G-8

ハニーポットにおける攻撃者の侵入ログの分析

Analysis of attacker intrusion log in honeypot

○齋藤利弥[1]，房安良和[2]，泉隆[3](日本大学)
*Toshiya Saito[1], Yoshikazu Fusayasu[2], Takashi izumi[3] (Nihon Univercity)

With the spread of IoT devices, it is predicted that the risk of cyber attacks on IoT devices will increase. In this paper, we will examine the configuration of honeypots that observe cyber attacks on IoT devices.

1. まえがき

近年 IoT デバイスが急速に普及し，2017 年には 270 億ほど存在する IoT デバイスが 2020 年には 400 億ほどに増加すると予測されている[1]．しかし，普及が進む一方，IoT デバイスを踏み台として利用するために，サイバー攻撃が行われている．例えば，国立研究開発法人情報通信研究機構(NICT)によるサイバー攻撃の観測では，全パケット中 26%が IoT デバイスへの攻撃であった[2]．この様な，IoT デバイスがサイバー攻撃の標的とされている現状に対して対策を行うためには，攻撃手法や使用されるマルウェアの分析等が必要となる．

IoT デバイスに対するサイバー攻撃の分析のために，先行研究[3]では，ハニーポットを用いたプロトコル別に不正アクセス件数や，ログイン試行で用いられたパスワード等についての分析が行われていた．しかし，ハニーポットへ行われたサイバー攻撃に対して，ログイン後に攻撃者が行ったコマンドの分析が不十分であることや，一部の攻撃者がマルウェアのダウンロードを行わずに攻撃を途中終了していた等の問題があった．

これらを踏まえ，本研究ではハニーポットに対するサイバー攻撃で発生した通信ログを収集し，攻撃手法の分析を通して，不正な通信の検知や分類を行うことを目的とする．

本稿では，構築するハニーポットの構成について検討を行った．

2. ハニーポット[4]

ハニーポットは，攻撃者に脆弱なシステムであると見せかけることで攻撃を誘い込み，侵入方法や侵入後の動作を分析するシステムである．

2.1. ハニーポットの構成

本研究で構築するハニーポット(Figure1)はホストマシンと QEMU を用いて i386 アーキテクチャの Linux(Debian)デバイスをエミュレーションし IoT デバイスと類似した環境を再現する．ハニーポットでは，攻撃手法を分析するため，平文で通信を行う Telnet を稼働させ，通信ログの取得及び攻撃手法の分析を行う．ハニーポットの構成を Table1 に示す．

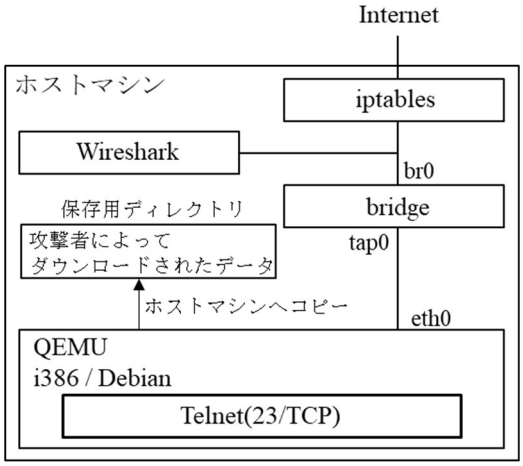

Figure 1. Diagram of honeypot

Table 1. Configuration of Honeypot

Category	Software	Purpose
Access controller	iptables	To restrict a communication from inside to outside (To prevent the occurrence of DoS attack)
Packet capture software	Wireshak	To collect communication packet(pcap file)
Server software	telnetd	To activate Telnet server
Shell after login	rbash	To restrict malware removal and execution

2.2. iptables の設定

ハニーポットは，攻撃者からの攻撃を受けるために設置する．そのため，攻撃者によって不正利用されないよう一部のコマンドに対して使用制限を設けている．しかし，ハニーポットがマルウェアに感染した場合等に備え，iptables を用いて外部への送信パケットに制限を設ける．iptables の設定を Figure2 に示す．

Figure2 内赤枠部では TCP，UDP，ICMP での通信に最初の 30 パケット以降は毎秒 1000 パケット以上の通信を破棄するように設定している．これにより，DDoS 攻撃を行う踏み台にされた場合にも閾値以上のパケットを破棄し，外部へ大量のパケット送信を防ぐ．

1：日大理工・学部・情報 2：日大理工・院(前)・情報 3：日大理工・教員・情報

```
iptables -P INPUT DROP
iptables -A INPUT -p tcp -m state --state ESTABLISHED,RELATED -j ACCEPT
iptables -P OUTPUT ACCEPT
iptables -A INPUT -i lo -j ACCEPT

iptables -N HASHCHECK
iptables -A HASHCHECK -m hashlimit --hashlimit-name hashcheck_t\
        --hashlimit 1000/s --hashlimit-burst 30 --hashlimit-mode dstip \
        -j ACCEPT

iptables -A HASHCHECK -m limit --limit 1/s -j LOG --log-prefix '[IPTABLES HASH DRO

iptables -A HASHCHECK -j DROP

iptables -A INPUT -p tcp -j HASHCHECK
iptables -A INPUT -p udp -j HASHCHECK
iptables -A INPUT -p icmp -j HASHCHECK

iptables -t nat -F

iptables -A INPUT -m state --state NEW -m tcp -p tcp --dport 23 -j ACCEPT
iptables -A INPUT -m state --state NEW -m tcp -p tcp --dport 22 -j ACCEPT
```

Figure 2. configuration of iptables

2.3. Wireshark の設定[5]

ハニーポットの稼働で発生する通信パケットの取得を目的に Wireshark の軽量な CUI 版である tshark を用いる．

2.3.1. 取得データの検討

発生する全てのパケットを取得することが可能であるが，その場合取得データサイズは膨大となる。そこで，キャプチャフィルタを用いて取得するパケットをあらかじめ観測対象である Telnet のみに絞った取得を行う．なお，解析に有益な通信が他プロトコルで行われる可能性を踏まえ，攻撃者によって入力されたコマンド情報等についても並行して取得する．また，それぞれのパケットキャプチャファイルの容量が大きくなりすぎることを避けるため，取得パケットは 30 分毎のパケットキャプチャファイルとして扱う．

2.3.2. 通信の分類方法の検討

取得した通信パケットを用いることで通信内容や，ログインの成否等の確認が可能である．ログイン試行での攻撃検知を行う場合，攻撃者によるログインの失敗の他に正規利用者の入力間違いによるログイン失敗が考えられるため，ログイン失敗を不正な通信と一概に分類することはできない．

そこで，先行研究において示されているログイン施行にて使用されやすいユーザ名やパスワードが存在することや，ハニーポットへの通信が多い地域傾向があることを用いることで攻撃の検知を提案する．例えば，入力されたログイン情報が攻撃時に多用されるものであることや，ログイン施行が連続して何度も行われるブルートフォース攻撃のような挙動であること，または IP アドレスから通信元地域を取得することで，不正なログイン試行の検知が可能と考える．

2.4. rbash の設定

ハニーポットは脆弱な端末であると見せかけた攻撃手法等の収集が目的である．しかし攻撃を受ける以上，攻撃者からの想定しない手法でマルウェアを実行される可能性があり，場合によっては攻撃の踏み台とされる危険がある．

その様な問題への対策として，ハニーポットで使用するユーザの bash は制限モードのみで使用する設定を施し，マルウェアを実行させず収集するために必要なコマンドのみを実行可能にする．

例としてファイルの圧縮・展開を行う「gzip」といったコマンド等は実行可能とするがジョブの実行を行う「crontab」やユーザを切り替える「su」といったコマンドはマルウェアを実行できてしまうため制限する．

ただし，単純にコマンドを実行不可とした場合，エラーメッセージ等からハニーポットであることが検知される可能性があるため，コマンドと同じ名称の関数を作成し，実際は違うコマンドを実行させる，これによって，正常にコマンドが実行された場合やエラー時のメッセージを模倣して表示し，実際のデバイスと近い反応を行うようにしている．一例を Figure3 に示す．

```
function busybox(){
        if [[ "$1" == sh ]] ; then
                if [[ "$2" == ""  ]] ; then
                PS1='$'
                return 0
        else
                rbash $2
        fi
fi
```

Figure 3. configration of command(ex. busybox)

Figure3 によって，攻撃者が busybox を介したコマンドの実行を試行した場合，rbash 下で実行される．

先行研究では，ブラックリスト的に使用不能とするコマンドを決定していたため，攻撃者に使用されると危険と考えられるコマンドが存在した．そのため，本研究ではホワイトリスト的に安全であると考えられるコマンドを順次追加する．安全な稼働が可能である一方，攻撃者にハニーポットであると検知されることを避けるため，攻撃者がマルウェアの実行できない範囲で必要とするコマンドを随時追加する．

3. まとめ

本稿では，構築するハニーポットの安全な稼働に向けた構成について検討した．

今後は攻撃者にハニーポットであると検知され難い環境の構築や，攻撃検知に向けたログの分析手法について検討を進める．

4. まとめ

[1] 総務省:「平成 30 年版情報通信白書のポイント」, http://www.soumu.go.jp/johotsusintokei/whitepaper/ja/h30/html/nb000000.html, 2019-09

[2] 国立研究開発法人情報通信研究機構:「NICTER レポート 2018」, http://www.soumu.go.jp/main_content/000467154.pdf,2019-09

[3] 房安良和・小寺建輝・泉隆：「ハニーポットを用いた IoT デバイスに対するサイバー攻撃の分析」,平成 30 年電気学会全国大会,3-089,2018-03

[4] 八木毅,青木一史,秋山満昭,幾世知範,高田雄太,千葉大紀：「実践サイバーセキュリティモニタリング」,コロナ社,2016-03

[5] 森久和昭：「サイバー攻撃の足跡を分析するハニーポット観察記録」,秀和システム, 2017-01