Computer Security Symposium 2017
23 - 25 October 2017

サイバー攻撃誘引基盤 STARDUST

津田 侑[1]　遠峰 隆史[1]　金谷 延幸[1]　牧田 大佑[1]　丑丸 逸人[1]　神宮 真人[1]　高野 祐輝[1]　安田 真悟[1]
三浦 良介[1]　太田 悟史[1]　宮地 利幸[1]　神薗 雅紀[1]　衛藤 将史[1]　井上 大介[1]　中尾 康二[1]

概要：標的型攻撃に代表される政府や企業等の組織を狙ったサイバー攻撃が社会問題となって久しい．このような攻撃に関連した各種ログ等には組織の機微情報が含まれることも多く，それらが組織外に公開されることは稀である．また，攻撃に用いられたマルウェアを解析しても，攻撃者による攻撃活動までは把握できず，攻撃の表層的な情報しか得られない．そこで，攻撃者の攻撃活動を迅速かつ詳細に把握するため，攻撃者を誘引して長期観測を可能にする観測基盤 STARDUST を提案する．STARDUST は組織を精巧に模倣したネットワーク環境である「並行ネットワーク」を数時間程度で自動構築でき，さらに誘引した攻撃者に察知されにくいステルス性の高い観測・分析を実現する．本稿では STARDUST の設計と実装を述べるとともに，いくつかの標的型攻撃キャンペーンについて，攻撃者誘引のケーススタディを示す．

キーワード：標的型攻撃対策，攻撃者誘引，並行ネットワーク

STARDUST: Large-scale Infrastructure for Luring Cyber Adversaries

Yu Tsuda[1]　Takashi Tomine[1]　Nobuyuki Kanaya[1]　Daisuke Makita[1]　Hayato Ushimaru[1]
Masato Jingu[1]　Yuuki Takano[1]　Shingo Yasuda[1]　Ryosuke Miura[1]　Satoshi Ohta[1]
Toshiyuki Miyachi[1]　Masaki Kamizono[1]　Masashi Eto[1]　Daisuke Inoue[1]　Koji Nakao[1]

Abstract: Cyberattacks, which target enterprise or government organizations, have been considered globally as a serious social problem. Although we need actual knowledge about these attacks to create practical countermeasures, it is difficult to obtain them because most of victims could not publicly disclose details of attacks (i.e., security logs). It is also hard to deeply understand actual behavior of human adversaries by only analyzing malware. Therefore, we develop STARDUST, which is an infrastructure enabling us to lure adversaries into a mimetic network and to monitor their whole behavior for long term. STARDUST can build an enterprise-scale mimetic network called "parallel-world network" within several hours, which is configured based on a real-world one. In addition, the parallel-world network has network-based and host-based monitoring functions to observe adversaries' activities. In this paper, we describe the design and implementation of STARDUST, and present several case studies of luring adversaries regarding to actual APT campaigns.

Keywords: Countermeasure against APT, Luring Cyber Adversaries, Parallel-world Network

1. はじめに

標的型攻撃に代表される政府や企業などの組織を狙うサイバー攻撃が社会問題となって久しい．このようなサイバー攻撃への対策技術を確立するためには詳細な知見を得る必要がある．そのひとつにセキュリティベンダが公開するレポートがある [1–3]．しかし，攻撃に関連した各種ログ等は機微情報が含まれることが多く，その詳細が組織外に公開されることは稀である [4, 5]．

一方で，サイバー攻撃で用いられたマルウェアを単一の仮想マシンやサンドボックス機器で解析することが一般的である．しかし，ここからは攻撃の表層的な面しか得られず，攻撃者が組織内に侵入した後の活動までは把握できない．また，特定の組織を狙ったサイバー攻撃を解析する場

[1] 国立研究開発法人 情報通信研究機構
National Institute of Information and Communications Technology

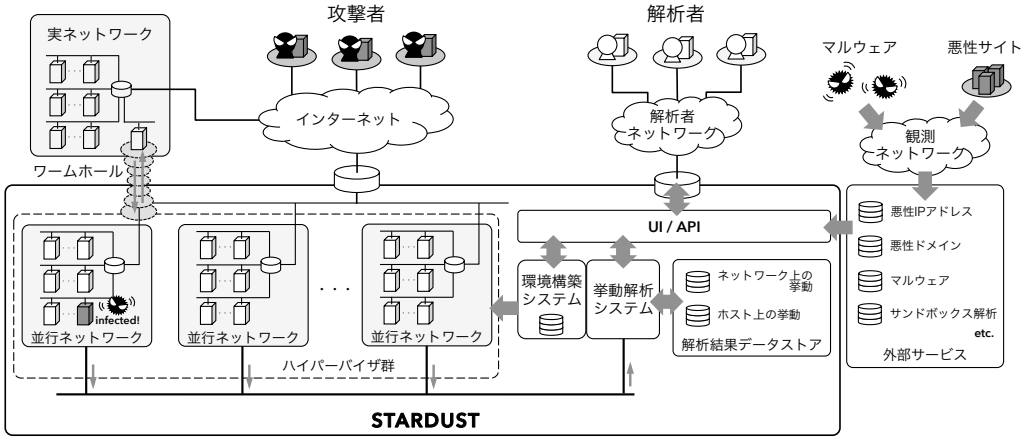

図 1　STARDUST のシステム概要

合，実際のネットワーク（以下，実ネットワーク）と解析環境の判別が攻撃者にとって容易である問題がある [6,7]，

そこで，攻撃者を解析環境内に誘引しその活動を長期に渡って観測可能な STARDUST を提案・実装する．STARDUST は解析者の用途に合わせ柔軟に解析用ネットワークを構築できる．解析用ネットワークは実ネットワークの設定や端末に導入されたアプリケーション，文書やメールなどまで含み，高精細な模倣ネットワーク（以下，並行ネットワーク）となる．また，並行ネットワークは解析ツールを別途導入せずに攻撃活動をリアルタイムかつ長期間観測できる機能を持つ．さらに，実・並行ネットワーク間を多段 NAT と VPN を利用したワームホールで接続することで，攻撃者の持つサーバとの通信が実ネットワークを介して発生し，実ネットワークと解析環境の判別を困難にする．

本稿では STARDUST の設計を第 2 章，実装を第 3 章で述べる．また，並行ネットワーク構築の性能評価を第 4 章に述べる．そして，STARDUST を用いたケーススタディを第 5 章に，それを基に第 6 章で考察する．最後に第 7 章に関連研究を述べ，第 8 章でまとめと今後の課題を述べる．

2. サイバー攻撃誘引基盤 STARDUST

2.1 システム概要

図 1 に STARDUST のシステム概要を示す．STARDUST では機能の追加や連携が容易な設計 [8] を適用する．

STARDUST を用いて解析する際に，まず解析者は環境構築システムを用いて並行ネットワークを構築する．並行ネットワークは複数の仮想マシンと仮想ネットワークから成り，そのネットワークトポロジは解析者自身で定義できる．仮想マシンには Windows や Linux，ソフトウェアルータといった OS が用意され，各々の仮想マシンでは，DHCP や DNS，Active Directory（AD）といったサービスが動作できる．また，実・並行ネットワーク間をワームホールで接続することで，攻撃者による実ネットワークと解析環境との判別を困難にする．

この後に解析者が並行ネットワーク上でマルウェアを実行し攻撃者の活動観測を開始する．観測結果は，挙動解析システムでネットワークおよびホストの視点で解析され，その結果は解析結果データストアに保存される．解析者は UI/API を介してリアルタイムに情報を取得できる．

本節のまとめに STARDUST を用いた解析の流れを示す．

(1) 初期解析: サンドボックス機器やマルウェア共有サービスでマルウェアの特徴を把握する．STARDUST は攻撃者を誘引して活動を長期に渡って観測することに焦点を当てており，より確実にマルウェアを並行ネットワーク上で動作させるために必要となる．

(2) 環境構築: 解析者はマルウェアの特徴や解析の要求に合わせて並行ネットワークを構築する．また，解析の状況に合わせて並行ネットワークを再構築できる．

(3) 挙動観測: 構築した並行ネットワーク上でマルウェアを実行する．観測するマルウェアおよび攻撃者の活動は逐次蓄積される．解析者は UI/API を介してリアルタイムに観測結果を閲覧できる．

次節以降では，環境構築システム，挙動解析システム，ワームホール，ユーザインタフェースの概要を述べる．

2.2 環境構築システム

環境構築システムでは 3 種類のテンプレートが用意され，解析者はその組み合わせで並行ネットワークを構築する．仮想マシンテンプレートは OS イメージとアプリケーションの組み合わせから成る．たとえば，“httpd” が導入された CentOS 7 や，“bind” が導入された Ubuntu 16.04 LTS のイメージなどがある．ネットワークテンプレートには，サブネットやその VLAN ID といったネットワーク情報を記述する．コンテンツテンプレートは，文書やメール，アプリケーションの設定といった並行ネットワークの模倣性を向上させるファイル群である．

また，並行ネットワーク構築のためのレシピを図 2 のように作成する．*vlan* にその名称と VLAN ID を記載する．

```
{
 "vlan": [
  { "ID": "server", "vlanid": "3133" },
  { "ID": "client", "vlanid": "3136" } ],

 "node": [
  { "ID": "dhcp",
    "hostname": "dhcp.example.com",
    "os": "CentOS6",
    "apps": [ "dhcp" ],
    "network": [ { "name": "eth0",
                   "vlan": "server",
                   "ipaddr": "10.140.8.6",
                   "netmask": "255.255.255.0",
                   "gateway": "10.140.8.254",
                   "dns1": "10.140.0.2" } ] },
  { "ID": "Win01",
    "hostname": "Win01.example.com",
    "os": "Win7x64",
    "apps": [ "office2010", "yarai" ],
    "contents": [ { "name": "重要書類",
                    "path": "C:\Users\tsuda" } ],
    "network": [ { "name": "eth0",
                   "vlan": "client",
                   "ipaddr": "dhcp" } ] } ]
}
```

図 2　並行ネットワーク構築のためのレシピの例

node には，並行ネットワーク上に設置する仮想マシンの設定（ID やホスト名，OS，導入するアプリケーション，コンテンツやネットワーク設定）を記述する．*network* の部分には，上述した *vlan* の指定や，IP アドレス，デフォルトゲートウェイ，DNS 等の情報を記述する．

さらに，並行ネットワークを再構築する際に，軽微な変更であれば並行ネットワーク全体を取り壊す必要はない．たとえば，解析者がインターネット接続に HTTP プロキシを利用しない場合は，プロキシサーバの停止およびソフトウェアルータの設定変更命令を投入するといった簡易的な手法で早急に解析者の要求を満たせる．

2.3　挙動解析システム

並行ネットワーク内でマルウェアを実行し攻撃者を誘引した際，STARDUST ではそれらの活動をステルス性の高い手法で観測する．挙動解析システムでは，ネットワーク上・ホスト上の両面から観測した情報を分析する．

2.3.1　ネットワーク上の活動の観測

並行ネットワーク上の仮想マシンは，攻撃者のサーバにインターネットを介して特に制限なく接続できる．そのため，攻撃者のサーバとマルウェア間の通信や悪性ツールの追加ダウンロード，並行ネットワークからのファイルアップロードなどの活動を観測できる．また，並行ネットワーク内におけるドメインコントローラやファイルサーバを探索するような攻撃者の活動も同様に観測できる．

STARDUST では，並行ネットワークに関する全てのネットワークトラフィックを仮想スイッチからミラーリングして保存する．このうち HTTP や DNS，ICMP といった解析作業で利用する主要なプロトコルを要約する．そのため，STARDUST を用いた解析をするときには解析者が膨大な pcap ファイルを全て閲覧する必要はない．

2.3.2　ホスト上の活動の観測

STARDUST は並行ネットワーク上の仮想マシン内部の情報を観測する機能も標準的に備える．STARDUST ではサーバホストとクライアントホストの 2 種類仮想マシンが提供されており，各々ホスト上の挙動を観測できる．

まず，Linux と Windows Server が搭載されるサーバホストでは OS 標準のログ収集機能を利用する．次に Windows が搭載されるクライアントホストにはエンドポイント対策製品を導入する．エンドポイント対策製品はクライアントホストの状態，たとえば起動中プロセスのツリー構造，ソケットやファイルの生成などが収集できる．これらのホスト上の活動の観測結果は，前節のネットワーク上の挙動観測結果と IP アドレスやドメイン名で関連付ける．

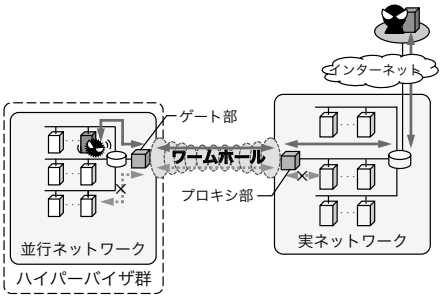

図 3　ワームホールの概要

2.4　ワームホール

攻撃者は実ネットワークと解析環境を判別し，攻撃活動を停止する [6,7]．STARDUST はワームホールを利用することで実・並行ネットワークを接続し攻撃者によるその判別を困難にする（図 3）．ワームホールの要求は以下の通りである．

- 並行ネットワーク上の仮想マシンは実ネットワークを介してインターネットに接続できる．
- 並行ネットワークでは実ネットワークの IP アドレスやデフォルトゲートウェイなどで同値の設定ができる．
- 並行ネットワーク上の仮想マシンは，実ネットワークのホストと接続できない．
- 並行ネットワークから実ネットワーク・インターネットへの攻撃活動を検知・遮断できる．

ワームホールはゲート部とプロキシ部の 2 つで構成される．ゲート部は並行ネットワーク内に，プロキシ部は実ネットワークに設置される．そして，インターネット接続のネットワークトラフィックはワームホールを介する．そのため，攻撃者のサーバには実ネットワークから接続する．さらに，実ネットワークの IP アドレスやデフォルトゲートウェイ等の情報を並行ネットワークに設定できる．

一方で，並行ネットワーク上での解析中に発生する通信が実ネットワークやインターネットに対して影響を及ぼす懸念がある．たとえば，DoS 攻撃やマルウェアの感染拡大が挙げられる．このような脅威に対応するためワームホールのゲート部に IDS/IPS を実装する．

表 1　仮想マシンテンプレート

OS	サービス・機能
Windows 7 Professional Windows 8.1 Windows 10 Pro	Microsoft Office 2010/2013/2016, エンドポイント対策製品, JRE, Adobe Flash/Acrobat, etc.
Windows Server 2008 R2	AD, ファイル共有
CentOS 6.5 CentOS 7.0 Ubuntu 12.04 LTS Ubuntu 14.04 LTS Ubuntu 16.04 LTS	HTTP (Apache), rsyslog, DHCP, HTTP プロキシ (Squid), POP, IMAP, SMTP (Postfix), DNS (BIND), FTP
VyOS 1.1.0	ルーティング, アクセス制御

2.5　ユーザインタフェース

解析者が簡便に STARDUST を操作できるように全機能を統合する UI を作成する．この UI では全ての並行ネットワーク上の仮想マシンの操作や解析結果の閲覧できる．また，解析フローを自動化するために API を備える．

3.　実装

3.1　環境構築システム

並行ネットワークは VMware vSphere [*1] と vSphere Distributed Switch [*2] を用いた仮想マシンと仮想ネットワークで構築した．これらを Alfons [9] を介して操作する．Alfons は模擬的なネットワークを高速に自動構築できるシステムである．

仮想マシンテンプレートには表 1 を利用できる．また，ネットワークトポロジは図 2 のようなレシピを記述することで設定できる．さらにコンテンツテンプレートを利用して Microsoft Office 文書や PDF ファイル，壁紙，ブックマーク，メールなどを仮想マシンに挿入できる．

3.2　挙動解析システム

ネットワーク上の活動の観測のために解析者が並行ネットワーク上の仮想マシンに設定やソフトウェアを加える必要はない．並行ネットワーク上のネットワークトラフィックは vSphere Distributed Switch でミラーリングされ，並行ネットワーク外のサーバで tcpdump で pcap ファイルとして保存される．さらに HTTP，DNS，ICMP といった主要なプロトコルを SF-TAP [10] で要約する．SF-TAP は L7 レベルでネットワークトラフィックを高速に解析できる基盤である．

ホスト上の活動の観測には OS 標準のログ収集機能やエンドポイント対策製品を用いた．Linux の仮想マシンでは rsyslog，Windows の仮想マシンでは Windows イベントログを利用した．これに加え，Windows が搭載された仮想マシンでは FFRI yarai [*3] を用いて起動プロセスやプロセスツリー，プロセスの通信状況を取得できる [11].

[*1] http://www.vmware.com/products/vsphere/
[*2] http://www.vmware.com/products/vsphere/features/distributed-switch
[*3] http://www.ffri.jp/assets/files/products/ctrg/yarai_E_brochure.pdf

3.3　ワームホール

第 2.4 節で挙げたワームホールの要求を満たすために多段 NAT と VPN を用いた．多段 NAT には iptables[*4] の *PREROUTING* をゲート部に，同様に *POSTROUTING* をプロキシ部に適用した．ゲート部とプロキシ部の間は，OpenVPN[*5] を用いて接続した．なお，このときゲート部とプロキシ部は並行ネットワークのルータとして機能する．そのため，ゲート部で TTL を合わせて詐称した．さらに，ゲート部では Snort [*6] と iptables を利用して DoS 攻撃やマルウェアの感染拡大を検知・遮断する機能を実装した．

3.4　ユーザインタフェース

STARDUST の UI は Ruby on Rails [*7] で実装した．この UI では観測結果の閲覧や VNC[*8] による仮想マシンの操作ができる．また，RESTful API が実装されており，解析者自身でこの API を用いて解析フローを自動化できる．

4.　性能評価

本章では，並行ネットワークの構築について性能を評価した．評価に用いた機器を表 2 に示す．表 2 の機器は全て 10 GbE のネットワーク機器で接続した．また，構築する並行ネットワークには大きさの異なる 3 種類を表 3 のように定義し，それぞれの構築に要した時間を計測した．

表 2　性能評価における機器の構成

システム	機器の仕様
環境構築システム	CPU: Intel Xeon (E5-2695 2.40 GHz) ×2 RAM: 96 GB NIC: Broadcom BCM57810 (10 GbE) OS: Ubuntu 14.04 LTS (Kernel 4.2.0) DISK: Fusion-io ioDrive2
ハイパーバイザ群	CPU: Intel Xeon (E5-2670 2.60 GHz) ×2 RAM: 96 GB NIC: Broadcom BCM57810S (10 GbE) OS: VMWare ESXi 5.1.1 DISK: FUJITSU NR1000 V3250 (NFSv4)

表 3　性能評価で構築する並行ネットワーク

	サブネット	仮想マシンの OS（サービス）
S	SERVER	CentOS 6.5×2 (DNS, DHCP)
	CLIENT	Windows 7 Professional 64bit×1
	-	VyOS 1.1.0 ×1 (ルータ)
M	DMZ	CentOS 6.5×2 (DNS, syslog)
	SERVER	CentOS 6.5×5 (DHCP, インドラ WWW, syslog, HTTP プロキシ, yarai コレクタ [11]), Windows Server 2008 R2×1 (AD)
	CLIENT	Windows 7 Professional 64bit×5
	-	VyOS 1.1.0×1 (ルータ)
L	DMZ	CentOS 6.5×2 (DNS, syslog), Ubuntu 12.04 LTS×3 (メール, WWW, FTP)
	SERVER	CentOS 6.5×5 (DHCP, インドラ WWW, syslog, HTTP プロキシ, yarai コレクタ [11]), Windows Server 2008 R2×3 (AD, ファイルサーバ ×2)
	CLIENT1	Windows 7 Professional 64bit×5
	CLIENT2	Windows 7 Professional 64bit×5
	-	VyOS 1.1.0×1 (ルータ)

[*4] http://www.netfilter.org/projects/iptables/
[*5] https://openvpn.net/
[*6] https://www.snort.org/
[*7] http://rubyonrails.org/
[*8] https://kanaka.github.io/noVNC/

表 4　各並行ネットワークの構築時間

並行ネットワーク & 仮想マシン	OS	Size (GB)	Min. (s)	Med. (s)	Max. (s)	Ave. (s)
S サイズ	—	—	990.8	1,060.5	1,087.7	1,045.7
M サイズ	—	—	4,023.5	4,208.2	4,275.3	4,163.7
L サイズ	—	—	6,990.7	7,040.4	7,188.0	7,074.7
DNS	CentOS 6.5	1.7	191.5	244.8	268.1	233.5
DHCP	CentOS 6.5	3.1	255.5	265.4	340.5	277.9
HTTP プロキシ	CentOS 6.5	2.3	450.7	470.9	517.8	476.9
イントラ WWW	CentOS 6.5	2.0	228.3	231.6	305.3	246.1
yarai コレクタ [11]	CentOS 6.5	8.7	418.7	464.6	471.7	457.3
Syslog	CentOS 6.5	2.3	246.7	256.3	258.8	253.5
WWW	Ubuntu 12.04 LTS	4.0	199.2	208.4	213.2	206.9
Mail	Ubuntu 12.04 LTS	4.9	231.0	237.5	307.6	249.4
FTP	Ubuntu 12.04 LTS	5.1	560.3	668.7	687.3	646.7
ルータ	VyOS 1.1.0	5.1	216.6	222.9	227.8	222.4
AD	Windows Server 2008 R2	7.0	280.6	287.0	297.0	287.8
ファイルサーバ	Windows Server 2008 R2	7.0	283.6	287.1	297.4	288.5
Window クライアント	Windows 7 Professional	7.8	304.1	306.0	312.9	307.0

表 4 に並行ネットワークの構築に要した時間を示す．なお，表中に記載の各仮想マシンの生成に要した時刻は L サイズの並行ネットワーク構築時に計測したものである．並行ネットワークの S サイズ，M サイズ，L サイズの構築にはそれぞれ平均で 1,045.7 秒，4,163.7 秒，7,074.7 秒要した．現在の Alfons の実装では，仮想マシン設置の並列実行やディスクのマウント変更などでさらなる高速化の余地がある．

5. ケーススタディ

5.1 概要

表 5 にケーススタディの概要を示す．本稿では日本や台湾の組織を標的とした DragonOK [2] を対象とした 3 件（Case 1〜3）の解析結果を述べる．また予備調査として，日本の政府組織に被害をもたらした BlueTermite [3] の解析事例を 1 件（Case 0）述べる．並行ネットワークは表 3 の M サイズにメール，FTP，WWW サーバ（Ubuntu 12.04 LTS）を DMZ に，ファイルサーバ（Windows Server 2008 R2）をサーバセグメントに追加したものを利用した．

攻撃者の活動はそれぞれ図 4 から図 8 に示す．攻撃者の活動は基本的に C&C 通信から得られた Windows の標準コマンドで，括弧付の文字列 (例：[download_file]，[list_files] など) はマルウェアに実装されたコマンドを表す．

以下に，ケーススタディの流れを示す．

(1) マルウェアの動的解析により C&C サーバのドメイン名を抽出し，*nslookup* を用いて名前解決できることを確認する．
(2) 並行ネットワークとは別のインターネット回線で *telnet* コマンドを用いて C&C サーバと接続できることを確認する．
(3) 以上の調査後に並行ネットワーク内でマルウェアを実行する．このマルウェアを実行した Windows クライントを以降では「感染ホスト」と呼ぶ．
(4) 感染ホストから C&C サーバへの通信が途絶える，もしくは攻撃者の活動が停滞すると観測を終了する．

5.2 Case 0

はじめに BlueTermite のマルウェア “Emdivi” を用いた観測結果について述べる．このケースでは並行ネットワーク上に文書やメールといったコンテンツは全く設置しない状態で攻撃者の活動を観測した．観測結果を図 4 に示す．

まず, 攻撃者は感染ホストに侵入し *ipconfig*, *net*, *tasklist*, *whoami* コマンドを利用し，感染ホストの設定や状態を確認していた．次に，*dir* コマンドでフォルダ内容を得た．これらの後に，攻撃者は感染ホストを *format* や *shutdown* コマンドを利用して停止させようとした．この結果より，攻撃者がこの感染ホストは解析環境の一部だと判別した可能性がある．そのため，次節以降のケースではいくつかの文書やメールといったコンテンツを設置することとした．

5.3 Case 1

ファイルサーバやデスクトップ上に日本語名で「公募 20XX」や「重要書類」,「作業中」といったフォルダや「総務部メンバー表.xlsx」というファイルを設置し，攻撃者の活動を観測した（図 5）．これら 27 件のコマンドは約 30 分間に実行されていた．攻撃者は *net*，*systeminfo*，*whoami*，*tasklist* コマンドを用いて，感染ホストの基本情報を取得していた．それから，*dir* コマンドでフォルダの内容を表示した．これらは Case 0 でも同様に見られた活動である．

このケースの特徴が 3 点挙げられる．まず，コマンドを手作業で実行していると考えられる点がある．図 5 中の *whomai* コマンドは Windows に搭載されておらず，その直後に *whoami* コマンドの実行からタイプミスだったことが伺える．2 点目として，約 30 分と短時間でコマンドが実行されていた点である．最後に，攻撃者は並行ネットワーク内の探索を試みる際に *net*，*netstat*，*ping*，*arp*，*tracert* コマンドというネットワーク関連の管理コマンドを実行していたことが挙げられる．

5.4 Case 2

日本の大学・研究機関を標的とした攻撃の解析事例を図 6 に示す．これら 36 件のコマンドは約 15 分以内に実行

表 5　ケーススタディの基本情報

#	解析日	対象	マルウェア（MD5）	C&C	並行ネットワークの状態
0	2015/08/04–08/04	BlueTermite	7af68ddba01ba2d69a8ef7c17430e5d0	JP	ドメイン参加
1	2016/03/25–04/11	DragonOK	251c0f90bfe9a302c471bf352b259874	US	ドメイン参加，ファイル・メール設置
2	2016/05/27–05/31	DragonOK	acc2e5f8abd7426574712fe6a13c2342	SG	ドメイン参加，ファイル・メール設置
3	2016/08/18–09/30	DragonOK	c938690a0558d070528a7cab4de0e9b3	US	ドメイン参加，ファイル・メール設置

```
ipconfig /all
tasklist /v
tasklist /v
net view /domain
whoami
net use
dir c:\users\ktakahashi\
dir c:\users\ktakahashi\Desktop
format c: /s
shutdown -t 0
format c:\
format c:
shutdown /s /t 0
```

図 4　Case 0. で観測したコマンド．

```
net view
systeminfo
whoami
tasklist
dir c:\users\nito\desktop\
dir "c:\program files\"
dir d:\
dir c:\users\nito\
dir c:\users\nito\documents\
dir c:\users\nito\downloads\
dir \x03"c:\Program Files (x86)\"
netstat -an
dir c:\users\nito\documents\\x03Credential\
ipconfig /all
whomai /groups | find /i "level"
whoami
whoami /groups
net group
net view
arp -a
netstat -ano
ping 10.136.8.4 -n 1
tasklist
netstat -an
net view
tracert
net view \\win05
```

図 5　Case 1. で観測したコマンド

```
ipconfig /all
cd Users\ktakahashi\Desktop
dir
[download_file] [総務部メンバー表.xlsx]
cd ??-??????????201605
dir
net view /domain
z:
dir
cd ??2016
dir
tasklist
net view
net user /domain
net user skawano /domain
whoami
net user ktakahashi /domain
net user
cd \
net view
dir \\SOUMU04\
z:
cd ???
cd ????
cd *2011
dir
cd..
cd *2016
dir
cd..
cd *2015
dir
net view /domain
net group "Domain computers" /domain
ping FS -n 1
net view \\10.136.8.10
```

図 6　Case 2. で観測したコマンド

されていた．さらに，マルウェアに実装された機能で「総務部メンバー表.xlsx」を攻撃者のサーバに送信していた．

このケースの特徴として，攻撃者は *cd* コマンドの実行に正規表現を利用していた．具体的には Windows 上では 1 文字マッチを表す “?” を利用したり（“??-??????????201605” など），1 文字以上マッチを表す “*” を利用した（“*2011”，“*2016”，“*2015”）．このことより，攻撃者は日本語入力をせずに「公募 2011」，「公募 2016」，「公募 2015」といったフォルダに移動する工夫をしていたと推察する．なお，並行ネットワークに利用した Windows 7 Professional 上では *cd ???* や *cd ????*の正規表現は解釈できない．

さらに，前ケース同様に攻撃者は *net* コマンドで並行ネットワーク内を探索しているが，これに加えて攻撃者がインタラクティブにコマンドを実行していた様子が散見された．まず，*net view* コマンドで感染ホストの参加ドメイン上のホストを探索している．このコマンドの結果には “FS” というホスト名を持つファイルサーバが含まれる．これに対して，*ping FS -n 1* というように “FS” に対して 1 度だけ ICMP パケットを送信していた．この活動により，感染ホストと “FS” とのネットワーク疎通性の確認を行い，“FS” の IP アドレス（ここでは，“10.136.8.10”）が得られる．そして，この直後に *net view \\10.136.8.10* と IP アドレスを指定してコマンドを実行していた．このように攻撃者が実行コマンドの結果を利用して次のコマンドを実行している様子が伺えた．

5.5　Case 3

最後のケースにおける観測結果を図 7 に示す．“conhost.exe” を実行し攻撃者を誘引した．このケースの攻撃者はこれまでと同様に 54 件の Windows コマンドの実行で並行ネットワーク内の設定や状況を調査した．他ケースと異なる点は，追加で Mimikatz を実行し AD 認証のためのチケットの取得を試みていたことである．

さらに，2016 年 8 月 29 日に “3.exe” が実行された．以降，解析終了までこのマルウェアを介して攻撃者は活動していた．この実行ファイルを並行ネットワークから持ち出し解析したところ，PlugX の亜種と考えられる特徴があった．そして，並行ネットワーク上でこれを実行したところ，図 8 のようなコマンド実行を観測できた．このケース中には攻撃者は “$Recycle.Bin” や’ “Startup” といったフォルダの中身を確認していた．また，マルウェアのキーロガー

*9　https://raw.githubusercontent.com/mattifestation/PowerSploit/master/Exfiltration/Invoke-Mimikatz.ps1

```
ipconfig /all
systeminfo
tasklist
          (略：上の 3 コマンドの繰り返しのため)
net view
net group /domain
net view /doamin
net view /domain
arp -a
dir "c:\program files"
dir c:\
dir "c:\Program Files (x86)"
net group /domain
net group "domain admins" /domain
ver
powershell IEX (New-Object Net.WebClient).
    DownloadString('URL *9'); Invoke-Mimikatz-
    DumpCerts
dir d:\
dir
dir *.txt
ipconfig /all
arp -a
net group /domain
net group "domain admins" /domain
tasklist
net start
systeminfo
dir C:\USERS
dir c:\\
net user
dir c:\users\ktakahashi
dir c:\users\ktakahashi\desktop
dir c:\users\ktakahashi\desktop\作 業 中
dir c:\users\ktakahashi\desktop /s
dir "c:\Program Files"
dir "c:\Program Files (x86)"
dir c:\windows\temp\3.exe
c:\windows\temp\3.exe
          (略：dir コマンドの実行の繰り返しのため)
systeminfo
dir "C:\Documents and Settings\Administrator\
    desktop"
          (略：dir コマンドの実行の繰り返しのため)
```

図 7　Case 3. で “conhost.exe” の実行時に観測したコマンド

```
[list_files] [C:\$Recycle.Bin]
[list_files] [C:\Users\ktakahashi\AppData\Roaming
    \Microsoft\Windows\Start Menu\Programs\
    Startup\]
[key_logger]
[screen_capture]
          (略:  以上が順不同で繰り返されるため)
```

図 8　Case 3. で “3.exe” の実行時に観測したコマンド

表 6　ピアソンの積率相関係数 (左から降順).

Cases	1 と 3	2 と 3	1 と 2	0 と 1	0 と 3	0 と 2
相関係数	0.79	0.52	0.49	0.38	0.35	0.16

やスクリーンキャプチャ機能が実行されたことが C&C 通信の解析によりわかった．

このケースでは C&C サーバの名前解決でループバックアドレスを返すようになり一度終了した．しかし，それから約 1 ヶ月後に再び別の IP アドレスを返すようになった．それから再度 STARDUST を用いて解析を試みたが，以前のように攻撃者が活動する様子は見られなかった．

5.6　ケーススタディのまとめ

STARDUST を利用することで攻撃者の活動を長期に渡って観測できた．最長のものが Case 3 でこれは 43 日間に渡った．既存研究 [12] では攻撃活動の観測が最長で 6 時間程度だが，STARDUST では攻撃者の長期誘引に足る模倣性を有した並行ネットワークを構築できたと考えられる．

観測した Windows 標準搭載コマンドの実行回数についてピアソンの積率相関係数を計算した結果を表 6 に示す．Case 1 と 3 について強い相関 (> 0.7) があり，Case 2 と 3，1 と 2 についてはやや相関 (> 0.4) があった．すなわち，DragonOK の活動について各々相関があった．その他に，攻撃グループが異なる Case 0 と 1 および Case 0 と 3 の間の相関係数についてはやや弱い相関 (> 0.2) があった．

また，Case 1 と 2 では短時間（それぞれ約 30 分，約 15 分ほど）でコマンドが実行され，コマンド実行時の利用オプションにも類似性が見られた．たとえば，*ping* コマンド実行時に *-n 1* オプションを利用していた．これらの結果を総合すると，攻撃者はマニュアル化された方法で標的組織のネットワークを調査していたと推測でき，それほど高度な技術を持ち合わせていない可能性がある．

6.　考察

Case 0 では並行ネットワーク上に文書やメールといったコンテンツを全く設置せずに解析を試みた．すなわち，この並行ネットワークは攻撃者にとって容易に解析環境と判別できる状態であった．攻撃者は最初に感染ホストのネットワーク設定やファイルの有無を調査した後にすぐに *format* や *shutdown* コマンドを用いてシステムを停止を試みた．このことから，攻撃者が並行ネットワークを実ネットワークだと錯覚するに足る「生活感」が必要であることが示唆される．そのため，これ以降のケースでは文書やフォルダ，メール，Windows の「最近使ったファイル」などに擬似的なコンテンツを挿入した並行ネットワークを構築した．その結果，長期間に渡って攻撃者を並行ネットワークに誘引できた．

また，STARDUST の観測結果から実ネットワークに適用可能な検知技術を確立できると考えられる．観測結果には *net*，*ipconfig*，*netstat*，*tasklist*，*systeminfo* などが多く散見された．これらは企業や政府組織の一般職員が普段の業務で頻繁に利用することはないコマンドである．そのため，これらのコマンド実行を検知する仕組みを組織に導入するだけでも一定の効果が得られると考えられる．同様に *ping* や *net* コマンドのようなネットワーク上の挙動の検知も有効であると考えられる．このように，STARDUST による観測結果を実ネットワークにフィードバックすることで，リアルタイムな検知技術を実現できる可能性がある．

7.　関連研究

7.1　マルウェアの動的解析

解析者は自身でカスタマイズした動的解析環境を持つ．仮想マシン上に Wireshark[*10] や Process Monitor[*11]，iNetSim[*12] などを動作させてマルウェアの活動を観測する．ま

[*10] https://www.wireshark.org/
[*11] https://technet.microsoft.com/en-us/sysinternals/processmonitor.aspx
[*12] http://www.inetsim.org/

た，DNS や HTTP 通信を対象とする Sandnet [13] や C&C 通信とホスト上の活動に着目する JACKSTRAWS [14] といった自動解析システムが研究されている．

STARDUST ではインターネット接続を確立した上で全てのネットワーク・ホスト上の活動を記録し続けられる．さらに，並行ネットワーク内部の通信も同様に記録できる．このため，マルウェアや攻撃者による並行ネットワーク内外の活動を合わせて観測できる．

7.2 攻撃者の活動の観測

まず高対話型クライアントハニーポットと STARDUST の差異を明確にする．多くのハニーポットは特定の脅威に焦点を当て的確かつ安全な観測を実現している [15, 16]．

STARDUST は，攻撃者を並行ネットワーク上で自由に活動させられる．攻撃者は並行ネットワーク内での探索活動や権限昇格，機密情報収奪など制約なく実行できる．ただし，並行ネットワークとインターネット間の通信のうち感染拡大や DDoS 攻撃といった並行ネットワーク外へ影響を及ぼす場合にはワームホールで検知・遮断できる．

次に実ネットワークを利用した欺瞞技術を述べる．CINDAM [17] は SDN で実ネットワークを欺瞞ネットワークに転換する．Wang らは実ネットワーク上のユーザ，ファイル，サーバの模擬活動による欺瞞を提案している [18]．

一方で STARDUST は実ネットワークを高精細に模倣した並行ネットワークを構築する．そして，これらのネットワークをワームホールで接続し，攻撃者による実・並行ネットワークの判別を困難にする．

8. おわりに

本稿ではサイバー攻撃誘引基盤 STARDUST を提案，実装した．STARDUST は解析機能を持つ並行ネットワークを数時間程度で自動構築できる．ケーススタディでは攻撃者が並行ネットワーク上で Windows のコマンドや別途ダウンロードしたツールを実行する様子を観測できた．これらの活動内容の類似性からマニュアル化された方法でネットワーク上やホスト内を調査している可能性も考えられる．

今後の課題として，並行ネットワーク構築の高速化や，よりステルス性の高い観測技術の開発，並行ネットワークの模倣性をより向上させるための擬似的な文書やユーザトラフィックの自動生成技術の確立等が挙げられる．さらに，STARDUST を用いた攻撃者誘引についてもその規模や並列性を拡大していく．

謝辞

STARDUST の研究開発に有益な助言や支援を頂いた福森大喜氏，竹久達也氏，岩村誠博士，吉岡克成准教授，清水雄介氏，岩崎圭佑氏に感謝の意を表する．

参考文献

[1] Mandiant. Mandiant APT1: Exposing One of China's Cyber Espionage Units, 2013.
[2] FireEye. Operation Quantum Entanglement, 2014.
[3] Kaspersky Lab. New Activity of the Blue Termite APT, 2015.
[4] Ari Juels and Ting-Fang Yen. Sherlock Holmes and The Case of the Advanced Persistent Threat. In *Proceedings of the LEET '12*, 2012.
[5] Stevens Le Blond, Adina Uritesc, Cédric Gilbert, Zheng Leong Chua, Prateek Saxena, and Engin Kirda. A Look at Targeted Attacks Through the Lense of an NGO. In *Proceedings of the 23rd USENIX Security Symposium*, pp. 543–558, 2014.
[6] Internet Identity Threat Intelligence Department. Exploring the Black Hole Exploit Kit, 2011.
[7] Kaspersky Lab. Unveiling "Careto" - The Masked APT. Technical report, 2014.
[8] James Lewis and Martin Fowler. Microservices - a definition of this new architectural term. `http://martinfowler.com/articles/microservices.html`.
[9] Shingo Yasuda, Ryosuke Miura, Satoshi Ohta, Yuuki Takano, and Toshiyuki Miyachi. Alfons: A Mimetic Network Environment. In *Proceedings of the TRIDENTCOM 2016*, 2016.
[10] Yuuki Takano, Ryosuke Miura, Shingo Yasuda, Kunio Akashi, and Tomoya Inoue. SF-TAP: Scalable and Flexible Traffic Analysis Platform Running on Commodity Hardware. In *Proceedings of the LISA '15*, pp. 25–36, 2015.
[11] 中里純二, 津田侑, 衛藤将史, 井上大介, 中尾康二. プロセスの出現頻度や通信状態に着目した不審プロセス判定. 電子情報通信学会技術研究報告, 第 115 巻, pp. 77–82, 2016.
[12] 寺田真敏, 堀健太郎, 成島佳孝, 吉野龍平, 萩原健太. 研究用データセット「動的活動観測 2015」. マルウェア対策研究人材育成ワークショップ 2015 (MWS2015), 2015.
[13] Christian Rossow, Cj Dietrich, Herbert Bos, Lorenzo Cavallaro, Maarten Van Steen, Felix C. Freiling, and Norbert Pohlmann. Sandnet: Network Traffic Analysis of Malicious Software. In *Proceedings of the BADGERS '11*, pp. 78–88, 2011.
[14] Gregoire Jacob, Ralf Hund, Ruhr-university Bochum, Christopher Kruegel, and Thorsten Holz. JACKSTRAWS: Picking Command and Control Connections from Bot Traffic. In *Proceedings of the 20th USENIX Security Symposium*, 2011.
[15] Alexander Moshchuk, Tanya Bragin, Steven.D. Gribble, and Henry.M. Levy. A Crawler-based Study of Spyware on the Web. In *Proceedings of the NDSS Symposium 2006*, 2006.
[16] Xuxian Jiang and Xinyuan Wang. Out-of-the-box Monitoring of VM-Based High-Interaction Honeypots. In *Proceedings of the RAID '07*, pp. 198–218, 2007.
[17] Seth Robertson, Scott Alexander, Josephine Micallef, Jonathan Pucci, James Tanis, and Anthony Macera. CINDAM: Customized Information Networks for Deception and Attack Mitigation. In *Proceedings of the SASOW*, pp. 114–119, 2015.
[18] Wei Wang, Jeffrey Bickford, Ilona Murynets, Ramesh Subbaraman, Andrea G. Forte, and Gokul Singaraju. Detecting Targeted Attacks By Multilayer Deception. *Journal of Cyber Security and Mobility*, Vol. 2, No. 2, pp. 175–199, 2013.