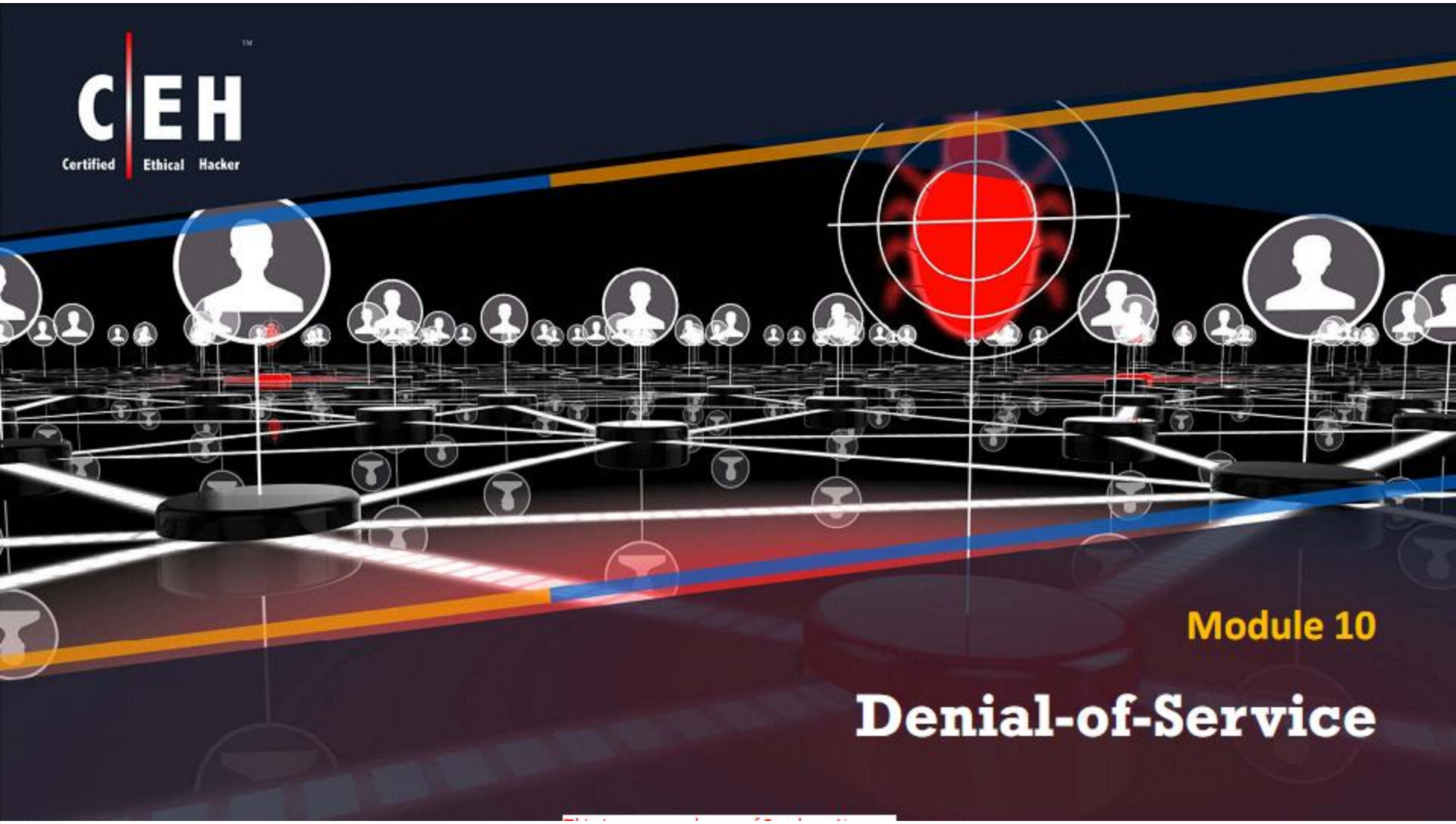
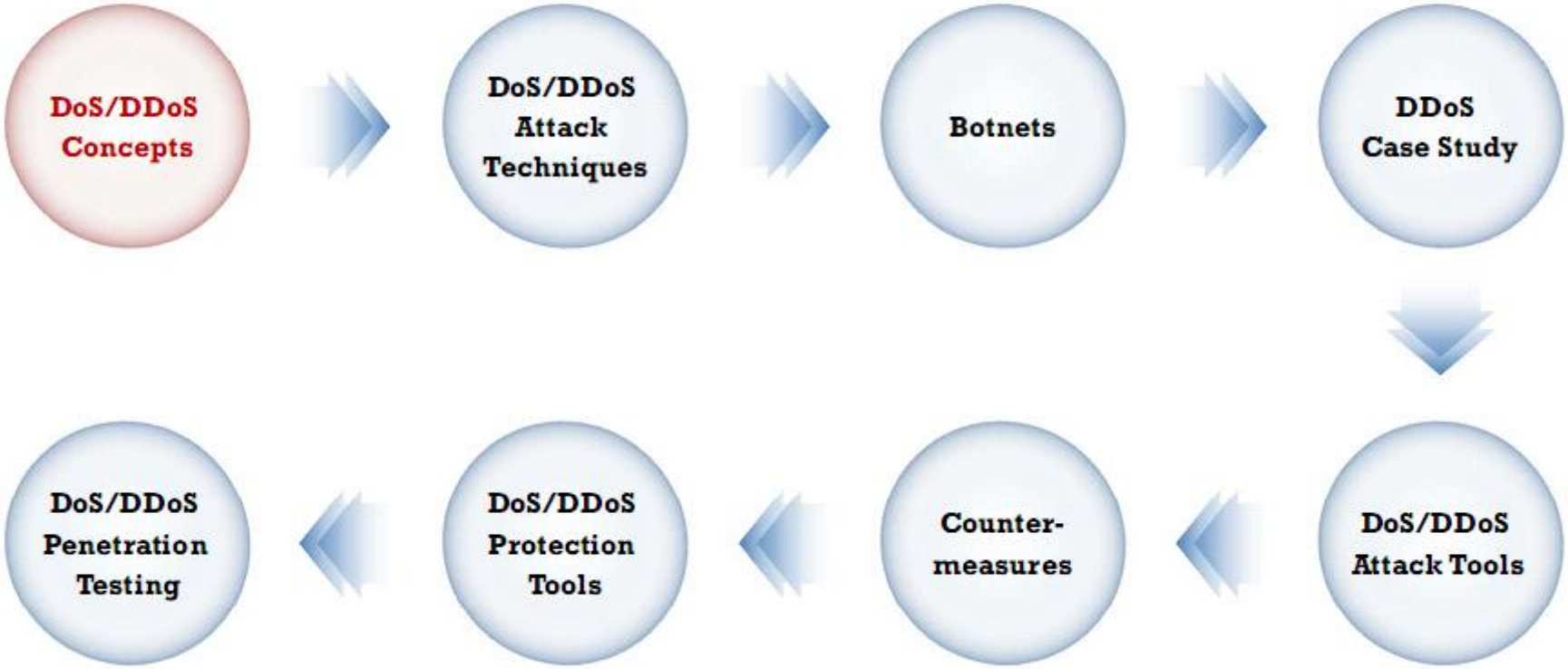

C|EH
Certified Ethical Hacker
Module 10
Denial-of-Service

Module Objectives

Module Objectives

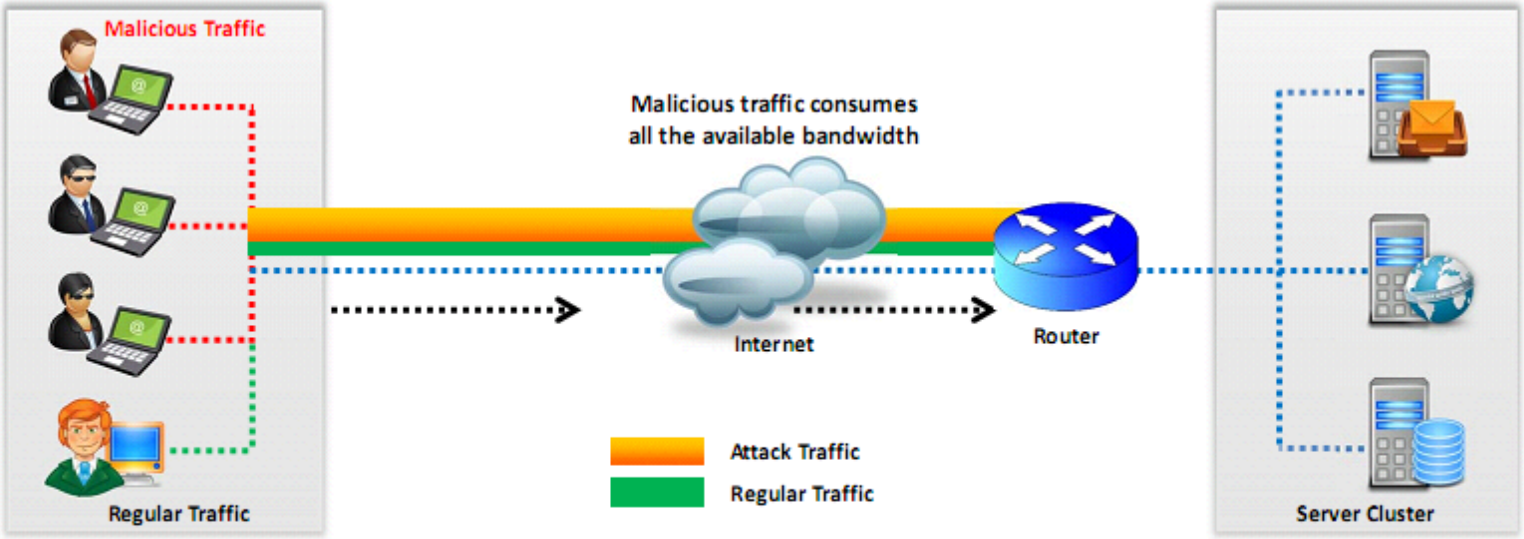
- Overview of Denial-of-Service (DoS) and Distributed Denial-of-Service (DDoS) Attacks
- Understanding Different DoS/DDoS Attack Techniques
- Understanding the Botnet Network
- Understanding Various DoS and DDoS Attack Tools
- Understanding Different Techniques to Detect DoS and DDoS Attacks
- Understanding Different DoS/DDoS Countermeasures
- Overview of DoS Attack Penetration Testing

Module Flow



What is a Denial-of-Service Attack?

- Denial-of-Service (DoS) is an attack on a computer or network that **reduces**, **restricts**, or **prevents** accessibility of system resources to its legitimate users
- In a DoS attack, attackers flood the victim system with **non-legitimate service requests or traffic** to overload its resources



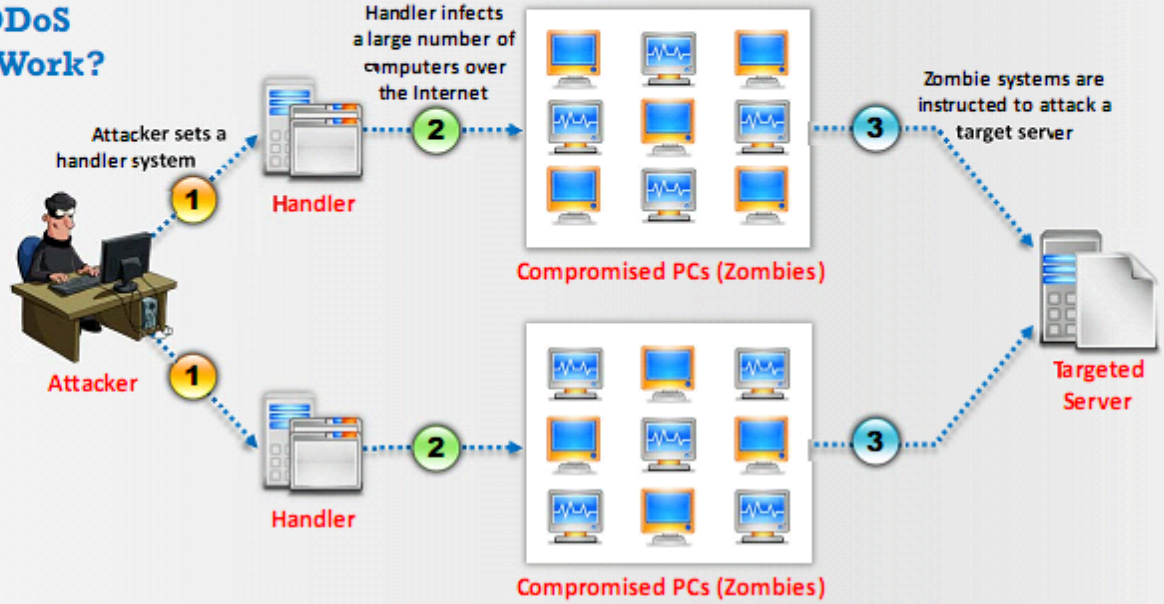
What is Distributed Denial-of-Service Attack?

- Distributed denial-of-service (DDoS) is a coordinated attack which involves a **multitude of compromised systems** (Botnet) attacking a single target; thereby causing denial of service for users of the targeted system

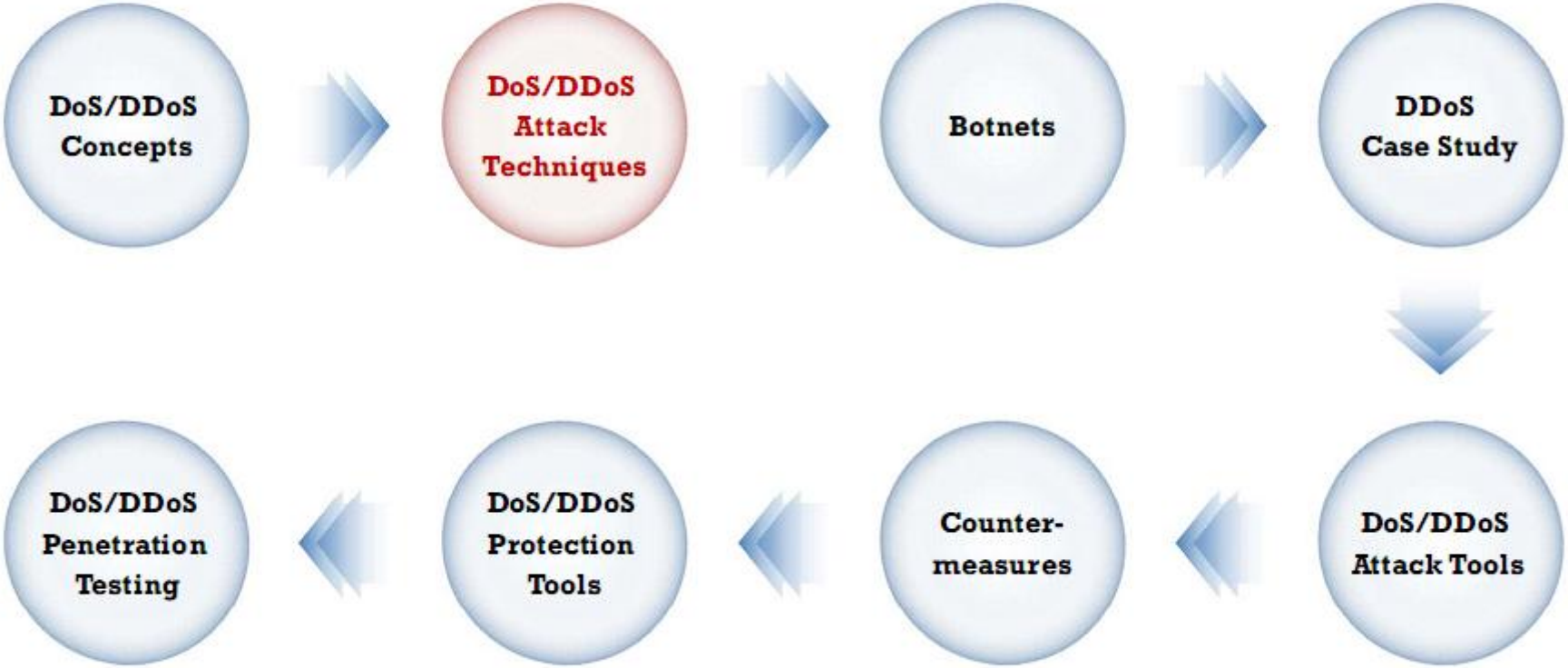
DDoS Impact

- Loss of Goodwill
- Disabled Network
- Financial Loss
- Disabled Organization

How DDoS Attacks Work?



Module Flow



Basic Categories of DoS/DDoS Attack Vectors

Volumetric Attacks

- Consumes the bandwidth of target network or service
- The magnitude of attack is measured in **bits-per-second (bps)**
- Types of bandwidth depletion attacks:
  - Flood attacks
  - Amplification attacks

Attack Techniques

- UDP flood attack
- ICMP flood attack
- Ping of Death attack
- Smurf attack

Protocol Attacks

- Consumes other types of resources like **connection state tables** present in the network infrastructure components such as **load-balancers**, **firewalls**, and **application servers**
- The magnitude of attack is measured in **packets-per-second (pps)**

Attack Techniques

- SYN flood attack
- Fragmentation attack
- ACK flood attack
- TCP state exhaustion attack

Application Layer Attacks

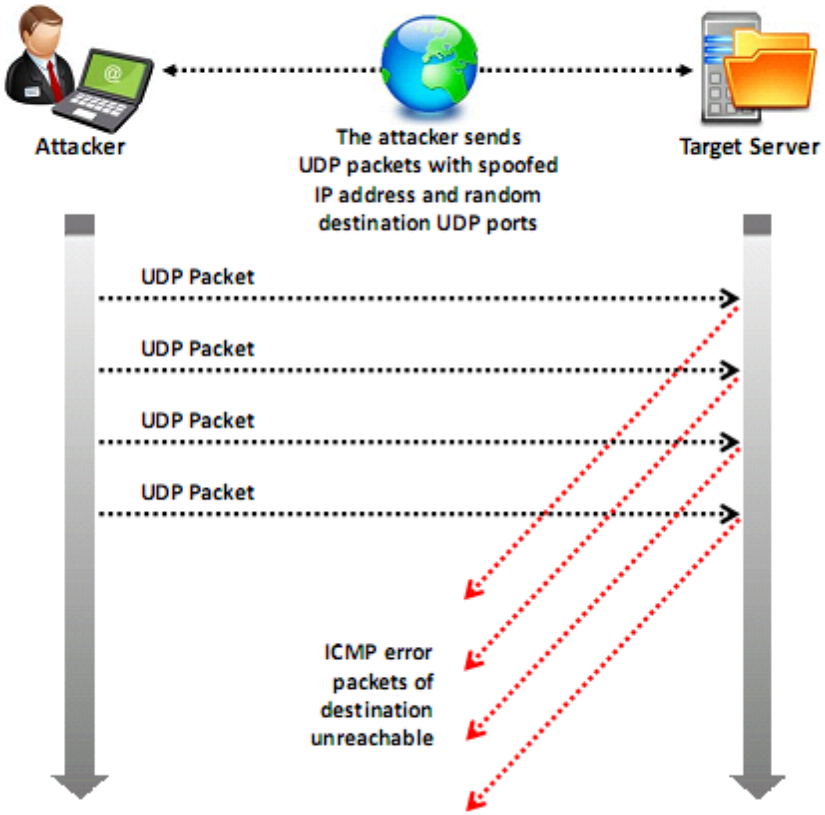
- Consumes the **application resources** or service thereby making it unavailable to other legitimate users
- The magnitude of attack is measured in **requests-per-second (rps)**

Attack Techniques

- HTTP GET/POST attack
- Slowloris attack

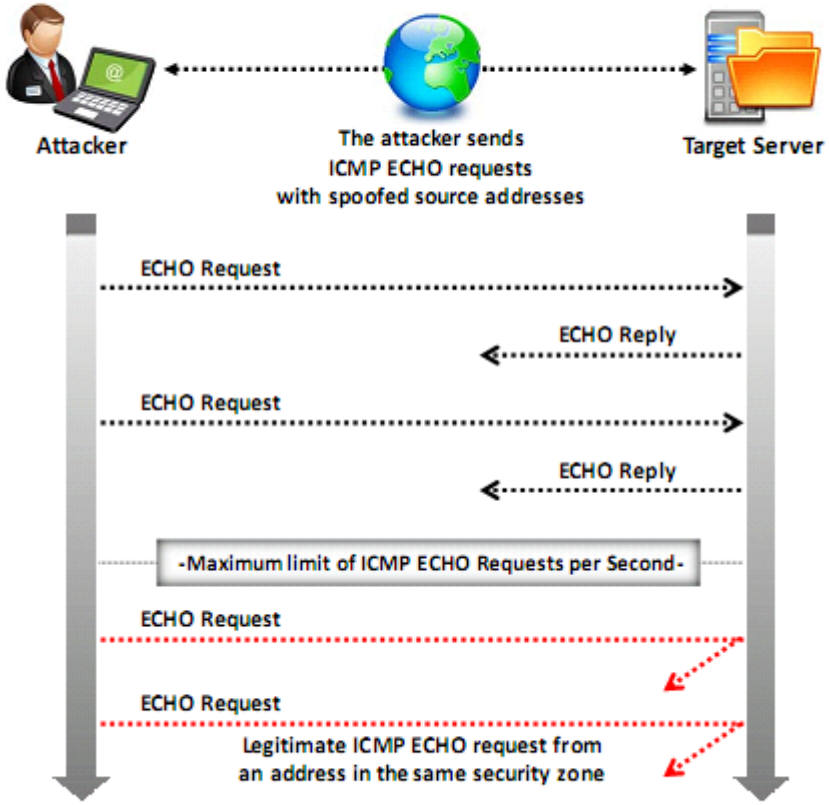
UDP Flood Attack

- An attacker sends **spoofed UDP packets** at a very high packet rate to a remote host on random ports of a target server using a large source IP range
- Flooding of UDP packets causes server to repeatedly check for **non-existent applications** at the ports
- Legitimate applications are inaccessible by the system and gives a **error reply** with an ICMP 'Destination Unreachable' packet
- This attack consumes **network resources** and available bandwidth, exhausting the network until it goes offline



ICMP Flood Attack

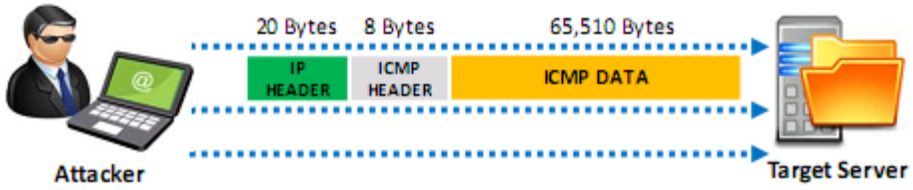
- Network administrators use ICMP primarily for IP operations, troubleshooting, and error messaging of **undeliverable packets**
- ICMP flood attack is a type of attack in which attackers send large volumes of **ICMP echo request packets** to a victim system directly or through reflection networks
- These packets signal the victim's system to reply and the combination of traffic saturates the bandwidth of the victim's network connection causing it to be overwhelmed and **subsequently stop** responding to legitimate TCP/IP requests
- To protect against ICMP flood attack, set a **threshold limit**, which when exceeded invokes the ICMP flood attack protection feature



Ping of Death and Smurf Attack

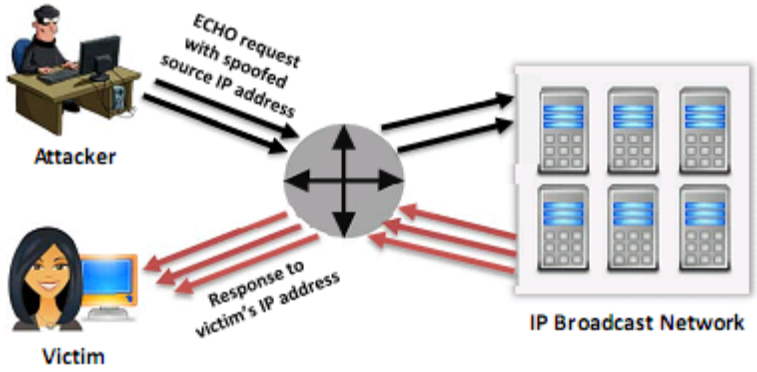
Ping of Death Attack

- In Ping of Death (PoD) attack, an attacker tries to crash, destabilize, or freeze the targeted system or service by **sending malformed or oversized packets** using a simple ping command
- For instance, the attacker sends a packet which has a size of 65,538 bytes to the target web server. This **size** of the **packet exceeds** the size **limit prescribed** by **RFC 791 IP** which is 65,535 bytes. The reassembly process by the receiving system might cause the system to crash



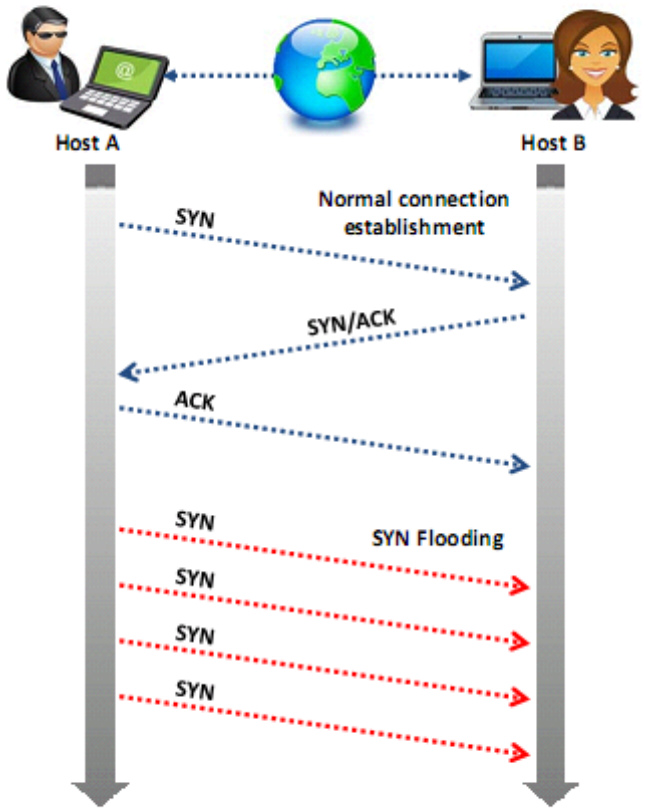
Smurf Attack

- In Smurf attack, the attacker spoofs the **source IP address** with the victim's IP address and sends **large number of ICMP ECHO request packets** to an IP broadcast network
- This cause all the hosts on the broadcast network to respond to the received **ICMP ECHO** requests. These responses will be sent to the victim machine, ultimately leading the machine to crash



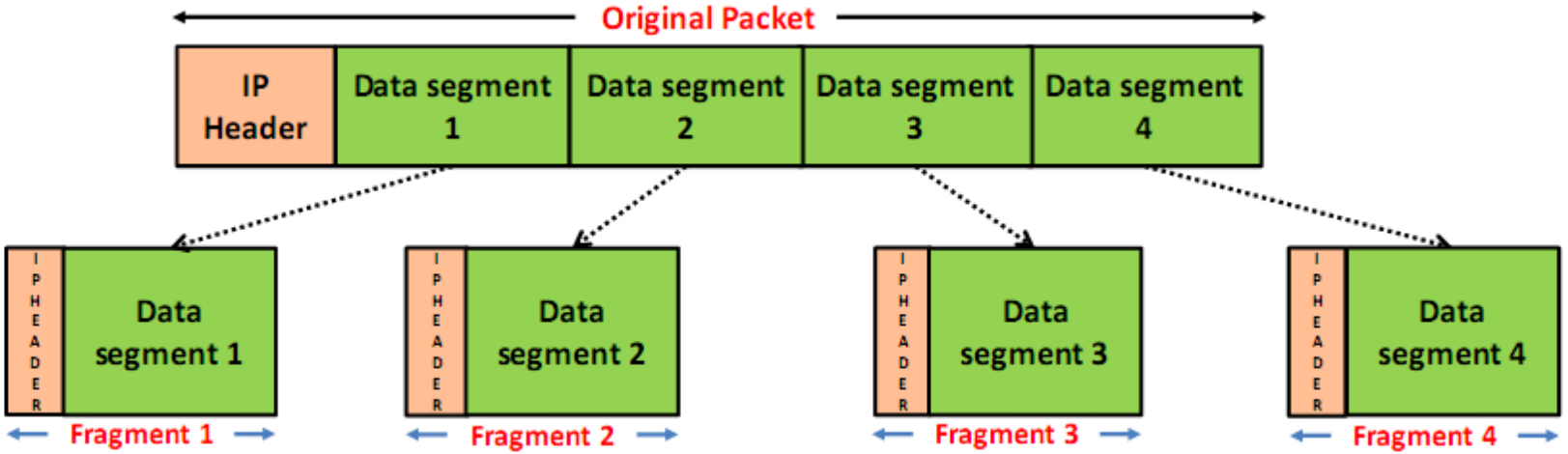
SYN Flood Attack

- The attacker sends a large number of **SYN request** to target server (victim) with **fake source IP addresses**
- The target machine sends back a **SYN ACK** in **response to the request** and waits for the ACK to complete the session setup
- The target machine **does not get the response** because the **source address is fake**
- SYN Flooding takes advantage of a flaw in the way most hosts implement the **TCP three-way handshake**
- When **Host B** receives the **SYN** request from Host A, it must keep track of the partially-opened connection in a "**listen queue**" for **at least 75 seconds**
- A malicious host can exploit the small size of the listen queue by **sending multiple SYN requests** to a host, but **never replying to the SYN/ACK**
- The victim's listen queue is quickly filled up
- This ability of **holding up** each incomplete **connection for 75 seconds** can be cumulatively used as a **Denial-of-Service attack**



Fragmentation Attack

- These attacks destroy a victim's ability to **re-assemble the fragmented packets** by flooding it with TCP or UDP fragments, resulting in reduced performance. Attacker sends large number of fragmented (1500+ byte) packets to a **target web server** with relatively small packet rate
- Since the protocol allows the fragmentation, these packets usually pass through the network equipments like routers, firewalls, IDS/IPS, etc. uninspected
- Reassembling and inspecting these large fragmented packets consumes excessive resources. Moreover the **content in the packet fragments** will be randomized by the attacker, which makes the process to consume more resource and leading the system to crash

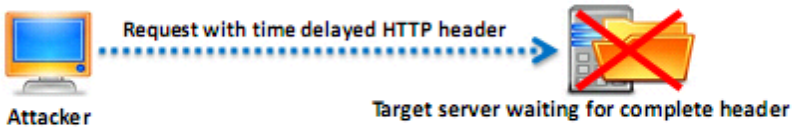


HTTP GET/POST and Slowloris Attacks

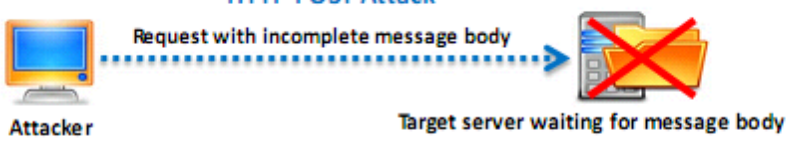
HTTP GET/POST Attack

- HTTP Clients such as web browsers, etc. connect to a web server through HTTP protocol to send HTTP requests. These requests can be either HTTP GET or HTTP POST
- In HTTP GET attack, the attackers use time delayed HTTP header to hold on to HTTP connections and exhaust web server resources
- In HTTP POST attack, the attacker sends the HTTP requests with complete headers but incomplete message body to the target web server or application making the server wait for the rest of the message body

HTTP GET Attack



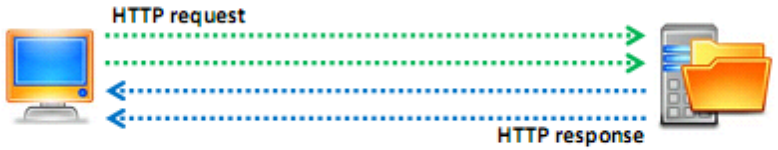
HTTP POST Attack



Slowloris Attack

- In the Slowloris attack, the attacker sends partial HTTP requests to the target web server or application
- Upon receiving the partial HTTP requests, the target server opens multiple open connections and keeps waiting for the requests to complete
- These requests will not be complete and as a result, the target server's maximum concurrent connection pool will be filled up and additional connection attempts will be denied

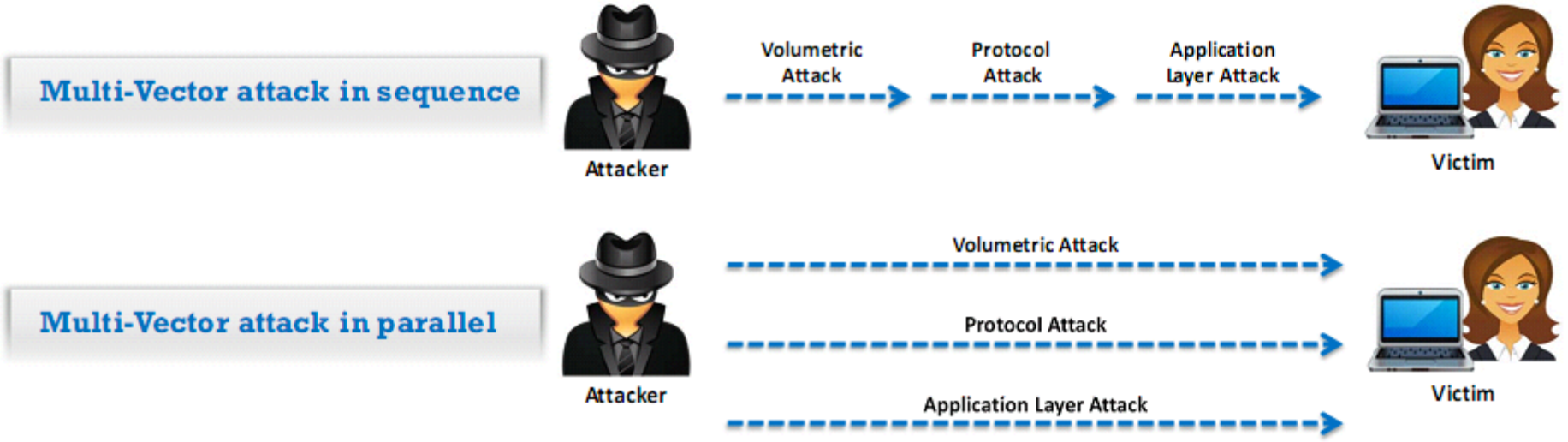
Normal HTTP request-response connection



Slowloris DDoS attack

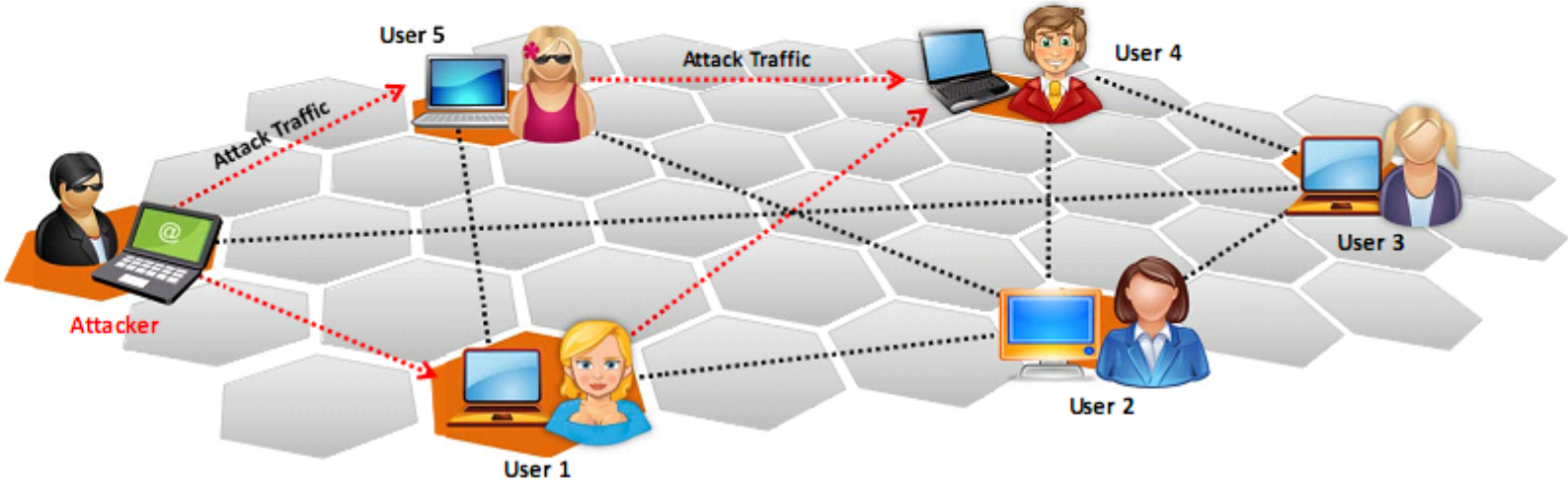
Multi-Vector Attack

- In multi-vector DDoS attacks, the attackers use **combinations of volumetric**, protocol, and application-layer attacks to take down the target system or service
- Attacker quickly changes from one form of DDoS attack (e.g.: SYN packets) to another (Layer 7), and so on
- These attacks are either **launched one vector at a time** or in parallel, in order to confuse a company's IT department and to make them spend all their resources and divert their focus to the wrong side



Peer-to-Peer Attacks

- Using peer-to-peer attacks, attackers **instruct clients of peer-to-peer file sharing hubs** to disconnect from their peer-to-peer network and to connect to the victim's fake website
- Attackers **exploit flaws** found in the network using DC++ (Direct Connect) protocol that is used for sharing all types of files between instant messaging clients
- Using this method, attackers launch **massive denial-of-service attacks** and compromise websites



Permanent Denial-of-Service Attack

Phlashing

Permanent DoS, also known as **phlashing**, refers to attacks that cause irreversible damage to system hardware

Sabotage

Unlike other DoS attacks, it **sabotages the system hardware**, requiring the victim to replace or reinstall the hardware

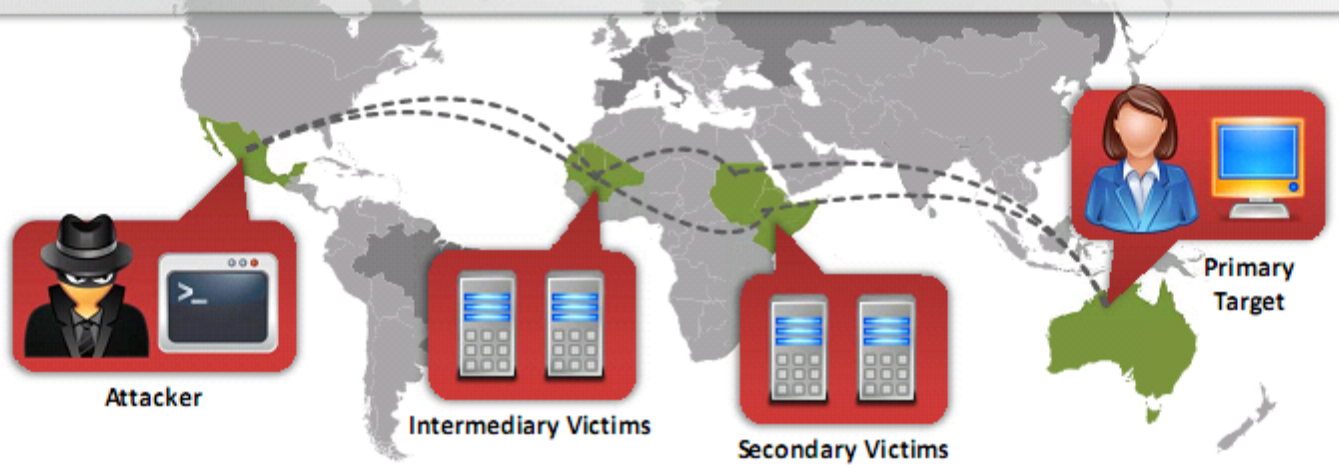
Bricking a system

- This attack is carried out using a method known as "**bricking a system**"
- Using this method, attackers send **fraudulent hardware updates** to the victims

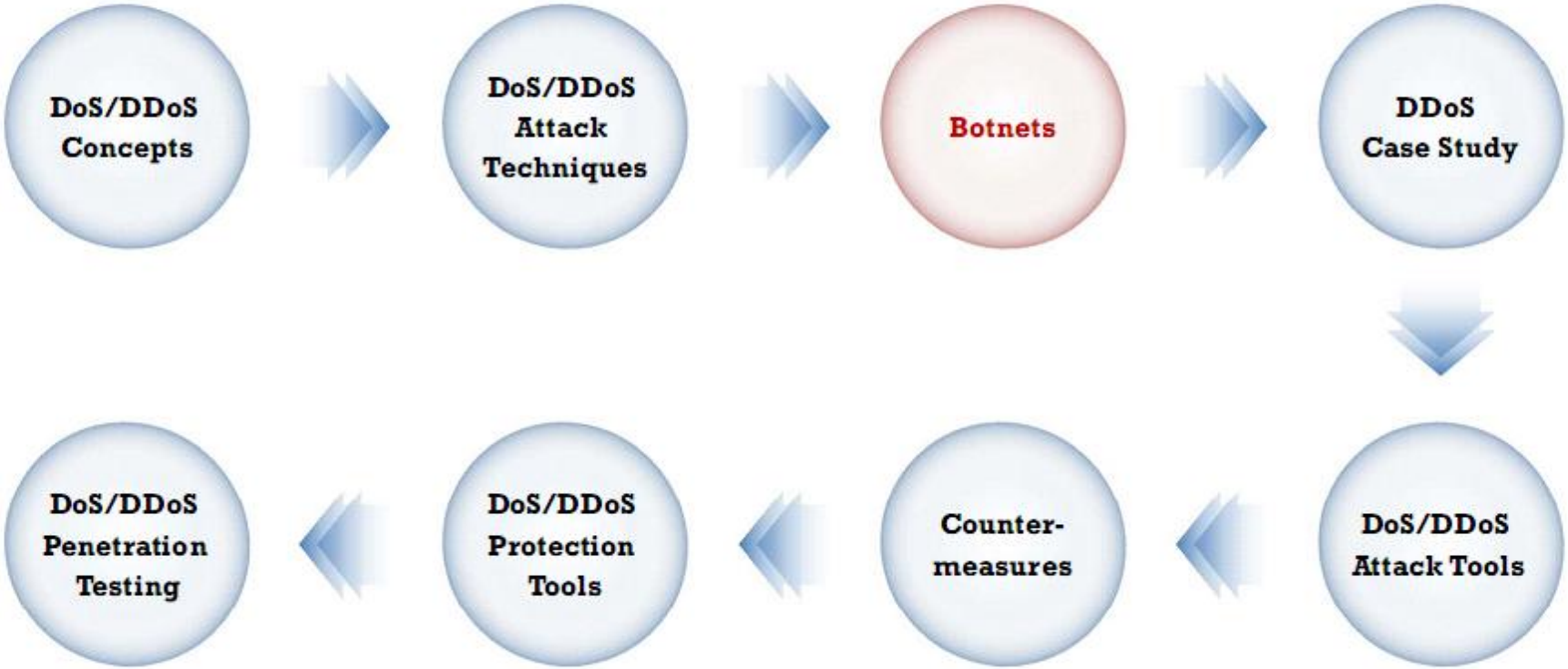
Process

Distributed Reflection Denial of Service (DRDoS)

- A distributed reflected denial of service attack (DRDoS), also known as spoofed attack, involves the **use of multiple intermediary and secondary machines** that contribute to the actual DDoS attack against the target machine or application
- Attacker launches this attack by sending requests to the intermediary hosts; these requests are then redirected to the secondary machines which in turn **reflects the attack traffic to the target**
- **Advantage**:
  - The primary target seems to be directly attacked by the secondary victim, not the actual attacker
  - Multiple intermediary victim servers are used, which results in increase in attack bandwidth

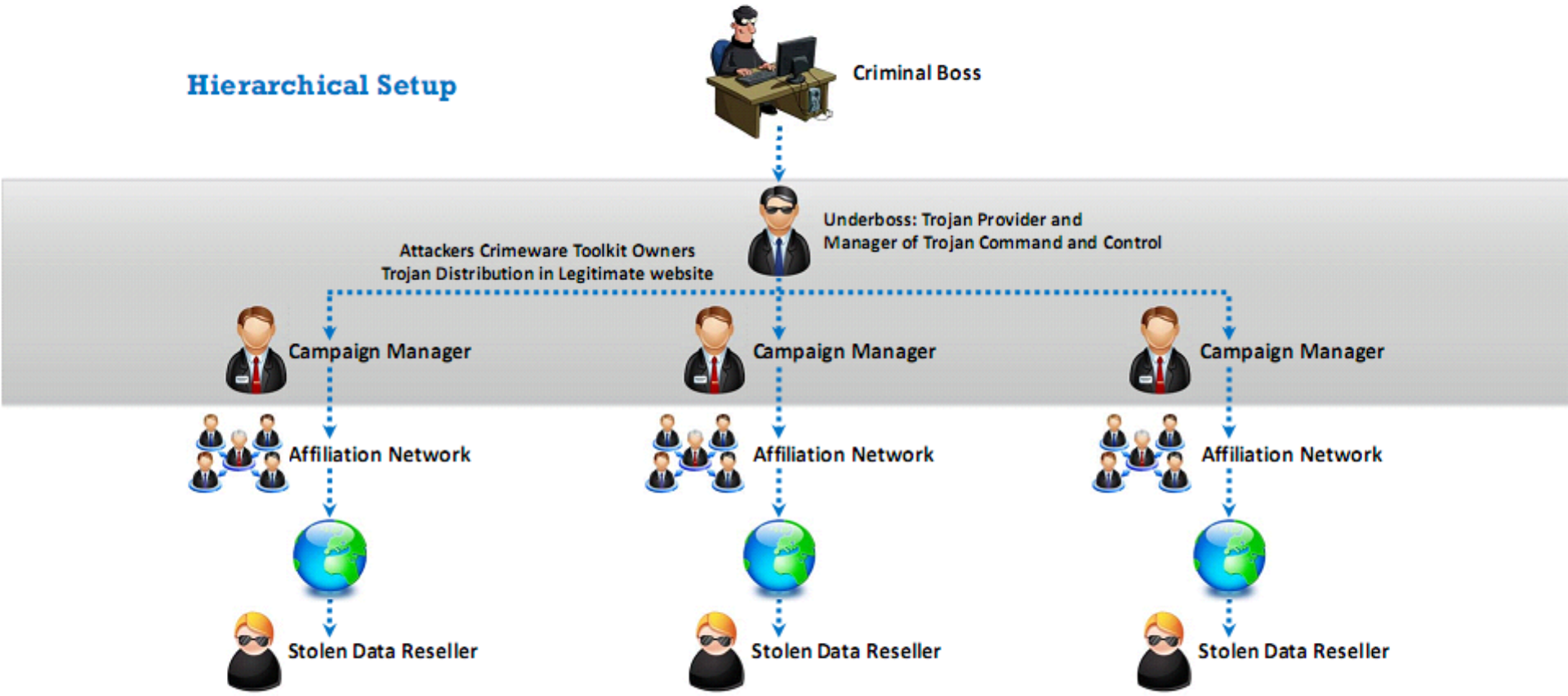


Module Flow



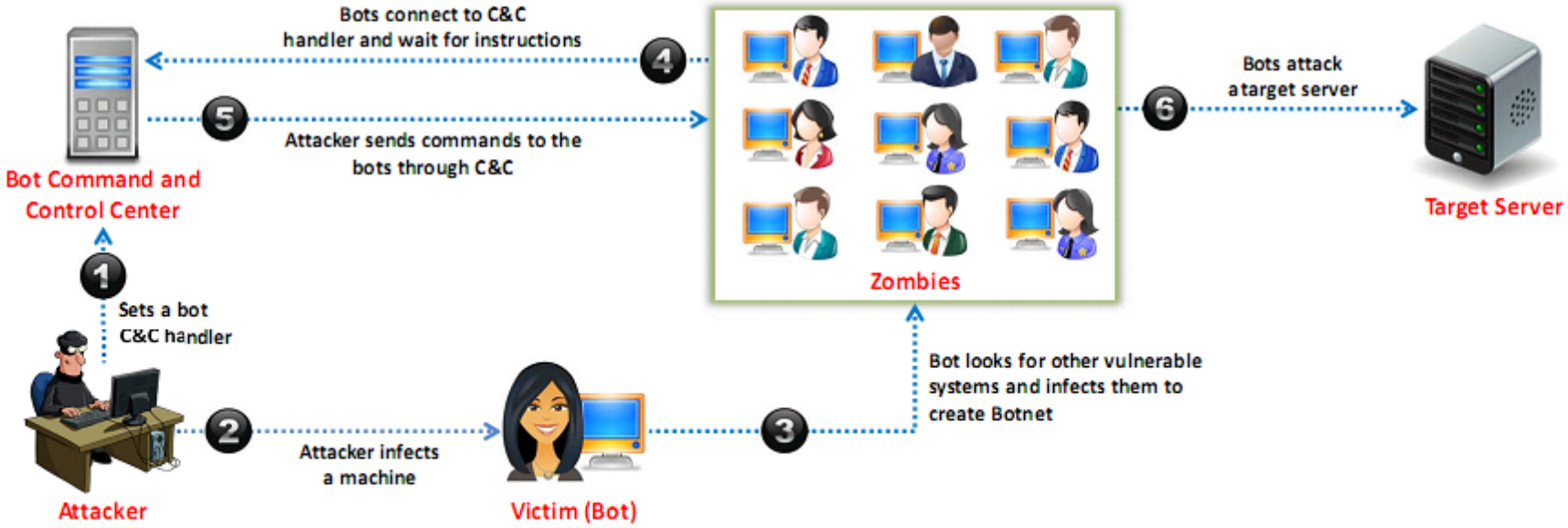
Organized Cyber Crime: Organizational Chart

Hierarchical Setup

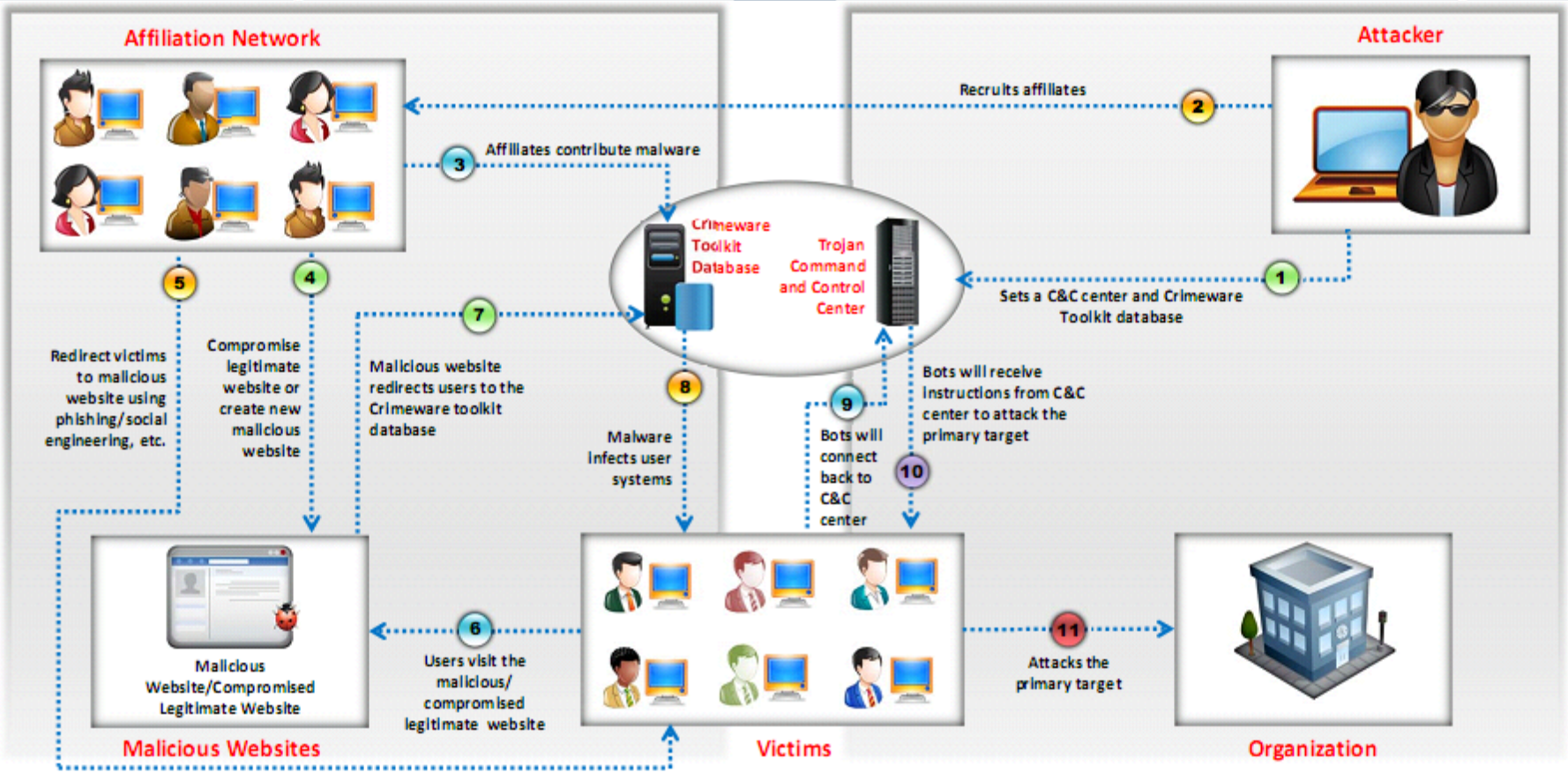


Botnet

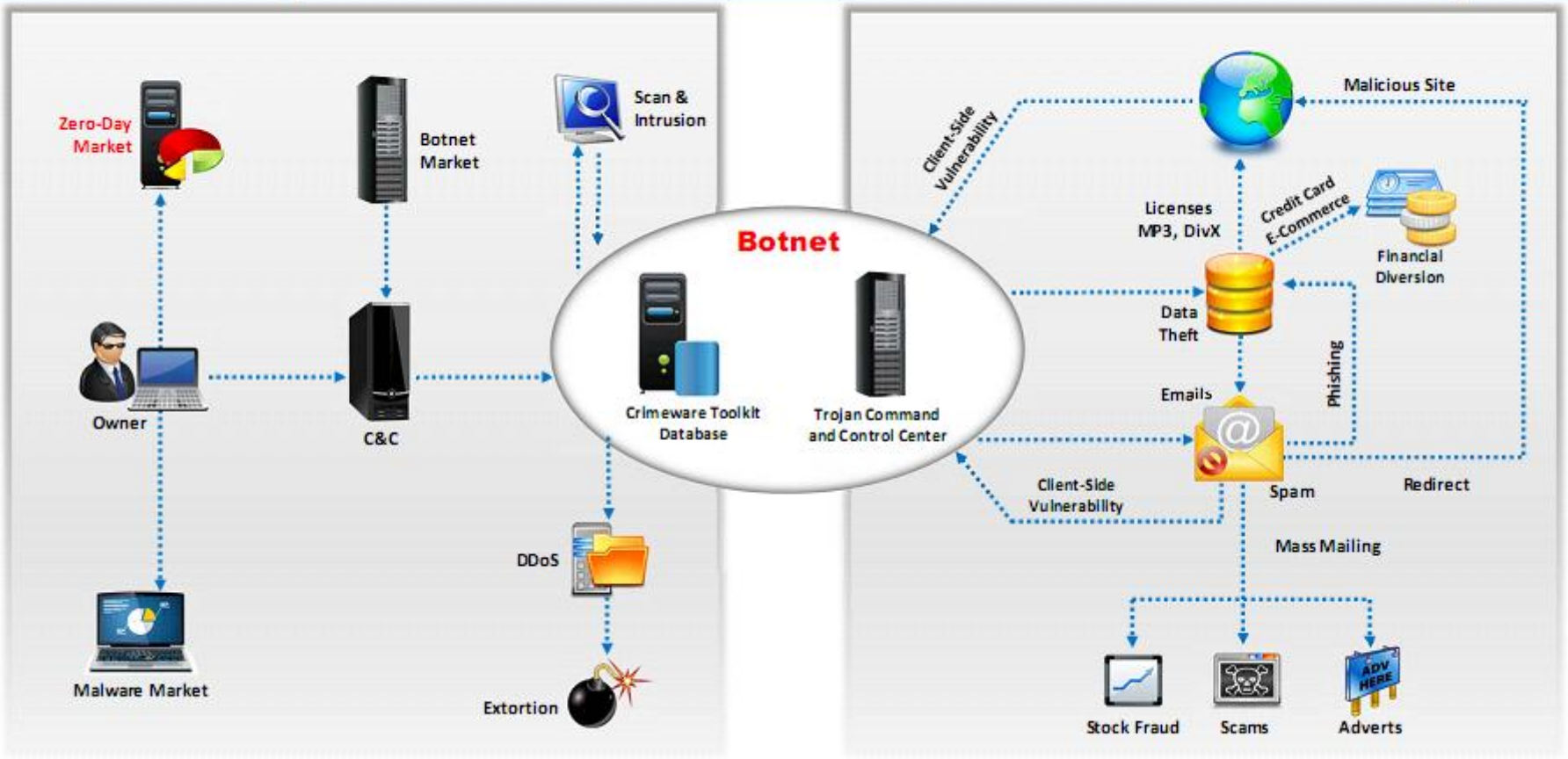
- Bots are software applications that **run automated tasks over the Internet** and perform simple repetitive tasks, such as web spidering and search engine indexing
- A botnet is a huge network of compromised systems and can be used by an attacker to **launch denial-of-service attacks**



A Typical Botnet Setup



Botnet Ecosystem



Scanning Methods for Finding Vulnerable Machines

Random Scanning	The infected machine probes **IP addresses** randomly from **target network IP range** and checks for vulnerability
Hit-list Scanning	Attacker first collects a list of **potentially vulnerable machines** and then scans them to find vulnerable machine
Topological Scanning	It uses the **information obtained on infected machine** to find new vulnerable machines
Local Subnet Scanning	The infected machine looks for **new vulnerable machines in its own local network**
Permutation Scanning	It uses **pseudorandom permutation list of IP addresses** to find new vulnerable machines

How Malicious Code Propagates?

Attackers use three techniques to propagate malicious code to newly discovered vulnerable system

Attacker places **attack toolkit on the central source** and a copy of the attack toolkit is transferred to the newly discovered vulnerable system

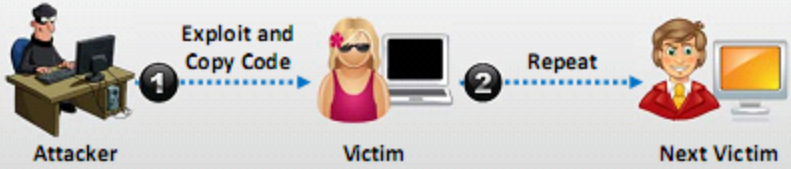
Central Source Propagation

Back-chaining Propagation

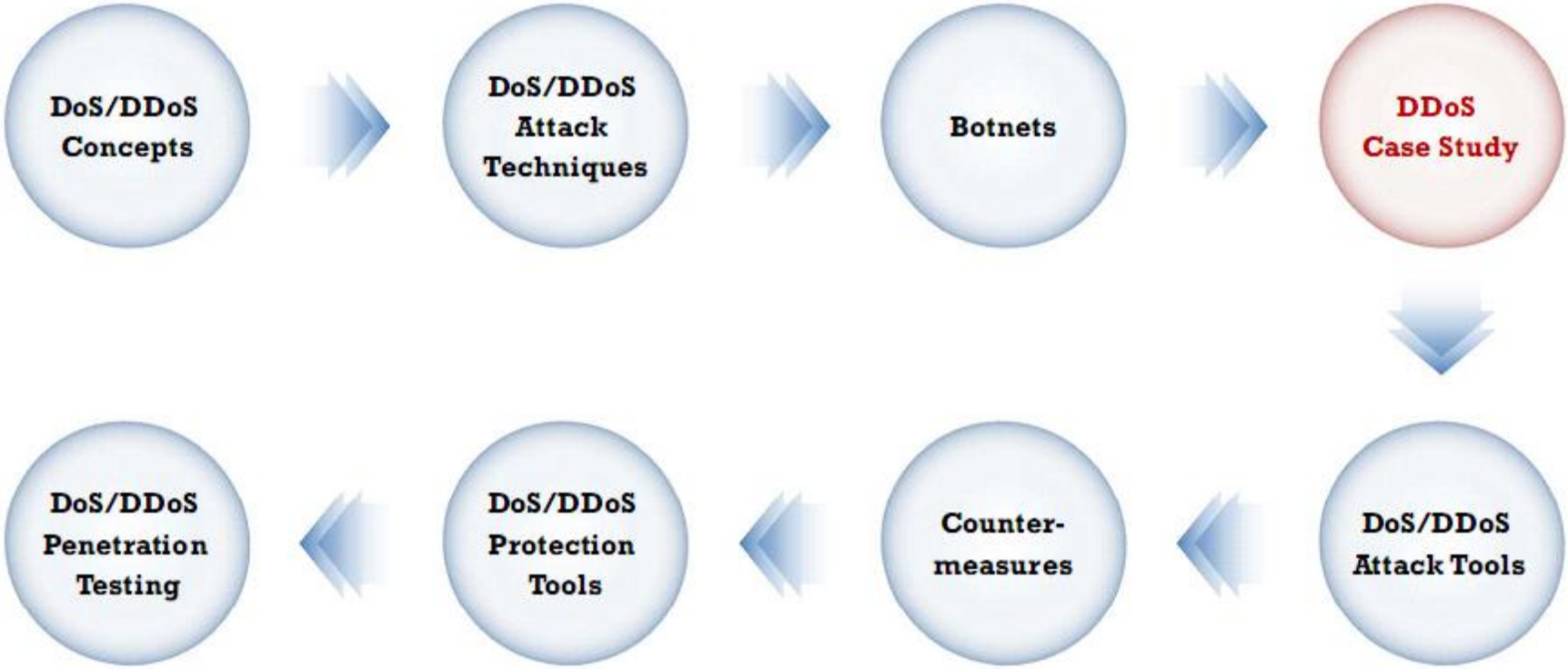
Attacker places **attack toolkit on his/her system itself** and a copy of the attack toolkit is transferred to the newly discovered vulnerable system

Attacking **host itself transfers** the attack toolkit to the newly discovered vulnerable system, exactly **at the time it breaks** into that system

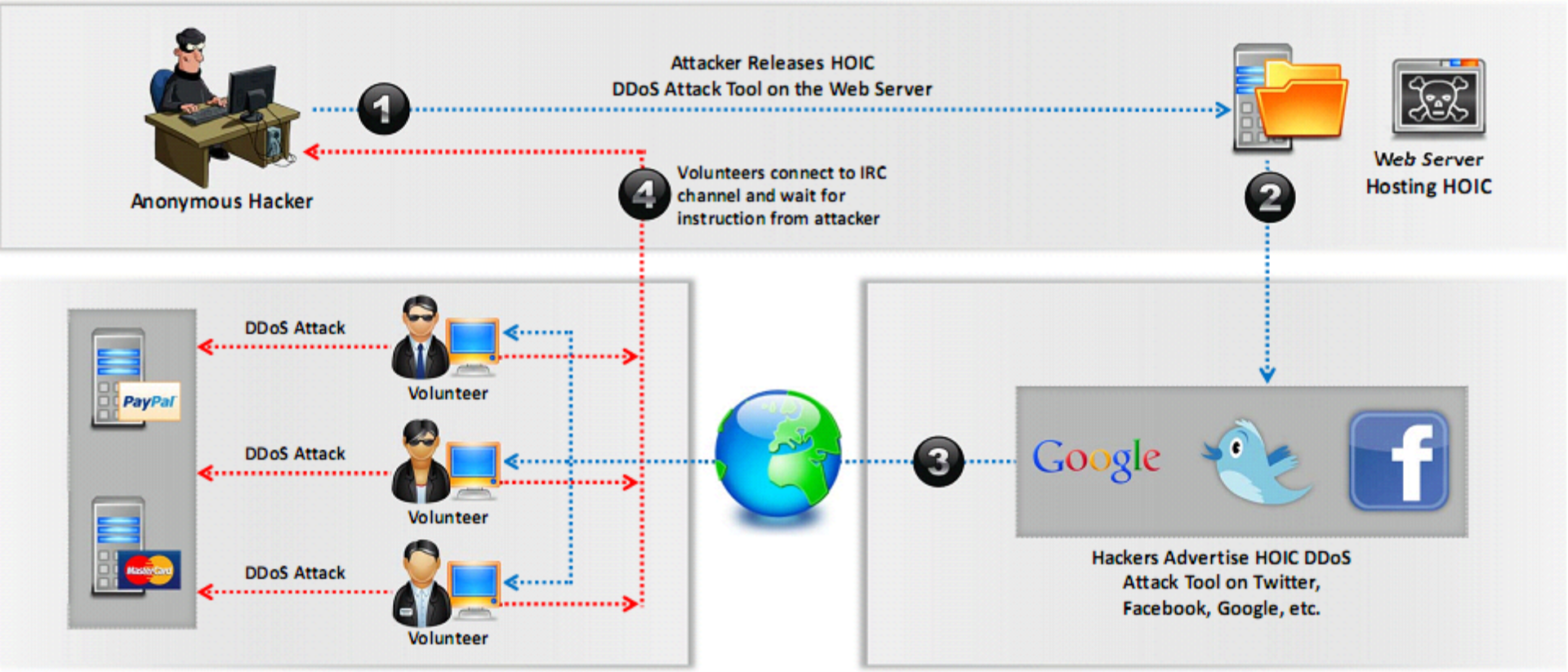
Autonomous Propagation



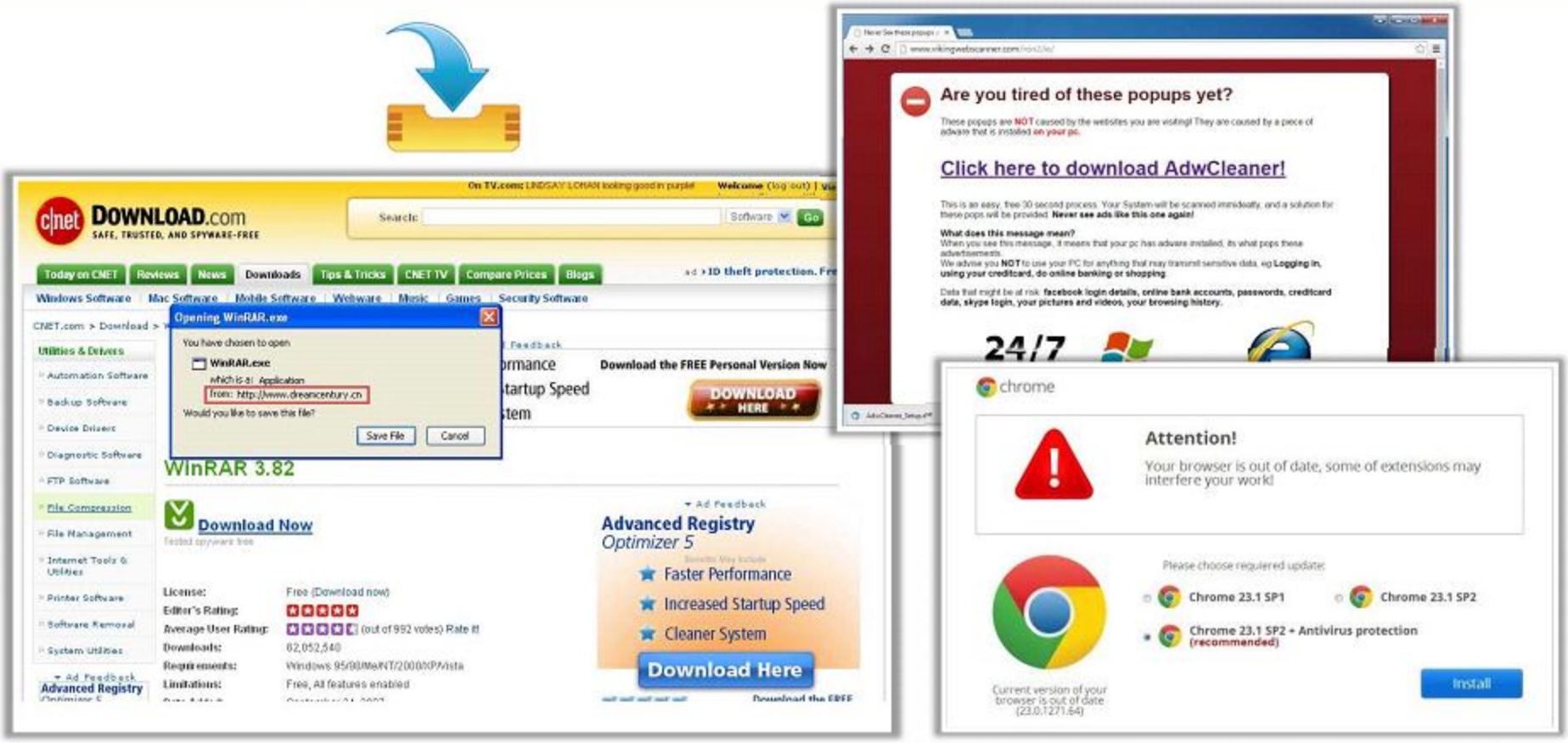
Module Flow



DDoS Attack



Hackers Advertise Links to Download Botnet



Use of Mobile Devices as Botnets for Launching DDoS Attacks

- Android device are passively **vulnerable to various malware** such as Trojan, bots, RATs, etc., which are often found in third-party application stores
- These unsecure android devices are becoming primary targets for the attackers in order to **enlarge their botnet** as they are **highly vulnerable to malware**
- Malicious android applications found in **Google Play store** and **drive-by download** are just a few examples of **infection methods**
- The attacker **binds the malicious APK server** to the android application package (**APK file**), **encrypts** it, and **removes unwanted features** and **permissions** before distributing the malicious package to a **third party app store** like Google Play Store
- Once the user is **tricked to download and install** such application, the victim's device will be taken over by the attacker, **enslaving the targeted device** into the **attacker's mobile botnet** to perform malicious activities like **launch DDoS attacks**, **web injections**, etc.

DDoS Case Study: Dyn DDoS Attack

- Dyn is a cloud based **Internet Performance Management (IPM)** organization that **provides DNS services** to many popular sites such as PayPal, Spotify, Twitter, Amazon, etc.
- The Dyn attack, which took place on 21st October, 2016, is one of the **largest data breaches** in history which overturned a large portion of the internet in the **United States and Europe** and affected plenty of services
- The source of the attack was the **Mirai botnets** and it was launched by exploiting vulnerabilities in insecure Internet-of-Things devices such as internet protocol (IP) cameras, printers, and digital video recorders
- This abrupt large volume of data originated from **various source IP addresses** and were destined for **destination port 53**, where the data packets were composed of **TCP and UDP packets**
- The objective of a Denial of Service (DoS) attack is to deny or disrupt authorized users from accessing a resource or service

Attack Timeline

1st Wave
Between 11:10 UTC to 13:20 UTC – Attack began to target the Dyn infrastructure in US-East region

Recovery
Mitigation efforts for first attack were fully deployed by 13:20 UTC

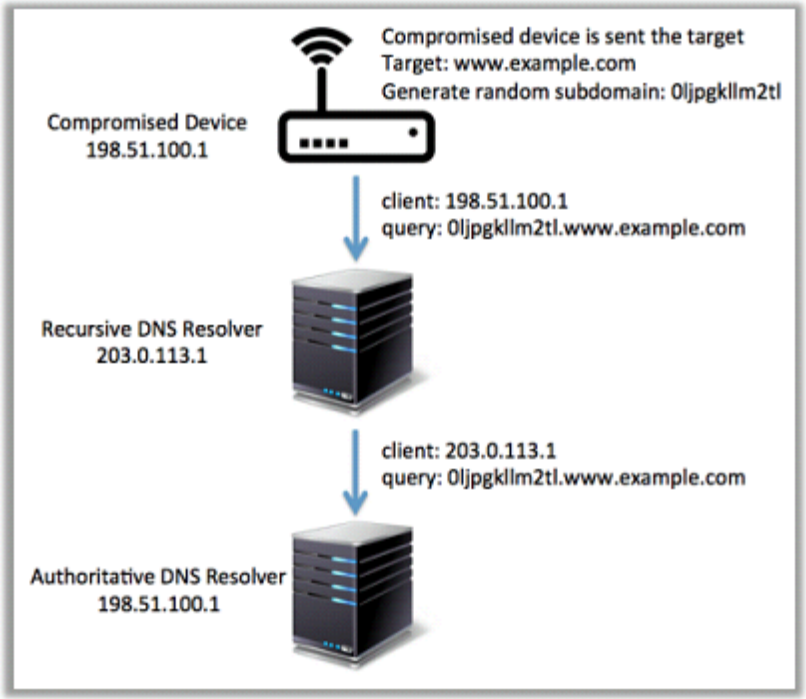
2nd Wave
Between 15:50 UTC and 17:00 UTC – Attack was targeting almost all the available Managed Infrastructures of Dyn around the globe

Recovery
Dyn substantially recovered from the second attack by 17:00 UTC

https://mycourses.aalto.fi

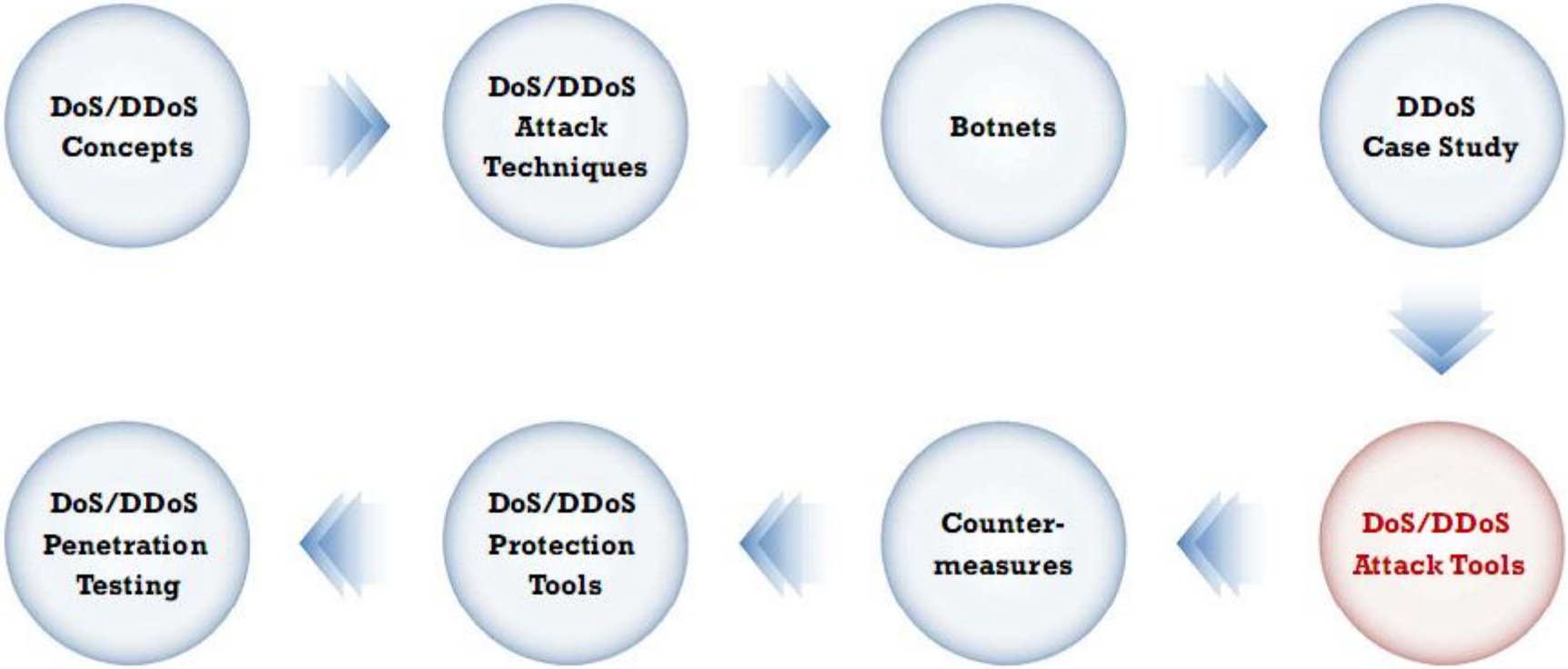
DDoS Case Study: Dyn DDoS Attack

- DNS protocol was used to perform DDoS attack on the DNS servers of the Dyn
- The attack vectors used to perform DDoS attack, which included recursive DNS query mechanism or DNS Waterfall Torture or authoritative DNS exhaustion attack
- Architecture of DNS server infrastructure consists of Recursive DNS resolver and Authoritative DNS resolver
- A recursive DNS resolver receives the DNS query from the bot to resolve a 12-digit pseudo random host from the domain of the authoritative resolver
- In the attack, it is ensured that the recursive DNS resolver fails to resolve the DNS record of random host, so that the query gets forwarded to the authoritative resolver. This mechanism removes the protection of caching layer from authoritative DNS resolvers
- The aim of this attack vector is to forward exceptionally large amount of DNS queries to the authoritative DNS resolver and exhaust the capacity of authoritative DNS resolver to resolve queries



https://mycourses.aalto.fi

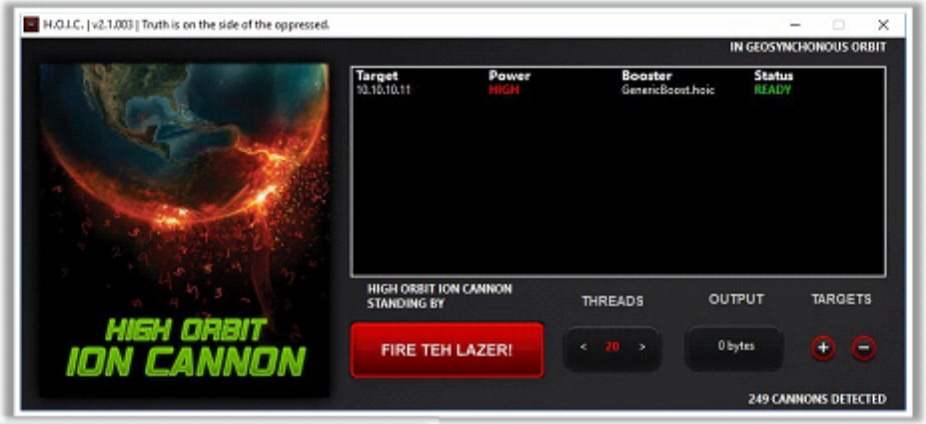
Module Flow



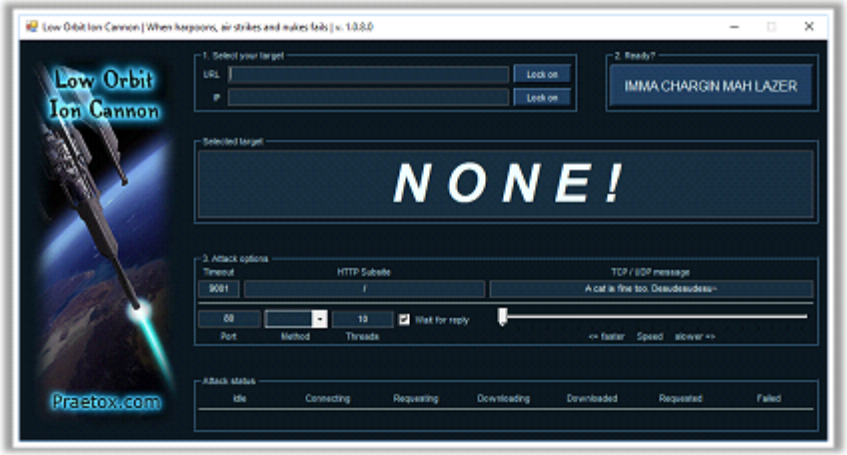
DoS/DDoS Attack Tools

High Orbit Ion Cannon (HOIC)

HOIC makes a DDoS to attack **any IP address** with a user selected port and a user selected protocol



https://sourceforge.net



Low Orbit Ion Cannon (LOIC)

LOIC can be used on a **target site** to flood the server with TCP packets, UDP packets, or HTTP requests with the intention of **disrupting the service** of a particular host

https://sourceforge.net

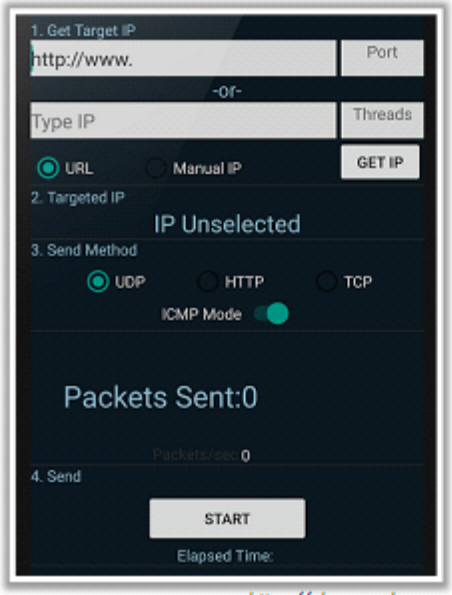
DoS/DDoS Attack Tools:

- HULK (*http://www.sectorix.com*)
- Blackhat Hacking Tools (*https://sourceforge.net*)
- DAVOSET (*https://packetstormsecurity.com*)
- Tsunami (*https://sourceforge.net*)
- R-U-Dead-Yet (*https://sourceforge.net*)

DoS and DDoS Attack Tool for Mobile

LOIC

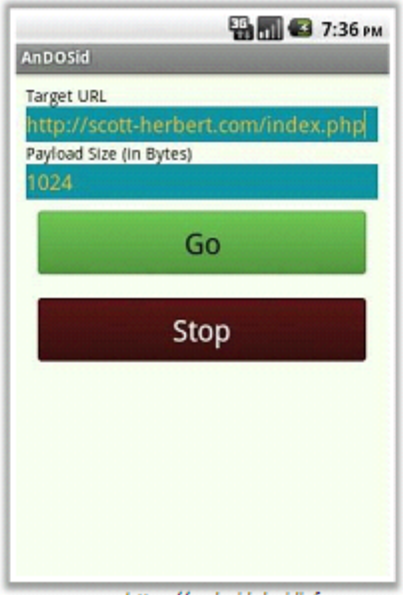
Android version of Low Orbit Ion Cannon (LOIC) software is used for **flooding packets** which allows attacker to **perform DDoS attack** on target organization



https://play.google.com

AnDOSid

AnDOSid allows attacker to simulate a DoS attack (a HTTP POST flood attack) and DDoS attack on a web server from mobile phones



https://andosid.droidinformer.org

DoS/DDoS Mobile Attack Tools

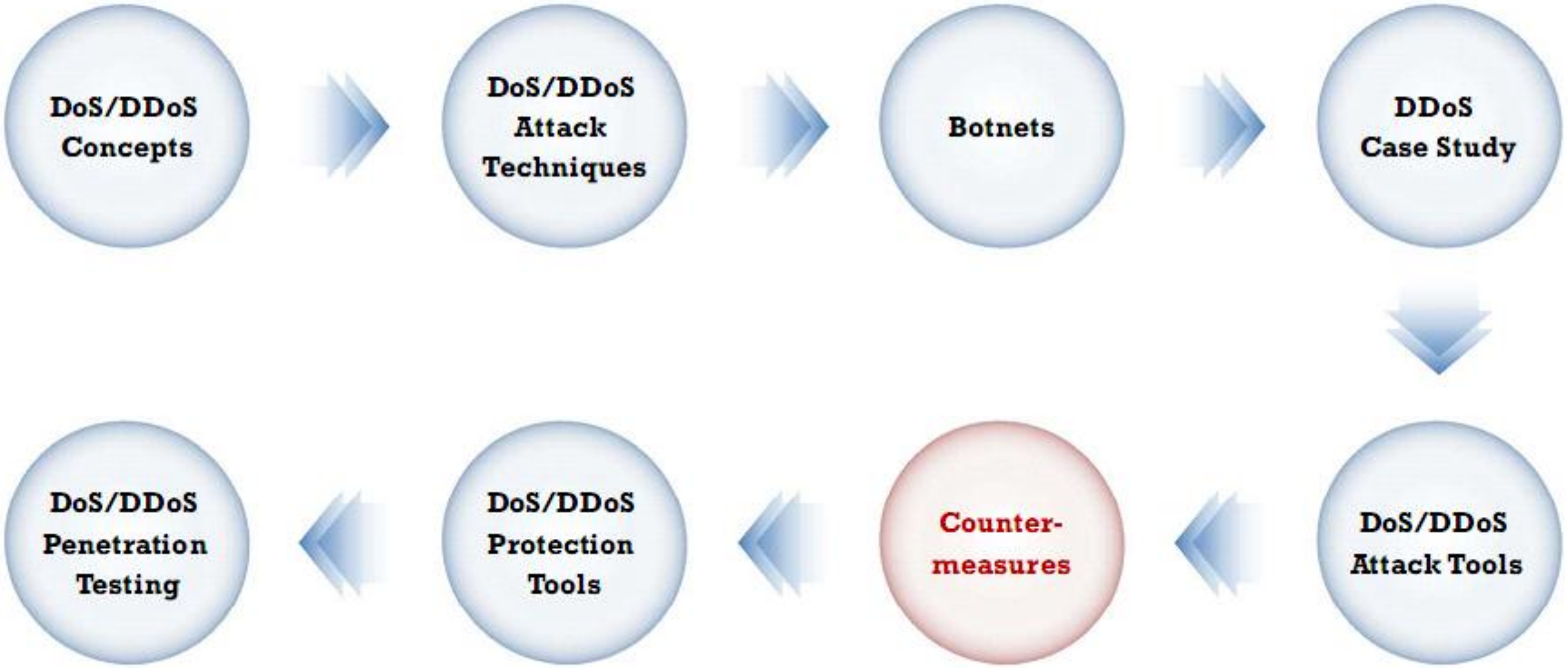
DDOS
https://play.google.com

DDoS
https://play.google.com

Packets Generator
https://play.google.com

PingTools Pro
https://pingtools.org

Module Flow



Detection Techniques

- Detection techniques are based on **identifying and discriminating illegitimate traffic increase** and flash events from legitimate packet traffic
- All detection techniques define an attack as an **abnormal and noticeable deviation** from a threshold of normal network traffic statistics

Activity Profiling

- Activity profiling is done based on the average packet rate for a network flow, which consists of consecutive packets with similar packet fields
- Activity profile is obtained by monitoring the network packet's header information
- An attack is indicated by:
  - An increase in activity levels among the **network flow clusters**
  - An increase in the overall number of **distinct clusters** (DDoS attack)

Sequential Change-point Detection

- Change-point detection algorithms isolate changes in network traffic statistics and in traffic flow rate caused by attacks
- The algorithms filter the **target traffic data** by address, port, or protocol and store the resultant flow as a time series
- Sequential change-point detection technique uses Cusum algorithm to identify and locate the **DoS attacks**
- This technique can also be used to identify the typical scanning activities of the network worms

Wavelet-based Signal Analysis

- Wavelet analysis describes an input signal in terms of **spectral components**
- Analyzing each spectral window's energy determines the presence of anomalies
- Wavelet-based signal analysis filters out the anomalous traffic flow input signals from background noise

DoS/DDoS Countermeasure Strategies

Absorbing the Attack

- Use additional capacity to absorb attack; it **requires preplanning**
- It also requires **additional resources**

Degrading Services

- Identify and keep **critical services** functional and stop non critical services

Shutting Down the Services

- Shut down all the services until the **attack has subsided**

DDoS Attack Countermeasures

01 **Protect Secondary Victims**

02 **Detect and Neutralize Handlers**

03 **Prevent Potential Attacks**

04 **Deflect Attacks**

05 **Mitigate Attacks**

06 **Post-attack Forensics**

Protect Secondary Victims and Detect and Neutralize Handlers

Protect Secondary Victims

- Monitor security on regular basis to remain protected from DDoS agent software
- Install anti-virus and anti-Trojan software and keep these up-to-date
- Increase awareness of security issues and prevention techniques in all Internet users
- Disable unnecessary services, uninstall unused applications, and scan all the files received from external sources
- Properly configure and regularly update the built-in defensive mechanisms in the core hardware and software of the systems

Detect and Neutralize Handlers

Network Traffic Analysis

- Analyze communication protocols and traffic patterns between handlers and clients or handlers and agents in order to identify the network nodes that might be infected by the handlers

Neutralize Botnet Handlers

- There are usually few DDoS handlers deployed as compared to the number of agents. Neutralizing a few handlers can possibly render multiple agents useless, thus thwarting DDoS attacks

Spoofed Source Address

- There is a decent probability that the spoofed source address of DDoS attack packets will not represent a valid source address of the definite sub-network

Prevent Potential Attacks

Egress Filtering

- Egress filtering **scans the headers of IP packets** leaving a network
- Egress filtering ensures that **unauthorized or malicious traffic** never leaves the internal network
- The packets will not reach the targeted address if they do not meet the necessary specifications

Ingress Filtering

- Ingress filtering **prevents source address spoofing** of Internet traffic
- It **protects from flooding attacks** which originate from the valid prefixes (IP addresses)
- It enables the originator to be traced to its true source

TCP Intercept

- TCP intercept feature in router protects TCP servers from a TCP SYN-flooding attack
- Configuring TCP Intercept **prevents DoS attacks** by intercepting and validating the TCP connection requests

Rate Limiting

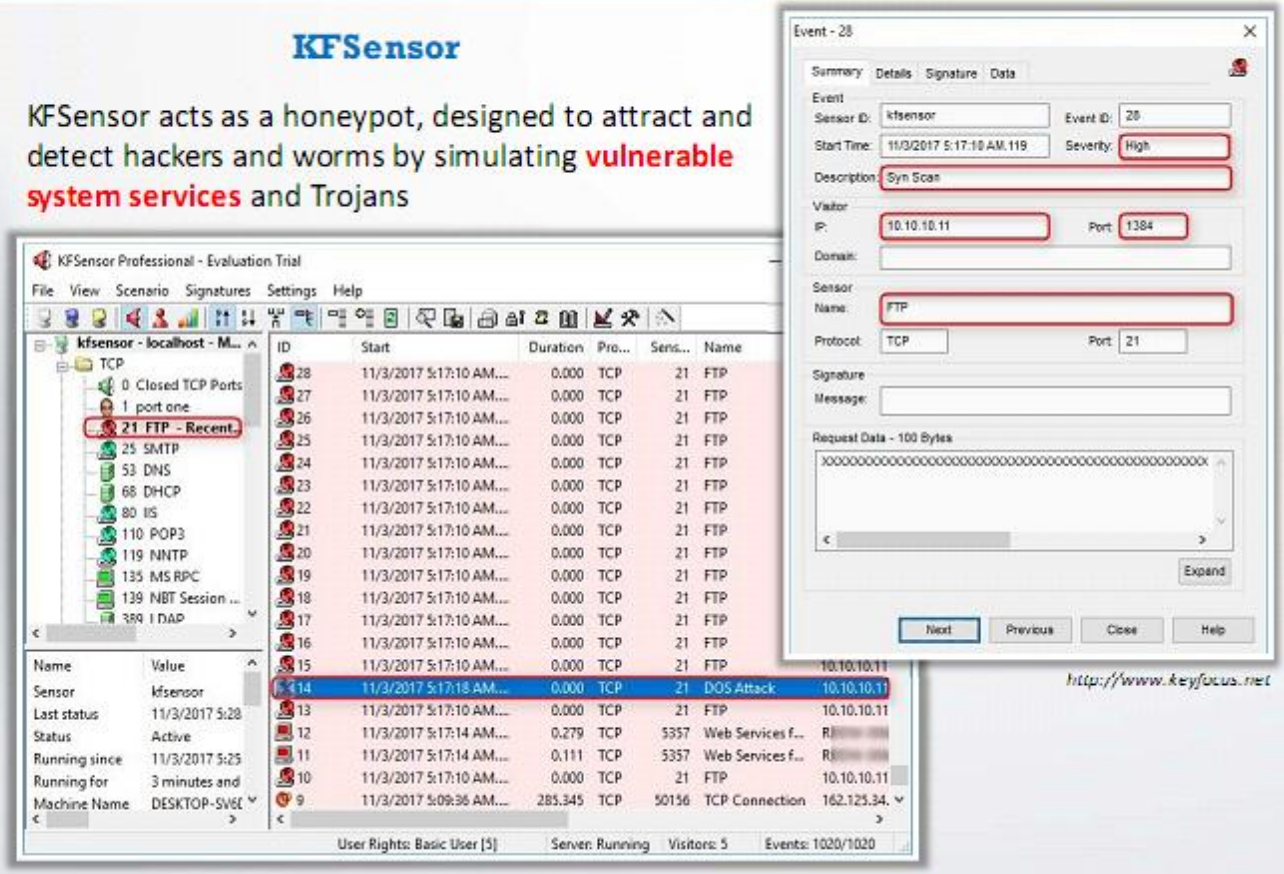
- Rate limiting **controls the rate of outbound or inbound traffic** of a network interface controller
- It **reduces the high volume inbound traffic** that cause DDoS attack

Deflect Attacks

- Systems that are set up with limited security, also known as **Honeypots**, act as an enticement for an attacker
- Honeypots serve as a means for **gaining information** about attackers, **attack techniques** and tools by storing a record of the system activities
- Use defense-in-depth approach with IPSes at different network points to divert **suspicious DoS traffic** to several honeypots

KFSensor

KFSensor acts as a honeypot, designed to attract and detect hackers and worms by simulating **vulnerable system services** and Trojans



http://www.keyfocus.net

Mitigate Attacks

Load Balancing

- Increase bandwidth on **critical connections** to absorb additional traffic generated by an attack
- **Replicate servers** to provide additional failsafe protection
- Balance load on each server in a **multiple-server architecture** to mitigates DDoS attack

Throttling

- Set routers to access a server with a logic to throttle **incoming traffic levels** that are safe for the server
- Throttling helps in preventing damage to servers by controlling the **DoS traffic**
- This method helps routers manage **heavy incoming traffic**, so that the server can handle it
- It filters legitimate user traffic from fake **DDoS attack traffic**

Drop Requests

- In this technique, servers and routers **drop packets** when load increases
- System induces requester to drop the request by making it to solve a difficult puzzle that requires a lot of **memory or computing** power before continuing with the request

Post-Attack Forensics

Traffic Pattern Analysis

- Traffic pattern analysis can help the network administrators to develop new **filtering techniques** for preventing the attack traffic from entering or leaving the networks
- Output of traffic pattern analysis helps in **updating load balancing** and **throttling countermeasures** to enhance efficiency and protection ability

Packet Traceback

- Packet Traceback is similar to **reverse engineering**
- It helps in identifying the true **source of attack** and taking necessary steps to block further attacks

Event Log Analysis

- Event log analysis helps in identifying the source of the **DoS traffic**
- This allows network administrators to recognize the type of DDoS attack or a combination of attacks used

Techniques to Defend against Botnets

RFC 3704 Filtering

RFC 3704 filtering limits the impact of DDoS attacks by denying traffic with **spoofed addresses**

Any traffic coming from unused or reserved IP addresses is bogus and should be filtered at the ISP before it enters the Internet link

Cisco IPS Source IP Reputation Filtering

Reputation services help in determining if an **IP or service** is a source of threat or not

Cisco IPS regularly **updates its database** with known threats such as botnets, botnet harvesters, malwares, etc. and helps in filtering DoS traffic

Black Hole Filtering

Black hole refers to network nodes where incoming traffic is discarded or dropped without informing the source that the data did not reach its intended recipient

Black hole filtering refers to **discarding packets** at the routing level

DDoS Prevention Offerings from ISP or DDoS Service

Enable IP Source Guard (in CISCO) or similar features in other routers to filter traffic based on the **DHCP snooping binding database** or IP source bindings, prevents a bot to send spoofed packets

DoS/DDoS Countermeasures

1. Use strong encryption mechanisms such as WPA2, AES 256, etc. for broadband networks to withstand against eavesdropping
2. Ensure that the software and protocols are up-to-date and scan the machines thoroughly to detect any anomalous behavior
3. Disable unused and unsecure services
4. Block all inbound packets originating from the service ports to block the traffic from reflection servers
5. Update kernel to the latest release
6. Prevent the transmission of fraudulently addressed packets at ISP level
7. Implement cognitive radios in the physical layer to handle jamming and scrambling attacks
8. Configure the firewall to deny external ICMP traffic access
9. Secure the remote administration and connectivity testing
10. Perform thorough input validation
11. Prevent use of unnecessary functions such as gets, strcpy, etc.
12. Prevent the return addresses from being overwritten

DoS/DDoS Protection at ISP Level

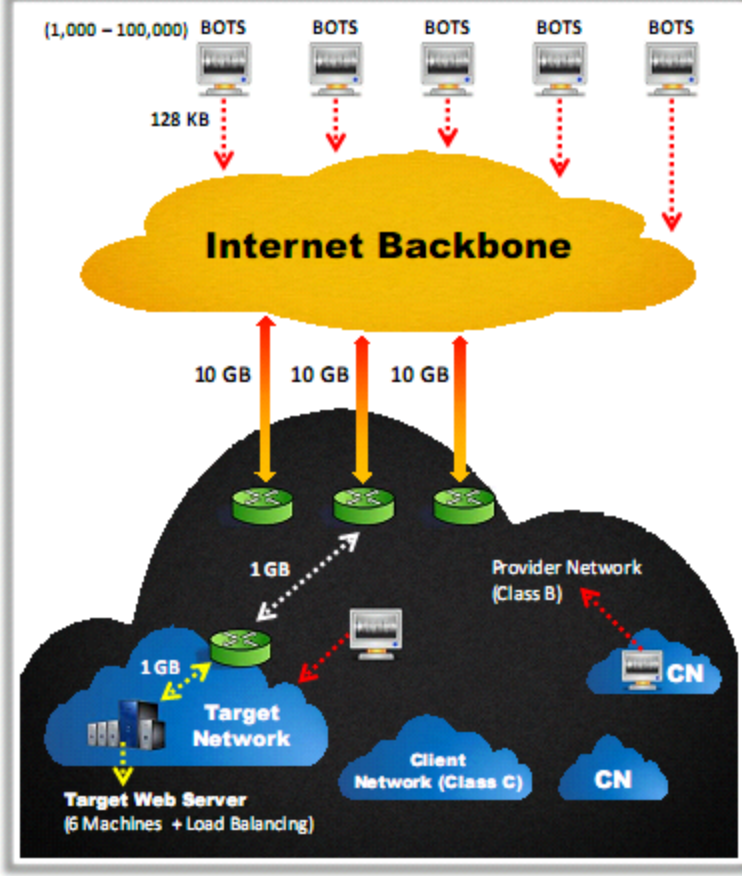
Most ISPs simply block all the requests during a **DDoS attack, denying even the legitimate traffic** from accessing the service

ISPs offer in-the-cloud DDoS protection for Internet links so that they do not become **saturated by the attack**

The in-the-cloud DDoS protection **redirects attack traffic** to the ISP during the attack and sends it back

Administrators can **request ISPs** to block the original affected IP and move their site to another IP after performing DNS propagation

http://www.cert.org



Enabling TCP Intercept on Cisco IOS Software

To enable TCP intercept on CISCO IOS, use these commands in global configuration mode:

Step	Command	Purpose
1	**access-list** access-list-number {**deny** | **permit**} **tcp any** destination destination-wildcard	Define an IP extended access list
2	**ip tcp intercept list** *access-list-number*	Enable TCP Intercept

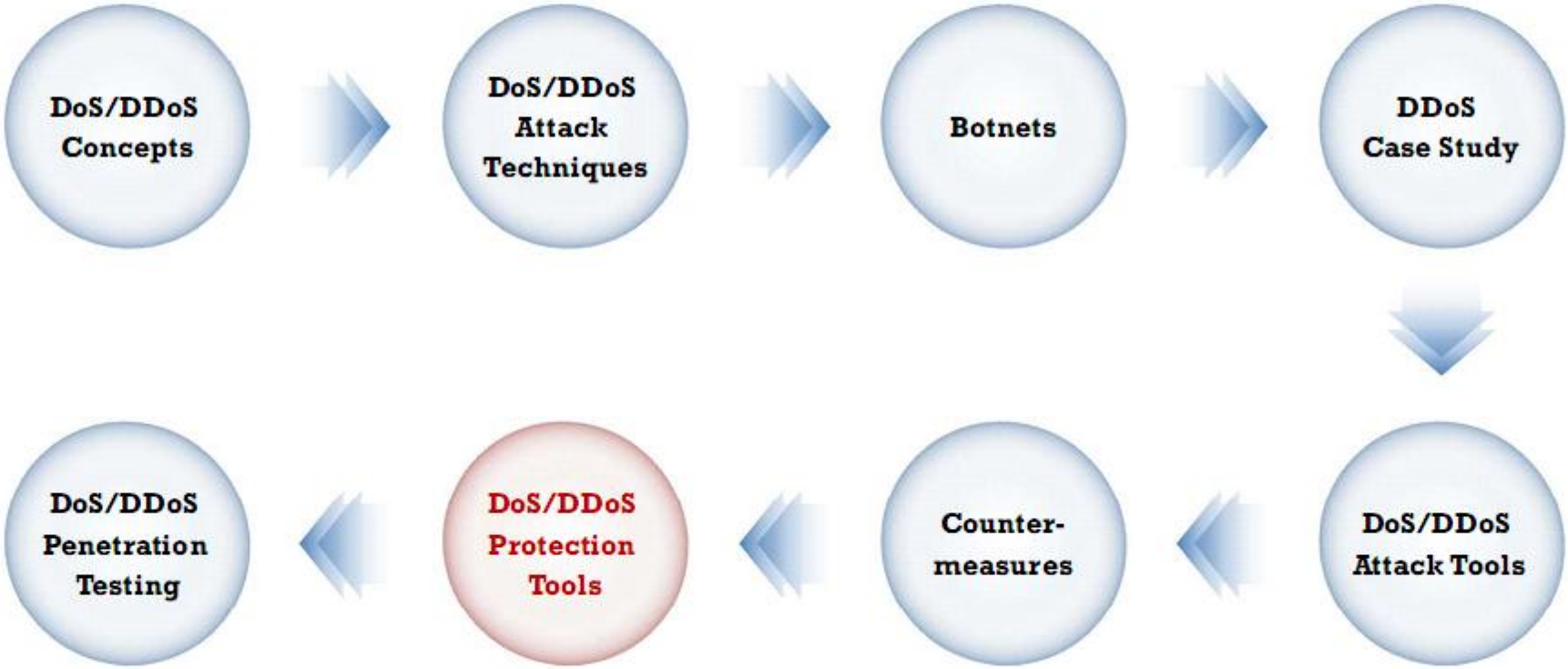
TCP intercept can operate in either active intercept mode or passive watch mode. The default is intercept mode

The command to set the TCP intercept mode in global configuration mode:

Command	Purpose
ip tcp intercept mode {intercept | watch}	Set the TCP intercept mode

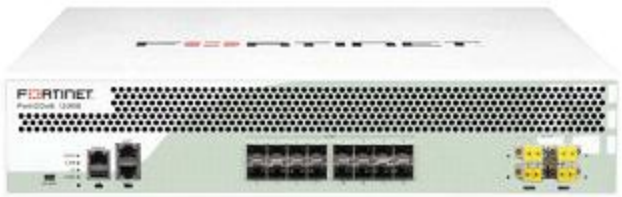
https://www.cisco.com

Module Flow



Advanced DDoS Protection Appliances

FortiDDoS-1200B



https://www.fortinet.com

DDoS Protector

https://www.checkpoint.com

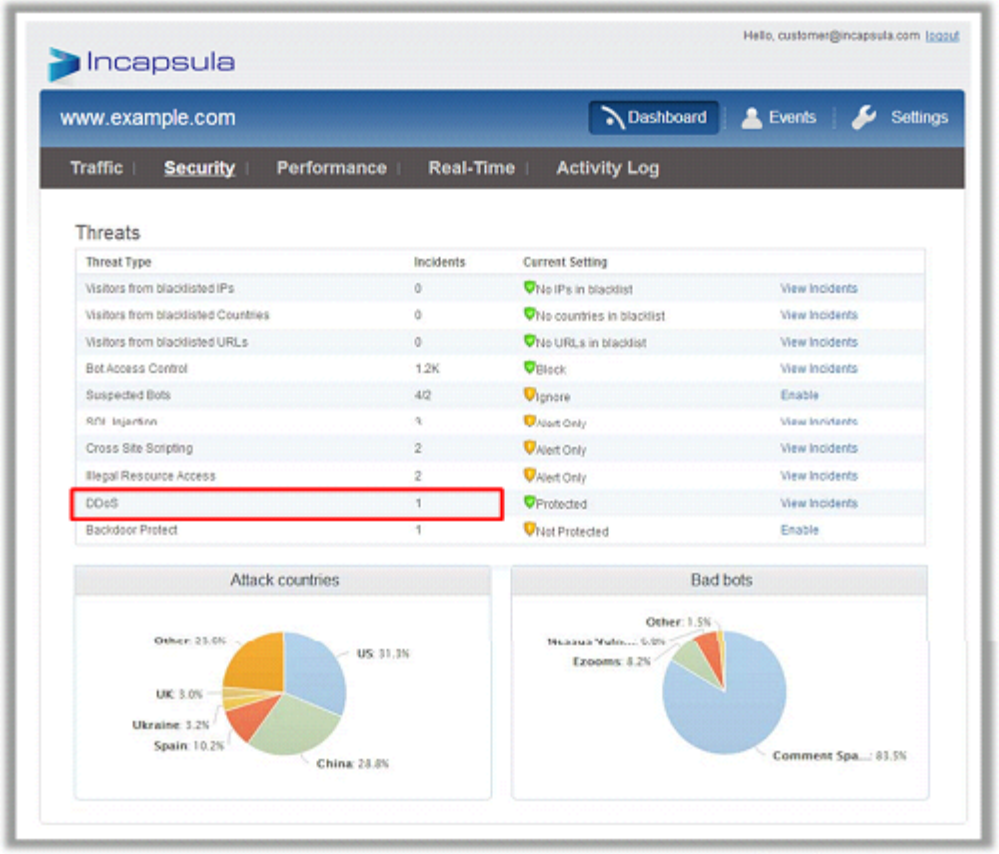
Cisco Guard XT 5650

https://www.cisco.com

A10 Thunder TPS

http://www.itp.net

DoS/DDoS Protection Tools



https://www.incapsula.com

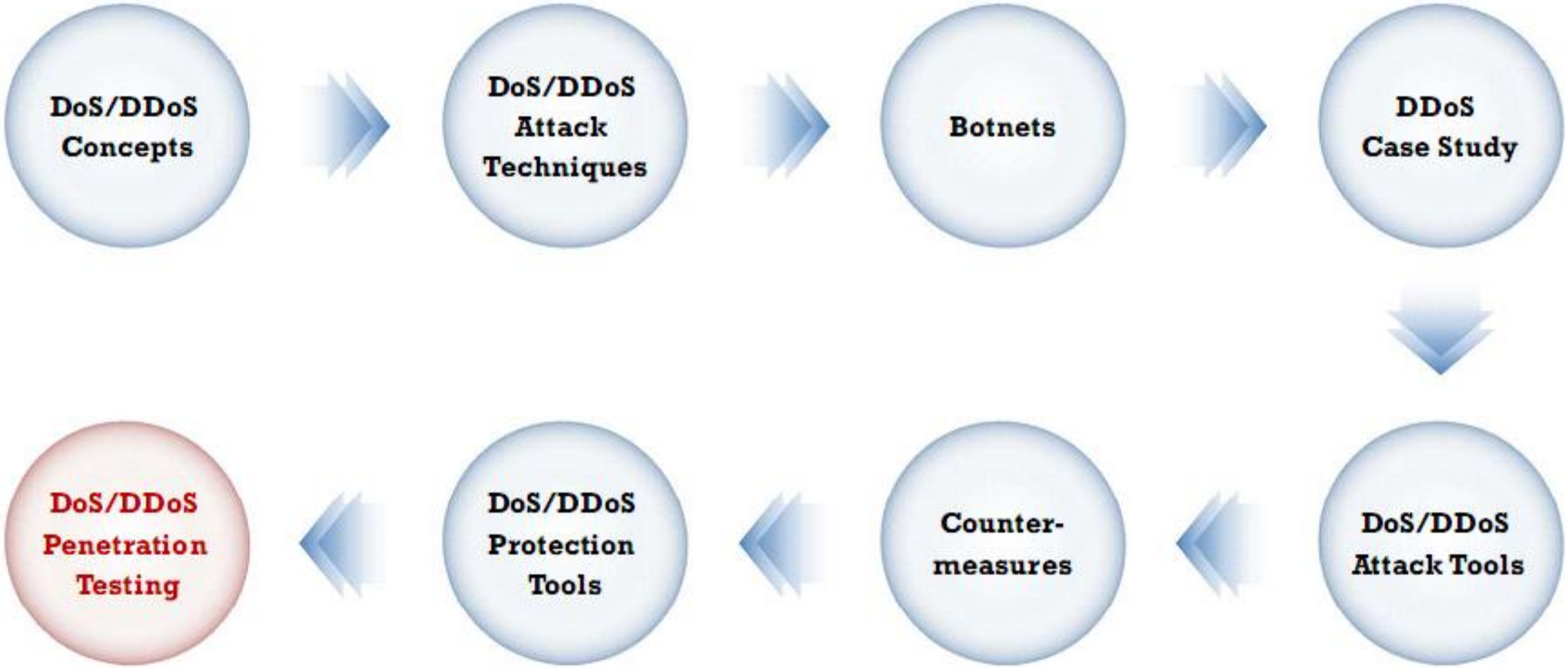
Incapsula DDoS Protection

Incapsula DDoS protection quickly mitigates any size attack without getting in the way of **legitimate traffic** or **increasing latency**

DoS/DDoS Protection Tools

- Anti DDoS Guardian (*http://www.beethink.com*)
- DDoS-GUARD (*https://ddos-guard.net*)
- Cloudflare (*https://www.cloudflare.com*)
- DOSarrest's DDoS protection service (*https://www.dosarrest.com*)
- DefensePro (*https://www.radware.com*)

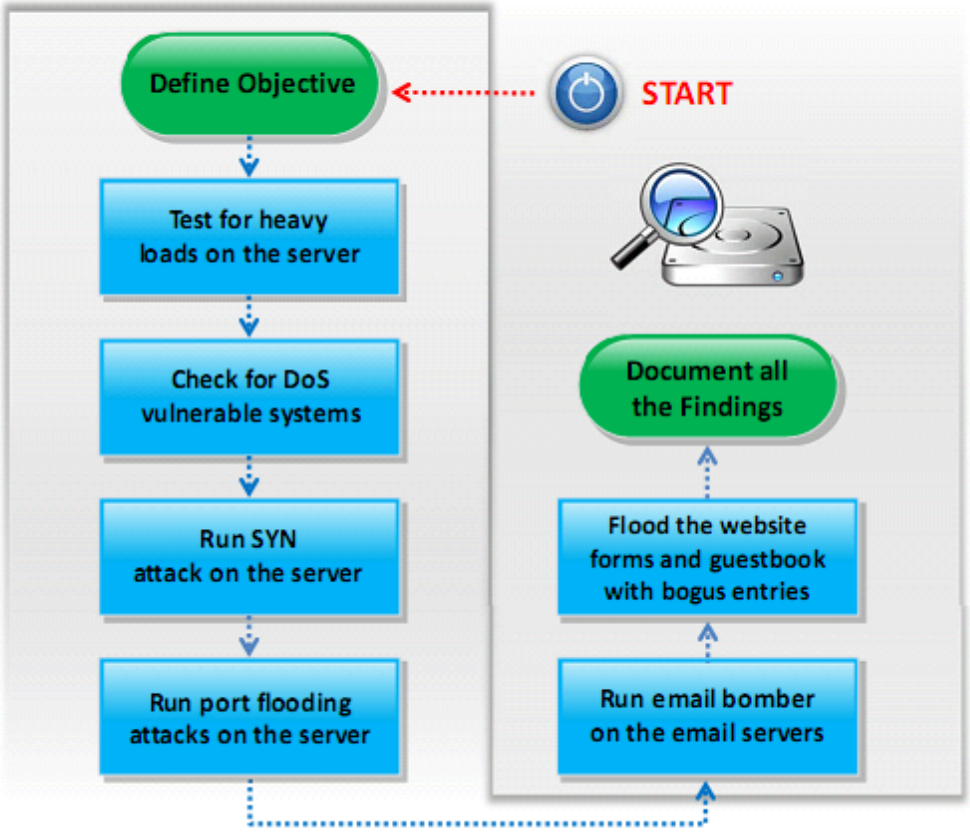
Module Flow



Denial-of-Service (DoS) Attack Pen Testing

- DoS attack should be incorporated into **Pen testing plans** to find out if the network server is susceptible to DoS attacks
- DoS pen testing determines a **minimum threshold for DoS attacks** on a system, but the tester cannot ensure that the system is resistant to DoS attacks
- The pen tester floods the target network with traffic similar to hundreds of people repeatedly requesting the service in order to **check the system stability**
- Pen testing results will help the administrators to determine and adopt suitable **network perimeter security** controls such as load balancer, IDS, IPS, Firewalls, etc.

Denial-of-Service (DoS) Attack Pen Testing (Cont'd)



- Test the web server using automated tools such as **Webserver Stress Tool** and **Apache JMeter** for load capacity, server-side performance, locks, and other scalability issues
- Scan the network using automated tools such as **Nmap**, **GFI LanGuard**, and **Nessus** to discover any systems that are vulnerable to DoS attacks
- Flood the target with connection request packets using tools such as **Dirt Jumper DDoS Toolkit**, **HOIC**, and **DoS HTTP**
- Use a port flooding attack to flood the port and increase the CPU usage by maintaining all the connection requests on the ports under blockade. Use tools **LOIC** and **Moihack Port Flooder** to automate a port flooding attack
- Use tools **Mail Bomber** to send a large number of emails to a target mail server
- Fill the forms with **arbitrary** and **lengthy** entries
- Document **all the findings** at each step of the DoS pen-testing methodology for **analysis** and future reference

Module Summary

- Denial of Service (DoS) is an attack on a computer or network that reduces, restricts, or prevents accessibility of system resources to its legitimate users
- A distributed denial-of-service (DDoS) attack involves a multitude of compromised systems attacking a single target, thereby causing denial of service for users of the targeted system
- Attacker uses various techniques to carry out DoS/DDoS attacks on the target but these attacks are basically categorized into: volumetric attacks, fragmentation attacks, TCP state-exhaustion attacks, and application layer attacks
- There are organized groups of cyber criminals who work in a hierarchical setup with a predefined revenue sharing model, like a major corporation that offers criminal services
- A botnet is a huge network of the compromised systems and can be used by an attacker to launch denial-of-service attacks
- Detection techniques are based on identifying and discriminating the illegitimate traffic increase and flash events from legitimate packet traffic
- The pen tester floods the target network with traffic similar to hundreds of people repeatedly requesting the service in order to check the system stability