L-017

DDoS 攻撃対策のための ISP 間連携フレームワークの構築
Inter-ISP collaboration framework for detection and defense of DDoS attacks

小西 崇之 *　　瀧本 栄二 *
Takayuki Konishi　　Eiji Takimoto

1　はじめに

インターネットが重要なインフラとなった現代において，Distributed Denial of Service (DDoS) 攻撃によってもたらされる損害は，非常に深刻なものとなっている．DDoS 攻撃とは攻撃元をインターネット上に分散させた Denial of Service (DoS) 攻撃のことであり，DoS 攻撃とは標的サービスの停止を目的とした攻撃の総称である．

現在の DDoS 攻撃対策の主流は，標的サービスの管理組織のファイアウォールで該当トラフィックを遮断する方法や，Internet Service Provider (ISP) に防御要請し ISP で対処する方法が一般的である．しかし，近年の DDoS 攻撃の大規模化と分散化により，所属 ISP のネットワークにおいての防御では効果が不十分であると考えられる．DDoS 攻撃を効果的に防御するためには，攻撃元に近い ISP において防御することが重要である．そのためには，ISP 間で連携して防御に必要な情報の共有とシグナリングが必要となる．また，各 ISP のネットワークで発生した異常トラフィック情報も合わせて共有することで，より迅速で正確な DDoS 攻撃検知を実現できると考えられる．

そこで本論文では，DDoS 攻撃対策のための ISP 間連携フレームワークを提案する．本フレームワークは，インターネット上に DDoS 攻撃情報を収集し提供する攻撃情報管理サーバを設置し，ISP との連携を促進する．また，通信プロトコルは現在 Internet Engineering Task Force (IETF) で標準化が進められている，DDoS 攻撃関連情報のリアルタイムシグナリングプロトコルの DDoS Open Threat Signaling (DOTS) [1] [2] [3] [4] [5] を使用することで，オープンなフレームワークとして使用できる．

* 立命館大学　Ritsumeikan University

2　DOTS

DOTS には，DOTS サーバと DOTS クライアントと呼ばれる二つのコンポーネントがある．DOTS のスコープはこの DOTS サーバと DOTS クライアントの間の通信のみである．DOTS の最も簡単な構成例としては，ISP などが持つ DDoS 攻撃防御装置・DOTS サーバ・DOTS クライアント・企業などの大規模な DDoS 攻撃の攻撃対象である．DOTS の最も簡単な構成例を図 1 に示す．

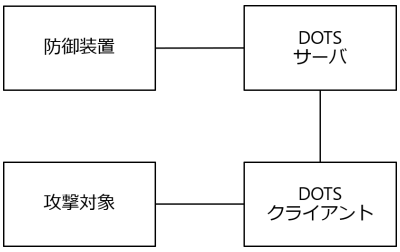

図 1　DOTS の最も簡単な構成例

DOTS において DOTS クライアントと DOTS サーバは，シグナルチャネルとデータチャネルという 2 種類のチャネルを確立する．2 種類のチャネルを図 2 に示す．

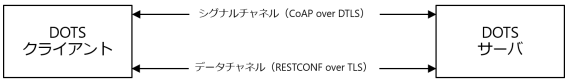

図 2　2 種類のチャネル

シグナルチャネルは，防御要請を送信するためのチャネルである．通信プロトコルは，Constrained Application Protocol (CoAP) [6] over Datagram Transport Layer Security (DTLS) [7] または CoAP over Transport Layer Security (TLS) [8] が用いられ，デフォルトポート番号は 4646 と定義されている．CoAP は HTTP

に似たメソッドを持っており，このメソッドの使い分けによって DOTS クライアントと DOTS サーバはシグナリングを行う．シグナルチャネルのデータ形式は全て JavaScript Object Notation (JSON) [9] を Concise Binary Object Representation (CBOR) [10] でシリアライズした上で送信される．

データチャネルは，シグナルチャネルを用いて防御要請を送信する前に，共有する設定情報などを交換するためのチャネルである．通信プロトコルは，Representational State Transfer Configuration Protocol (RESTCONF) [11] が用いられている．RESTCONF は，TLS 上に確立された Network Configuration Protocol (NETCONF) [12] データストアを操作するために HTTP をベースに作られたプロトコルである．

3　提案フレームワーク

提案フレームワークでは，DOTS クライアントと DOTS サーバの機能を持つノードを各 ISP に設置し，攻撃情報管理サーバと通信を行う．提案フレームワークの構成を図 3 に示す．

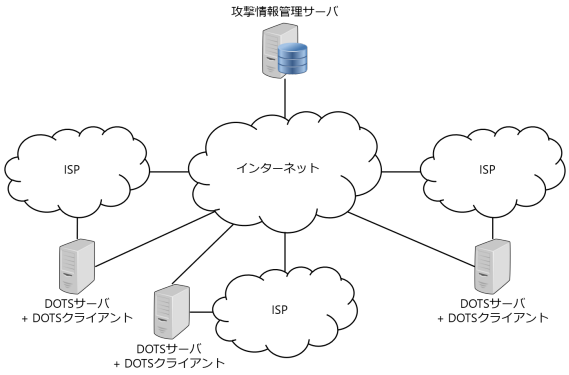

図 3　提案フレームワークの構成

攻撃情報管理サーバには防御要請配信機能とトラフィック情報管理機能という二つの機能がある．防御要請配信機能とは，攻撃情報管理サーバが ISP から受信した防御要請を，他の各 ISP に一斉に送信する機能である．トラフィック情報管理機能とは，攻撃情報管理サーバが各 ISP から受信した DDoS 攻撃と断定できないがその可能性があるトラフィックの情報をデータベースに保存し，アラートの送信や ISP からのトラフィック情報の問い合わせに対して応答する機能である．攻撃情報管理サーバの構成を図 4 に示す．

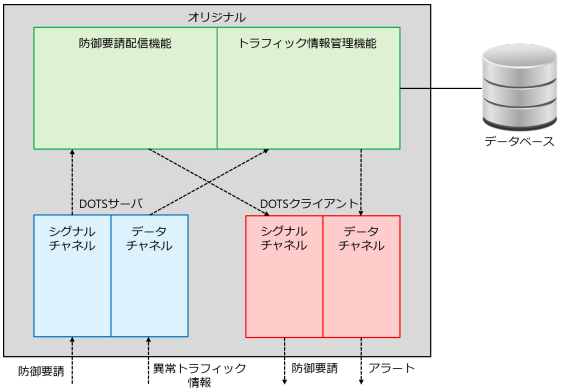

図 4　攻撃情報管理サーバの構成

3.1　ユースケース

本項では，提案フレームワークの三つのユースケースについて説明する．

3.1.1　防御要請配信

ISP が下流ネットワークの管理組織からの防御要請など，何らかの方法で DDoS 攻撃を検知した際に，DOTS クライアントのシグナルチャネルを用いて攻撃情報管理サーバに防御要請を送信する．防御要請を受信した攻撃情報管理サーバは，他の各 ISP にシグナルチャネルを用いて防御要請を送信する．防御要請を受信した各 ISP は，該当トラフィックが自身のネットワークを通過していた場合，防御装置に防御指示を出す．防御要請配信のユースケースを図 5 に示す．

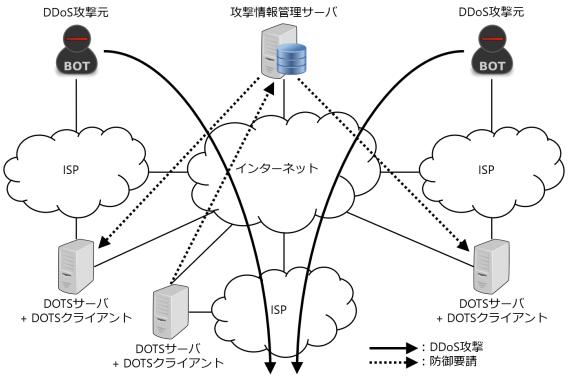

図 5　防御要請配信のユースケース

3.1.2　異常トラフィック情報報告

ISP が自身のネットワークにおいて異常なトラフィックを検知した際に，DOTS クライアントのデータチャネルを用いて攻撃情報管理サーバに異常トラフィック情報を送信する．本論文において異常トラフィック情報とは，特定の IP アドレスへのトラフィックの急激な増加や，存在しない IP アドレスに対して同じポート番号

あてのトラフィックの増加など，通常観測されないトラフィックだが DDoS 攻撃と断定できないトラフィックの情報と定義する．異常トラフィック情報の内容については，トラフィックの送信元・宛先 IP アドレスや送信元・宛先ポート番号などが考えられる．そのため，データモデルは組織間のセキュリティインシデント情報の交換のためのデータモデルである Incident Object Description Exchange Format (IODEF) [13] を使用する．攻撃情報管理サーバは，受信した異常トラフィック情報をデータベースに保存しておく．同時期に異常トラフィック情報が急増した場合，DDoS 攻撃が発生した可能性が高いと判断し，各 ISP にアラートを送信する．異常トラフィック情報報告のユースケースを図 6 に示す．

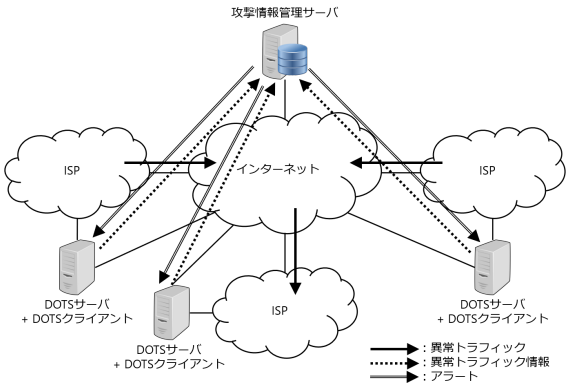

図 6　異常トラフィック情報報告のユースケース

3.1.3　異常トラフィック情報問い合わせと通知

ISP が自身のネットワークにおいての DDoS 攻撃の検知などに，他の ISP が送信した異常トラフィック情報が必要であれば，攻撃情報管理サーバに問い合わせをし，異常トラフィック情報を取得できる．異常トラフィック情報問い合わせのユースケースを図 7 に示す．

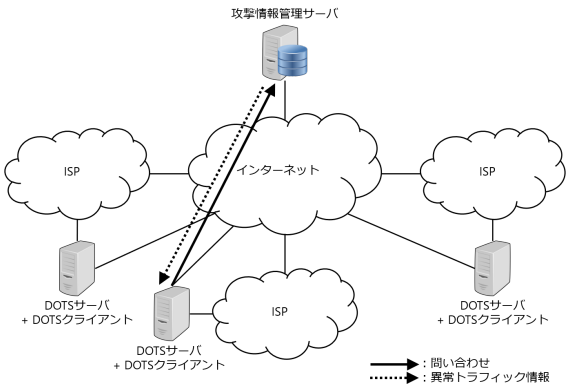

図 7　異常トラフィック情報問い合わせのユースケース

また，攻撃情報管理サーバに問い合わせをした際に，ISP は取得したい異常トラフィック情報の条件を攻撃情報管理サーバに登録すると，攻撃情報管理サーバが条件を満たす異常トラフィック情報を受信した場合，受信した異常トラフィック情報を ISP に送信する．条件の例としては，自身のネットワーク宛である場合や送信元 IP アドレスの中に自身のネットワークのアドレスが含まれている場合などである．特定異常トラフィック情報通知のユースケースを図 8 に示す．

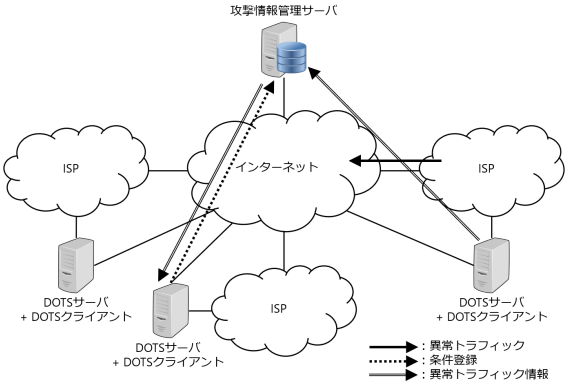

図 8　異常トラフィック情報通知のユースケース

4　考察

DDoS 攻撃は標的だけでなく近隣 ISP にも影響が及ぶため，攻撃元に近い ISP で迅速に防御することが望ましい．提案フレームワークは，攻撃情報管理サーバの防御要請配信機能により他の ISP への防御要請を実現しており，標的ネットワークでの攻撃検知からインターネット全体への防御要請を 3 回の通信で行うことができる．よって，提案フレームワークは ISP 間で連携して防御に必要な情報の迅速な共有とシグナリングを可能にし，有効であると考えられる．また，これまでの DDoS 攻撃検知や予測技術は，ISP 内をカバーするものであり得られる情報が予測精度を高めるには不十分となる可能性がある．誤った判断を避けるためには，インターネットを網羅する情報が必要である．提案フレームワークは，トラフィック情報管理機能によりそのような情報を提供する手段を提供するため，攻撃検知からの防御の自動化・高速化のみならず予測と対策により攻撃自体を未然に防ぐことが期待できる．今後は，異常トラフィック情報のデータモデルは IODEF を使用するが，どのデータを必須とし，どのデータをオプションとするかといった具体的検討が必要である．

5 おわりに

本論文では，効果的な DDoS 攻撃対策をするには，ISP 間で連携して防御に必要な情報の共有とシグナリングが必要であると考え，DDoS 攻撃対策のための ISP 間連携フレームワークの構築を提案した．そして，提案フレームワークについて考察を行った．今後は，異常トラフィック情報のどのデータを必須とし，どのデータをオプションとするかといった具体的な検討を課題とする．

謝辞

本研究は JSPS 科研費 JP18K18045 の助成を受けたものです．本研究を進めるにあたり，立命館大学情報理工学部の毛利公一教授には，研究の進め方などの様々なご指導をいただきました．深く感謝いたします．

参考文献

[1] A. Mortensen, K.T. Reddy, and R. Moskowitz, “DDoS Open Threat Signaling (DOTS) Requirements,” RFC 8612, May 2019. https://rfc-editor.org/rfc/rfc8612.txt.

[2] A. Mortensen, K.T. Reddy, F. Andreasen, N. Teague, and R. Compton, “Distributed-Denial-of-Service Open Threat Signaling (DOTS) Architecture,” Internet-Draft draft-ietf-dots-architecture-14, Internet Engineering Task Force, May 2019. Work in Progress. https://datatracker.ietf.org/doc/html/draft-ietf-dots-architecture-14.

[3] K.T. Reddy, M. Boucadair, P. Patil, A. Mortensen, and N. Teague, “Distributed Denial-of-Service Open Threat Signaling (DOTS) Signal Channel Specification,” Internet-Draft draft-ietf-dots-signal-channel-34, Internet Engineering Task Force, May 2019. Work in Progress. https://datatracker.ietf.org/doc/html/draft-ietf-dots-signal-channel-34.

[4] M. Boucadair and K.T. Reddy, “Distributed Denial-of-Service Open Threat Signaling (DOTS) Data Channel Specification,” Internet-Draft draft-ietf-dots-data-channel-29, Internet Engineering Task Force, May 2019. Work in Progress. https://datatracker.ietf.org/doc/html/draft-ietf-dots-data-channel-29.

[5] R. Dobbins, D. Migault, S. Fouant, R. Moskowitz, N. Teague, L. Xia, and K. Nishizuka, “Use cases for DDoS Open Threat Signaling,” Internet-Draft draft-ietf-dots-use-cases-17, Internet Engineering Task Force, Jan. 2019. Work in Progress. https://datatracker.ietf.org/doc/html/draft-ietf-dots-use-cases-17.

[6] Z. Shelby, K. Hartke, and C. Bormann, “The Constrained Application Protocol (CoAP),” RFC 7252, June 2014. https://rfc-editor.org/rfc/rfc7252.txt.

[7] E. Rescorla and N. Modadugu, “Datagram Transport Layer Security Version 1.2,” RFC 6347, Jan. 2012. https://rfc-editor.org/rfc/rfc6347.txt.

[8] E. Rescorla, “The Transport Layer Security (TLS) Protocol Version 1.3,” RFC 8446, Aug. 2018. https://rfc-editor.org/rfc/rfc8446.txt.

[9] T. Bray, “The JavaScript Object Notation (JSON) Data Interchange Format,” RFC 8259, Dec. 2017. https://rfc-editor.org/rfc/rfc8259.txt.

[10] C. Bormann and P.E. Hoffman, “Concise Binary Object Representation (CBOR),” RFC 7049, Oct. 2013. https://rfc-editor.org/rfc/rfc7049.txt.

[11] A. Bierman, M. Björklund, and K. Watsen, “RESTCONF Protocol,” RFC 8040, Jan. 2017. https://rfc-editor.org/rfc/rfc8040.txt.

[12] R. Enns, M. Björklund, A. Bierman, and J. Schönwälder, “Network Configuration Protocol (NETCONF),” RFC 6241, June 2011. https://rfc-editor.org/rfc/rfc6241.txt.

[13] R. Danyliw, “The Incident Object Description Exchange Format Version 2,” RFC 7970, Nov. 2016. https://rfc-editor.org/rfc/rfc7970.txt.