分散フィルタリングによる DNS リフレクター攻撃対策拡張手法の提案

M2015SC005 平松 剛
指導教員：河野　浩之

1　はじめに

Akamai 社の 2015 年度第 4 四半期のインターネットセキュリティレポート [1] によるとネットワークを利用する攻撃の中で，DNS リフレクター攻撃は 92%も前期より増加しており，現在の攻撃手法のトレンドであるといえる．DNS リフレクター攻撃とは，世界中に存在する設定不備の状態で運用されている，オープンリゾルバとよばれる DNS サーバを悪用することで攻撃者にとってより効率的な DDoS 攻撃をすることができる手法である．

この攻撃への対策を講じている先行研究 [3] には，ネットワーク機器への負荷による処理速度の低下や輻輳といった問題がある．そこで，我々が提案する手法ではその問題を解決するために分散フィルタリングを実行する．一般的に高い性能のルータが配置される BGP ルータでパケットフィルタリングを実行することで処理速度の低下を防ぎ，ネットワーク内部のある一点で処理していた既存手法とは異なり，分散的にフィルタリング処理をすることができる．この検証にはネットワークシミュレータ ns-3 の拡張版である ns-3 DCE 上に BGP で経路交換をするネットワークを実装し，フィルタリングの性能を評価する．

本研究論文は全 5 章で構成される．2 章では DNS リフレクター攻撃への対策として，提案されている手法について解説する．3 章では我々が提案する BGP ルータでフィルタリングを実行する手法を説明する．4 章では提案する手法に基づいてシミュレーションをし，5 章でその実験について評価，考察をする．

2　先行研究

この章では，DNS リフレクター攻撃への既存対策手法及び先行研究について記す．DNS リフレクター攻撃への対策として一番効果的な手法は，外からの名前解決の問い合わせにも応答するという適切なアクセスコントロールが設定されていないキャッシュサーバを全て無くすことであるが，それは現実的ではない．そこで攻撃への対策として様々な手法が考案されている．

2.1　DNS Amplification Attacks Detector

DAAD(DNS Amplification Attacks Detector)[3] とは，Kambourakis らによって考案された DNS の名前解決の仕組みに基づいて，DNS リフレクター攻撃を検出するシステムであり，パケットフィルタリングと併用することで攻撃からの被害を低減することができる．DAAD とネットワークフィルタリングを用いた被害の低減手法を図 1 に示し，図中の番号で示す緩和処理の流れを以下で説明する．

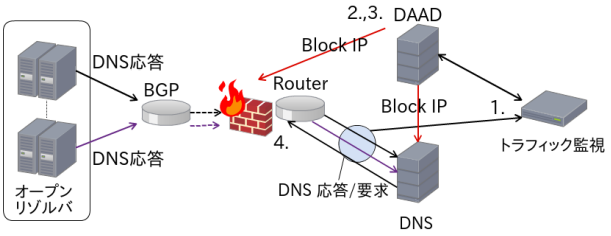

図 1　DAAD とフィルタリングによる被害低減手法

1. 名前解決のパケットを監視して要求/応答とポート番号を DAAD に登録する．
2. 名前解決では要求と応答は常に 1:1 であるので、データベースを照合することで応答だけの異常なパケットが一定値以上出現した場合，攻撃されていると判断する．
3. 応答だけを送信してきている IP アドレスをフィルタリングするようにルータやファイアウォールに設定する．
4. ルータやファイアウォールでフィルタリングを実行することで被害を緩和する．

Di らによって提案された手法 [2] は，Bloom Filter を用いることで DAAD を拡張し，データベースから効率的に検索することができる．これによって機器にかかる負荷が軽減された．

2.2　Remote Triggered Brack Hole Filtering

Remote Triggered Brack Hole Filtering(RTBH)[4] とは，BGP を用いたフィルタリング手法である．BGP の設定であらかじめ null Routing となるプレフィックスを設定しておき，攻撃が起こった際に攻撃を受けているプレフィクスの Nexthop を設定しておいたプレフィックスに向けて広報する．これによって攻撃パケットの向かう先は null となり，攻撃パケットが破棄される．

2.3　先行研究の課題

DAAD を使った手法では，Di らによる拡張でデータベースにかかる負荷が解消されたが，通信帯域にかかる負荷については解消されていない．また，RTBH では正常なパケットもブラックホールに巻き込んで破棄してしまうという問題点がある．

3　DAAD 拡張手法の提案

本章では，RTBH のように BGP を使うことで分散フィルタリングを可能にした提案手法について記す．3.1 では想定する環境について示し，3.2 では分散フィルタリング

の処理の概要について述べる．

3.1 想定する環境

被害端末は，インタんーネットにおける中規模 AS のネットワークの中に存在する端末の一つであり，所属する AS と隣接する AS との間で BGP による経路交換がされている．

3.2 分散フィルタリングの処理概要

本提案手法では，DAAD による DNS リフレクター攻撃の検知，検知後の解析によるフィルタリングパターンの判断，フィルタリングルールを BGP で流すことによる BGP Flowspec を用いた分散フィルタリングという順序で実行することで，DNS リフレクター攻撃による被害を緩和する．図 2 に簡単な処理の流れを示す．

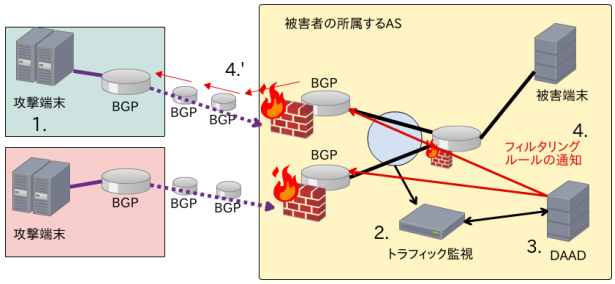

図 2 フィルタリング処理の流れ

3.3 DNS リフレクター攻撃の発生

パケット監視システムと DAAD によって DNS リフレクター攻撃が検知されるまでは，それぞれのネットワーク内で動作する機器は通常通りの機能を果たしている．攻撃を受けていると検知した場合に次の処理へ．

3.4 攻撃パターンの判別

DNS リフレクター攻撃では，同じ内容の問い合わせが繰り返し送信されてくることから被害者のネットワークに所属するユーザからの正常な DNS 問い合わせにできるだけ影響が出ないようなフィルタリングルールを設定することが可能である．DAAD のデータから，パケットサイズや送信先/送信元の IP アドレス，送信先ポート番号を調べる事で，あらかじめ決めておいた異常なパケットのルールのどれに近いか判定し，次の処理へと進む．ここでのルールとは，正常な通信をできる限り阻害しないようなフィルタリングルールを設定するためのものである．悪意のないユーザにできるだけ影響が出ないように設定した場合のルール例は以下の通りである．

1. 監視の結果，攻撃元が 1 つの場合に送信先/元 IP アドレスとポート番号，パケットサイズでフィルタリングする．
2. 監視の結果，攻撃元は複数だが，送信先ポート番号が一定である場合に，送信先 IP アドレスとポート番号，パケットサイズでフィルタリングする．
3. 監視の結果，攻撃元も，送信先ポート番号も複数である場合に，送信先 IP アドレスとパケットサイズでフィルタリングする．

DNS リフレクター攻撃では，送信元ポート番号が UDP53 と固定であること，送信先ポート番号に指定されるのが UDP0，53 がほとんどであり，パケットサイズが 4000byte 程度と攻撃の統計がでているのでルールが適用しやすい．

3.5 BGP Flowspec による分散フィルタリング

各ルータでは，判定されたフィルタリングルールが適用され次第攻撃パケットの送信元に対する帯域制限や，攻撃パケットの破棄など境界ルータでフィルタリングを実行していく．DAAD によって攻撃の発信元である DNS サーバを特定することで BGP にのせてフィルタリングルールを攻撃発信元にできるだけ近い，被害者から遠く離れたルータへ届ける．これにより，他のネットワークへの被害も最小限に抑えることができる．

しかし，現在広域ネットワークでの実装はすすんでおらず，その原因としては上位 ISP にあたるトランジット ISP が下位 ISP から流れてくるフィルタリングルールを信用できないという理由が主である．

BGP Flowspec の問題点解決

その問題を解決するために，我々の提案手法では，先行文献の RTBH で使用されているルートリフレクタとよばれるルータを用いて，同じ規模の ISP 同士で VPN を張ることでフィルタリングルールを伝えることを提案する．信頼できる同規模 ISP へルールを送り，その ISP が信頼する別 ISP へと伝えていくことでルールを広げていく．これならばメリット/デメリットがお互いに同じ規模であるので成立しやすいのではないかと考える．

4 分散フィルタリングの検証

本章では，実験に使用したツールと提案手法の有効性を示すために行うシミュレーション及び既存手法との比較方法について記す．

4.1 シミュレーション環境

本研究では，実験環境としてネットワークシミュレータである ns-3[6] 上に，提案手法に必要な攻撃者の踏み台となっている DNS サーバと BGP によって経路が交換されている AS，被害者のネットワークを実装する．通常 ns-3 では BGP を再現することができないが，ns-3 のフレームワークであり Linux カーネルに実装されたネットワークスタックを利用することができる Direct Code Execution(DCE)[7] を用いることで AS 間での通信を再現する．できる限り現実での環境に近づけるために BGP によるルーティングを実行した状態で攻撃シナリオをシミュレーションする．

4.2 シナリオファイル概要

ns-3 では DNS を再現することができないので，実験ではシミュレーションシナリオとして，DNS サーバ役のノードから DNS 応答パケットとして UDP パケットを大量に送信することで DNS リフレクター攻撃を擬似的に再

現する．実験に使用するシナリオファイルは，ns-3 DCE に付属している田崎創氏によって開発された BGP 用のサンプルファイルを参考に記述した．ns-3DCE ではルーティングソフトウェアである quagga を用いて BGP を動作させている．構築した実験環境を図 3 に示す．図 3 中の番号はすべてシナリオで記述したノードの番号と一致している．

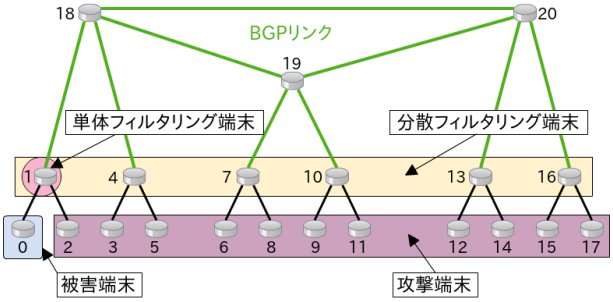

図 3　実験環境のノード配置図

攻撃端末はインターネット上のあらゆる場所に点在しているため，被害端末であるノード [0] 以外の端末であるノード [2,3,5,6,8,9,11,12,14,15,17] から攻撃パケットを送信する．シミュレーションを実行する上での各種パラメータの値を以下の表 1 に示す．

表 1　各種パラメータ

シミュレーション時間	60[sec]
ネットワークノード合計数	21 ノード
攻撃実行端末数	11
BGP 経路を構成するリンク帯域幅	100Mbps
その他経路を構成するリンク帯域幅	10Mbps
攻撃端末の攻撃レート	10Mbps
攻撃パケットの宛先ポート	UDP 53
エラー発生率	0.00001%
シミュレーション試行回数	20 回

4.3　提案手法の評価基準

実験では，BGP Flowspec を実際に実行することは困難であるので，境界ルータとして想定してあるネットワークノードで，あらかじめ設定しておいたフィルタリングルールに基づいてフィルタリングを実行する．攻撃がはじまり，分散フィルタリングが開始された後にフィルタリングルータから被害端末へのパケットロス率，スループットについても調査し，既存の手法である単体でのフィルタリング時と比較する．スループットは 2 種類の方法で測定をする.1 つめの手法は，ns-3 のパケットトレース機能を用いて測定する手法である．受信ソケットへ流入してきたパケットを 1 秒毎に区切ってファイルに出力していくことで計測区間のスループットを算出する．検証では TCP,UDP 共に計測することで攻撃パケットと正常なパケットのスループットを表示する.2 つめの手法は iperf を利用する方法であり，後述するパケットロスと同様の手法である．パケットロス率は ns-3 で計測する機能が DCE モジュールを導入することによって使用できなくなるために，逆に ns-3 DCE を導入することで使用可能となった iperf を実行することでその値を計測する．

4.4　シミュレーション結果

提案手法の有効性を検証するために行なったシミュレーションの結果であるスループット，パケットロス率の測定結果を示し既存手法と比較した結果を評価する．

シミュレーション実行環境

シミュレーションシナリオを実行した環境は以下の表 2 の通りである．ns-3 DCE では Linux カーネルを使用しており，今回のシミュレーションで使用するフィルタリング機能などの設定はデフォルトでは組み込まれていないので，menuconfig で細かい設定を行い，カーネルを再構築して使用した．

表 2　シミュレーション実行環境

OS	Ubuntu 14.04 LTS 64bit
CPU	Intel Core i7 920 @ 2.67GHz × 8
メモリ (RAM)	6GB
ns-3 バージョン	ns-3.26
DCE バージョン	DCE-1.9
カーネル	Linux 2.6.36

4.5　TCP スループットの比較

図 4 に示されているのは，ns-3 の機能を用いて出力されたグラフである．Dist が提案手法である分散フィルタリングによるスループットであり，Single が既存手法である単体でのフィルタリングとなっている．分散フィルタリングの TCP スループットの平均値は，一部分ランダムエラー要素によって単体フィルタリングに負けている部分もあるが，UDP スループットが減少し始めている部分で総合的に見て単体フィルタリングよりも高い値であることがわかる．

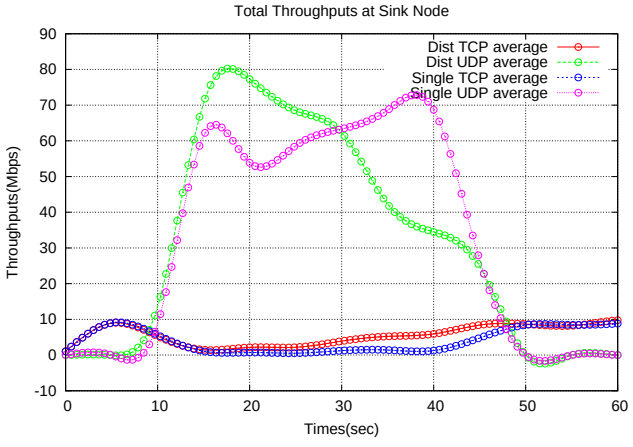

図 4　スループット比較図

単体フィルタリングではシミュレーション開始から 45 秒，分散フィルタリングでは遠距離から 20 秒，30 秒，45 秒の時にフィルタリングを開始しており，その結果が UDP のスループットが正しく減少していることがわかる．

4.6 iperf による計測区間

単体フィルタリング

単体でのフィルタリングを実行する端末はノード [1] であり，攻撃パケットがこのルータのみで処理されるために，ノード [1] とノード [18] の間が特に通信の負荷がかかるボトルネックリンクとなることが予想される．

分散フィルタリング

分散フィルタリングを実行する端末はノード [1，4，7，10，13，16] の 6 箇所であり，トランジット ISP を模しているノード [18，19，20] でのパケット流量が減少するはずである．そこで，そのポイントを押さえる観測位置として表 3 に示したノードで iperf のサーバ/クライアントを動作させる．

表 3 スループット, パケットロス率測定ノード一覧

種別	トランジット	ボトルネック
Server	nodes[0]	nodes[1]
Client	nodes[1，4，7，10，13，16]	nodes[18]

パケットロス率の計測結果

単体フィルタリング計測結果

```
    Bandwidth    Lost/Total   Datagrams
9.23 Mbits/sec   1058/ 8921   (12\%)
```

iperf の結果，単体フィルタリングの場合 12%のパケットロスということがわかる．また，帯域幅は 9.23Mbits/sec で利用することができている．

分散フィルタリング計測結果

```
    Bandwidth    Lost/Total   Datagrams
9.80 Mbits/sec   581/ 8921   (6.5\%)
```

iperf の結果，単体フィルタリングの場合 6.5%のパケットロスということがわかる．また，帯域幅は 9.8Mbits/sec で利用することができている．

比較結果

パケットロス計測結果では提案手法である分散フィルタリングが 6.5%，既存手法である単体フィルタリングで 12%と，送信パケットに占める割合で 5.5%の減少となった．スループットでは，分散フィルタリングでは帯域幅が 9.80Mbits/sec を計測したのに対して単体フィルタリングの場合では 9.23Mbits/sec を計測したので，約 6%スループットが向上した．従って，提案手法では既存手法の問題であったスループットの向上と通信路の負荷によるパケットロスについて改善することができたといえる．

5 おわりに

本研究では，DNS リフレクター攻撃の対策として正確性の高い DAAD という攻撃の検知手法を利用することで正確なフィルタリングルールを導き出した．そのルールをもとに BGP Flowspec[5] を用いることで被害端末のネットワークから遠く離れた攻撃元に近い BGP ルータへフィルタリングルールを届けることで，分散フィルタリングをする手法について提案をした．

インターネット上に無数に点在する DNS サーバからのパケットが集中する前の段階である BGP ルータ上でフィルタリングを行うことで，分散フィルタリングを実施し，単体でのフィルタリングでの問題であるフィルタリング機器付近でのネットワークへの影響，また機器への負荷の面で緩和をすることができた．具体的には，iperf で計測したパケットロス率では，送信パケットに占める割合で既存手法に比べて 5.5%の減少を確認できた．また，スループットでは約 6%の改善を確認することができた．

今後の課題としては，分散フィルタリングのルールの伝達手段として，トランジット ISP などの自分より上位の ISP にそのルールいかに信用してもらうのか．上位 ISP へ BGP によるルールの伝達が可能になれば一度にかなりの ISP へルールを伝達することが可能になる．また，そうなった場合には伝達の速度もかなり高速化することができると考えられるため，早急に安心してルールを配布できる仕組み作りをしていかなければならない．

参考文献

[1] Akamai Technologies: Q4 2015 State of the Internet - Security Report (2015).

[2] Di Paola, S. and Lombardo, D.: *Protecting against DNS Reflection Attacks with Bloom Filters*, pp. 1–16, Springer Berlin Heidelberg (2011).

[3] Kambourakis, G., Moschos, T., Geneiatakis, D. and Gritzalis, S.: Detecting DNS amplification attacks, *International Workshop on Critical Information Infrastructures Security*, Springer, pp. 185–196 (2007).

[4] Kumari, W. and McPherson, D.: Remote Triggered Black Hole Filtering with Unicast Reverse Path Forwarding (uRPF), RFC 5635 (Informational) (2009).

[5] Marques, P., Sheth, N., Raszuk, R., Greene, B., Mauch, J. and McPherson, D.: Dissemination of Flow Specification Rules, RFC 5575 (Proposed Standard) (2009). Updated by RFC 7674.

[6] National Science Foundation: ns-3, http://www.nsnam.org/ (Accessed Jan. 2017).

[7] 田崎 創：ns-3 Direct Code Execution(ns-3-dce), *https://www.nsnam.org/overview/projects/direct-code-execution/* (Accessed Jan. 2017).