Vol.2017-CSEC-77 No.9
Vol.2017-IOT-37 No.9
2017/5/26

組織内発ダークネット宛通信分析による ネットワーク管理者支援の提案

山門 彩[1]　佐藤 聡[2,a)]　新城 靖[3]

概要：
ダークネット宛通信発生の要因には様々なものがあり，特に，組織内ネットワークから組織内のダークネット宛通信には，すぐに解消すべき事項が含まれている可能性がある．そのような通信を選別し，その問題を解決するには機器の利用者とアプリケーションの特定が必要であり，その作業は管理者にとって手間がかかる．本研究ではその作業をスムーズに行うことのできる方法を提案し，管理者の負担を軽減することを目的とする．まず，ブラックホールルータにて破棄される組織内ネットワークからの通信のログ (ヘッダのみ) を分析し，ネットワーク管理者に不適切通信を行っている調査対象候補機器を通知する．さらに調査対象候補機器の利用者の同意を得て，その通信をハニーポットを用いて調査対象機器に対して疑似応答を返し，その一連のやりとりのパケットを収集および分析することで不適切通信の原因特定を試みる．

キーワード：ダークネット，ハニーポット

1. はじめに

組織内ネットワークでは，利用されていない IP アドレス領域が存在する．例えば，現在筑波大学で利用可能な IP アドレスとして 2 つのクラス B の IP アドレス空間 131,072 個が割り当てられている．それらの IP アドレスは有効的に利用されているが，しかしながらその中には利用されていない IP アドレス領域も存在する．本論文では，この利用されていない IP アドレス領域のことを**ダークネット**と呼ぶ．

従来，ダークネットに送られたパケットは組織内コアルータからパケットを破棄するためのサーバ，すなわちブラックホールサーバへと転送され，破棄されていた．これは組織内コアルータと大学の境界に設置されている境界ルータの間でパケットのループを避けるための方策である．破棄する際には，その通信の発信元 IP アドレス，宛先 IP アドレス，発信元ポート番号, 宛先ポート番号，プロトコルといったパケットヘッダに含まれている 5 タプルの情報をログとして記録している．以後，この論文では，このログのことを，**通信ログ**と呼ぶ．

通常のネットワーク利用ではダークネットにパケットは到着しないはずであるので，ダークネット宛てのパケットの中には不適切な通信が含まれている可能性が高いといえる．特に，組織外を発信元とするパケットは攻撃だと思われる．我々は, そのような組織外からのパケットを解析する研究を行ってきた [1]. この論文では, 組織内を発信元とするパケットの解析について述べる．

組織内を発信元とするパケットは機器の設定ミスや DNS の設定ミスによるもの, マルウェアの感染によるものが考えられる．これらの不適切な通信は重大なセキュリティリスクにつながりかねないため，早期発見し問題を解決したいという要求がある．それを実現するためには，ネットワーク管理者はダークネット宛通信を発生させている機器の利用者に伝え，利用者はダークネット宛通信の発生原因を特定し，通信を止める必要がある．しかし，機器の利用者はパケットヘッダの情報と同じ情報だけを含む通信ログだけで，発生原因の原因を特定することは非常に困難である．利用者は，ダークネット宛通信を発生させているアプリケーションが分かれば，原因を特定することの助けになる．一般的には，これらの作業については，ネットワーク管理者がその通信のパケットを解析し，パケットの情報

[1] 筑波大学 システム情報工学研究科 コンピュータサイエンス専攻
Department of Computer Science, Graduate School of System and Information Engineering
[2] 筑波大学 学術情報メディアセンター
Academic Computing and Communications Center, University of Tsukuba
[3] 筑波大学 システム情報工学域 情報工学域
Division of Information Engineering, Faculty of Engineering, Information and Systems, University of Tsukuba
[a)] akira@cc.tsukuba.ac.jp

からアプリケーションを特定している．この作業は，ネットワーク管理者に非常に多くの労力を発生させている．したがって，発信源の特定とその通信を止めるためには，その通信の情報から，その通信を発生させているアプリケーションを特定する仕組みは有益であると考えられる．

そこで，本研究では組織内ダークネット宛て通信のアプリケーションを特定することで，ネットワーク管理者を支援する．これにより，不適切な通信を減少させて組織内ネットワークのセキュリティ向上につなげる．なお，本研究において“不適切な通信”とは以下に示す 2 点であると定義する．

- 機器の設定ミスによる通信
- マルウェアによる通信

本研究ではダークネット宛の通信ログとハニーポットを用いて組織内ネットワークの不適切利用の発生源の特定を支援する手法を提案する．まず，パケットヘッダと同様の情報を持つ通信ログを分析し，ネットワーク管理者に不適切通信を行っている調査対象候補機器を通知する．そして，調査対象候補機器の利用者の同意を得て，ハニーポットを用いて調査対象機器に対して疑似応答を返す．その一連のやりとりのパケットを収集および分析することにより，その通信を発生させているアプリケーションを特定する．その情報を利用者に伝えることにより，利用者がより簡単に不適切通信を止めることができる．

2. 組織外からのダークネット宛通信観測

我々は, 組織外から組織内ダークネット宛の通信を解析する研究を行っている．文献 [1] では HTTPS を擬態可能なハニーポットを実装し, それを筑波大学のダークネットに設置し, HTTPS トラフィックの収集と分析した結果について述べた．文献 [1] のシステムでは HTTPS 以外にも TLS 上で動作するプロトコルに対しても拡張して利用可能である．そこで，TLS 上で動作する複数のプロトコル，HTTPS，IMAPS，POPS を擬態可能なハニーポットを設置し，2015 年 12 月 2 日から 2015 年 12 月 31 日までの 30 日間で TLS 及び HTTPS，IMAPS，POPS のトラフィックを収集した．その時のネットワーク構成を図 1 に示す．また，表 1 に収集したデータの概要を示す．なお，括弧内の割合は各項目の合計値に対する，それぞれのプロトコルの数値の割合を示す．

TLS アクセスとは TLS 層でホストがパケットを送信した接続であり，TLS セッション確立とは TLS ハンドシェイクが正常に終了した接続である．また，上位プロトコル通信数とは，TLS ハンドシェイクが正常に終了し，その後 HTTPS，IMAPS，および POPS 等の TLS 上のプロトコルで通信を行った接続である．TLS アクセスに対して TLS ハンドシェイクを正常に終了した接続の割合は，HTTPS が 51.5%，IMAPS が 10.6%，POPS が 30.9%であった．また，それぞれのプロコトルで，TLS アクセスに対して上位プロトコルで通信を行った接続の割合は HTTPS が約 9.8%，IMAPS が約 1.4%，POPS が約 2.09%となっている．外部ホストとは， ダークネットの外にあり，収集期間内に TLS アクセスを行ったホストの数であり，ダークネット内サーバとは， ダークネットの内にあり，ハニーポットがエミュレートしたサーバである．

また，各プロトコルの TLS メッセージの種類ごとの解析を行った結果，HTTPS と POPS に対しては Heartbleed 脆弱性 [2] を狙った攻撃が見受けられた．さらに，HTTP リクエストを解析した結果，次のような攻撃を発見した．

- Vtiger 社が開発を主導するコミュニティ駆動型のオープンソース CRM ソフトウェア vtiger CRM[3] を標的としたスキャン
- SAP ジャパン株式会社 [4] のポータルソフトウェアを利用している Web ページのスキャン

文献 [1] のシステムおよび本節で説明した拡張を行ったシステムでの調査では，組織外から組織内ダークネット宛の通信を監視しているが，組織内から組織内ダークネット宛の通信はブラックホールルータにて通信ログを記録して破棄していた．

3. 組織内からダークネット宛通信の予備調査

不適切な通信は組織外から組織内だけでなく，組織内から組織内へも行われるが，それが問題になる場合がある．例えば，設定ミスによって設定された宛先 IP アドレスが，単なる未使用 IP アドレスなら問題無いが，その後その IP アドレスが使用されることとなった場合，情報流出リスクが有る．具体例として，印刷データを送信するプリンタの IP アドレスを未使用 IP アドレスにしただけでは問題無いが，その後もその誤った設定を放置し，かつて未使用であった IP アドレスが他人に使用されることとなった場合，印刷データの流出が発生してしまう可能性がある．

組織内から組織内宛の不適切な通信として，ポートスキャンもある．ポートスキャンとは，多数の対象ホストの TCP または UDP ポートに次々と接続を試みてポートの状態を調査する行為である．ポートスキャンの発生原因として考えられるのはボット感染による探索行動によるものがある．ボットに感染した場合，ユーザの気づかぬ間に攻撃の踏み台にされる．

2 章で述べた解析ネットワークでは組織外から組織内ダークネット宛て通信のみを観測対象としており，組織内から組織内ダークネット宛て通信はブラックホールサーバで破棄していた．しかしながら組織内からの通信の中にも不適切な通信が含まれている可能性がある．そこで，筑波大学の学内用ブラックホールサーバの 2016 年 8 月 20 日の通信ログ 24 時間分を調査した．その結果，組織内からダークネットに宛てて約 342 万件，1 秒あたり約 40.2 件の

表 1 組織外からダークネット宛通信の観測結果の概要 (30 日間)

項目名	合計	HTTPS	IMAPS	POPS
TLS アクセス	10,193,353	9,351,530(91.67%)	344,659(3.46%)	497,164(4.87%)
TLS セッション確立	5,008,246	4,817,015(96.18%)	37,486(0.75%)	153,745(3.07%)
上位プロトコル通信数	932,534	917,311(98.37%)	4,820(0.52%)	10,403(1.11%)
外部ホスト数	5,302	4,731	2,884	2,885
ダークネット内サーバ数	20,378	20,378	20,378	20,376

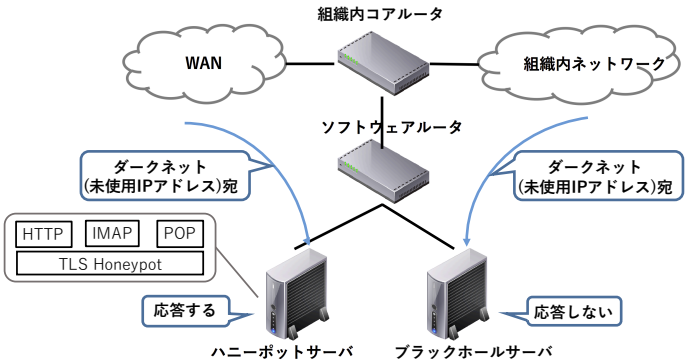

図 1 組織外からダークネット宛通信の監視ネットワーク構成

通信があったことがわかった．筑波大学に割り当てられた全 IP アドレス 131,068 個のうち，調査を行った時点での未使用 IP アドレスはすくなくとも 2 万個である．表 2 に調査結果の概要を示す．

外部ホストとは，組織内のネットワークでダークネットの外にあり，調査期間内にダークネット内のホストに対して TCP または UDP ポートにアクセスを行ったホストである．ダークネット内サーバとは，ダークネット内にあり，外部ホストからアクセスされたサーバである．

これらの通信の発生原因として考えられるのは，ネットワーク機器の設定ミスや，管理者による IP アドレスが未使用であることの確認 (ネットワーク管理者オペレーション)，またポートスキャンなどである．

上記期間中のダークネット宛通信をプロトコルと宛先ポート番号ごとに集計を行った．表 3 に TCP の宛先ポートごとにグルーピングし，パケット数の多い順にソートしたものの上位 6 位を，表 4 に UDP のそれを示す．なお，動的プライベートポート番号 (49152-65535) の“標準的なアプリケーション”については，参考文献 [5] に記載されているアプリケーション名であり，ポート番号とアプリケーションの 1 対 1 対応を示すものではない．ウェルノウンポート (1-1023) であればポート番号からアプリケーションを特定しやすいが，それ以外のポートについてはアプリケーションを特定するにはパケットの解析が必要になる．

表 2 組織内からダークネット宛通信の予備調査結果の概要 (24 時間)

項目名	TCP	UDP
アクセス数	183,976	3,292,597
外部ホスト	201	348
ダークネット内サーバ	397	1,315

表 3 および表 4 に示したポート番号の通信のうち，ウェルノウンポート番号及び登録済みポート番号である 9100

表 3 TCP 宛先ポート別通信数上位 10 位

ポート番号	標準的なアプリケーション	パケット数	割合 [%]
9100	LPR - RAW Printing	65553	35.6
57666	Xsan. Xsan Filesystem Access	41442	22.5
50442	Xsan. Xsan Filesystem Access	20656	11.2
80	HTTP	19984	10.9
50751	Xsan. Xsan Filesystem Access	10303	5.60
10050	Zabbix agent	6096	3.31
445	Microsoft-DS SMB file sharing	3827	2.08
139	NetBIOS NetBIOS Session Service	3621	1.97
22	SSH	3061	1.66
515	Line Printer Daemon (LPD)	3040	1.65
その他	—	6393	3.47

表 4 UDP 宛先ポート別通信数上位 10 位

ポート番号	標準的なアプリケーション	パケット数	割合 [%]
161	SNMP	1774771	53.9
8610	Canon MFNP Service	528135	16.0
3756	Canon CAPT Port	441098	13.4
8612	Canon BJNP Port 2	353445	10.7
514	Syslog	42459	1.29
53	DNS	39577	1.20
8611	Canon BJNP Port 1	29434	0.894
137	NETBIOS Name Service	17028	0.517
123	Network Time Protocol	16611	0.504
3289	ENPC	11714	0.356
その他	—	38255	1.16

番/tcp ポート宛，80 番/tcp ポート宛，10050 番/tcp ポート宛，161 番/udp ポート宛，8610 番/udp ポート宛，3756 番/udp ポート宛，8612 番/udp ポート宛，514 番/udp ポート宛，および 53 番/udp ポート宛のパケットについて Wireshark を用いて解析した．その結果，実際に次のような不審な通信を発見した．

(1) 機器の設定ミスが原因のものとみられる通信 (DNS に対する不審な問い合わせ)
(2) プリンタの探索行動
(3) SNTP プロトコルや Zabbix Agent プロトコルに対するポートスキャンとみられる通信

(1) については，Wireshark でパケットを分析した結果，DNS 問い合わせであることがわかった．その旨を対象機器の管理者に伝達することで DNS サーバの設定ミスが原因であることが判明し，問題解決に至った．(2) については，パケットのデータ部に MFNP(プリンタの MFNP ポートの意) の文字列が含まれていたため，プリンタと判断した．自宅等のネットワークプリンタのプライベート IP アドレスを設定した機器によるプリンタの探索行動が原因であることが推測される．機器の設定ミスの発生原因として考えられるのは機器管理者の知識不足によるものや，DNS サーバ管理者の設定ミスなどが考えられる．(3) については 1 つの送信元 IP アドレスから複数の連番の IP アドレスの同一ポートに対してアクセスを試みていたためポートスキャンであると推測される．

調査した 24 時間文の通信ログの中には同一送信元 IP アドレスから同一宛先 IP アドレスの異なる複数の宛先ポート宛の通信や，同一送信元 IP アドレスから異なる複数の宛先 IP アドレスの同じポート番号宛の通信が含まれていた．前者の場合，特定のサーバのポートスキャン，後者の場合は特定ポートを狙ったポートスキャンというリスクがある．具体的には，前者のケースは，1 つの送信元 IP アドレスから 1 つの宛先 IP アドレスの 48 種類の異なるポートに 4 回ずつアクセスしていたものがあった．また，後者のケースでは 1 つの送信元 IP アドレスから 33 種類の異なる宛先 IP アドレスの 8610 番ポートに，1 つの宛先あたりそれぞれ 2741 2743 回のアクセスがあった．

UDP については，コネクションレスであるため受け取ったパケットをそのまま解析すればよい．しかし，TCP のパケット解析を行うためにはについてはハンドシェイクを確立させ，ファーストパケットを受け取る必要があり，そのためには何らかの応答を返す必要がある．それを実現するために，本研究ではハニーポットを用いる．

4. 組織内からダークネット宛通信の解析の提案手法

図 2 に組織内からダークネットへの通信を解析するために用いた提案手法のネットワーク構成図を示す．この図では，次の 2 の外部ホストのリストを用いる．

調査候補リスト ダークネット宛に多数の通信を行った外部ホストを自動的に登録する．

調査対象リスト ハニーポット等による詳細な解析を行う外部ホストをネットワーク管理者が外部ホストの管理者の同意を得た上で手動で登録する．

また，以下に動作順序を示す．

(1) 組織外および組織内の全ての送信元からダークネット宛の通信は組織内コアルータよりソフトウェアルータ VyOS[6] へ転送される．
(2) VyOS では組織外からの通信は組織外用ハニーポットへと転送する．組織内からの通信については調査対象リストを参照し，調査対象であれば，組織内用ハニーポットサーバに転送，そうでなければ調査対象機器選定サーバに転送する．
(3) 調査機器対象選定サーバは，4.1 節で述べる分類アルゴリズムを基に不審な通信を行っている調査候補リストを作成する．
(4) 組織内用ハニーポットサーバでは Honeyd で擬似応答を返し，送信元との一連のパケットを収集し Snort を用いて不適切通信を行っているアプリケーションの特定を行う．

調査対象機器選定サーバへ送られたパケットは，5 タプル (送信元 IP アドレス，送信元ポート番号，宛先 IP アドレス，宛先ポート番号，プロトコル番号) の通信ログを記録した後に廃棄する．本研究では，その通信ログを分析することで調査候補リストを作成する．さらにその機器の利用者の同意を得て，その機器を調査対象リストに登録する．調査対象リストにある外部ホストからの通信に対しハニーポットを用いて疑似応答を返し，それら一連のパケットを分析することで不適切通信の原因を特定する．本研究ではポートスキャンや設定ミスを引き起こしているアプリケーションの特定に Snort を用いる．

本研究では Niels Provos らによって開発されたオープンソースのハニーポットソフトウェア Honeyd[7] を用いる．Honeyd は Honeyd を動作させているホストの IP アドレス宛ての通信以外の通信に対しても応答可能である．

4.1 通信ログを用いた調査候補リストの作成

本研究では，通信ログを用いて調査対象候補機器のリストを作成する．この時，単位時間あたりの通信数と，宛先ダークネット内サーバ数の 2 つのパラメタを用いる．単位時間あたりに宛先ダークネット内サーバ数等に関係なく多数の通信を行っていた場合，調査対象候補リストに加える．また，ある外部ホストから多数のダークネット内サーバに対して通信があった場合は単純なポートスキャン攻撃であると推測できる．また，ある外部ホストから少数のダークネット内サーバに対して，多数のポートに通信をしている

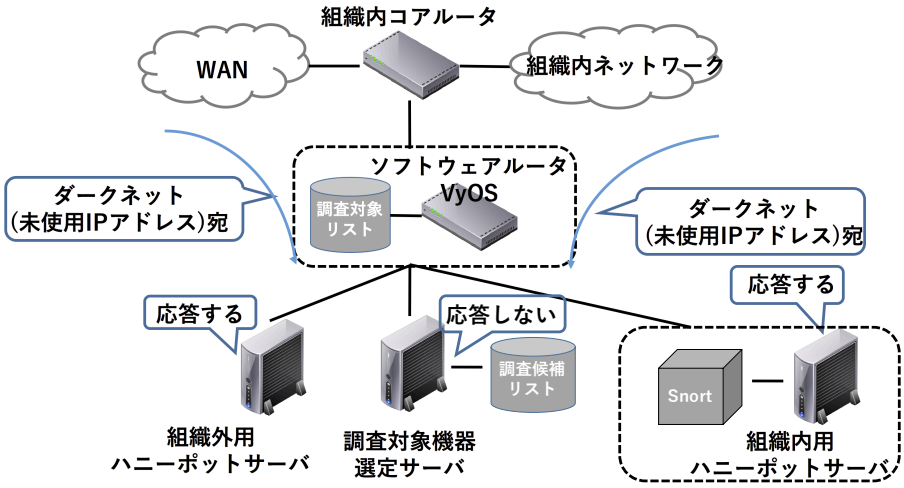

図 2 提案手法のネットワーク構成図

場合はポートスキャン攻撃であると推測できる．これらのポートスキャンをしている機器も，リストに加える

Algorithm 1 に調査対象候補機器の分類アルゴリズムを示す．このアルゴリズムでは，ある一定時間 T で区切り，

Algorithm 1 Rule of Classification

```
T ⇐ unit of time
t_pktn ⇐ threshold of pktn{total number of packets per T}
t_dsthn ⇐ threshold of dsthn{total number of original destination hosts }
t_dstpn ⇐ threshold of dstpn{total number of original destination ports }
tstart ⇐ start time
tend ⇐ end time
n ⇐ number of original hosts per T
while tstart < t < tend do
    for i = 0 to n do
        pktn[i] + +{total number of packets per T of host i}
        dsthn[i] + +{total number of original destination hosts per T of host i}
        dstpn[i] + +{total number of original destination ports per T of host i}
        if pktn[i] > t_pktn then
            Target device
        else if dsthn[i] > t_dsthn then
            Scan
        else if dstpn[i] > t_dstpn then
            Port scan
        end if
    end for
end while
tstart ⇐ tstart_new
tend ⇐ tstart_new + T
```

その領域をスライドさせながらその間のパケットを順繰りに調査していく．ある外部ホストから受信した一定時間の間のパケット総数を $pktn$ とする．調査対象候補機器と認定するためのパケット総数のしきい値を t_pktn とし，$pktn$ が t_pktn を超えた場合，その外部ホストを調査対象候補機器と認定する．ある外部ホストから受信した一定時間の間のダークネット内サーバ数を $dsthn$ とする．調査対象候補機器と認定するための宛先ダークネット内サーバ数のしきい値を t_dsthn とし，$dsthn$ が t_dsthn を超えた場合，そのホストを調査対象候補機器と認定する．ある外部ホストから受信した一定時間の間の宛先ダークネット内サーバのポート番号数を $dstpn$ とする．このポート番号は，重複を含まないポート番号である．調査対象候補機器と認定するための宛先ポート番号数のしきい値を t_dstpn とし，$dstpn$ が t_dstpn を超えた場合，そのホストを調査対象候補機器と認定する．ある一定時間 (tstart~tend) の調査が終了すると，tstart に tend の直後の時刻 tstart_new を代入する．tend には tstart_new に T を加算したものを代入する．これらのしきい値は，ネットワーク管理者が処理可能な件数のアラートをあげるようなものになるよう，実験的に決定する．

4.2 スキャンの通知

4.1 節で述べたアルゴリズムでは，ポートスキャンの検知までを行い，アプリケーションの特定は行わない．ポートスキャンを検知すれば，ネットワーク管理者は対象機器の利用者にポートスキャンを行っている旨と，ウイルス感染の可能性を通知し，利用者にウイルススキャンの実行を依頼する．

4.3 機器の設定ミスの特定

本研究では、前節で述べた通り，ハニーポットにより，疑似応答させることにより，不適切な通信のアプリケーションを特定する．特定のプロトコルでは，プロトコルに従って応答させることにより，原因の特定を行う．それ以外の TCP 通信については，ハニーポットでは，通信相手より送られてきた SYN パケットに対して，SYN-ACK パケット

```
1 alert tcp $INTERNAL any -> any 10050 (
    msg:"ZABBIX Agent"; content :"ZBXD"
     ; sid:1000000;)
2 alert tcp $INTERNAL any -> any 80 (msg
    :"HTTP"; sid:1000001;)
3 alert tcp $INTERNAL any -> any 22 (msg
    :SSH; sid:1000002;)
4 alert tcp $INTERNAL any -> any 10050 (
    msg":Canon MFNP "Port; content ":"
    MFNP ; sid:1000003;)
```

図 3 Snort のルール (一部)

を返信する．これにより通信相手は，その応答として ACK パケットを送信してくる可能性がある．この ACK パケットを解析することにより，アプリケーションの特定を試みる．なお，ACK パケットを受け取ったハニーポットはすぐさま FIN パケットを送る．

本研究では，これらの一連の通信を，Snort やファイアウォール製品の Intrusion Detection System (IDS) により分析を行う．

Snort[8] とはネットワーク型 Intrusion Detection System/Intrusion Prevention System であり，予め用意した攻撃パタンと一致したパケットを観測するとアラートをあげる．パタンは予め用意されたセットのほか，ユーザが作成することが可能である．特定のアプリケーションが発生させるパケットを解析し，それらを識別するためセットを準備することにより，アプリケーションの識別が可能となる．ファイアウォールには，非常に多くのアプリケーションを識別できるものもある [9]．

本研究では，ソフトウェアルータとハニーポットサーバとの間の通信を IDS により監視することによって，通信を発生させているアプリケーションの識別を行う (図 2)．

表 3 および表 4 に示したアプリケーションを識別するための Snort のルールの一部を図 3 に示す．このルールでは Zabbix Agent については，パケットに ZBXD の文字列が含まれるため [10]，10050 番ポート宛でなおかつ ZBXD の文字列が含まれるものを Zabbix Agent と認定している．しかしながら，MFNP のアプリケーションが発生するパケットにおいて必ず MFNP の文字列が含まれているかどうかの調査は終了していない．

4.4 ハニーポットによる不適切通信の原因の特定

調査対象リストに登録されている外部ホストからの通信に対し，ハニーポットで調査対象機器に対して擬似応答を返し，それら一連のやりとりのパケットを分析することで通信の発生原因を特定する．たとえば，80 番ポート (HTTP)，および 443 番ポート (HTTPS) 宛の通信に関しては，HTTP リクエストの Host:ヘッダにはリクエスト先サーバ名が設定される [11]．ハニーポットから DNS サーバへ Host:ヘッダの内容を問い合わせる．もし Host:タグの FQDN に対応する DNS 問い合わせの回答の IP アドレスが存在しない場合，クライアントは実在しない FQDN へアクセスを試みている，すなわち，クライアントの設定ミス，またはアクセスミスであるといえる．一方，Host:タグの FQDN に対応する DNS 問い合わせの回答の IP アドレスが存在する場合，クライアントは実在する FQND へアクセスを試みているのに未使用 IP アドレスへルーティングされる，すなわち，DNS サーバの設定ミスとなる．このような DNS の設定ミスを検知する方法は，パケット内にアクセス先の DNS 情報を含むプロトコルである SMTP においても検知可能となる．

また，HTTP リクエストの User Agent:ヘッダを見ることでブラウザアプリケーションの識別も可能である．

4.5 システムの評価

作成したシステムをネットワーク管理者に実際に利用してもらい，不適切な通信の発生源の特定，利用のしやすさや検知の精度について評価を得る．

5. 関連研究

本研究と関連する，ハニーポットを用いた不適切通信の検知の研究として Young Hoon Moon らの研究 [12] がある．この研究では，マルウェアを収集するモジュール 1 と収集したマルウェアの振る舞いを分析するモジュール 2 の，2 つのモジュールを用いる．モジュール 1 では不審な URL や IP アドレス，不審なバイナリファイルを収集する．モジュール 2 ではまず既知のマルウェアか否かを調査し，その後 Windows OS の VM で振る舞い検査を行う．その結果をプレ検知マネジメントシステムに保存する．

Young Hoon Moon らの研究ではマルウェアの検知のみを行っているのに対し本研究では設定ミスや管理者オペレーションなどの悪意のない通信についても検知を試みている点が，相違点である．

6. おわりに

本稿ではダークネット宛の通信に着目することで組織内ネットワークの不適切利用を検知する手法について述べた．まずは通信ログ (ヘッダ) のみを用いて不適切通信を行っている調査対象候補機器を特定し，その機器の利用者の許可を得た上でハニーポットを用いて一連のパケットを収集分析することで原因を特定する．

今後の課題としては，まず 3.1 節で述べた，調査対象候補機器選定のためのしきい値を決定することがあげられる．また，不適切通信の原因となるアプリケーションの特定精度の向上もあげられる．

参考文献

[1] 佐藤聡, 佐藤聖, 中井央, 新城靖. TLS/SSL プロトコルを対象とした 汎用ハニーポットシステムの実装と HTTPS による収集結果. 情報処理学会研究報告. IOT, インターネットと運用 技術, Vol.2015-IOT-29, No18, pp1-8, May 2015.
[2] Heartbleed Bug. http://heartbleed.com/, accessed: 2017/04/17.
[3] Vtiger CRM — Software for Sales, Support and Marketing. https://www.vtiger.com, accessed 2016/1/12.
[4] SAP Software Solutions — Technology & Applications. http://go.sap.com/index.html, accessed 2016/1/12.
[5] Online TCP UDP port finder - adminsub.net. http://www.adminsub.net/tcp-udp-port-finder, accessed: 2017/4/14.
[6] VyOS. http://vyos.net/wiki/Main Page, accessed: 2016/11/19.
[7] Developments of the Honeyd Virtual Honeypot. http://www.honeyd.org/, accessed: 2016/11/19.
[8] Snort - Network Intrusion Detection & Prevention System. https://www.snort.org/, accessed: 2016/11/19.
[9] paloalto NETWORKS. https://www.paloaltonetworks.jp/technologies/appid, accessed:2017/04/17.
[10] Docs/protocols/zabbix agent/2.0 - Zabbix.org. https://www.zabbix.org/wiki/Docs/protocols/zabbix_agent/2.0, accessed 2017/04/17.
[11] Hypertext Transfer Protocol – HTTP/1.1 https://tools.ietf.org/html/rfc2616, accesse 2017/04/17.
[12] Young Hoon Moon, Eunjin Kim, Suh Mahn Hur, and Huy Kang Kim. “Detection of botnets before activation: an enhancedhoneypot system for intentional infection and behavioral observation of malware”, Security and Communication Networks 2012.