築時間が異なるということが分かった．負荷を与えなかったときとバックグラウンドプロセスの優先度を指定しなかったときは同程度の時間で IPsec コネクションが確立するが，バックグラウンドプロセスの優先度が高くなった場合には構築時間が長くなる．また，バックグラウンドプロセスの優先度が低いときは，負荷を与えなかったときと同程度で確立されることが分かった．宛先ノードでは、３節

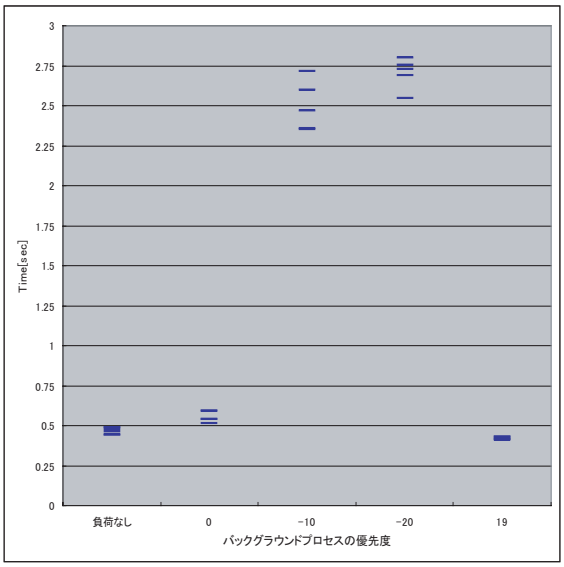

図 8 宛先ノード C に負荷を与えた場合

で述べたように IKE による ISAKMP SA の確立や IPsec SA の確立が行われている．これらの処理には負荷が大きく影響し，その優先度により応答性能が決まる．IPsec の鍵交換をつかさどる IKE デーモンである PLUTO は 10，それ以外のプロセスは 0 と設定されていることから，バックグラウンドプロセスの優先度が 0 よりも高いと IPsec コネクション確立にかかる時間が長くなると考えられる．

7. ま と め

本稿では，汎用 OS を用いて構築されたマルチホップネットワークにおいて，ストリーミング配信を行っているノードを想定し，マルチメディアデータのエンコーディングにより負荷を与えた際のセキュアコネクション構築時間を測定して比較した．その結果から，中継ノードにおいて CPU に負荷を与えた場合には影響はほとんどないが，宛先ノードにおいて CPU に与える負荷を変化させるとセキュアコネクション構築時間も変化するということが分かった．今後は，ホームネットワークや ITS などへの応用を目指し，環境によって認証強度やレスポンスタイムを選択できるような方式を検討していく．

参 考 文 献

1) MANET：http://www.ietf.org/html.charters/manet-charter.html
2) OLSR：http://hipercom.inria.fr/olsr/
3) TBRPF：http://www.ietf.org/rfc/rfc3684.txt
4) AODV：http://www.aodv.org/
5) DSR:http://www.ietf.org/internet-drafts/draft-ietf-manet-dsr-10.txt
6) Greg Kroab-Hartman:LINUX カーネル クイックリファレンス，オライリー・ジャパン
7) Preemptible Kernel：http://monoist.atmarkit.co.jp/fembedded/rtos03/rtos03b.html
8) 日経 Linux 2003 年 11 月号，日経 BP 社
9) 鎌田 美緒, 小口 正人：“ マルチホップルーティングプロトコルを用いたセキュアコネクション構築管理手法 ”，DEWS2007, C2-7, 2007 年 2 月
10) 宇野 美穂子，小口 正人：“ マルチホップネットワークにおけるセキュリティ実現方式の応答時間に関する評価 ”，DEWS2008, D7-5, 2008 年 3 月
11) http://www.jp.realnetworks.com/
12) BUFFALO WLI-PCM-L11GP：http://buffalo.jp/products/catalog/item/w/wli-pcm-l11gp/
13) olsrd：http://www.olsr.org/
14) Linux openswan:http://www.openswan.com/