原著論文

リフレクター近傍リクエストフィルタリングによるDRDoS攻撃回避

矢板祐樹*・森口一郎**

要旨：DRDoS攻撃のうちDNSamp攻撃では，リフレクターと呼ばれる踏み台サーバによってデータ量が大幅に増幅され，標的サーバやそのネットワークに負荷をかける．本研究では，増幅発生前の不正クエリの段階で攻撃パケットを遮断することが可能な，ISPが実装することを想定したシステムを開発した．本システムでは，プロバイダネットワークと利用者の境界に同じ機能を持つ機器を設置する．これらの機器の中で，リフレクター近傍の機器ではDNSクエリの増加を検知し，フィルタリングを行う．しかし，これにより標的サーバからリフレクターへの正規クエリを遮断する可能性がある．そのため，標的サーバ近傍の機器で正規通信に対してポート番号の変換を行うことにより，リフレクター近傍の機器での攻撃通信と正規通信の識別を可能とした．これにより，通信増加が正規クエリであった場合には通信を阻害することなく，DNSamp攻撃を約1.2秒程度で回避することができた．

キーワード：DRDoS攻撃，DNSamp攻撃，ISP，オープンDNSリゾルバ

A DRDoS Attack Avoidance Method Based on the Request Filtering of a Reflector Neighborhood

Yuki YAITA*and Ichirou MORIGUCHI**

Abstract: The DNSamp attack, a DRDoS attack type, amplifies the amount of data via steppingstone servers named “reflectors” and attacks target servers and their networks. In this study, we developed an attack prevention system that blocks malicious queries before amplification, which should be implemented in ISPs. This system consists of several machines with the same function, which are located at borders between ISP and user network. Among these servers, the machines located at the reflector neighborhoods detect and filter increasing DNS queries. However, these machines may block harmless queries to reflectors from victim servers through this filtering action. To avoid this, machines at victim neighborhoods convert the destination port number to distinguish between malicious traffic and harmless traffic. Therefore, we could avoid DNSamp attack in approximately 1.2 s, without blocking harmless packets even if they are transmitted rapidly.

Keywords: DRDoS attack, DNSamp attack, ISP, open DNS resolver

* 東京情報大学　総合情報学部　総合情報学科（2022年３月卒業予定）
Faculty of Informatics, Tokyo University of Information Sciences
** 東京情報大学　総合情報学部　総合情報学科
Faculty of Informatics, Tokyo University of Information Sciences

2021年10月12日受付
2022年１月26日受理

1．はじめに

近年，分散した多数のPCを用いて同時に大量のパケットを送信しサーバのネットワークリソースを枯渇させサービスを停止させるDDoS攻撃（Distributed Denial of Service Attack）がインターネット上で多く発生している[1]．このような現状の中で，すべてのDDoS攻撃のうちおよそ7割を占めるのがDRDoS攻撃（Distributed Reflection Denial of Service Attack）である[2]．これは攻撃者が送信元IPアドレスを詐称したリクエストパケットをリフレクターと呼ばれる別のサーバに対して大量に送信し，その返答を攻撃対象に集中させることでネットワークリソースを枯渇させるDDoS攻撃である．その中でもDNS（Domain Name System）サーバをリフレクターとして使用し，データ量の増幅を行うものをDNSamp（amplification）攻撃と呼ぶ．

DNSamp攻撃のうち，設定に不備のあるキャッシュDNSサーバを使用する手法の対策法はDNSサーバ管理者が適切な設定を行うことであり，これを推進する取り組みにより設定に不備のあるDNSサーバは年々減少傾向にある[3]．しかし，全てのDNSサーバ管理者が適切な設定を行うことは事実上不可能であるため，根本的な対策法とは言えない．別の対策手法として，IPS（Intrusion Prevention System）やファイアウォールを使用し被害者側で通信制御を行う方法がある．しかし，大規模な攻撃によりこれらの通信制御機器に負荷がかかり，停止した場合に攻撃が成功してしまうという問題がある．このような理由から，現在DNSamp攻撃を防止するための有効な手法は存在していない．

この問題に対する研究として，野口，後藤らは，送信されるクエリパケットの特徴分析を行い，リフレクターにクエリが到達する前に特徴に基づき攻撃検知を行い，IPアドレスの詐称を判定してフィルタリングを行った[4]．しかし，この手法はOpenFlowネットワークを使用し，IPアドレス詐称判定等を行っているためOpenFlowネットワーク以外の適用が困難である．別の特徴をもつ研究として，桂井，中村，高橋らは，複数のルータで分散してフィルタリングを行う手法[5]や，さらに高い耐久性がある上位のルータでフィルタリング[6]を行った．しかし，いずれも情報量が増加したレスポンスを対象に遮断を行うため，さらに大規模な攻撃があった場合に対処しきれない可能性がある．また，玄，村山らは，ISP間の境界のファイアウォールにてIPフラグメントが発生するような不正巨大DNSレスポンスの遮断を行う手法を考案している[7][8]．しかし，桂井，中村，高橋らの研究と同様の理由で，さらに大規模な攻撃があった場合に対処しきれない可能性がある．

本システムはISPが実装することを前提として，利用者とプロバイダネットワークの境界に独自機器を設置し，リフレクター近傍でDNSクエリの増加を検知した場合に，リフレクターに対する53番ポートへの通信を遮断する．さらに，被害サーバ近傍独自機器に被害サーバとリフレクターのIPアドレス及び任意のダイナミックポート番号を通知し，被害サーバからリフレクターへの正規クエリには被害サーバ近傍独自機器で宛先ポート番号の変換を行い，リフレクター近傍独自機器で逆宛先ポート番号変換を行う．これにより，正規クエリに対してはパケット破棄が行われないため，正規通信に影響を与えずに攻撃回避が可能である．なお，本手法では増幅前の不正クエリの段階で遮断を行うため，プロバイダネットワークの負荷軽減が可能であり，さらに，フラグメントIPパケットへの対応を行う必要がない．

本研究ではこのシステムによりDRDoS攻撃回避が可能であることを明らかにした．また，攻撃回避のトリガーに通信増加を使用しているため，攻撃回避を開始するまでの間は攻撃トラフィックが約1.2秒間被害サーバに到達することがわかった．

2．DNSamp攻撃の手法

DNSamp攻撃では権威DNSサーバまたは設定に不備のあるキャッシュDNSサーバがリフレクターとして使用される[9]．このうち，権威DNSサーバでの対策は同一送信先への同一名に対する応答を制限するRRL（Response Rate Limiting）の導入が進められている．これに対し，設定に不備のあるキャッシュDNSサーバでの対策は，その設定を適切なものにすることである．

キャッシュDNSサーバの利用は通常の場合，内部ネットワークからの問い合わせに限られる．即ち，キャッシュDNSサーバでは内部ネットワークからの問い合わせのみを許可するようにアクセス制限を

行う必要がある．しかしながら，DNSサーバの設定やネットワークの構成に不備があると，キャッシュDNSサーバが外部からの問い合わせに対して応答してしまう場合があり，このような状態のDNSサーバをオープンDNSリゾルバと呼ぶ．即ち，オープンDNSリゾルバの問題点は，内部ネットワークに関係ないドメインに対して外部からの名前解決を行ってしまうことである．

DNSamp攻撃ではこの問題点を悪用する場合があり，攻撃者はオープンDNSリゾルバに対して，巨大なTXTレコードを返答するドメインに関するクエリを送信する．さらに，このDNSクエリの送信元IPアドレスを被害サーバのIPアドレスに詐称することにより，攻撃者は送信したクエリよりも数十倍大きいデータ量のDNSレスポンスを被害サーバに集中させ[10]，ネットワークリソースを枯渇させる（図１）．このように，攻撃者はオープンDNSリゾルバが外部からの再帰的問合せを受け付けていることを悪用し，攻撃パケットを増幅及び反射させることで，攻撃者の特定から逃れつつ，攻撃を大規模化させることを目的としている．

なお，オープンDNSリゾルバ状態になっているDNSサーバは，DNS機能を持つブロードバンドルータ等の一般利用者向け機器の不具合によるものが多く，このような機器ではアクセスログを残さないことが多い[11]．そのため，攻撃者を特定することが困難であるという特徴があり，悪質な攻撃である．

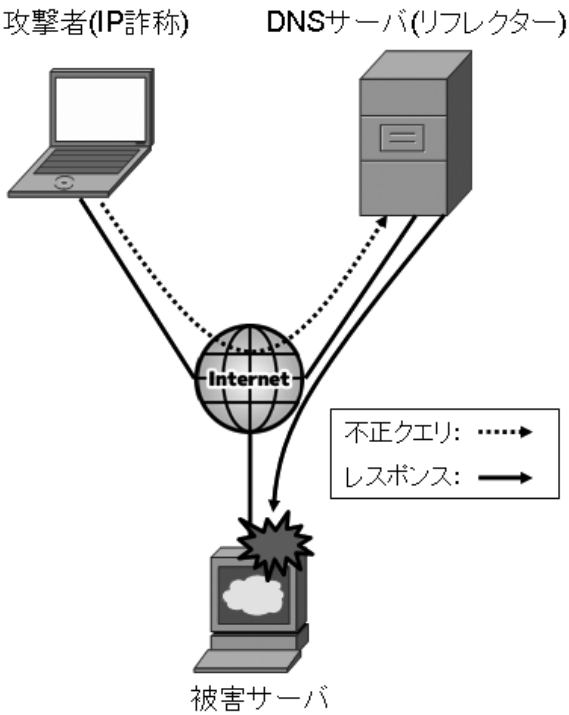


図１　DNSamp攻撃の仕組み

３．本システムの概要

DNSamp攻撃のようなDRDoS攻撃を回避する場合に考えられるのが，IPSやファイアウォール等を使用した被害者側での境界型防御である．しかし，DRDoS攻撃では攻撃対象にアクセスを集中させ，ネットワークリソースを枯渇させるという特徴があるため，大規模攻撃ではこれらの通信制御機器がダウンしてしまう可能性がある．また，ISPで送信元IPアドレス詐称パケットを遮断する技術であるイングレスフィルタリングでは，ISP内部の利用者のIPアドレス詐称は識別可能だが，外部から流入するIPアドレス詐称パケットの識別は困難な場合がある[12]．これらの場合に攻撃が成功してしまうため，本システムではリフレクターによってデータ量が増幅するより前にポート番号変換によって不正クエリを判定し，攻撃を遮断することを目標としている．

３．１　本システムの攻撃回避

本システムではISPが実装することを前提とし，プロバイダネットワークと利用者の境界に独自機器を設置する（図２）．これらの独自機器は全て同じ機能を備えているが，攻撃が発生した場合には，攻撃に対するそれぞれの独自機器の位置によって異なる働きをする．

図２はDNSamp攻撃に対する本システムの動作を表した構成図であり，攻撃者はIPアドレスを被害サーバのものに詐称したDNSクエリをリフレクターに送信している．これによりリフレクターは被害サーバに返答を行うため，攻撃が成立する．このような攻撃に対して本システムでは，リフレクター近傍の独自機器Bのインターネット側インターフェースで，特定の送信元IPアドレスからの通信増加を検知する．検知後，該当送信元IPアドレスからリフレクターへのDNSクエリの遮断を行うことで攻撃回避を行うが，通信増加が正規通信であった場合にはそのトラフィックを遮断するべきではない．そのため，被害サーバ近傍の独自機器Cで正規クエリの宛先ポート番号を任意のダイナミックポート番号に変換することで独自機器Bで正規クエリと不正ク

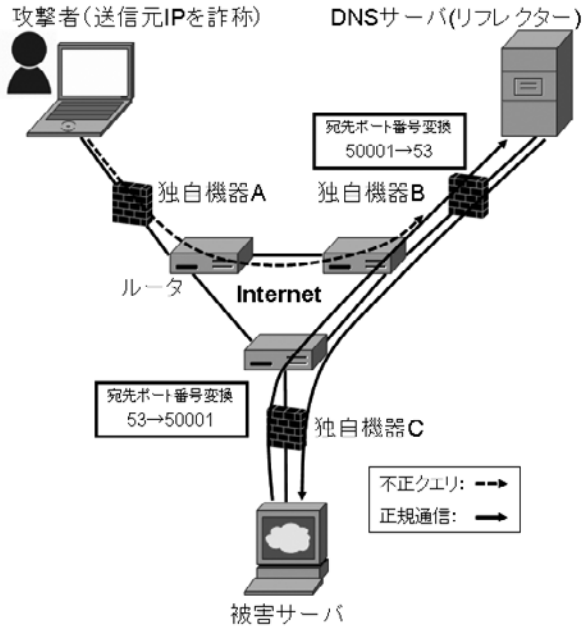


図2　本システムの構成

独自機器Aはポート番号変換を行わないため，攻撃トラフィックのみ，宛先ポート番号が53番のまま到達する

エリとの区別を行い，その後に宛先ポート番号の逆変換を行う．このような処理を行うためにリフレクター近傍独自機器は通信増加検知後，増加したパケットの詐称された送信元IPアドレスをもとに被害サーバ近傍の独自機器Cを探索する．この探索が成功した場合には，独自機器Bから独自機器Cに対して攻撃に使用されているリフレクターのIPアドレスと被害サーバのIPアドレス及び任意のダイナミックポート番号を通知する．なお，ここで通知を行っている任意のダイナミックポート番号は独自機器Bがランダムに生成したものである．

このように，本システムの攻撃回避処理では，トラフィック監視による通信増加検知と，ポート番号変換による不正クエリと正規クエリの識別を行っている．このうち，通信増加検知では複雑なルールを使用していないため，正規クエリが増加した場合にも検知が行われる場合がある．しかし，正規通信ではポート番号の変換が適切に行われるため，正規クエリの誤遮断が発生することはない．

3.2　本システムの独自機器

前節で述べた攻撃回避の特徴から，本システムでは攻撃回避に関する全ての処理を独自機器で行っている．そのため，独自機器の処理は多岐にわたる．

独自機器に求められる要件は，大きく分けて5つ存在する．1つ目の要件は，ルーティングが可能であることである．これは，独自機器間でIPアドレスやポート番号の送受信を行うため，独自機器自体がIPアドレスを持つ必要があるためである．次に，2つ目の要件は，ポート番号の書き換えが可能であることである．これは，前節で述べた攻撃クエリと正規通信の区別を行うためのものであり，IPアドレスは変換せずにポート番号のみを変換する必要がある．さらに，3つ目の要件は，IPアドレス別の通信監視が可能であることである．これは，特定の送信元IPアドレスからの通信増加を検知するために必要なものである．次に，4つ目の要件は被害サーバ近傍独自機器を探索することが可能であることである．これによって，正規通信に対してのみ宛先ポート番号変換を可能にする．最後に，5つ目の要件が，ポート番号及びIPアドレスの通知機能である．

本システムの独自機器ではこれらの機能が全て適切に動作するように実装する必要があるため，既存ツールや自作プログラムを組み合わせて構築した．なお，独自機器のOSはVineLinux6.3を使用したため，ルーティング機能に関してはLinux上で動作するソフトウェアルータであるQuaggaを使用した．

3.3　独自機器のポート番号変換

独自機器では攻撃クエリと正規通信を判別するために，ポート番号の変換を行う必要があるが，raw socketやlibpcapでは通過するパケットのコピーを行うため，一度該当パケットを遮断した上で，ポート番号を書き換えたパケットを再度送信することになる．この場合，パケット送信の際に送信元IPアドレスの詐称を行うことになってしまい問題となることから，本システムではポート番号変換にiptablesを使用した．

iptablesの機能の1つにNAPT（Network Address Port Translation）がある．NAPTではIPアドレス及びポート番号の書き換えを行うことから，独自機器のポート番号変換に類似した処理を行っていると言えるため，iptablesのNAPT機能を応用し，ポート番号の変換を行った．これにより，ローカルプロセスでの処理をせずにポート番号変換処理を行うことを実現した．

例として，図3のような攻撃を回避する場合に適用すべきルールを示す．まず，表1と表2を参考に

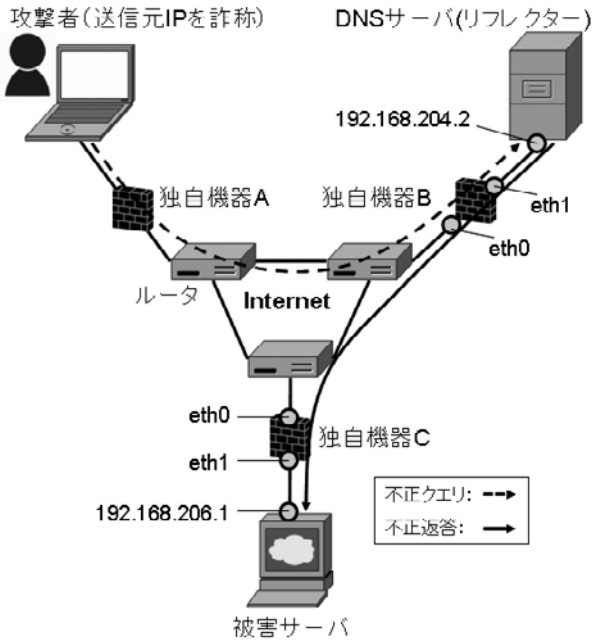


図３　iptablesのルール説明時に想定する攻撃

表１　iptablesのルール適用に使用したオプション

オプション	説明
-t	マッチングテーブル指定
-A	ルール追加チェイン指定
-D	ルール削除チェイン指定
-i	受信インターフェース指定
-s	送信元IP指定
-d	宛先IP指定
-p	プロトコル指定
--dport	宛先ポート番号指定
-j	ターゲット指定
--to-destination	NAPT変換後の宛先情報指定

表２　iptablesのルール適用に使用したターゲット

ターゲット	説明
ACCEPT	パケット通過を許可
DROP	パケット破棄
DNAT	宛先情報変更

独自機器Cの利用者側インターフェースに対して図４に示すルールを適用する．このルールでは，NAPTによる変換前の送信元IPアドレスと変換後の送信元IPアドレスを同じ値とすることで，ポート番号の変換のみを行っている．さらに，表１と

```
# iptables -t nat -A PREROUTING -i eth1 -d ¥
  192.168.204.2 -s 192.168.206.1 -p udp ¥
  --dport 53 -j DNAT --to-destination ¥
  192.168.204.2:50001
```

図４　独自機器Cの利用者側インターフェースでのiptablesのルール

```
# iptables -t nat -A PREROUTING -i eth0 -d ¥
  192.168.204.2 -s 192.168.206.1 -p udp ¥
   --dport 50001 -j DNAT --to-destination ¥
   192.168.204.2:53
# iptables -t raw -A PREROUTING -i eth0 -d ¥
  192.168.204.2 -s 192.168.206.1 -p udp --dport 53 ¥
  -j DROP
```

図５　独自機器Bのインターネット側インターフェースでのiptablesのルール

表２を参考に変換するポート番号を反転させたルールと不正クエリを破棄するためのルールを独自機器Bのインターネット側インターフェースに対しても適用する（図５）．このように，NAPT機能を応用することで宛先ポート番号の変換を行ったが，iptablesではNAPTを使用する上で通過パケットのトラッキングを行っており，その上限数が決まっている．トラッキング上限数を超過した場合にはパケットが破棄されるため，本システムでは上限超過が発生しないように設定を行った．また，独自機器Aでは宛先ポート番号が変換されないため，不正クエリとして被害サーバIPアドレスからのリフレクターに対する53番ポートへの通信を破棄するように設定した．

３．４　攻撃回避プログラム

前節で述べたiptablesによるポート番号変換は独自機器で通信を監視し，通信増加があった場合に適用されるものである．そのため，通信増加を検知し，攻撃回避を行うプログラムを作成する必要がある．

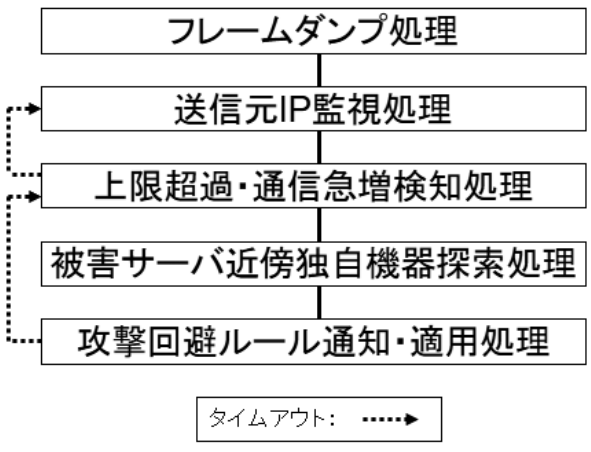
本研究では，攻撃回避プログラムにJava及びShell Scriptを使用した．その上で，攻撃回避ではリフレクター近傍と被害サーバ近傍の２点で処理を行うため，プログラムも２つに分けて作成した．このうち，被害サーバ近傍用プログラムは，必要な情報を受け取り，ルールを適用するという単純なものであるのに対し，リフレクター近傍用プログラムではトラ

フィックの監視や通信増加検知等，必要な処理が多い．そのため，リフレクター近傍用プログラムを中心に説明を行う．また，各独自機器ではこれら２つのプログラムが同時に実行されている状態となる．

リフレクター近傍用攻撃回避プログラムの処理は大きく５つに分けられる（図６）．１つ目の処理では，独自機器を通過するパケットを監視するためにフレームのダンプを行う．さらに，２つ目の処理ではダンプしたフレームの送信元IPアドレスを監視し，新たなIPアドレスからの通信であれば個別の監視スレッドを起動する．次に，３つ目の処理では個別の送信元IPの通信増加を検知する．通信増加を検知した場合，４つ目の処理で被害サーバ近傍の独自機器の探索を行う．最後に，５つ目の処理ではリフレクター近傍独自機器が，被害サーバ近傍独自機器に対してIPアドレス及びポート番号の通知を行う．

まず，フレームダンプ処理では各独自機器がインターネット側から受信する53番ポートへのアクセスを全て監視するためにlibpcapを使用した．なお，本プログラムではJavaを使用しているため，Javaでlibpcapを使用するためのラッパーであるJnetPcapライブラリを使用した．

次に，DRDoS攻撃を行うための不正クエリは送信元IPアドレスが被害サーバのIPアドレスに詐称されているため，送信元IPアドレスごとの監視を行うように処理をスレッド化し，監視中のアドレスを配列で管理した．その上で，独自機器のメモリ領域の限界を考慮し，送信元IPアドレスの監視上限



※該当パケットが1分間到達しない場合タイムアウト

図６　攻撃回避プログラムの主な処理の流れ

数は10万とし，上限を超過した場合には最も監視開始時間が古いアドレスを削除することで，上限超過した場合に配列の要素数を超過することなく処理を行うようにした．

さらに，攻撃回避のトリガーとなる通信増加検知処理では２つの基準を設けた．DRDoS攻撃の特徴は大量の不正クエリを送信することであるため，これを検知するために独自機器のインターネット側インターフェースが受信する特定IPアドレスからのDNSクエリに上限通信速度を設け，その速度を超過した場合に通信増加とみなした．加えて，DRDoS攻撃では攻撃開始後に不正クエリが急増するという特徴があるため，上限通信速度を超えないトラフィック急増の検知を行った．このように，検知基準は２つ設けているが，どちらで検知した場合でも次に行われるのは被害サーバ近傍独自機器探索処理である．しかし，通信速度上限を超過するような著しく大きいトラフィックが到達している場合，これによる輻輳や帯域超過により独自機器探索が行えない可能性があるため，通信速度上限超過を検知した場合にのみ，一時的に該当送信元IPアドレスからの53番ポートへの通信を遮断する．そのため，２つの検知基準の処理順について，先に通信速度上限超過検知を行い，その後，通信急増検知を行うことにより，通信速度上限を超過しているにも関わらず，通信急増を検知してしまうことをなくした．なお，独自機器探索に伴う一時的なトラフィックの遮断は独自機器間のIPアドレス及びポート番号共有後に終了し，前節で述べた宛先IPアドレスを含めたルールに置き換わる．また，通信増加検知処理中に１分間，該当送信元IPアドレスからの53ポートへの通信がなかった場合，その送信元IPアドレスの監視は終了する．

上記のように，リフレクター近傍独自機器で通信増加を検知した後，リフレクターと被害サーバのIPアドレスと任意のダイナミックポート番号を被害サーバ近傍独自機器に通知する．しかし，通信増加検知の時点では，被害サーバ近傍独自機器のIPアドレスはわからないため，リフレクター近傍独自機器は増加したパケットの詐称された送信元IPアドレスを使用し，被害サーバ近傍独自機器を探索する．この探索には，ターゲットマシンまでの経路をルータレベルで表示可能なツールであるtraceroute

の技術を応用した．具体的には，あらかじめ全独自機器のインターネット側のIPアドレスを各独自機器に登録しておき，被害サーバへの経路中に該当IPアドレスが存在すれば探索成功とする．この際，ICMPではポート番号を持たないため，通常のプロセス管理が不可能であることから，ICMPパケットのデータ部に存在するエラーを起こしたパケットの情報を使用し，プロセス管理を行った．もし，経路中に独自機器が存在しなかった場合は，攻撃回避システム未対応IPアドレスとして被害サーバのIPアドレスを登録し，それ以降は該当IPアドレスについての監視を行わないようにした．

最後に，攻撃回避ルール通知及び適用処理ではポート番号の変換を行うために，リフレクター近傍独自機器から被害サーバ近傍独自機器に対して，リフレクターと被害サーバのIPアドレス及び任意のダイナミックポート番号の通知を行う．前述の通信増加検知では送信元IPアドレスのみを監視していたが，この段階では該当送信元IPアドレスからの通信が増加していることはわかっているため，宛先IPアドレスの監視を行い，同じ送信元IPアドレスから新たな宛先IPアドレスを持つDNSクエリが到達した場合にルールの通知を行う．なお，この通知が適切に行えない場合は攻撃回避が困難になることから，通知の信頼性を担保するためにTCPを使用した．そのため，被害サーバ近傍用プログラムは新規ルール設定リクエストやルール削除リクエストに関するメッセージを受け取るために，TCPの19990番ポートを常にオープン状態にしておき，その情報をもとにiptablesのルール適用を行う．

３．５　独自機器のアーキテクチャ

前節までに述べた攻撃回避システムの各処理はiptablesやlibpcap等の様々なツールを使用することで実現している．また，独自機器はTCP/IPの複数の層にまたがり機能を提供している（表３）．そのため，独自機器がどの段階でどのような処理を行っているかを示す（図７）．まず，独自機器がNIC（Network Interface Card）でフレームを受信した直後に行われる処理が，libpcapによるフレームのダンプである．これにより，パケットのヘッダ情報を確認することが可能であることから，通過するフレームが持つMACアドレスやIPアドレス，ポート番号の監視を可能にしている．次に，行われるのがiptables

表３　TCP/IPのプロトコルスタックと独自機器の主な機能

層	ツール	主な機能
アプリケーション	−	攻撃回避ルール通知
トランスポート	iptables	ポート番号変換
インターネット	Quagga	ルーティング
ネットワークインターフェース	libpcap	トラフィック監視

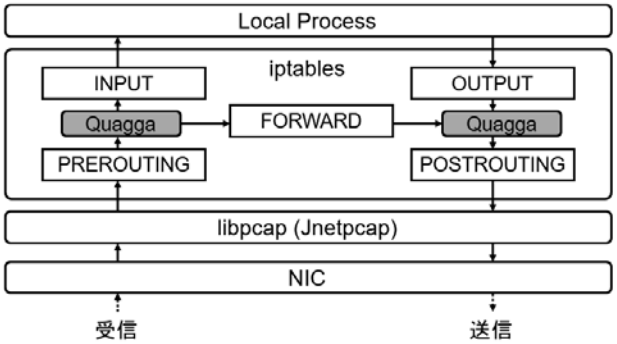


図７　独自機器のアーキテクチャ

によるポート番号の変換やトラフィックの遮断である．また，iptablesは転送を行うパケットか，ローカルプロセスに対してのパケットかによって適用するチェインを分けているため，各チェインによる処理が行われる途中でルーティングの処理をQuaggaが行っている．さらに，ローカルプロセスでは独自機器間の通信により，攻撃回避に使用するIPアドレスやポート番号の通知を行う機能を提供している．

攻撃回避に使用するiptablesのルールはJavaのRunTimeクラスを使用し，シェルコマンドを実行することでルール適用を行っている．また，独自機器はiptablesによる攻撃パケットの遮断よりも先に，libpcapにより全てのトラフィックの監視を行うことができるため，攻撃回避により，攻撃パケットの遮断が行われている最中にも，攻撃トラフィックを監視し，攻撃の終了を検知することが可能となっている（図７）．

４．本システムの評価実験

本研究では，本システムの有効性を確認するために実験ネットワーク環境を構築し，DNSamp攻撃の再現を行った．本章では，これらの作成や構築の方法と実験条件等を述べる．

また，本研究では仮想環境であるVMWare Player

上で環境構築を行ったため，実験ネットワーク環境についてもVMwareの仮想ネットワークであるVMnetを使用した．このことから，ホストマシンのハードウェアの限界を考慮し，通信帯域を10Mbpsに制限した．なお，ホストマシンのOSはWindows8.1 Enterpriseであり，仮想環境上のOSは，ルータやDNSサーバ，攻撃者マシン等の全ての機器でVineLinux6.3を使用した．そのため，ルータに関しては独自機器と同様にQuaggaを使用して構築した．

4．1　DNSサーバ

本研究では，DNSamp攻撃のうち攻撃者がオープンDNSリゾルバに対して送信元IPアドレスを詐称したクエリを送信し，その返答を標的サーバに集中させる手法を再現する．そのため，攻撃者が使用するオープンDNSリゾルバや巨大な返答を行うDNSサーバの構築を行う．

```
options {
        directory "/var/named";
        auth-nxdomain no;
        allow-query{ any; };
        forward first;
        forwarders { 192.168.204.1; };
};
```

図8　オープンDNSリゾルバのnamed.confのoptionsステートメント

```
acl authoDNSnet{
        192.168.204.0/24;
        127.0.0.0/8;
};
options {
        directory "/var/named";
        auth-nxdomain no;
        recursion yes;
        allow-query{ any; };
        allow-recursion { authoDNSnet; };
};
zone "authoritative.example.ne.jp" IN {
        type master;
        file "authoritative.zone";
};
```

図9　権威DNSサーバのnamed.conf

オープンDNSリゾルバの構築にはVineLinuxにデフォルトでインストールされているBIND 9.9.7を使用し，named.confのoptionsステートメントを図8のように記述し起動した．ここで，allow-queryオプションでは引数をanyとしているため，どこからのクエリでも受け取ることが可能である．次に，forwardオプションではfirstを指定することで，フォワーダから回答が得られない場合に再帰的に問い合わせを行う設定となっている．さらに，forwardersオプションでは問い合わせに対する回答を持っていない場合にオープンDNSリゾルバが問い合わせを行うDNSサーバのIPアドレスを指定している．

さらに，実際のDNSamp攻撃では，攻撃者が巨大返答を行うDNSサーバに対してオープンDNSリゾルバを介して問い合わせを行うことにより増幅を行う．そのため，オープンDNSリゾルバが問い合わせるDNSサーバを構築する必要がある．

今回は実験環境の簡略化のため，権威DNSサーバを1つ構築した．そこに巨大な返答を行う設定を行い，オープンDNSリゾルバのフォワーダとすることでDNSamp攻撃を再現した．なお，権威DNSサーバもオープンDNSリゾルバ同様，BIND 9.9.7を使用し，named.confを図9のように記述した．ここで，acl（access control list）ステートメント及びoptionsステートメントのallow-recursionオプションにより，再帰的な名前解決を許可するIPアドレスを指定しているため，オープンDNSリゾルバとなることはない．また，zoneステートメントにより示しているzoneファイルには巨大なレコードを持たせている．しかし，通常のAレコードでは巨大なレコードを作成することができないため，メールの送信ドメイン認証技術であるSPFに使用されるレコードを持たせている．

4．2　SPF（Sender Policy Framework）

DNSamp攻撃に悪用されることが多い巨大返答を可能にする技術としてSPFがある．SPFは送信者の送信元メールアドレスをDNSによって認証する仕組みである．

メールを送受信するにあたって使用されるメールアドレスはヘッダfrom及びtoとエンベロープfrom及びtoであり，エンベロープfrom及びtoに関しては，サーバ間のやり取りで使用されるため基本的に偽装することはできないが，主にユーザ間で利用さ

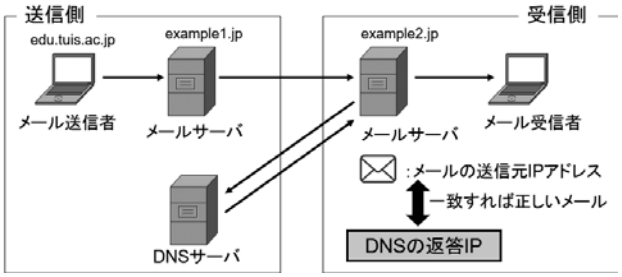


図10　SPFの仕組み

れるヘッダfromに関しては偽装が可能である．このような特徴から，なりすましメールではヘッダfromとエンベロープfromが異なることが考えられる．実際に旧来のなりすましメール検知はこれにより行われていたが，近年はクラウドサービスの普及により正規のメールであってもヘッダfromとエンベロープfromが異なるケースが多い．そこで使用されるなりすまし誤認対策技術がSPFである．

SPFではメールサーバのメール受信時にエンベロープfromのドメインのSPFレコードをDNSサーバに問い合わせる（図10）．問い合わせを受けたDNSサーバはメールサーバのIPアドレスを返し，実際に受信したメールの送信元IPアドレスと比較し，正しければ正規のメールとして受信者に送信する．一方，もしIPアドレスが一致しなかった場合は該当メールを破棄する．このようにSPFではメールを送信する可能性のあるメールサーバのIPアドレスを事前にDNSサーバに登録しておくことで，なりすましの誤認を防止している．この際に特徴的なことは，メールサーバが複数存在する場合に１つのレコードに対して複数のIPアドレスを登録する必要があることである．そのため，SPFはTXTレコードを使用した巨大な返答を行うことが可能となっており，このことから攻撃者がDNSamp攻撃に悪用するケースが多くなっている．

４．３　攻撃プログラム

前節で構築したリフレクターに対して大量の不正クエリを送信することが可能な攻撃プログラムをraw socketを使用しJavaで作成した．

生成する不正クエリパケットはIPヘッダ，UDPヘッダ，データ部からなり，IPヘッダでは送信元IPアドレスの詐称を行う．また，データ部ではDNSのフォーマットに従いデータを生成するが，その中のAdditionalセクションでは攻撃倍率の向上を図るためにOPT疑似レコードを付加し，EDNS（Extenstion mechanisms for DNS）を使用可能にした．本攻撃プログラムでは，これらのフォーマットに従いIPペーロード長が74byteのクエリパケットを生成した．

また，DNSamp攻撃ではリフレクターによってデータ量を増幅させるため，クエリの段階では帯域制限値である10Mbpsを超えぬようにした．具体的には，実験環境で攻撃プログラムを使用する際の一攻撃者あたりの不正クエリ送信速度を約4Mbpsとし，その場合のリフレクター近傍での不正クエリ到達間隔は約227.47マイクロ秒となった．

４．４　独自機器の通信増加検知

３章４節で述べた通信増加検知は，通信速度超過検知と通信急増検知に分けられる．このうち，通信速度上限超過の上限速度は実験ネットワーク環境を考慮し，2Mbpsとした．この際の通信速度計測をJavaで行う場合に，スレッドを使用して一定時間おきに処理を行うことが可能だが，これでは処理が煩雑化するため，該当フレームを2Mbit受信するために経過した時間が１秒を下回った場合に通信上限超過として検知を行った．また，通信急増検知では，該当フレームを2Mbit受信するために経過した時間を時系列順に配列で管理し，ある点を境に該当フレームの受信速度が５倍以上になった場合に通信急増として検知を行った．

４．５　実験条件

本システムでは正規通信を阻害せずにDRDoS攻撃を回避することを目標としている．そのため，攻撃回避の有効性と攻撃回避による正規通信の阻害が発生しないかを確認するための実験を行った．

このうち，攻撃回避の有効性を確認する実験では，攻撃開始時刻を設け，何秒間で攻撃回避が可能であるかを被害サーバでの攻撃トラフィック受信速度を元に確認した．なお，この実験は使用するリフレクターの台数や攻撃開始のタイミングを変更し，２種類の条件で行った．

また，正規通信の阻害が発生しないかを確認するための実験では，本システムによる攻撃回避中に被害サーバからリフレクターに対してdigコマンドによる問い合わせを一定時間行い，名前解決の成功率を算出し，攻撃回避が正規通信に与える影響を確認した．この実験でのdigコマンドによる問い合わせ

は連続して行われるが，名前解決の成功またはタイムアウト後に次の問い合わせを行うため，digコマンドの発生回数や発生間隔はネットワークの余剰帯域幅やDNSサーバにかかる負荷等に依存する．

5．実験結果

本システムによりDRDoS攻撃を回避することが可能かを確認するために，前章までに作成，及び構築を行ったルータやDNSサーバ，独自機器，攻撃プログラムを使用して実験を行なった．

5．1　分散攻撃に対する有効性

実際のDRDoS攻撃では，複数の攻撃者及び，複数のリフレクターを使用し，被害サーバに負荷をかけ，ネットワークリソースを枯渇させる．そのため，攻撃者とリフレクターを共に2台用意し，攻撃実験を行った（図11）．なお，各攻撃者は別々のリフレクターに対して不正クエリを送信する（表4）．また，権威DNSサーバは2つのリフレクターで同一サーバとしている．

このような環境で実験を行った結果，攻撃回避システムを動作させなかった場合の被害サーバには，

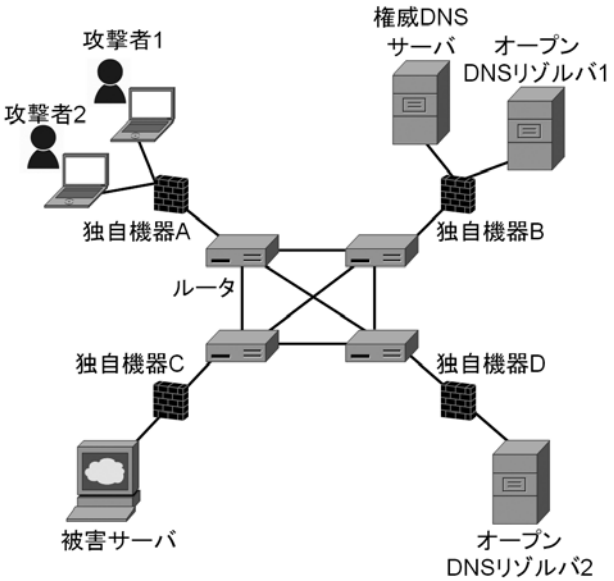


図11　分散攻撃に対する有効性確認実験用環境

表4　分散攻撃に対する有効性確認実験での攻撃者とリフレクターの組み合わせ

攻撃者	リフレクター
攻撃者1	オープンDNSリゾルバ1
攻撃者2	オープンDNSリゾルバ2

攻撃開始直後の9秒付近から攻撃パケットが到達しており，受信データ速度が急激に上昇している（図12）．その後は，帯域制限値である10Mbpsに達し，不安定に受信データ速度が推移している．この理由は，実際の受信データ速度は約10Mbpsであるが，到達したフレームは被害サーバのNICの受信バッファに一度格納され，その後，実際の受信データ速度とは異なるタイミングで出力がされるため，計測時に受信データ速度の推移が不安定になって

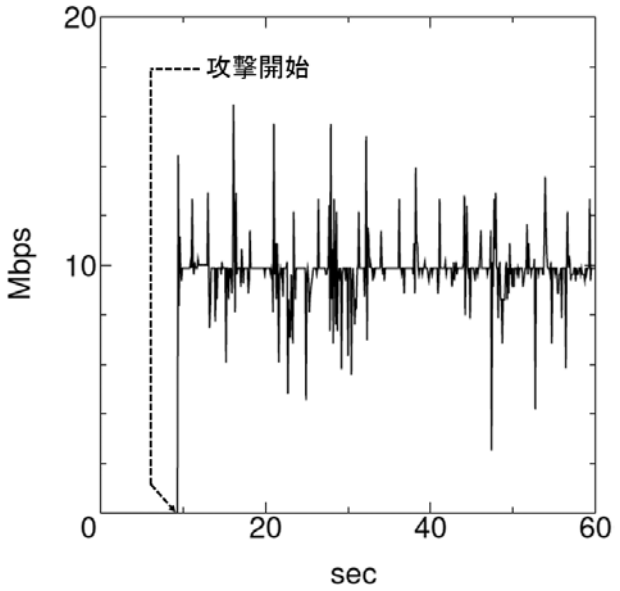


図12　対策なしの際の被害サーバでのDNSレスポンス受信速度

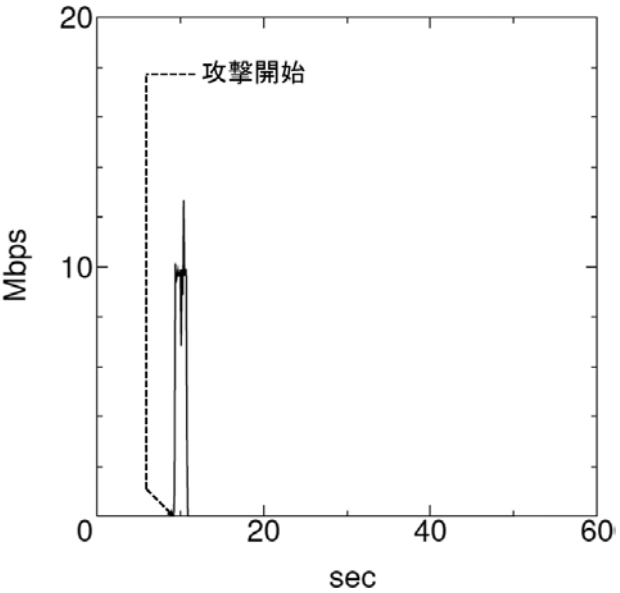


図13　対策ありの際の被害サーバでのDNSレスポンス受信速度

いると考えられる．これに対し，攻撃回避システムを動作させた場合は，９秒付近から攻撃パケットを1.6秒間受信したが，その後は，攻撃パケットを受信することはなかった（図13）．このことから，独自機器Bが通信増加検知を行うまでの間は攻撃パケットが到達するが，それ以降は本システムによって攻撃を回避することが可能であることがわかった．

５．２　リフレクターIP追加時の有効性

本システムのリフレクター近傍独自機器では通信増加検知後から宛先IPアドレスに関しても監視を行い，新たな宛先IPアドレスを持つパケットが到達した場合には，それに合った攻撃回避ルールの通知とその適用を行う．そのため，通信増加検知後に新たな宛先IPアドレスが追加された場合の有効性を確認する必要がある．このことから，攻撃者用マシンを２台用意し，同じリフレクター近傍独自機器を経由するように実験環境を構築した（図14）．また，各攻撃者が異なるDNSサーバをリフレクターとして使用し，１分間の時間差をつけて攻撃を開始した（表５）．

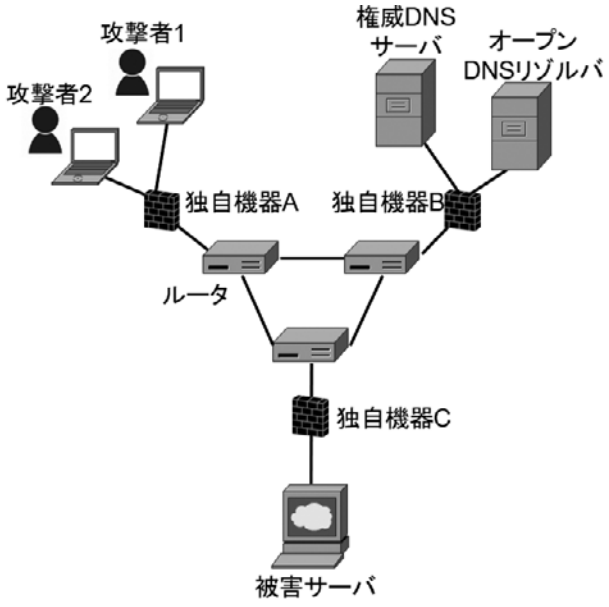


図14　リフレクターIP変化時の有効性確認実験用環境

表５　リフレクターIP追加時の有効性確認実験での攻撃者とリフレクターの組み合わせ

攻撃者	リフレクター
攻撃者１	オープンDNSリゾルバ
攻撃者２	権威DNSサーバ

この結果，被害サーバでは，攻撃パケットを攻撃開始直後の９秒付近から0.8秒間受信した（図15）．このトラフィックは攻撃者１による攻撃であり，前節の実験と同様に独自機器Bの通信増加検知までは攻撃パケットが到達していることがわかる．なお前節の攻撃回避と比較して攻撃パケットの到達時間が半分になっている理由は，攻撃者が１台であることから，通知及び適用を行うルールが半分になっていることや，不正クエリによる帯域の圧迫が少ないことが原因であると考えられる．さらに，その１分後の69秒付近で攻撃者２による攻撃を開始したが，その際には被害サーバに攻撃パケットは到達しなかった．これは，既に独自機器Bで通信増加検知が終了していることから，迅速に攻撃回避ルールが適用されたためである．

以上の結果から，攻撃回避中にリフレクターのIPアドレスが変化した場合には，通信増加検知前と比較して迅速に攻撃を回避できることがわかった．

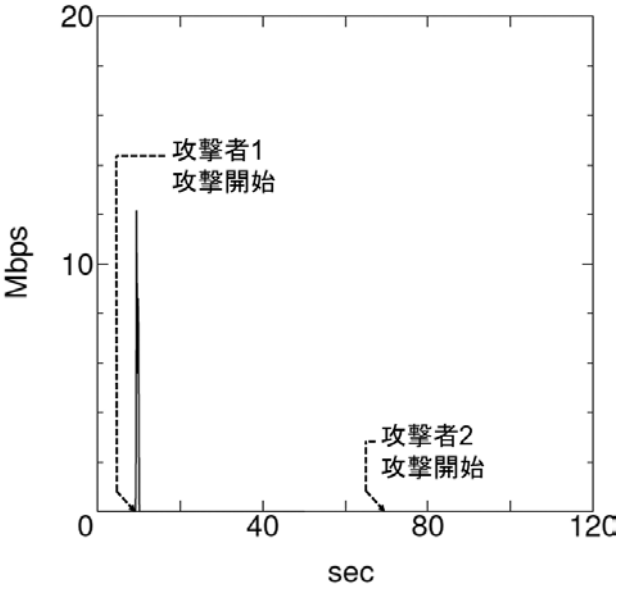


図15　リフレクターIP変化時の被害サーバでのDNSレスポンス受信速度

表６　攻撃及び対策の有無と正規通信の名前解決成功率と成功回数

環境	成功率	成功回数
攻撃なし	100％	166777／166777
攻撃あり＋対策なし	11.54％	15／130
攻撃あり＋対策あり	100％	128007／128007

5.3　正規通信の成功率

本システムでは，通信増加が正規通信であった場合は宛先ポート番号変換が適切に行われるため，通常の通信を阻害しない特徴がある．この確認のために前節で構築した実験環境を使用し，攻撃回避中に被害サーバからオープンDNSリゾルバに対して問い合わせを行った（図10）．問い合わせにはdigコマンドを30分間実行し続けるスクリプトを作成し，名前解決の成功回数と成功率を算出した（表6）．

結果として，対策ありの場合では名前解決の成功率が100パーセントであったことから，正規通信を阻害せずに攻撃回避を行うことが可能であることがわかった．しかし，攻撃なしの場合と比較し名前解決の成功回数が約23パーセント減少しているが，これは攻撃クエリによる独自機器Bとインターネット側のルータ間の回線の圧迫が原因であると考えられる．また，対策なしで攻撃を行った場合は，攻撃による帯域超過により問い合わせがタイムアウトを起こすまで待つため，名前解決の回数が著しく少なくなっている．

6．まとめ

本研究では，DRDoS攻撃のうちDNSamp攻撃を再現した上で，独自機器による宛先ポート番号の変換により，攻撃クエリと正規通信を区別し，攻撃を回避するシステムを構築した．これにより，DRDoS攻撃の特徴である正規通信と不正クエリの判別が困難であるという問題を解決した上で，複雑なルールを使用せずに攻撃回避を行うことを実現した．

本システムを用いることで，攻撃開始から約1.2秒程度で攻撃の回避が可能であることがわかった．また，リフレクターとなっているDNSサーバに対して，被害サーバから大量の名前解決要求を行った場合でも問題なく通信が行えたため，正規通信に影響を与えずに攻撃回避が可能であることがわかった．しかし，攻撃回避開始のトリガーに通信増加検知を使用しているため，検知までの間は被害サーバに攻撃パケットが到達した．このことから，攻撃回避が行われるまでは被害サーバやネットワークへの過負荷によってサービスが停止する可能性があるが，数秒で攻撃回避がなされるため，本システムを導入することにより，迅速なサービス再開を行うことが可能である．なお，本研究の実験では，実験ネットワークの構築に仮想環境を使用しているため，実際のISPと比較し，ネットワークの規模が小さい．そのため，実際にISPに実装する際にはより強力なスケーラビリティが求められると考えられる．

また，本研究では実験環境としてDNSamp攻撃を再現したが，本システムはポート番号とIPアドレスのみで攻撃と正規通信の判別が可能であるため，送信元IPアドレスを詐称した不正クエリを大量に送信する特徴のあるNTPamp攻撃等の多くのDRDoS攻撃に有効であると考えられる．

しかし，本システムは攻撃者が詐称した送信元IPアドレスを元に被害サーバ近傍独自機器を探索するため，被害サーバと攻撃者の近傍の独自機器が同一である場合に，不正クエリにも宛先ポート番号の変換を行ってしまうことから，攻撃回避を行うことが困難である．さらに，本システムはISPが実装することを前提としているが，このシステムに未対応のISPから不正クエリが到達する場合や，すでにリフレクターによって増幅されたパケットが該当ISP内に入ってくる場合には攻撃回避を行えない可能性がある．このことから，今後はISPだけでなく，複数のISPを繋ぐ役割のあるIX（Internet eXchange）への応用や，各ISP間の経路情報に基づいた攻撃回避ルールの適用方法を検討する必要があると考えられる．また，本システムでは通信増加を攻撃回避のトリガーとしているが，より精度の高い攻撃検知を行うためには，攻撃の実データセットに基づく攻撃トラフィックのシミュレートと，それに基づく攻撃検知アルゴリズムの検討が必要である．

【引用文献】

［1］国立研究開発法人　情報通信研究機構　サイバーセキュリティ研究所　サイバーセキュリティ研究室：NICTER観測レポート2020，https://www.nict.go.jp/cyber/report/NICTER_report_2020.pdf（参照 2021-11-30）．

［2］A10 Networks: How to Defend Against Amplified Reflection DDoS Attacks, https://www.a10networks.com/blog/how-defend-against-amplified-reflection-ddos-attacks/（参照 2021-11-30）．

［3］大井貴晴，落合秀也，江崎浩：オープンDNSリゾルバの現状把握手法の提案と評価，マルチメディア，分散協調とモバイルシンポジウム2018論文集，pp. 1126-1133（2018）．

［4］野口大貴，後藤滋樹：DDoSリフレクション攻撃の

分析と防御法，コンピュータセキュリティシンポジウム2016論文集，pp. 1183-1190（2016）.
［5］桂井友輝，中村嘉隆，高橋修：経路変更を用いた分散フィルタリングによるDNS amp攻撃への対策手法の提案，研究報告マルチメディア通信と分散処理（DPS），pp. 1-6（2015）.
［6］桂井友輝，中村嘉隆，高橋修：ネットワーク帯域への影響を考慮したDNSamp攻撃に対する攻撃パケットフィルタリング手法の提案，情報処理学会研究報告（Web），vol.2016-CSEC-72, no.26，https://ipsj.ixsq.nii.ac.jp/ej/?action=repository_uri&item_id=157862&file_id=1&file_no=1（参照 2021-11-30）.
［7］玄英哲，村山純一：DNSアンプ攻撃対策としての分散型仮想FirewallとNATの連携技術，電子情報通信学会　情報ネットワーク研究会　技術研究報告，vol.117，no.460，IN2017-91，pp. 11-14，2018年３月.
［8］玄英哲，首藤裕一，村山純一：DNSアンプ攻撃対策としての仮想Firewallにおけるフラグメントパケット処理方式，電子情報通信学会　情報ネットワーク研究会　技術研究報告，vol.116，no.45，IN2016-16，pp. 85-88，2016年５月.
［9］JPRS：技術解説：「DNS Reflector Attacks（DNSリフレクター攻撃）」について，https://jprs.jp/tech/notice/2013-04-18-reflector-attacks.html（参照 2021-11-30）.
［10］CISA: Alert（TA14-017A）UDP-Based Amplification Attacks, https://us-cert.cisa.gov/ncas/alerts/TA14-017A（参照 2021-11-30）.
［11］総務省：電気通信事業におけるサイバー攻撃への適正な対処の在り方に関する研究会第一次とりまとめ，https://www.soumu.go.jp/main_content/000283608.pdf（参照 2021-11-30）.
［12］IIJ：DNSオープンリゾルバ問題，https://www.iij.ad.jp/dev/report/iir/pdf/iir_vol21_internet.pdf（参照 2021-11-30）.