Vol.2018-IOT-40 No.31
2018/3/6

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

山門 彩[1]　佐藤 聡[2,3,a)]　新城 靖[3]

概要：組織内発ダークネット宛通信の要因は様々であるが，それが問題になる場合がある．そのような通信を選別し，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，それは管理者にとって手間がかかる．本研究ではその作業の自動化を提案し，管理者を支援することを目的とする．まず，組織内ネットワークからダークネット宛通信のログ (ヘッダのみ) を異なり数分析で分析し，閾値を超える通信数を行っている機器を，不適切通信を行っている調査対象候補機器としてネットワーク管理者に通知する．さらに，調査対象候補機器の利用者にペイロード取得および分析の同意を得たのち，アプリケーションプロトコルから受け取る最初のメッセージを収集および分析することで不適切通信の原因を自動的に特定する．筑波大学学内ネットワークにおいて実際にシステムを運用した結果，管理者の仕事のうち利用者の特定，ペイロード解析許可の取得，ペイロード調査によるアプリケーション特定に対する支援につながることがわかった．

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，その中には利用されていない IP アドレス領域 (ダークネット) も存在する．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ブラックホールサーバへ転送され，破棄されていた．この際には，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．組織外を発信元とするパケットは攻撃だと思われる [1] が，組織内を発信源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．文献 [2] の予備調査では実際に機器の設定ミスが原因の通信やプリンタの探索行動，ポートスキャンとみられる通信を発見した．

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，ネットワーク管理者にはそれらを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，従来のネットワーク管理では，パケットヘッダの情報と同じ情報だけを含む通信ログしか利用できない．ネットワーク管理者も機器の利用者も，通信ログだけでは発生原因の原因を特定することは非常に困難である．ネットワーク管理者も利用者も，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．従来のネットワーク管理では，必要な場合，ネットワーク管理者がその通信のパケットを解析し，パケットの情報からアプリケーションを特定しなければならない．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．

そこで，本研究では組織内ダークネット宛通信のアプリケーションの特定や不適切通信の通知を自動化をするこ

[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 Systems 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 システム情報工学系
Faculty of Engineering, Information and Systems, University of Tsukuba
[a)] akira@cc.tsukuba.ac.jp

とで，ネットワーク管理者を支援する．これにより，不適切な通信に早期に対処し，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と IDS(Intrusion Detection System) の Snort 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異なり数分析で分析し，閾値を超える通信数を行った機器を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として通知する．そしてネットワーク管理者は調査対象候補機器の利用者に対してペイロード取得および分析の同意を得る．本研究ではアプリケーションプロトコルに従って交わされる最初のメッセージ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自動的に特定する．管理者はその特定結果を利用者に伝え，利用者自身が不適切通信を止める．このような自動化により管理者の負担を軽減する．

2. 組織内発ダークネット宛通信分析の自動化の設計と実装

2.1 本研究が対象とするネットワーク

本研究の対象は，筑波大学の学内ネットワークである．従来，筑波大学ではダークネット宛通信は組織内コアルータからパケットを破棄するためのブラックホールサーバへとルーティングしヘッダ情報のみを保存しパケットを破棄していた．破棄する際には，1 章で述べた通信ログを記録している．

本研究において，ネットワークの**利用者**とは組織内ネットワークを利用する者，**管理者**とは組織内ネットワークを管理する者であると定義する．筑波大学ではプライバシ保護の為の規則である“国立大学法人筑波大学情報セキュリティ実施要領”で利用者に無断で通信を調査することを禁止している．しかし，アプリケーションの特定にはペイロードを調査する必要がある．そこで，本研究では利用者より同意を得られた発信元 IP アドレスからの通信に限定してペイロードを調査する．

2.2 ダークネット宛通信発生時の管理者の仕事

本研究では，ダークネット外にあるクライアントがダークネット内のサーバにアクセスすることを想定する．このクライアントを**ダークネット外クライアント**と呼ぶ．ダークネット外クライアントからダークネット宛通信が生じた場合，管理者は次のような作業を行う．

(1) 管理者は，パケットのヘッダの発信元 IP アドレスを調査し，利用者を特定する．

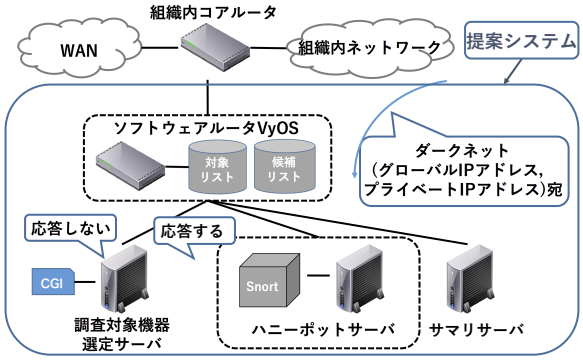

図 1 提案手法のネットワーク構成図

(2) 管理者は，利用者にダークネット宛通信が生じていることを通知し，調査のためのペイロード解析の許可を得る．また，ウイルススキャンを依頼する．

(3) 管理者はパケットのペイロードの調査を行い，アプリケーションを特定する．

(4) 管理者は，利用者にアプリケーションを通知する．

最終的に利用者が機器の設定を変更し，問題を解決する．本研究では，これらの管理者の作業のうち，(1) と (2) と (3) を支援するような仕組みを実装する．

2.3 ダークネット宛通信分析および利用者への通知

図 1 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るシステムの構成を示す．この図では，次の 2 つのダークネット外クライアントのリストを用いる．

候補リスト ダークネット宛に閾値を超える数の多数の通信を行ったダークネット外クライアント．システムが自動的に登録する．

対象リスト ハニーポット等による詳細な解析を行うダークネット外クライアント．利用者がペイロード取得に対する同意の意思表示を行うため，本研究で実装した Web ページの URL(以下：**同意 URL**) から同意ボタンを押すことで登録する．

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[*1] へ転送される．

(2) VyOS では組織内からの通信について，対象リストを参照し，それに含まれてい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，1 章で述べた通信ログを記録する．また，2.4 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 TCP に関しては Honeyd で TCP コネクションを確立させるための擬

[*1] http://vyos.net/wiki/Main Page

発信元IPアドレス	宛先IPアドレス	宛先ポート番号
192.0.2.100	192.0.2.200	45431
192.0.2.100	192.0.2.201	45431
192.0.2.100	192.0.2.202	45432
192.0.2.100	192.0.2.203	45432

異なり数分析

—	4	2

図 2 通信ログに対する異なり数分析

似応答を返す．発信元からアプリケーションプロトコルの最初のメッセージを収集し Snort を用いてアプリケーションの特定を行い，その結果を管理者に通知する．

管理者には．アプリケーションの特定結果を利用者に通知する．利用者は不適切通信であれば，それを止める．このようにして，不適切通信を減少させ，セキュリティリスク低下につなげる．

2.4 通信ログを用いた候補リストの作成

2.1 節で述べた規則を遵守しながら，候補リストを作成するために**異なり数分析**という手法を用いる [3]．異なり数分析とは，任意に選択した項目が一致する部分集合，すなわち集約フローを作成したのち一致しない項目の種類の値を計測する分析手法である．本節では，通信ログから異なり数分析により候補リスト作成方法を述べる．

図 2 に通信ログの例を示す．図 2 では通信ログの中から発信元 IP アドレスが一致する部分集合を取り出した時の他の項目，つまり宛先 IP アドレスと宛先ポート番号の異なり数を求めている．発信元 IP アドレスが一致する部分集合をとったとき，宛先 IP アドレスの種類の値は 4 であるので，宛先 IP アドレスの異なり数は 4 である．また，同様に，宛先ポート番号の種類の値は 2 であるので，宛先ポート番号の異なり数は 2 である．

異なり数分析では異なり数の「閾値」を定め，異なり数がその閾値を超えたものについて焦点を当てる．文献 [3] では宛先ポート番号の異なり数と宛先 IP アドレスの異なり数の組み合わせで通信を分類する方法を提案している．

本研究で異なり数分析の結果，発見できる不適切通信を表 1 に示す．本研究では発信元 IP アドレスを固定した時の，他の項目の固定の組み合わせで通信を 4 種類に分類する．4 種類のうち，3 種類については自動的に不適切通信であると判定できる．「固定」は集約フローを作成するために選択する項目を表し，「—」は一致しない項目の種類の値を計測する項目を表す．なお，本研究では，従来の異なり数分析の手法を拡張して，表 1 の 4 種類のうち特定の物に対しては，閾値に到達するまでに要した時間を考慮する．“時間の考慮”の項目において“◯”は“考慮する”，“×”は“考慮しない (従来と同じ)”の意である．

表 1 の項番 1 は，同一発信元 IP アドレスから同一宛先 IP アドレスの同一宛先ポート番号に閾値を超える多数の通信が観測された場合に該当する．表 1 の項番 2 は，同一発信元 IP アドレスから同一宛先ポート番号の，閾値をこえる多数の宛先 IP アドレス宛に通信が観測された場合に該当する．表 1 の項番 3 は，同一発信元 IP アドレスから閾値をこえる多数の宛先 IP アドレスや宛先ポート番号宛に通信が観測された場合に該当する．表 1 の項番 4 は，同一発信元 IP アドレスから同一宛先 IP アドレス番号の，閾値をこえる多数の宛先ポート番号宛に通信が観測された場合に該当する．これは，ポートスキャンである．表 1 の項番 1，2，4 の通信に関しては，閾値に到達するまでに長時間要したとしても通常のネットワーク利用で発生するとは考えにくい通信であるため，閾値到達までの時間は考慮せずに不適切通信と認定し，候補リストに追加する．

本研究で従来の異なり数分析の手法を拡張して閾値に到達するまでに要した時間時間を考慮する．その目的は表 1 における項番 3 のケースに対応することである．項番 3 については，ポートスキャンでも特定アプリケーションのスキャンでも無く，自動的に不適切通信とは認定できない．しかし，短時間で閾値に到達した場合，何らかの問題が発生している可能性が考えられるので，その場合にのみ候補リストに追加する．閾値に到達するまでに要した時間が長い場合，候補リストには追加せず無視する．ある発信元 IP アドレスが複数の項番に該当する場合，最初に該当したものに対して管理者に通知が届く．

異なり数分析では，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．

図 3 に候補リスト作成および利用者への通知の流れを示す．ソフトウェアルータ Vyos の通信ログには候補リスト，対象リスト，その他のダークネット外クライアントから学内ダークネット宛通信が含まれる．その通信ログを rsyslog を用いて調査対象機器選定サーバへ転送する．調査対象機器選定サーバでは，それを異なり数分析を行うプログラム Vicar へ渡すための形式に変換する．通信ログを syslog_parser.sh で整形し，それを Vicar へ渡す．

Vicar はアイテム集合 (発信元 IP アドレス，宛先 IP ア

表 1 異なり数分析の結果発見できる不適切通信

項番	発信元 IP アドレス	宛先 IP アドレス	宛先ポート番号	分類	時間の考慮
1	固定	固定	固定	特定アプリケーションによる特定ホストへの通信	×
2	固定	—	固定	特定アプリケーションによるスキャン	×
3	固定	—	—	ネットワーク管理者による調査が必要な通信	○
4	固定	固定	—	マルウェアによるポートスキャン	×

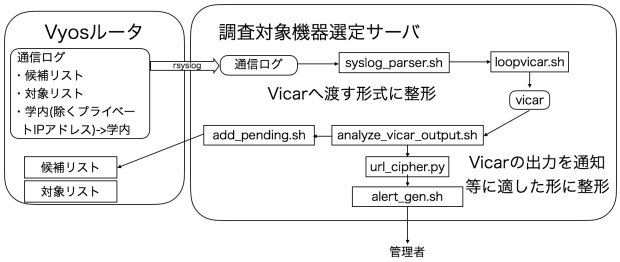

図 3 候補リスト作成および利用者への通知

```
1  alert udp $INTERNAL any -> any 53 (msg
       :"DNS MX"; content:"|000f0001|";
       nocase; sid:2000012;)
2  content:"|a2|"; nocase; sid:2000026;)
3  alert udp $INTERNAL any <> any 161 (
       msg:"SNMP Set-Request"; content
       :"|020100|"; content:"|a3|"; nocase
       ; sid:2000027;)
```

図 4 アプリケーションを特定するための Snort のルール (一部)

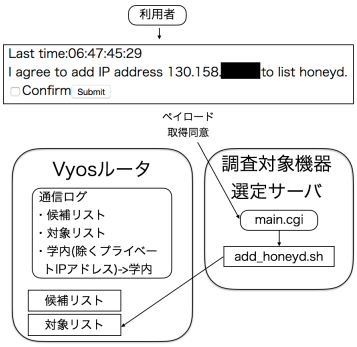

図 5 対象リスト作成

ドレス，宛先ポート番号などの組) の数え上げアルゴリズムを実装しているプログラムである．本研究では Vicar を一部改変し，ログに含まれる時刻も出力する．ログ中の時刻フィールドの位置を予め引数として受け取り，アイテム集合を読む際に，時刻フィールドを読み込み，時刻を記録する．アイテム集合の出現数が閾値に到達した時に記録した時刻と，そのアイテム集合が最初に入力された際に記録した時刻との差を異なり数と一緒に出力する．Vicar の出力結果を analyze_vicar.sh で分析したり利用者へ通知するスクリプトやルータの候補リストへ追加するスクリプトを呼び出す．

2.5 アプリケーションの特定

本研究では，TCP と UDP を調査対象とする．TCP 通信については，ハニーポットソフトウェア Honeyd[*2] を用いることで，TCP のコネクションを確立させる．次に，クライアントからアプリケーションプロトコルの最初のメッセージが送られてくると，それを保存し，TCP コネクションを切断する．UDP についてはコネクションレスであるため，そのままアプリケーションプロトコルの最初のメッセージを解析する．HTTP や DNS などのアプリケーションを擬態するためのプログラムは使用しない．本研究では TCP および UDP の全てのポートで通信を待ち受ける．

本研究では，アプリケーションプロトコルの最初のメッセージの分析を Snort により行う．Snort[*3] 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ことができる．パタンは付属のもののほか，ネットワーク管理者が作成することが可能である．Snort はポート番号やペイロードの内容に応じて細かくルールを設定できる利点がある．パケット解析時間やドキュメントの豊富さの観点より，本研究では Snort を採用する．

本研究で作成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 4 に示す．このルールでは DNS と SNMP のリクエストを特定するための記述をしている．1 行目の DNS のルールでは，「組織内のグローバル IP アドレスから任意の UDP53 番ポート宛通信で，ペイロードに 0x000f0001 が含まれていれば DNS の MX レコードに対するクエリパケット」を識別するための記述をしている．このように，宛先ポート番号と，ペイロード中に含まれる文字列などに着目してルールを作成することでアプリケーションやそれに対する詳細な識別が可能である．本研究では，管理者は簡単に Snort ルールを追加していくことで以後のアプリケーション特定を効率的に行えるようにする．また，分析時点で Snort のルールで定義されていないパケットについては TCP は”TCPANY”，UDP は”UDPANY”というルールを当てはめる．

本研究では，Snort の分析結果のサマリを管理者に示し，ダークネット宛通信の原因の特定を手助けする．サマリは，Snort の alert ログから日時，発信元 IP アドレス，宛先 IP アドレス，宛先ポート番号，Snort で識別したアプリケーションのみを切り出し，発信元 IP アドレスをキーにソートした後，重複を取り除いて作成する．つまり，発信元 IP アドレスでソートされた，日時，発信元 IP アドレス，宛先ポート番号，アプリケーション名，宛先ポート番号の組が重複無く含まれるものである．

Snort の分析結果のサマリは一定時間ごと (例：1 時間ごと) に調査対象機器選定サーバへ転送され，管理者がアクセス可能なページに表示可能にする．

図 5 に対象リスト作成の流れを示す．なお，以下筑波大学の IP アドレスを表記する際には第三オクテット以下を伏せる．候補リスト追加通知メール (以下：**通知メール**を受け取った利用者が同意 URL をクリックすると，図 5 に

*2 http://www.honeyd.org/
*3 https://www.snort.org/

示したような URL の有効期限と候補リストに掲載された IP アドレスを対象リストに追加することに同意するか否かを決定するためのページを表示する．もし利用者が同意ボタンを押した場合，IP アドレスを対象リストに追加するためのスクリプトが呼び出され，それにより IP アドレスが対象リストに追加される．

3. 評価

3.1 実験環境

作成したシステムを筑波大学の学内ネットワークで運用し，不適切通信の検知や機能・性能評価を行った．実験対象となるダークネットは未使用グローバル IP アドレスおよびプライベート IP アドレスである．管理者に通知メールを送信し，ペイロード調査結果から不適切通信の原因特定を行った．

3.2 実験結果の概要

2017 年 12 月 9 日に作成したシステムの運用を開始し，2018 年 1 月 31 日現在まで運用を継続している．集計日の 2018 年 1 月 12 日時点で候補リストには 1,542 個の IP アドレス，対象リストには 107 個の IP アドレスが含まれる．管理者への通知メールは候補リストの件数と同様，1,542 件であった．アクセスされたダークネット内サーバの内グローバル IP アドレスが 180 個，プライベート IP アドレスが 21,414 個であった．2017 年 12 月 19 日から計測を開始したパケット数は，全パケットが約 2 億 478 万件であり，そのうちハニーポットへルーティングされたものは約 100 万件であった．

2018 年 1 月 12 日にソフトウェアルータ VyOS で観測された 24 時間分のパケット内，発信元 IP アドレスが候補リスト掲載のものは約 890 万個，対象リスト掲載のものは約 7 万 4 千個，両リスト外のものは約 1 万 6 千個であった．

2018 年 1 月 12 日にハニーポットサーバで観測されたダークネット内サーバ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 2 にダークネット内サーバの 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を，表 3 に UDP のそれの上位 6 位を示す．サマリの件数は 3,711 件であり，そのうち Snort ルールによって識別できたのが 2,117 件であった．識別率は約 57%である．これにより 2.2 節に示した管理者の仕事のうち 3 を支援することが出来た．特定できなかった通信のうちの殆どが動的プライベートポート番号 (49152-65535) であった．

また，図 6 に 2017 年 12 月 9 日から 2018 年 1 月 12 日までの 1 日ごとの候補リストに含まれる IP アドレス数の推移を示す．図 6 より，集計開始後 5 日は 1 日に 100〜200 件のペースで増加し，その後は徐々に増加数が減少していくことが分かる．また，筑波大学の休業日付近の 12 月 29 日から 1 月 8 日の間は増加数が 20 件に満たないが，授業開始日の 1 月 9 日以降は増加数が 40 件程度に増加している．

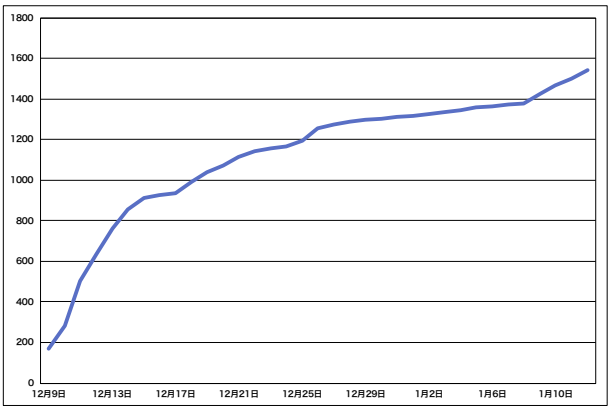

図 6 日毎の候補リストの IP アドレス数の推移

3.3 発見された不適切通信の分析

実際に発見された不適切通信のうち，利用者から同意を得てペイロード調査を行ったものを以下に示す．

- 特定プライベート IP アドレス宛通信
- Microsoft-DS Active Directory に対する通信
- DNS サーバ宛通信

この事例では 2 章で述べた仕組みの，通信ログを用いた候補リストの作成，対象リストの作成および Snort によるアプリケーション特定が有効に動作した．このうち，最初のものは Snort にルールがなくアプリケーション特定を自動的には行うことができなかった．

3.3.1 候補リスト作成

2.4 節で述べたように，本研究では Vicar を用いて候補リストを作成することが出来た．これにより，2.2 節に示した管理者の仕事のうち 1 と 2 を支援することが出来た．本研究では Vicar の閾値を暫定的に 500 件とした．適切な設定値を定める必要がある．この閾値を調整することでより通信数の多いもののみを抽出したり，その逆も行うこと

表 2 ダークネット内サーバ TCP 宛先ポート別通信数

ポート番号	アプリケーション	パケット数	割合 [%]	ダークネット外クライアント数
80	HTTP	158	56.0	5
8014	Continuation or non-HTTP	113	40.1	25
445	Microsoft-DS SMB file sharing	8	2.84	3
8080	Continuation or non-HTTP	3	1.06	1
その他	—	0	0	0

表 3 ダークネット内サーバ UDP 宛先ポート別通信数上位 6 位

ポート番号	アプリケーション	パケット数	割合 [%]	ダークネット外クライアント数
8610	Canon MFNP Service	37,713	55.1	2
53	DNS	28,395	41.5	15
514	Syslog	868	1.27	8
161	SNMP	850	1.24	9
123	Network Time Protocol	435	0.636	8
138	NetBIOS Datagram Service	120	0.175	8
その他	—	11	0.0161	3

が可能である．最適な閾値を求めることは，今後の課題である．

3.3.2 対象リスト作成

2.5 節で述べたように，利用者が通知メールのリンクをクリックすることで対象リストに当該 IP アドレスを追加することが出来た．これにより 2.2 節に示した管理者の仕事のうち 2 の利用者から許可を得る作業の支援が出来た．対象リスト作成は通知メールのリンクより 1 つずつ行うものであったため，数が多い場合には手間がかかる．今後．複数アドレスを一度に登録したり，ネットワークアドレスで登録可能に拡張したいと考えている．

4. 関連研究

本研究と同様，組織内からのダークネット宛通信分析の研究として Song らの研究 [4] がある．この研究では，IDS アラートログに含まれている発信元ホストのうち，ダークネットに 対しても通信を行ったホストに着目する. そのホストについて IDS アラートの種類とウィルススキャンの結果を比較し, 相関を考察している．Song 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．

ネットワーク管理者を支援することを目的とした 2 本の研究がある．高橋ら [5] は能動的情報資源 (Active Information Resource：AIR) の概念に基づくネットワーク管理支援システム (AIR-based Network Management Support System：AIR-NMS) を用いて，様々な障害への対策案をネットワーク管理者に提示する方法を提案している．Kim ら [6] はテレコミュニケーション管理ネットワーク (Telecommunication Management Network：TMN) とタイムトリガ型メッセージトリが型オブジェクト (Time-triggered Message-triggered Object：TMO) モデルを統合したリアルタイム分散ネットワーク管理 (Real-time Distributed Network Management：RTDNM) というネットワーク管理システムを提案した．これら 2 本の論文ではシステムが管理者に対策案を提案し，それを管理者が実行する．本研究ではネットワーク管理者がシステムに対して行っていた行動を自動化し，ネットワーク管理者は利用者とのやりとりを主に行う．

5. まとめと今後の課題

本稿では，ダークネット宛の通信ログとハニーポットの Honeyd と IDS(Intrusion Detection System) の Snort を用いて組織内発ダークネット宛通信を自動的に分析する手法について述べた．作成したシステムを運用して実験を行った結果，いくつかの不適切通信を検知し，アプリケーションの特定を行うことができた．

今後の課題として，アプリケーションプロトコルを擬態するスクリプトをハニーポットに実装し, アプリケーションプロトコルの 2 つめ以降のメッセージも受け取ることが挙げられる．本研究ではハニーポットを TCP コネクション確立の目的のみで用いたが，アプリ ケーションによってはサーバ側が先にメッセージを送信するものもあるので，アプリケーショ ンプロトコルも擬態するハニーポットを用いることでより詳細なアプリケーション特定が行 えるという期待がある．また，Vicar の閾値の適切な値を決定することも課題である．

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, インターネットと運用技術 (IOT), Vol.2015-IOT-29, No18, pp1-8, May 2015.

[2] 山門彩, 佐藤聡, 新城靖. 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. 情報処理学会研究報告, インターネットと運用技術 (IOT), Vol.2017-IOT-37, No9, pp1-7, May 2017.

[3] Yusuke SHOMURA, Yoshinori WATANABE, Kenichi YOSHIDA. Analyzing the Number of Varieties in Frequently Found Flows. IEICE TRANS. COMMUN., VOL.E91-B, NO.6 JUNE 2008.

[4] Jungsuk Song, Younsu Lee, Jang-Won Choi, Joon-Min Gil, Jaekyung Han and Sang-Soo Choi. Practical In-Depth Analysis of IDS Alerts for Tracing and Identifying Potential Attackers on Darknet. Sustainability. Vol.9, No.2, pp.262, 2017.

[5] 高橋 優介, 三杉大輔, 高橋晶子, 笹井一人, 阿部亨, 木下哲男. 能動化された知識の組織化によるネットワーク障害管理支援方式. 情報処理学会研究報告書. CSEC, コンピュータセキュリティ, Vol.2010-CSEC-48, No.5, pp1-8, Feb 2010.

[6] M.H.Kim, S.Lim, and J.Kim. Modeling of Real-Time Distributed Network Management based on TMN and the TMO Model. Proceeding of the Eighth International Workshop on Object-Oriented Real-Time Dependable Systems. pp.56-63, Jan. 2003.