SR2I203 Hacking : méthodes et pratiques

Davide GALLITELLI Carlotta CASTELLUCCIO Axel FOTSO

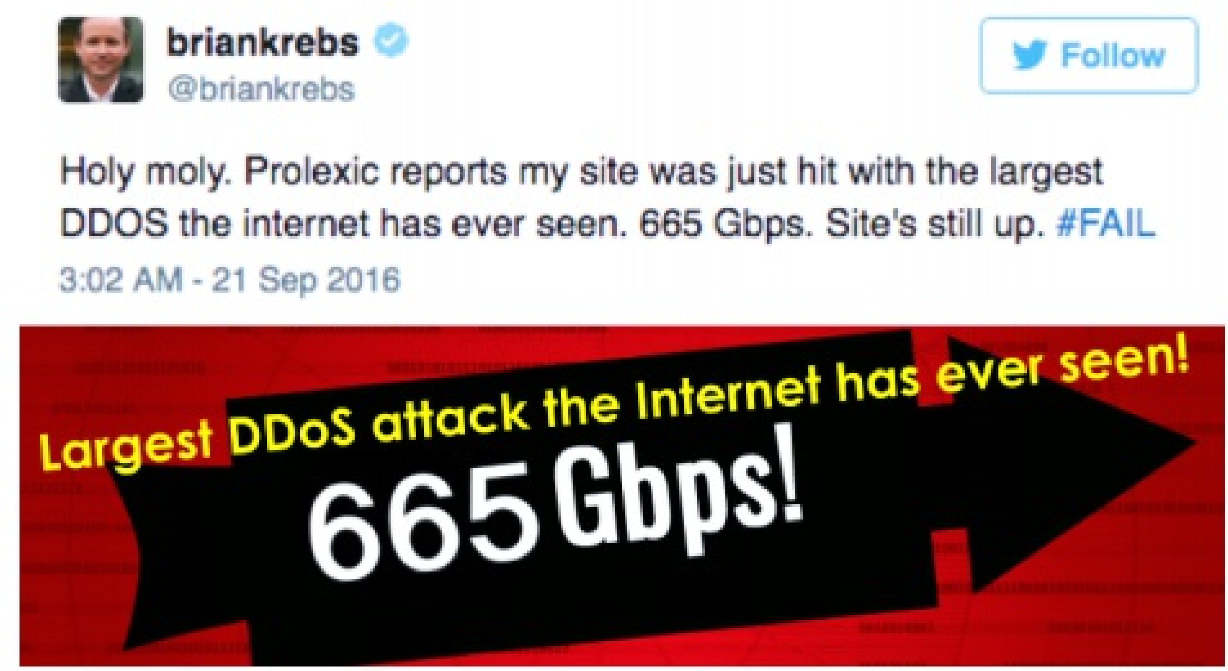
Mirai Botnet - Analysis and Simulation

Introduction

IoT devices are an ever-growing category of network devices, including printers, routers, security cameras, smart TVs, etc. Those devices have particularly susceptible to malware attacks and are becoming increasingly attractive targets for cybercriminals because of lack of security.

Recently, IoT devices have been used to create **large-scale botnets**, which can deliver highly destructive Distributed Denial of Service (DDoS) attacks. One of these malwares, **Mirai**, brought to light the problem of IoT Security, and an increased attention to the topic of interconnected devices.

In Tuesday, September 20th 2016, KrebsOnSecurity.com blog was targeted by an extremely large and unusual Distributed Denial-of-Service attack (DDoS) of over **660 Gbps of traffic**. The attack seems to have been designed to knock offline the website of the investigative cybercrime journalist Brian Krebs in retaliation for the arrest of the owners of vDOS attack-for-hire service. The attack did not succeed, but according to Akamai it was nearly **double the size of the largest attack they had ever seen** and it orders of magnitude more traffic than is typically needed to knock the most of sites offline.

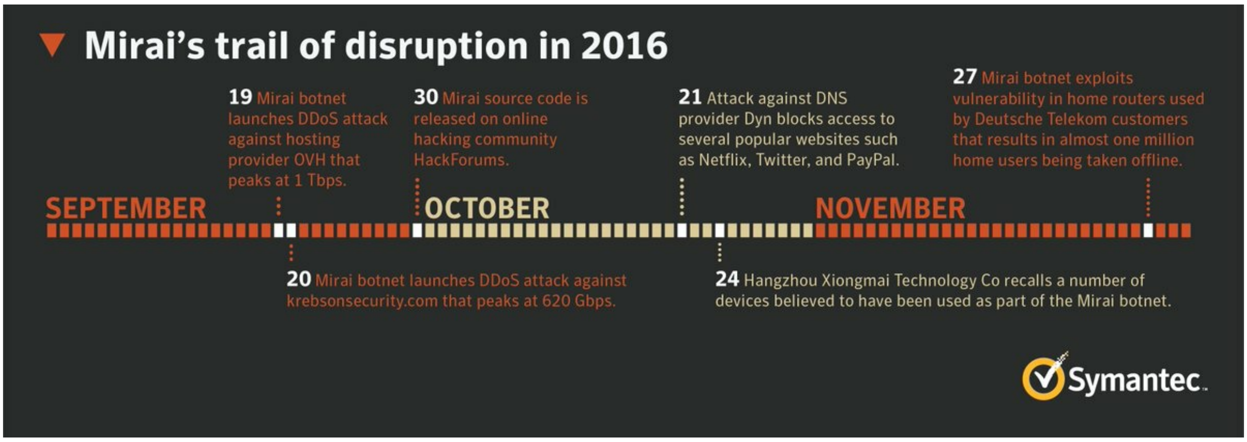


briankrebs
@briankrebs

Holy moly. Prolexic reports my site was just hit with the largest DDOS the internet has ever seen. 665 Gbps. Site's still up. #FAIL

3:02 AM - 21 Sep 2016

In the same month, an attack sharing the same technical characteristics was launched against the **French webhost OVH**, breaking the record for the largest recorded DDoS attack with at least **1.1 Tbps of traffic.** Multiple attacks have been registered since then, especially after the code for the malware has been release on September 30th 2016.



Mirai's trail of disruption in 2016

SEPTEMBER

19 Mirai botnet launches DDoS attack against hosting provider OVH that peaks at 1 Tbps.

20 Mirai botnet launches DDoS attack against krebsonsecurity.com that peaks at 620 Gbps.

30 Mirai source code is released on online hacking community HackForums.

OCTOBER

21 Attack against DNS provider Dyn blocks access to several popular websites such as Netflix, Twitter, and PayPal.

24 Hangzhou Xiongmai Technology Co recalls a number of devices believed to have been used as part of the Mirai botnet.

NOVEMBER

27 Mirai botnet exploits vulnerability in home routers used by Deutsche Telekom customers that results in almost one million home users being taken offline.

The most interesting aspect of this attack is that it was not performed by using traditional reflection/amplification DDoS, but with **direct traffic** instead: the attack was carried out by a **Botnet** (or Zombie Network) of hacked devices. While the total number of devices involved was not known for sure, it was sure that hundreds of thousands of compromised devices were related to the Internet of Things (mainly home routers, IP security cameras, Digital Video Recorder boxes and printers). The IoT devices became infected with malware by very **simple Telnet dictionary attacks** and were made part of the botnet that would then deliver the DDoS attack.

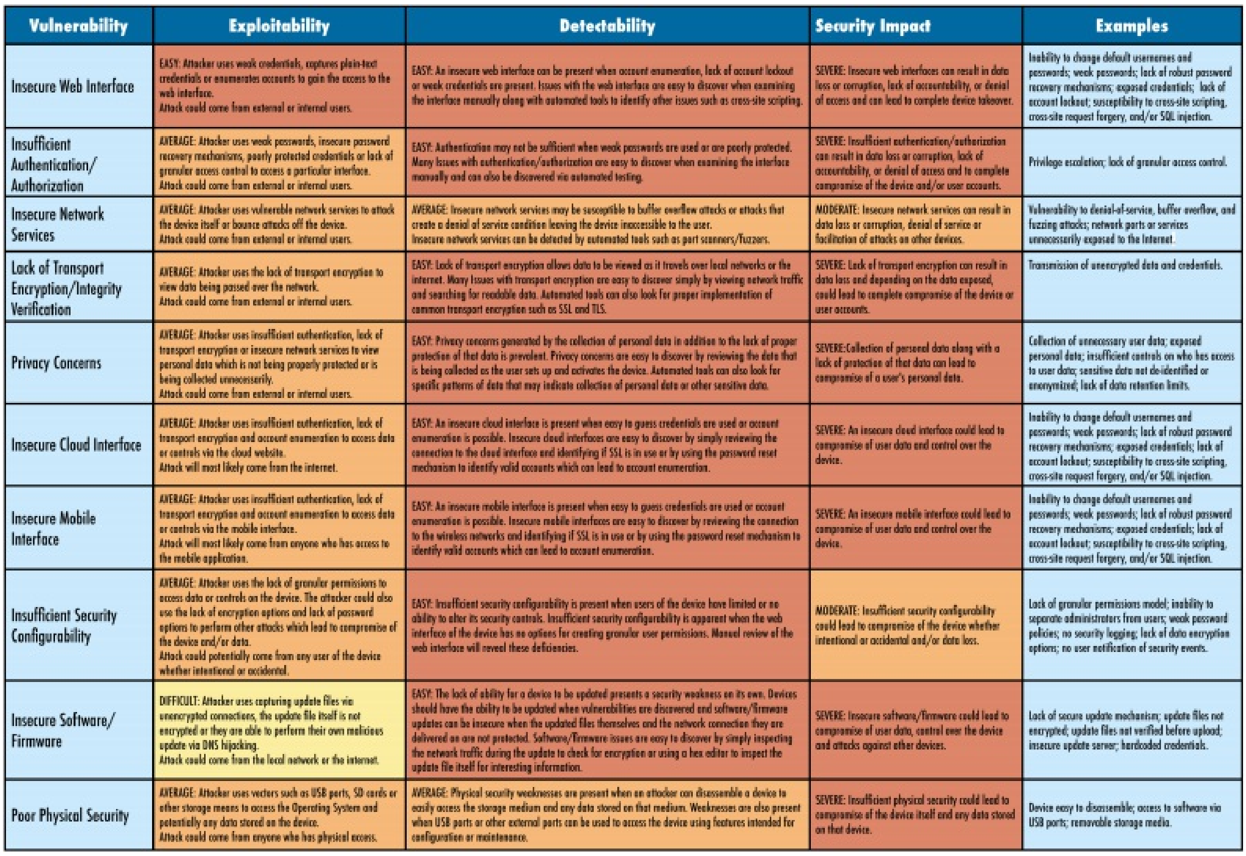
IoT Security Problem

The more connected devices, the bigger the threat. Typically IoT devices are poorly secured (sometimes, not secured at all) and the interconnected nature of these smart objects means that every poorly secured device that is connected online, it potentially affects the security and resilience of the network.

The main problem with IoT devices is that the majority of them has lack of even elementary security and they present some interesting features which make them an ideal target for hackers. To name a few, those devices:

- are highly scalable
- are always online
- are connected to fast Internet networks
- are highly heterogeneous
- might connect to Internet other offline objects
- can be phisically unprotected
- might not require particular permissions (such as root access or user interaction)

The IoT Security problem has been analyzed also by the Open Web Application Security Project and they identified the **10 most common IoT vulnerabilities** which are shown in the following table:

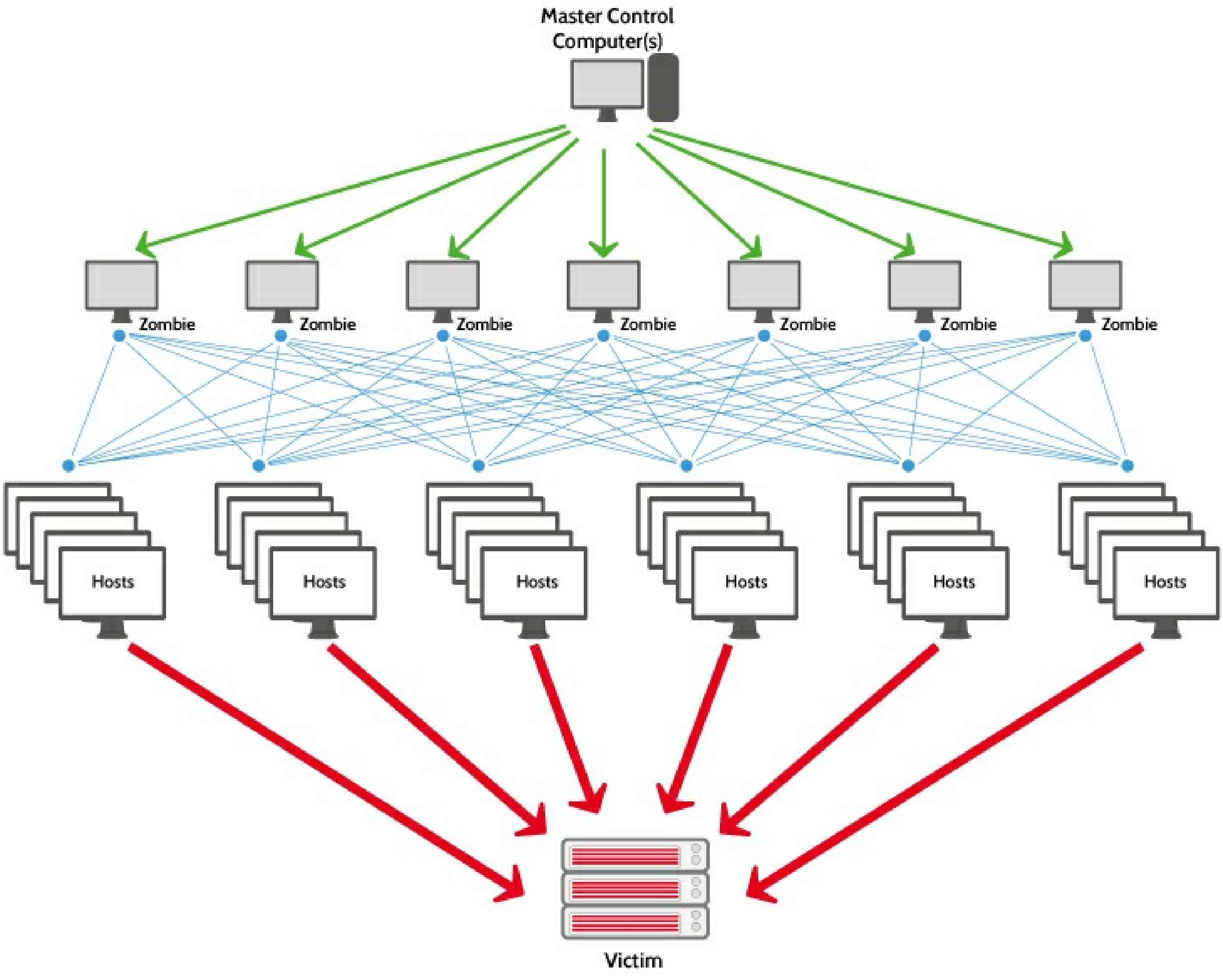


Vulnerability	Exploitability	Detectability	Security Impact	Examples
Insecure Web Interface	EASY: Attacker uses weak credentials, captures plain-text credentials or enumerates accounts to gain the access to the web interface. Attack could come from external or internal users.	EASY: An insecure web interface can be present when account enumeration, lack of account lockout or weak credentials are present. Issues with the web interface are easy to discover when examining the interface manually along with automated tools to identify other issues such as cross-site scripting.	SEVERE: Insecure web interfaces can result in data loss or corruption, lack of accountability, or denial of access and can lead to complete device takeover.	Inability to change default usernames and passwords; weak passwords; lack of robust password recovery mechanisms; exposed credentials; lack of account lockout; susceptibility to cross-site scripting, cross-site request forgery, and/or SQL injection.
Insufficient Authentication/ Authorization	AVERAGE: Attacker uses weak passwords, insecure password recovery mechanisms, poorly protected credentials or lack of granular access control to access a particular interface. Attack could come from external or internal users.	EASY: Authentication may not be sufficient when weak passwords are used or are poorly protected. Many Issues with authentication/authorization are easy to discover when examining the interface manually and can also be discovered via automated testing.	SEVERE: Insufficient authentication/authorization can result in data loss or corruption, lack of accountability, or denial of access and to complete compromise of the device and/or user accounts.	Privilege escalation; lack of granular access control.
Insecure Network Services	AVERAGE: Attacker uses vulnerable network services to attack the device itself or bounce attacks off the device. Attack could come from external or internal users.	AVERAGE: Insecure network services may be susceptible to buffer overflow attacks or attacks that create a denial of service condition leaving the device inaccessible to the user. Insecure network services can be detected by automated tools such as port scanners/fuzzers.	MODERATE: Insecure network services can result in data loss or corruption, denial of service or facilitation of attacks on other devices.	Vulnerability to denial-of-service, buffer overflow, and fuzzing attacks; network ports or services unnecessarily exposed to the Internet.
Lack of Transport Encryption/Integrity Verification	AVERAGE: Attacker uses the lack of transport encryption to view data being passed over the network. Attack could come from external or internal users.	EASY: Lack of transport encryption allows data to be viewed as it travels over local networks or the internet. Many Issues with transport encryption are easy to discover simply by viewing network traffic and searching for readable data. Automated tools can also look for proper implementation of common transport encryption such as SSL and TLS.	SEVERE: Lack of transport encryption can result in data loss and depending on the data exposed, could lead to complete compromise of the device or user accounts.	Transmission of unencrypted data and credentials.
Privacy Concerns	AVERAGE: Attacker uses insufficient authentication, lack of transport encryption or insecure network services to view personal data which is not being properly protected or is being collected unnecessarily. Attack could come from external or internal users.	EASY: Privacy concerns generated by the collection of personal data in addition to the lack of proper protection of that data is prevalent. Privacy concerns are easy to discover by reviewing the data that is being collected as the user sets up and activates the device. Automated tools can also look for specific patterns of data that may indicate collection of personal data or other sensitive data.	SEVERE:Collection of personal data along with a lack of protection of that data can lead to compromise of a user's personal data.	Collection of unnecessary user data; exposed personal data; insufficient controls on who has access to user data; sensitive data not de-identified or anonymized; lack of data retention limits.
Insecure Cloud Interface	AVERAGE: Attacker uses insufficient authentication, lack of transport encryption and account enumeration to access data or controls via the cloud website. Attack will most likely come from the internet.	EASY: An insecure cloud interface is present when easy to guess credentials are used or account enumeration is possible. Insecure cloud interfaces are easy to discover by simply reviewing the connection to the cloud interface and identifying if SSL is in use or by using the password reset mechanism to identify valid accounts which can lead to account enumeration.	SEVERE: An insecure cloud interface could lead to compromise of user data and control over the device.	Inability to change default usernames and passwords; weak passwords; lack of robust password recovery mechanisms; exposed credentials; lack of account lockout; susceptibility to cross-site scripting, cross-site request forgery, and/or SQL injection.
Insecure Mobile Interface	AVERAGE: Attacker uses insufficient authentication, lack of transport encryption and account enumeration to access data or controls via the mobile interface. Attack will most likely come from anyone who has access to the mobile application.	EASY: An insecure mobile interface is present when easy to guess credentials are used or account enumeration is possible. Insecure mobile interfaces are easy to discover by reviewing the connection to the wireless networks and identifying if SSL is in use or by using the password reset mechanism to identify valid accounts which can lead to account enumeration.	SEVERE: An insecure mobile interface could lead to compromise of user data and control over the device.	Inability to change default usernames and passwords; weak passwords; lack of robust password recovery mechanisms; exposed credentials; lack of account lockout; susceptibility to cross-site scripting, cross-site request forgery, and/or SQL injection.
Insufficient Security Configurability	AVERAGE: Attacker uses the lack of granular permissions to access data or controls on the device. The attacker could also use the lack of encryption options and lack of password options to perform other attacks which lead to compromise of the device and/or data. Attack could potentially come from any user of the device whether intentional or accidental.	EASY: Insufficient security configurability is present when users of the device have limited or no ability to alter its security controls. Insufficient security configurability is apparent when the web interface of the device has no options for creating granular user permissions. Manual review of the web interface will reveal these deficiencies.	MODERATE: Insufficient security configurability could lead to compromise of the device whether intentional or accidental and/or data loss.	Lack of granular permissions model; inability to separate administrators from users; weak password policies; no security logging; lack of data encryption options; no user notification of security events.
Insecure Software/ Firmware	DIFFICULT: Attacker uses capturing update files via unencrypted connections, the update file itself is not encrypted or they are able to perform their own malicious update via DNS hijacking. Attack could come from the local network or the internet.	EASY: The lack of ability for a device to be updated presents a security weakness on its own. Devices should have the ability to be updated when vulnerabilities are discovered and software/firmware updates can be insecure when the updated files themselves and the network connection they are delivered on are not protected. Software/Firmware issues are easy to discover by simply inspecting the network traffic during the update to check for encryption or using a hex editor to inspect the update file itself for interesting information.	SEVERE: Insecure software/firmware could lead to compromise of user data, control over the device and attacks against other devices.	Lack of secure update mechanism; update files not encrypted; update files not verified before upload; insecure update server; hardcoded credentials.
Poor Physical Security	AVERAGE: Attacker uses vectors such as USB ports, SD cards or other storage means to access the Operating System and potentially any data stored on the device. Attack could come from anyone who has physical access.	AVERAGE: Physical security weaknesses are present when an attacker can disassemble a device to easily access the storage medium and any data stored on that medium. Weaknesses are also present when USB ports or other external ports can be used to access the device using features intended for configuration or maintenance.	SEVERE: Insufficient physical security could lead to compromise of the device itself and any data stored on that device.	Device easy to disassemble; access to software via USB ports; removable storage media.

Botnets and DDoS

A **denial-of-service attack (DoS attack)** is a cyber-attack where the perpetrator seeks to make a machine or network resource unavailable to its intended users by temporarily or indefinitely

disrupting services of a host connected to the Internet. Denial of service is typically accomplished by flooding the targeted machine or resource with superfluous requests in an attempt to overload systems and prevent some or all legitimate requests from being fulfilled. In a **distributed denial-of-service attack (DDoS attack)**, the incoming traffic flooding the victim originates from many different sources. This effectively makes it impossible to stop the attack simply by blocking a single source.
From these definitions, one can easyly understand that **Distributed Denial of Service (DDoS) attacks** constitute one of the major threats and among the hardest security problems in today’s Internet and thier impact can be proportionally severe. On the

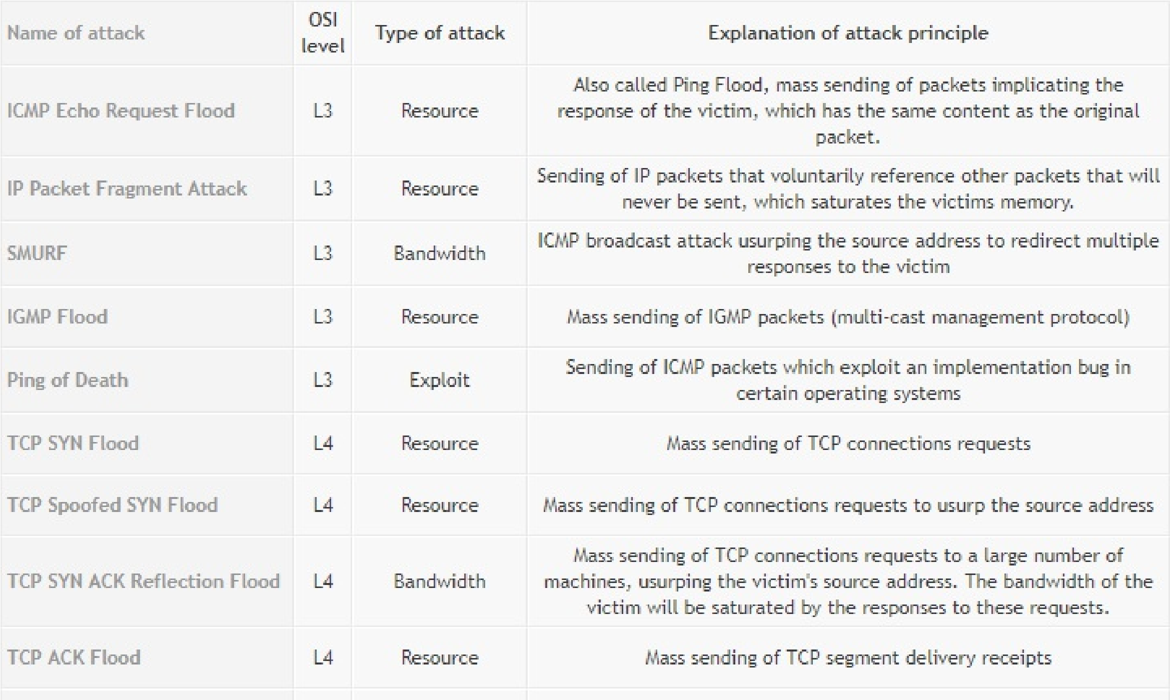


Types of DDoS attacks

DDoS attactks can be implemented using three main stategies:

- **Traffic attacks**: Traffic flooding attacks send a huge volume of TCP, UDP and ICPM packets to the target. Legitimate requests get lost and these attacks may be accompanied by malware exploitation.
- **Bandwidth attacks**: This DDos attack overloads the target with massive amounts of junk data. This results in a loss of network bandwidth and equipment resources and can lead to a complete denial of service.
- **Application attacks**: Application-layer data messages can deplete resources in the

application layer, leaving the target's system services unavailable.
Different types of attacks fall into categories based on the traffic quantity and the vulnerabilities being targeted.
Here is a list of the most popular types of DDoS attacks:



Name of attack	OSI level	Type of attack	Explanation of attack principle
ICMP Echo Request Flood	L3	Resource	Also called Ping Flood, mass sending of packets implicating the response of the victim, which has the same content as the original packet.
IP Packet Fragment Attack	L3	Resource	Sending of IP packets that voluntarily reference other packets that will never be sent, which saturates the victims memory.
SMURF	L3	Bandwidth	ICMP broadcast attack usurping the source address to redirect multiple responses to the victim
IGMP Flood	L3	Resource	Mass sending of IGMP packets (multi-cast management protocol)
Ping of Death	L3	Exploit	Sending of ICMP packets which exploit an implementation bug in certain operating systems
TCP SYN Flood	L4	Resource	Mass sending of TCP connections requests
TCP Spoofed SYN Flood	L4	Resource	Mass sending of TCP connections requests to usurp the source address
TCP SYN ACK Reflection Flood	L4	Bandwidth	Mass sending of TCP connections requests to a large number of machines, usurping the victim's source address. The bandwidth of the victim will be saturated by the responses to these requests.
TCP ACK Flood	L4	Resource	Mass sending of TCP segment delivery receipts

What is a Botnet?

Occasionally referred to as a "**zombie army,**" a **botnet** is a group of hijacked Internet-connected devices, each injected with malware used to control it from a remote location without the knowledge of the device's rightful owner. From the point of view of hackers, these botnet devices are computing resources that can be used for any type of malicious purposes—most commonly for spam or DDoS attacks.

How is a botnet controlled?

A core characteristic of a botnet is the ability to receive updated instructions from the bot herder. The ability to communicate with each bot in the network allows the attacker to alternate attack vectors, change the targeted IP address, terminate an attack, and other customized actions. Botnet designs vary, but the control structures can be broken down into two general categories:

1. **The client/server botnet model**
   These botnets operate through Internet Relay Chat networks, domains, or websites. Infected clients access a predetermined location and await incoming commands from the server. The bot herder sends commands to the server, which relays them to the clients. Clients execute the commands and report their results back to the bot herder.
2. **The peer-to-peer botnet model**
   To circumvent the vulnerabilities of the client/server model, botnets have more recently been designed using components of decentralized peer-to-peer filesharing. Embedding the control structure inside the botnet eliminates the single point-of-failure present in a botnet with a

centralized server, making mitigation efforts more difficult. P2P bots can be both clients and command centers, working hand-in-hand with their neighboring nodes to propagate data.

How do IoT devices become a botnet?

No one does their Internet banking through the wireless CCTV camera they put in the backyard to watch the bird feeder, but that doesn't mean the device is incapable of making the necessary network requests. The power of IoT devices coupled with weak or poorly configured security creates an opening for botnet malware to recruit new bots into the collective. An uptick in IoT devices has resulted in a new landscape for DDoS attacks, as many devices are poorly configured and vulnerable. If an IoT device's vulnerability is hardcoded into firmware, updates are more difficult. To mitigate risk, IoT devices with outdated firmware should be updated as default credentials commonly remain unchanged from the initial installation of the device. Many discount manufacturers of hardware are not incentivized to make their devices more secure, making the vulnerability posed from botnet malware to IoT devices remain an unsolved security risk.

Just to have an idea of the consequences of theses attacks, remeber the famous **"Dyn Botnet DDos cyberattack"** that took place on October 21, 2016. The victim was the servers of Dyn, a company that controls much of the internet's domain name system (DNS) infrastructure. It remained under sustained assault for most of the day, bringing down sites including Twitter, the Guardian, Netflix, Reddit, Paypal, CNN and many others in Europe and the US.

Bibliography

C. Kolias, G. Kambourakis, A. Stavrou, J.Voas,"DDoS in the IoT: Mirai and Other Botnets", IEEE Computer Society, 2017

M. Antonakakis, T. April, M. Bailey, M. Bernhard, E. Bursztein, J. Cochran, Z. Durumeric, J. A. Halderman, L. Invernizzi, M. Kallitsis, D. Kumar, C. Lever, Z. Ma, J. Mason, D. Menscher, C. Seaman, Nick Sullivan, K. Thomas, Y. Zhou, "Understanding the Mirai Botnet", Proceedings of the 26th USENIX Security Symposium, August 16–18 2017 ,Vancouver (Canada)

B. Herzberg, D. Bekerman, I. Zeifman, "Breaking Down Mirai: An IOT DDoS Botnet Analysis", Blog (https://WWW.INCAPSULA.COM/BLOG/CATEGORY/BLOG), October 2016

R. Graham, "Mirai and IoT Botnet Analysis", RSA Conference 2017, February 13-17, San Francisco

S. Jasek, "Mirai botnet: intro to discussion", OWASP, Krakow, 2016/11/15

H. Sinanovi´c, S. Mrdovic, "Analysis of Mirai Malicious Software", University of Sarajevo

N. B. Said, F. Biondi, V. Bontchev, O. Decourbe,T. Given-Wilson, A. Legay, J. Quilbeuf,"Detection of Mirai by Syntactic and Semantic Analysis", HAL Id: hal-01629040, https://hal.inria.fr/hal-01629040, 5 Nov 2017

B. Botticelli, “IoT Honeypots: State of the Art”, Seminar in Advanced Topics in Computer Science, Università di Roma Sapienza, September 2, 2017