

アクティブネットワーク技術を利用したDDoS攻撃対策システムの構築 及び評価

森川 裕介[†] 柏 大^{††} 松本 真弥[†] 重野 寛[†] 岡田 謙一[†]
松下 温^{†††}

† 慶應義塾大学理工学部 〒223-8522 神奈川県横浜市港北区日吉3-14-1

†† NTT情報流通プラットフォーム研究所 〒239-0847 神奈川県横須賀市光ヶ丘1-1

††† 東京工科大学 〒192-0982 東京都八王子市片倉町1404-1

E-mail: †{morikawa, matumoto, shigeno, okada, on}@mos.ics.keio.ac.jp, ††kashiwa@nttmhs.tnl.ntt.co.jp

あらまし 本研究ではインターネットセキュリティにおいて問題視されているDDoS (Distributed Denial of Service) 攻撃に着目し、これに対し効果的に対策を行うシステムを提案する。このシステムは、アクティブネットワーク技術を利用してルータが連携することで、悪意のあるユーザのトラヒックの帯域を制限し、正規ユーザのトラヒックに影響を及ぼさない、Active Shaping モデルを利用している。本論文では上記のモデルを実現するシステムデザインについて述べ、プロトタイプシステムを用いた仮想 DDoS 攻撃に関しての評価結果について述べる。

キーワード インターネットセキュリティ, DDoS 攻撃, アクティブネットワーク

Design and evaluation of countermeasure system against DDoS attacks using active network technology

Yusuke MORIKAWA[†], Dai KASHIWA^{††}, Shin-ya MATSUMOTO[†], Hiroshi SHIGENO[†], Ken-ichi OKADA[†], and Yutaka MATSUSHITA^{†††}

† Faculty of Science and Engineering, Keio University Hiyoshi 3-14-1, Kouhoku-ku, Yokohama-shi,
223-8522 Japan

†† NTT Information Sharing Platform Laboratories Hikarigaoka 1-1, Yokosuka-shi, 239-0847 Japan

††† Tokyo University of Technology Katakura-cho 1404-1, Hachioji-shi, 192-0982 Japan

E-mail: †{morikawa, matumoto, shigeno, okada, on}@mos.ics.keio.ac.jp, ††kashiwa@nttmhs.tnl.ntt.co.jp

Abstract In this paper, we propose a countermeasure system against DDoS(Distributed Denial of Service) attacks. In the system, cooperating routers using Active Network technology with each other, malicious traffic is shaped and only legitimate traffic can be communicating. This paper describes a model of the system realizing proposed architecture and the evaluation result about the effectiveness of the prototype when causing a false DDoS attack.

Key words Internet Security, DDoS Attack, Active Network

1. 導入

インターネットを利用しビジネスを展開する場合、一時的なサービスの停止が企業にとって大きな損害となる。例えば、直接的な金銭的損失ばかりでなく企業イメージの低下による間接的な損失にもつながることが考えられる。このようなことから、サービスを常に提供できる状態を維持するというサービスの可用性の確保が、インターネットを利用したサービスを提供する

場合において非常に重要となる。

このような中で2000年の2月、アメリカの大手電子商取引サイト-Yahoo, Amazon.com, eBayなどが分散型サービス停止(DDoS: Distributed Denial of Service)攻撃を受け、長時間サービスが停止した[1]。これ以降、DDoS攻撃はインターネットセキュリティにおいて脅威となる存在の一つとなった。DDoS攻撃では、攻撃者は攻撃用のプログラムをネットワーク上でセキュリティに問題がある多数のホストに侵入させ、ター

ゲットとなるサーバやネットワークに対して各ホストが一斉に大量のパケットを送信するものである。攻撃の際にはいわゆる「踏台」が大量に利用されることが多い。近年では安価で手軽な高帯域型常時接続サービスも開始されており、セキュリティの知識に乏しい個人ユーザのホストが増加するため、踏台となる端末が増加し、今後も被害が増加することが考えられる。DDoS攻撃の種類として、「TCP (SYN) Flood」、「UDP Flood」、「Smurf」などが確認されている。また、DDoS攻撃自体の特徴として攻撃する多数のホストが広く分散化していることが挙げられるため、ユーザ側で対策を行うのでは効率が悪く、広域型の効果的な対策手法が早急に求められている。攻撃に対し、各サイト毎に攻撃パケットをフィルタリングする手法も既に普及されている。しかし、フィルタリングを行うノードに非常に大きな負荷がかかってしまうこと、踏台となっているホストから攻撃を受けているサイトまでのネットワーク帯域を無駄に消費してしまうこと、攻撃パケットのみを十分に特定することができず誤って正規パケットまで廃棄してしまうこと、などが問題としてあげられている。

上記で述べたとおり、多数の分散したホストから攻撃が行われる DDoS 攻撃に対し、サイト毎に対策を行うことは非効率的と思われる。そこでこの論文では、広域型の DDoS 攻撃対策システムについて述べる。提案システムは、ISP バックボーンなどのネットワーク側で対策を行う。提案システムでは、アクティブネットワーク技術を利用したネットワークノード（ルータ）が自律的に行動し、攻撃を検出するとまずローカルで帯域制限を行う。それと同時に広域で対策を行い無駄な帯域の消費を抑制する。さらに、不正トラヒックを制限することで正規ユーザのトラヒックにまで影響を及ぼす問題についても考慮し、不正なトラヒックのみをブロックするシステムを実現している。

本論文は、以下のよう構成になっている。まず、2. 章で DDoS 攻撃とそれに関連する研究について述べる。3. 章では、DDoS 攻撃の被害を低減させる帯域制限モデルについて述べる。4. 章では、3. 章のモデルを実装した提案システム全体について説明する。5. 章ではテストベッドと提案システムのプロトタイプを利用した評価実験について述べる。そして、最後 6. 章ではまとめと今後の課題について述べる。

2. 関連研究

DDoS 攻撃は、その名の通り DoS (Denial of Service) 攻撃を分散化したものである。DoS 攻撃は特定のサーバあるいはネットワークに対し単一ホストから大量のパケットを送信する攻撃で、ネットワークの帯域を消費させ輻輳を生じさせたりサーバの処理量を増加させることでダウンさせたりする。DoS 攻撃に対しては、多くの効果的な対策がすでに提案されている。例えば IDS (Intrusion Detection System : 侵入検知システム) の登場により、リアルタイムのトラヒック監視や統計的分析による攻撃トラヒックの検出が可能となった。現状では、この IDS とファイアウォールを組み合わせることで攻撃トラヒックを防ぐ手法が広く利用されている[2][3]。[4] では、アクティブネットワーク技術を利用したネットワークノードにおいて

パケットをキャプチャし、トラヒックを監視する手法が提案されている。[5] では、パケットの異常性に応じて帯域を制限すると共にモバイルエージェントを配置させるモデルを提案している。

また、DDoS 攻撃に対しては、分散された攻撃者からの攻撃をネットワーク全体で防御する方法として、分散しているネットワークノードが不正なトラヒックを制限する方法が提案されている。[6] では、攻撃の追跡とフィルタリングを組み合わせたアプリケーション層のプロトコルとアーキテクチャを提案している。[7] では、輻輳を引き起こすトラヒックの帯域を制限するフレームワークを提案している。これらの研究では、ゲートウェイノードを過負荷にさせたり、バックボーンと対象サイト両方のネットワーク帯域を無駄に消費せたりすることを避ける機構を提供している。

しかし、これらの方には共通した問題が存在する。ほぼ全ての攻撃パケットは通常のパケットと区別がつきにくいため、輻輳しているトラヒックをフィルタリングすることにより、攻撃パケットだけでなく正規パケットまでが制限されてしまう、という問題である。[8] で述べられているように、インターネットバックボーンを流れる攻撃パケットのうち 90% 以上が正規パケットも利用している TCP プロトコルを利用している。すなわち、フィルタリングなどを行うことで、誤って正規パケットまでもブロックしてしまう可能性がある。このような、DDoS 攻撃そのものによる直接的な被害ばかりではなく、攻撃を回避する場合に生じる間接的な被害にも対処する必要がある。

3. Active Shaping Model

DDoS 攻撃による被害を低減するために、[9] において対策モデルを提案した。本章では、モデルの概要について述べる。

3.1 アクティブネットワーク技術

提案モデルを実現するため、ネットワークノード（ルータ）にアクティブネットワーク技術[10][11] を導入した。アクティブネットワークの基本的な概念は、1995 年頃に DARPA の研究プロジェクトにより提唱されている。従来のネットワークノードはパケットを蓄積しフォワーディングを行うだけである。しかし、アクティブネットワークのノードはパケットに対しアプリケーション層までの処理を行うことが可能となっており、パケット毎、あるいはアプリケーション毎に柔軟な処理を行うことが可能となる。また、ノード上のプログラムは柔軟に追加したりすることも可能となる。アクティブネットワークに関しては、様々な研究がなされている[12][13][14]。この中で提案モデルでは DDoS 攻撃対策として Active Node 方式を利用しておらず、ネットワークノードが自律的に行動すること、実行時に動的にプログラムモジュールをダウンロードすること、あるいは他のネットワークノードと協調して動作をすること、などの処理をユーザの要求に従って柔軟に提供することが可能となっている。

本モデルでは、AR (Active Router) と呼ばれる高機能ネットワークノードが、輻輳を検出した時にまず帯域制御を行い、各トラヒックの追跡を行った後、トラヒックを分析しさらに制

限を行うあるいは制限を解除する、などの機能を提供する。これらの機能はアクティブネットワーク技術を利用することで実現されており、悪意を持った攻撃トラヒックのみを帯域制限することが可能となっている。なお本論文において、traffic shaping とはトラヒックの帯域を制限する動作をあらわす。

3.2 基本概念

図 1 に提案モデルの概要を示す。AR はバックボーンネットワークのエッジに配置しており、バックボーンネットワークとユーザネットワークを、あるいは別のドメインに属するバックボーンネットワークとを接続している。また、ターゲット（攻撃を受けるホスト）に最も近い AR を特に root node と呼ぶことにする。

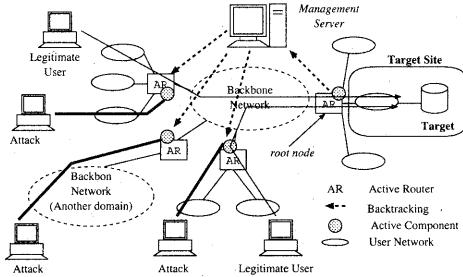


図 1 DDoS 攻撃対策モデルの概要

root node は Flooding 型の DDoS 攻撃（例えば UDP Flood 攻撃など [1]）に特化した攻撃検出機能を有しており、あらかじめユーザのポリシーにより決定された検出条件に従って、保護すべきサイトに流入してくるトラヒックを監視している。検出機能が何らかの不正なトラヒックパターンを検出すると、そのノードは出力キューを操作することでトラヒック量を削減する。この機能を Local Shaping と呼ぶ。その後、root node は管理サーバに対し攻撃情報を伝え、管理サーバは上流の AR にトラヒックシェーピングの為のプログラムを配布する。この機能を Backtracking と呼び、図 1 の破線で描かれている。この Local Shaping と Backtracking を組み合わせることで、トラヒックの輻輳を自律的に上流の AR で防ぐことが可能となり、ターゲットとなるサイト等が過負荷になったりターゲットサイトやバックボーンの帯域を無駄に消費することが避けられる。また、今回のシステムでは攻撃と疑わしいトラヒック（suspicious traffic）を正規トラヒック（legitimate traffic）、または攻撃トラヒック（malicious traffic）にユーザの要求に従って分類する機能も有している。

3.3 トラヒック制御モデル

AR におけるトラヒック制御モデルを図 2 に示す。このルータがアクティブネットワーク技術を利用していていることで、自律的な動作や他の AR との連携を可能としている。AR のそれぞれのインターフェースは 3 つのクラス（normal class, suspicious class, malicious class）と、流入してくるトラヒックを 3 つのクラスに選別するフィルタを所有している。normal class は正規と分類されたトラヒックを扱い、全く制限を行われない。通

常の状況では全てのトラヒックが normal class と判断される。root node が攻撃を検出すると、suspicious class として疑わしいトラヒックを扱う。suspicious class に分類されたトラヒックは、suspicious キューにより制限を行われる。suspicious キューにおける帯域制限値はユーザによって決定される。AR は多くのユーザにより共有されており、またユーザによって制限値は異なっているため、ユーザ毎に suspicious キューを作るようしている。malicious class は攻撃とみなされるトラヒックを扱い、malicious キューによって制限を行われる。malicious class はユーザが攻撃とみなしたトラヒックのみを扱うため、制限値は非常に小さいかあるいは 0 となる。

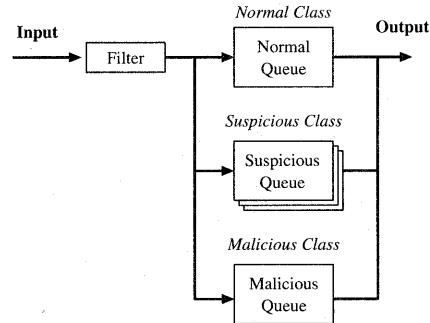


図 2 3 キューモデルによる Active Router でのトラヒック制御

4. システム設計

本章では、アクティブシェーピングモデルを実現するシステムのアーキテクチャ、トラヒック管理モデル、及び対策機能を含めたシステムの設計について述べる。

4.1 システムアーキテクチャ

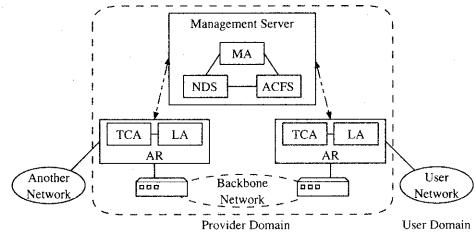


図 3 提案システムの概要

提案システムのアーキテクチャを図 3 に示す。提案システムは以下の 2 つのドメインに分けられる。

- プロバイダドメイン：DDoS 攻撃に対する対策機能を提供する実行主体の集合。

- ユーザドメイン：提供されるネットワークを利用するユーザの集合。

プロバイダドメインはバックボーンネットワーク・AR・管理サーバから構成されている。管理サーバはドメイン中の全ての AR を管理する。管理サーバは Management Agent (MA)，

Active Component Factory Server (ACFS), Naming Directory Server (NDS) という 3 つのコンポーネントから構成されている。また、AR は Local Agent (LA), Traffic Control Agent (TCA) という 2 つのコンポーネントから構成される。上記に述べた各コンポーネントの関係を図 4 に示す。これらのコンポーネントがお互い連携することで、対策機能が実現される。各コンポーネントの主な機能を表 1 に示す。また、各コンポーネントに記載されている主なメソッドについても以下節で述べる。

表 1 各コンポーネントの主な機能

コンポーネント	主な機能
MA	AR 間の全通信を仲介し、トラッピングの追跡及び分類の機能を中央で管理する。また、バックボーンネットワーク管理者のためのインターフェースなどを提供する。
ACFS	ユーザによりカスタマイズされた AC (Active Component) を登録するサーバ。ユーザの要求により AC を AR に配備し、管理する。
NDS	管理サーバ及び AR 上のコンポーネントへの参照を解決する。
LA	アクティブノード上のコンポーネントが外部にある管理サーバと通信するための、インターフェースを提供する。
TCA	トラッピングの監視、攻撃の検出、Backtracking の開始、帯域制御といった基本的な対策機能を提供する。
PAC	ユーザの要求に従い、suspicious トラッピングを normal トラッピングまたは malicious トラッピングへと分類する。

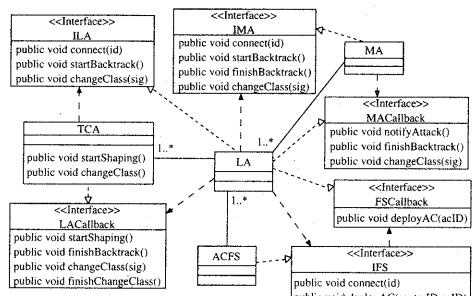


図 4 コンポーネント間の関係を示すクラス図

4.2 DDoS 攻撃対策機能

提案システムにおける各コンポーネントの動作の詳細について、対策のシナリオに沿って説明する。

4.2.1 DDoS 攻撃の検知と Local Shaping

攻撃が生じると、ターゲットが過負荷になりダウンすることやターゲットサイトのネットワーク帯域を無駄に消費すること、そしてターゲットネットワークにおける輻輳の発生を避けるために、root node の TCA は攻撃を検出し輻輳しているトラッピングをローカルで制限し始める。検出に関しては、監視トラッピングの統計的な分析により行っている。具体的には、ある閾値を超えるトラッピングが長時間連続して生じた場合に攻撃とみなしえ出を行う。各ルータに存在する TCA は Flooding 型の DDoS 攻撃を検出する機能を有しており、root node の TCA が攻撃を検出すると *startShaping()* メソッドを呼び出し Local Shaping を始める。TCA には属性・スループット閾値・検出間隔からなる検出条件があり、これらの条件はユーザによってあらかじめ定義される。表 2 には検出条件の例が示されている。それぞれの条件には宛先アドレス (Dest) 属性が必ず含まれている。

表 2 検出条件の例

属性	スループット閾値	検出間隔
{Dest=192.168.10.16/32, Protocol=TCP, Port=80}	600packet/s	10sec
{Dest=192.168.10.16/32, Protocol=UDP}	500packet/s	20sec
{Dest=192.168.10.0/24}	800packet/s	20sec

TCA は、ユーザが決定した suspicious class の制限値に出力帯域を制限する suspicious キューを作り、条件にマッチするトラッピングを normal class から suspicious class へと変更する。その時、signature と呼ばれる攻撃情報を記したものを作成する。signature には送信元アドレス、宛先アドレス、プロトコル、サービスポート番号などが記述されており、この情報を元に後述の Classification が行われる。

4.2.2 Backtracking

root node の TCA が Local Shaping を始めると同時に、攻撃の発生を管理サーバへと伝達する。この際 root node の LA が *startBacktrack()* メソッドを呼び出し、MA へとメッセージを伝達し、管理サーバが Backtracking 処理を開始する。同時に、root node の LA は管理サーバに対し上記の Local Shaping にて得られた signature を管理サーバへと通知する。root node が過負荷になることやネットワーク帯域を無駄に消費することを避けるため、管理サーバ上の ACFS は上流の AR へと制御プロセス (PAC) を配備する。この時、*notifyAttack()* メソッドにより以下の属性を持ったバケットトラックメッセージを送信する。

- (a) メッセージ ID
- (b) 攻撃 ID
- (c) suspicious signature の数
- (d) suspicious signature
- (e) suspicious トラッピングの制限値
- (f) 防御サイト ID
- (g) 終了条件

(a)(b)(c)(d) の値は root node の TCA によって Backtracking が開始される時に設定され、Backtracking が終了するまで保持される。(f)(g) の値はユーザの要求に従い TCA によって構成され、上記の値同様 backtracking が終了するまで保持される。(e) の値はユーザの要求に従いあらかじめ root node の TCA によって設定され、その後、MA がより小さい値へと設定しなおす。例えば S を (e) の初期値とし MA によって設定し

なおされた値を S' とすると、以下のような式が成り立つ；

$$S' = kS \quad (0 \leq k < 1)$$

ここで k は減衰係数であり、AR の総数とネットワーク管理者に従って決定されたトラヒック管理ポリシーに従って定義される。 k の値が大きくなると、バックボーンネットワークを流れる suspicious トラヒックの量が多くなり、帯域を無駄に消費することになる。このため、 k の値はなるべく小さい値を選択する。

4.2.3 Classification

Backtracking が終わると、各 AR は攻撃トラヒックを特定し正規トラヒックを保護するため、各 AR に配置された PAC がユーザのポリシーに従った高機能なトラヒックの分析とクラス分けを行う。既存の攻撃の場合、攻撃トラヒックの特性は既に知られているため、これらの条件にあてはまるトラヒックを特定しブロックする。また root node で得た管理サーバから受信した signature の情報を元に、各 AR はホストアドレス毎にトラヒックの分析を行う。より攻撃元に近いところで監視を行い対処するため、DDoS 攻撃を個々の DoS 攻撃とみなして対応することが可能となる。PAC はユーザサイトに適した条件・アルゴリズムを利用し、suspicious class のトラヒックを normal class か malicious class に分類する。これを Classification と呼ぶ。この時、normal class と malicious class は suspicious class より高い優先度を持っており、図 2 のフィルタによって選別される。管理サーバの ACFS が *deployAC()* メソッドを利用し AR へと配備させた PAC は、*changeClass()* メソッドを利用してトラヒックのクラスを変更する。このようにして、正規なトラヒックは帯域を回復させることができ、不正なトラヒックのみが制限される。

4.3 プロトタイプの実装

提案アーキテクチャの実用性とその効果を検証するために、PC/AT 互換機上で AR および管理サーバのプロトタイプを実装した。

プロトタイプの環境を表 3 に示す。各コンポーネント (MA, ACFS, NDS, LA, TCA) は Java 言語により実装されており、管理サーバと AR の間の通信は RMI により実現されている。また、トラヒック管理に必要なライブラリ関数は C++ 言語を利用し実装されている。カーネルにより提供されている CBQ (Class Base Queuing) を利用することで、トラヒックの判別と帯域制限が可能となっている。

表 3 プロトタイプの環境

ハードウェア仕様	
CPU	Pentium III 866Mhz
RAM	128MByte
ネットワーク	Ether NIC
ソフトウェア仕様	
OS	Linux kernel2.4
Programing language	各コンポーネント：Java (JDK1.3.1) 帯域制御機構：C++ (gcc ver.2.9.6)
帯域制御機能	CBQ (kernel による実装)

5. 評価

本章では仮想的に DDoS 攻撃を生じさせた時の、提案システムの効果について評価する。

図 5 に実験環境を示す。図が示すとおり、正規ユーザと不正ユーザが一つのネットワークに存在している (AR2 が管理するネットワーク)。また、コアネットワークは一つのマシン (CR : Core Router) から構成されており、管理サーバもそのマシンに配置してある。CR に関する全てのリンクは 100Mbps となっており、コアネットワークの端には 3 つの AR が配置されている。アクセシングを想定し、ターゲットとなるサイトと AR1 とのリンクのみ帯域が 10Mbps となっている。

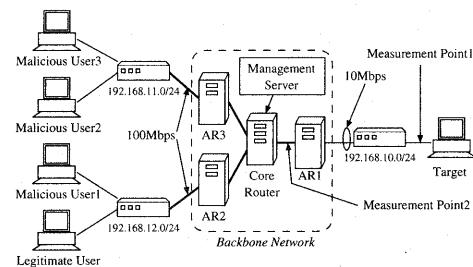


図 5 評価実験の環境

まず、PAC を配布し Classification 用のコンポーネントをダウンロードするのに必要な時間を測定した。PAC (17.5kbyte) を配布するのに必要な時間は平均 329msec であった。また、Classification 用のコンポーネント (8.8kbyte) をダウンロードするのに必要な時間は平均 259msec であった。通常 DDoS 攻撃は長時間行われるため、これらの結果から提案システムに基づいたプロトタイプシステムは実用的であると考えられる。

次に、以下のシナリオに従い疑似 DDoS 攻撃トラヒックを発生させ、その時のプロトタイプシステムの性能を評価した。

- ・ 時刻 0 で、正規トラヒック（正規ユーザによるトラヒック）が送信される。
- ・ 時刻 20 で、不正トラヒック（不正ユーザ 1 によるトラヒック）が送信される。
- ・ 時刻 40 で、残る不正トラヒック（不正ユーザ 2, 3 によるトラヒック）が送信される。

なお、攻撃トラヒックとしては典型的な DDoS 攻撃として知られる UDP Flood を利用した。実験に利用した入力フローについてのパラメータ値は表 4 に示されている。また、表 5 には実験における Classification のパラメータと帯域制限値が示されている。

表 4 実験に利用した入力フロー

分類	位置	プロトコル	入力量	入力パターン
攻撃フロー	User1 (AR2)	UDP	3Mbps	連続送信
	User2 (AR3)	UDP	5Mbps	連続送信
	User3 (AR3)	UDP	10Mbps	連続送信
正規フロー	User (AR2)	TCP	1.5Mbps	連続送信

表 5 トラヒック分類パラメータと帯域制限値

機能	パラメータ種別	パラメータ値
トラヒック分類	UDP 連続転送帯域	1Mbps
	UDP 連続転送間隔	15sec
帯域制限	初期 Suspicious 制限値	3Mbps
	調整 Suspicious 制限値	1Mbps
	Malicious 制限値	300kbps

そして、図 5 の 2 つの測定点におけるフロー毎のスループットの時間推移を測定した。この時に利用した DDoS 攻撃の検出条件は、「Attribute={Dest=192.168.10.0/24, Protocol=UDP}, Throughput threshold=7.5Mbps, Duration threshold=20sec」である。ボトルネックリンクに設定した測定点 1 の結果を図 6 に、バックボーンネットワークに設定した測定点 2 の結果を図 7 に示す。

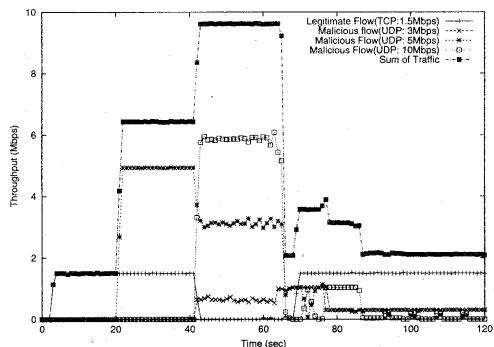


図 6 ボトルネックリンクにおけるスループットの時間推移

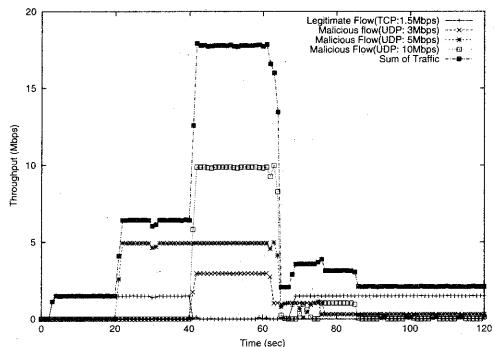


図 7 バックボーンリンクにおけるスループットの時間推移

図 6 の時刻 40 に疑似 DDoS 攻撃トラヒックによって輻輳が引き起こされ、そのため正規トラヒックのスループットが一時的に減少している。時刻 65 において全てのフローが一度減少しているが、これは root node において Local Shaping 处理が行われたためであり、短期的なものである。そして、Backtracking と Classification 处理が行われたおかげで、正規ユーザのトラヒックのみが回復している。

図 7 の時刻 40 では、3 つの攻撃フローによりボトルネックリンクである AR1-ターゲット間の帯域 10Mbps を超えたため、輻輳が発生し正規ユーザのトラヒックが落ち込んでいることがわかる。しかし時刻 60 において、AR1 が攻撃を検出しトラヒックの制御 (Local Shaping, backtracking) を行うことでき、コアネットワークを流れる不正トラヒックのみが制限された。その後、全ての不正トラヒックが Classification により帯域をさらに制限され、正規トラヒックだけが回復している。この結果により、提案システムでは不正なトラヒックのみを制限し、コアネットワークの帯域を無駄に消費させないことが示される。

以上の結果により、提案システムでは、バックボーンネットワークを流れる不正トラヒックを減少させ、正規ユーザのトラヒックまでも制限されてしまう問題を改善できることが証明された。

6. まとめと課題

本論文では DDoS 攻撃対策システムについて説明し、そのシステムデザインとプロトタイプの設計について述べた。我々のシステムでは、分散しているネットワークノードがお互いに連携し、自律的に Local Shaping, Backtracking, Classification などの機能を果たす。提案システムでは、Local Shaping と Backtracking によりトラヒックの輻輳を避けることができ、Classification によりユーザのポリシーに従って正規ユーザのトラヒックを改善することが可能となる。これらの機能を実現するためアクティブネットワーク技術を利用した。アクティブネットワーク技術を利用してネットワークノードが高機能になり、ユーザのポリシーに従った様々な機能を実現することが可能となる。また、プロトタイプを実装しこれを利用した評価実験を行うことでシステムの有効性を示した。

今後の課題としては以下のことが考えられる。

(1) 他の DDoS 攻撃への対応

本提案では、DDoS 攻撃の中でも Flooding 型攻撃に対してのプロトタイプを実装し評価実験を行った。しかし、実際にはこれは DDoS 攻撃の中の一つにすぎず、他の DDoS 攻撃に対しての効果を評価する必要があると思われる。

(2) 拡張性に関する評価

今回の評価実験では、テストベッドを構築しプロトタイプの評価を行った。しかし、テストベッドでは拡張性に関して制限されてしまうため、大規模なネットワーク環境においても評価を行う予定である。具体的にはシミュレータなどを利用し、大規模環境における効果や拡張性についての評価を行う。

文 献

- [1] Lee Garber "Denial-of-Service Attack Rip the Internet" IEEE Computer, vol.33(4), pp.12-17, Apr. 2000.
- [2] R.L.David "Evaluating intrusion detection systems: The 1998 darpa off-line intrusion detection evaluation" Proceedings of DARPA Information Survivability Conference and Exposition, 2000.
- [3] E.H.Spafford and D.Zamboni "Intrusion detection using autonomous agents" Computer Networks, Philadelphia, PA, pp:547-570, 2000.
- [4] T.Hasegawa, S.An, K.Nakao and F.Kubota "Programming

- Remote Traffic Monitoring Method Using Active Network Approach" Proceedings of IWAN2001, Philadelphia, PA, 2001.
- [5] E.Y.Chen "Aegis: an active-network-powered defence mechanism against ddos attacks" Proceedings of IWAN2001, Philadelphia, PA, 2001.
 - [6] D.Schnackenberg, K.Djahandari and D.Sterne "Infrastructure for intrusion detection and response" Proceedings of the DARPA Information Survivability Conference and Exposition(DISCEX), South Carolina, 2000.
 - [7] R.Mahajan, S.M.Bellovin, S.Floyd, J.Ioannidis, V.Paxson and S.Shenker "Controll high bandwidth aggregates in the network - draft" 2001.
 - [8] D.Moore, G.M.Voelker and S.Savage "Inferring Internet Denial-of-Service Activity" Proceedings of 10th USENIX Security Symposium, 2001.
 - [9] D.Kashiwa, E.Y.Chen and H.Fuji "Active Shaping: A Countermeasure against DDoS Attacks" Proceeding of ECUMN2002, Colmar, France, 2002.
 - [10] D.Tennenhouse, J.M.Smith, W.D.Sincoskie, D.J.Wetherall and G.J.Minden "A Survey of Active Network Research" IEEE Communications Magazine, Jan. 1997.
 - [11] K.Psounis "Active Networks: Applications, Security, Safety, and Architectures" 1st Quater 1999.
 - [12] B.Schwartz, W.Zhou and A.W.Jackson "Smart Packets for Active Networks" BBN Technology, 1998.
 - [13] D.Wetherall, J.V.Guttang and D.L.Tennenhouse "ANTS: A Toolkit for Building and Dynamically Deploying Network Protocols" Proceeding of IEEE OPENARCH '98, Apr. 1998.
 - [14] C.Gunter, S.Nettles and J.Smith "The SwitchWare Active Network Architecture" IEEE Network Magazine, vol.12, no.3, May/June 1998.