1W-06

IoTデバイスへのサイバー攻撃に関する検討

小寺 建輝† 泉 隆‡

日本大学†‡

1. はじめに

近年,IoT デバイスの普及が進む一方,そのセキュリティには注意が払われることが少なく,現在多くの IoT デバイスがサイバー攻撃の被害に遭っている.2016 年秋頃には,「Mirai」と呼ばれるマルウェアが流行し,このマルウェアに感染した IoT デバイスのボットネットによる仏 OVH への DDoS 攻撃のデータ量は 1.5Tbps に達したと言われており[1],史上最大級の DDoS 攻撃が行われた.また,この Mirai は 2016 年 10 月 1 日に攻撃者本人によってソースコードが公開され,多くの人がそれを利用して亜種が次々と作成された.その結果,Mirai に感染している IoT デバイスは Mirai のソースコード公開前には 21 万 3000 台だったが公開後は 49 万 3000 台以上まで増加している[2].また,Mirai と同等の勢力をもつマルウェアとして「bashlite(別名 Gafgyt 等)」がある.bashlite は Mirai よりも前の 2014 年にその存在が確認されており,100 万台の IoT デバイスのボットネットが構築されていると言われている[3].Mirai,bashlite の共通点として,どちらのマルウェアも telnet を利用して感染し,DDoS 攻撃を行うことが目的であることがあげられる.

本研究では,ハニーポットを構築してこれらのマルウェアの収集を行った.本報告では,ハニーポットで収集した bashlite に対して静的解析を行った結果を報告する.

2. bashlite[3]

bashlite は,Linux で動作する ELF 形式の実行ファイルであり telnet を利用している IoT デバイスに対して感染する.また,IoT デバイスで採用されている様々な CPU アーキテクチャ(x86,x86_64,ARM,SuperH,PPC,M68K,MIPS,MIPSEL,Sparc 等)ごとに存在する.

ハニーポットでの観測によると,攻撃者によって telnet でのログイン認証に成功された IoT デバイスは,wget,curl,tftp,ftpget などを利用してシェルスクリプトファイルをダウンロードして実行するコマンドが送信される.そのシェルスクリプトファイルには,複数の CPU アーキテクチャの bashlite をダウンロードし,成功するまでそれらを一つずつ実行するスクリプトが記述されている.これは,IoT デバイスが様々な CPU アーキテチャを採用しているためである.

bashlite が実行された後は,攻撃者の C&C サーバに接続され,ボット化する.攻撃者からは「DDoS 攻撃(UDP フラッド攻撃,TCP フラッド攻撃等)の実行」「感染活動」「DDoS 攻撃停止」などの指令が与えられる.

3. bashlite の解析

ハニーポットでは,150 個のハッシュ値(SHA-256)の異なる bashlite を取得することができた.その内訳は図 1 に示すように,ARM や i386 で動作するものが多く,他は MIPS64 の 1 個を除き同様であった.また,ファイルの平均サイズは約 120KB(最大 251KB,最小 67KB)であった.今回解析するマルウェアは表 1 に示す i386 で動作する 27 個の bashlite を対象とする.

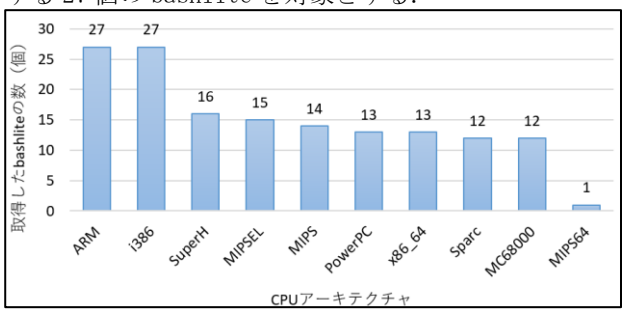

図 1. CPU アーキテクチャ別取得した bashlite の数

表 1. 解析対象の bashlite(SHA-256)

mal_ID	SHA-256
1	cd4d9f367ceb8af98b09d5ad560cb9f1342486c1cc1a9abdef39325fdbc2ba7f
2	f94beb6b9fcaccc87cc88d86d9b56cf9b2ebac04b6b3b8c95b33c5014175ffba
3	376a1e007ae745d5636256ed6af92e45776e1c8868e6dd0631193630ff217e22
4	60915403a72b8a0322d88b775a2af7f6fa183cf933a5c85103437c908a0fad0e
5	fc6dc6990fc4387ba57a32bb6ee5bc57b1cc2c3034885275f0c83534cd64cc2b
6	493c7ed0e104584efb502fe93a2142636355e46e8c3144ebbc929c563632c551
7	0668cfb10ab1e773561c31c2feb62df61d5c48f3f5b0036e35fe1fe9f2813f16
8	23083297df1af5a18b85175739c56643f25c500d30371217433307f2719ac81b9
9	e806484648fb41f037123281e8a4cf6bb0264cf734dda57622dc629e75a3e13c
10	c45407ca61c8ad43515a6a726502922270b172c0d5ed40131ba59b3c4d074048
11	6be0ffc59f2bf5a9e7394188198610013bd7165c536e10a67c8225b62b5d2a17
12	16f6e8fa362b21af751306804ac303bdf653c90ce64b5e59720b443736d432de
13	1827c0cf4e7be386658844b71856c3349d1ba0b129c367cf6eb1a03f45da2a0b
14	59c2b38d6021cfa31591a67f45a7bbca63629fa66c3ae9df38166496d2c34140
15	3c5b7b6f64b63b9db5a40a4226608e69f738539144dec839173c0ef113d96cd8
16	7873bc42f317aa5a996b03cf573219efdaf812a83a14af433c9d9a5768d2eed6
17	8b67b12b2d2c93b9fa0244c620b6521d598a47b45bf9820e1c0b054369d96397
18	1550445619474a35d00743cb8d72f89f76492a5bba085f50e8a704185ec9a4f4
19	3781df04172780cf0b00da30d6cb9ef297b65bada28c8d9bc5332ce604013b52
20	a3f2104f5cd14903b03ff2ff09035a9ddb65da5a57394e634e0360f09db9dd96
21	ac89d6fb5940f2403d9180363e63071868bfb83d0a77a96f39184ae787b3c77b
22	2ba710de60490f4c067b48c18bad77d96c5348a2573a46f45e9d7722d683f502
23	6dbe51c554952d739603cbc0f8cb02c3d4eb25355fa6611e65468c5a04f2145d
24	4eccbf0866fc3659607c6bbe0c79350ff369ca9049d7fbef943f59eb4b5e427e
25	a93be642a79293145d5f888560c215fa97043d8294511ae5a3d71aeeae5452cb
26	2cf6588e4efb9037277d6bbde81fdf2d2de8ffc42041092c8f593177d8cbeaa8
27	0504ca8f196f16300b7c9f2c5dc5f9a8946c410af77e36d6302e407c349b2a10

Study for Cyber Attacks on IoT devices
†Tateki Kodera・Department of Computer Engineering, College of Science and Technology, Nihon University
‡Takashi Izumi・Department of Computer Engineering, College of Science and Technology, Nihon University

3.1 bashliteの実行直後の挙動

bashlite(mal_ID=1,2,3,10,26を除く)がIoTデバイス上で実行されると,まずGoogle Public DNSサーバー(8.8.8.8:53)にリクエストを送信し,IoTデバイスがインターネットに接続されているかを確認し,接続が確認されるとIoTデバイスのIPアドレスとMACアドレスを取得する.その後,攻撃者のC&Cサーバに接続しコマンド指令を受け取ることによりIoTデバイスがボット化することが分かった.Google Public DNSサーバやC&CサーバのIPアドレスはbashlite本体にハードコードされていた.また,これらのbashliteは「/proc/cpuinfo」からCPUの情報,「/proc/net/route」からネットワークの情報を収集することが確認できた.

3.2 C&Cサーバからのコマンド指令

C&Cサーバから受け取るコマンドやその応答には,mal_IDごとに違いが見られた.その結果を表2に示す.

表2. C&Cサーバから受け取るコマンド

Command	Description	Reply	Command format	mal_ID
PING	接続の確認	"PONG!"	-	all
SCANNER or TELNET	<arg1>=ON: スキャンの実行 <arg1>=OFF: スキャンの停止	no reply or <arg1>!=ON | OFF: "SCANNER ON | OFF" <arg1>=ON: "PROBING" "SCANNER STARTED!" or "STARTING TELNET" or "SCANNER" or "SCANNER Starting On" <arg1>=OFF: "REMOVING PROBE" or "SCANNER STOPPED!" or "STOPPING TELNET" or "SCANNER" or "SCANNER OFF"	SCANNER <arg1> TELNET <arg1>	all
GETLOCALIP	IPアドレスの取得	"My IP: %s"	-	1以外
UDP	UDPフラッド攻撃の実行	-	UDP <target> <port(0 for random)> <netmask> <size(1-65500)> <time poll interval>	10,26以外
TCP	TCPフラッド攻撃の実行	-	TCP <target> <port(0 for random)> <time> <netmask> <flags(syn,ack,psh,rst,fin,all)> <size> <time poll interval>	10,26以外
HTTP or HTTPFLOOD	HTTPフラッド攻撃の実行	no reply or "HTTP %s Flooding %s:%d for %d seconds"	-	1,2,3,12,13,17以外
HOLD	特定期間の攻撃の 中断・遅延	-	HOLD <target> <port> <time>	4,9,17,25,27
JUNK	JUNKコマンドによる フラッド攻撃の実行	-	JUNK <target> <port> <time>	4,9,17,25,27
CNC	-	-	CNC <target> <port> <time>	4,9,12,13,17,25,27
STD	-	-	STD <target> <port> <time>	1,9,14,15,17,18,19, 25以外
KLLATTK or KILL	攻撃スレッドの停止	Success: "Killed %d." Failure: "None Killed."	-	all
LOLNOGTFO or FUCKOFF or COOLMEMS or GTFOFAG or LUCKYLILDUDE	ボットの停止	-	-	all
UPDATE or UPDATEBIN	アップデート	no reply or "UPDATING BINS" or "Updating Bins"	-	2,3,4,12,13,27
FATCOOK	次のコマンドを順に実行: "rm -rf /var/log/wtmp" "rm -rf /tmp/*" "history -c" "rm -rf ~/.bash_history" "rm -rf /bin/netstat" "history -w" "killall -9 perl" "service iptables stop"	-	-	4,9,25,27
SPOOF	-	"KILLING YOUR BOATS"	SPOOF <subnet>	12,13
DOUSPOOF	-	"FUC YEA I DO (%s)"	-	12,13
BOTKILLZ	-	-	-	12,13
PYTHON	-	-	-	4,27
COMBO	-	-	-	4,9,17,25,27

3.3 感染活動

C&CサーバからSCANNERコマンドやTELNETコマンドを受信したIoTデバイスは,256のランダムなIPアドレスをスキャンする.スキャンして接続が成功した場合はtelnet経由で接続し,マルウェア内に記述されているユーザ名/パスワードのリストを用いて辞書攻撃でログイン試行をする. 辞書攻撃によりIoTデバイスへのログインに成功すると感染しているIoTデバイスは,自身がbashliteに感染した時と同様にシェルスクリプトファイルをダウンロードして実行するコマンドを対象デバイスに送信し感染させることが確認できた.

また,辞書攻撃を行う際によく利用されるユーザ名/パスワードのリストを表3に示す.

表3. ユーザ名/パスワードのリスト

username	mal_ID
root	all
guest	1,2,3,4,6,9,10,11,12,13,14,17,18,19,22,23,25,26,27
admin	1,4,6,9,10,11,12,13,14,16,17,18,19,22,23,25,26,27
support	1,2,3,6,9,11,14,17,18,19,22,23,25
telnet	4,9,12,13,14,16,18,19,25,27
cisco	6,9,11,14,18,19,22,23,25
user	6,9,11,17,22,23,25
login	6,9,11,17,22,23,25
netgear	9,14,18,19,25
pi	9,14,18,19,25
ubnt	2,3,9,25

password	mal_ID
root	all
guest	1,2,3,4,6,9,10,11,12,13,14,15,17,18,19,22,23,25,26,27
1234	1,2,3,4,6,9,11,12,13,14,15,16,17,18,19,22,23,25,27
admin	1,2,3,6,9,10,11,12,13,16,17,22,23,25,26
123456	1,2,3,6,9,10,11,12,13,16,17,22,23,25,26
support	1,2,3,6,9,11,14,15,17,18,19,22,23,25
telnet	4,9,12,13,14,15,16,18,19,25,27
12345	1,2,3,6,10,11,17,22,23,26
password	6,9,11,15,17,18,19,22,23,25
cisco	6,9,11,14,15,18,19,22,23,25
user	6,9,11,17,22,23,25

表3より,全てのbashliteは辞書攻撃にユーザ名/パスワードとして「root」を用いることが分かった.次いで「guest」「admin」などが多く利用されていた.また,特徴的なユーザ名/パスワードとして「pi(RaspberryPi のユーザ名)」「cisco(ルータの製品名)」「netgear(ルータの製品名)」「ubnt(Ubiquiti のユーザ名/パスワード)」等も利用されていた.表3に示した以外にも,「Vizxv(Dahua 監視カメラのパスワード)」「D-Link(D-Link スイッチのユーザ名/パスワード)」「dreambox(DreamboxTV 受信機のパスワード)」といったユーザ名/パスワードを確認でき,監視カメラ,ネットワーク機器,TV等のIoTデバイスを対象としていることが考えられる.

4. まとめ

本報告では,ハニーポットを構築して収集したbashliteに対して静的解析を行うことで,bashliteが実行された直後の挙動やC&Cサーバから受け取るコマンド,感染活動の手法について分析を行った.

今後は,引き続きハニーポットを運用し,マルウェアの収集を行うと共に,bashlite以外のマルウェアについても解析を行っていく.

参考文献

[1]US-CERT,"Heightened DDoS Threat Posed by Mirai and Other botnets", https://www.us-cert.gov/ncas/alerts/TA16-288A,2017-01
[2]Level3 communications,"Attack of Things!",http://blog.level3.com/security/attack-of-things,2017-01
[3]threatpost,"BASHLITE Family of Malware Infects 1 Million IoT Devices", https://threatpost.com/bashlite-family-of-malware-infects-1-million-iot-devices/120230/,2017-01