How I dealt with a DDoS attack that killed a Game Server

Pranay Meshram

Your local neighbourhood NSQ host

Some information presented may be incomplete.
Please don’t be harsh with questions!

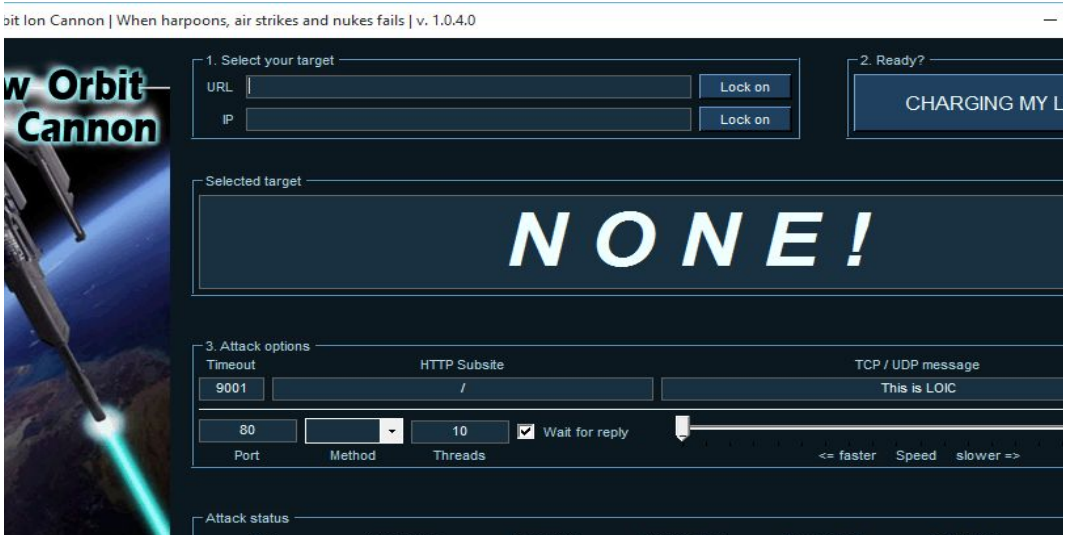
Introduction to Service Attacks

- **A service attack is a type of cyber attack that aims to disrupt or disable the normal functioning of a computer system or network.**
- **It works by overwhelming it with traffic or requests.**
- **The goal of a service attack is to prevent legitimate users from accessing the system or service.**
- **Two major types:**

  DDoS - Distributed Denial of Service
  DOS - Denial of Service

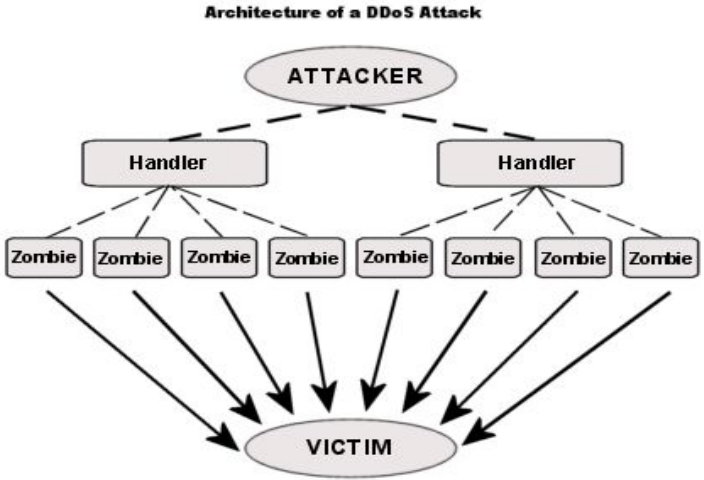
- **Famous Example - Mirai botnet.**

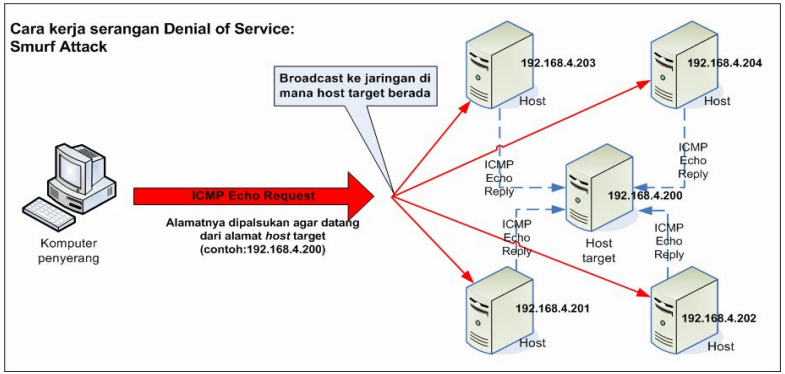
  - Attacked OVH with 1.1 Tbps per second of traffic or 17.2 million requests per second.
  - Costed $40-120k per hour
  - Connected to 600k devices.



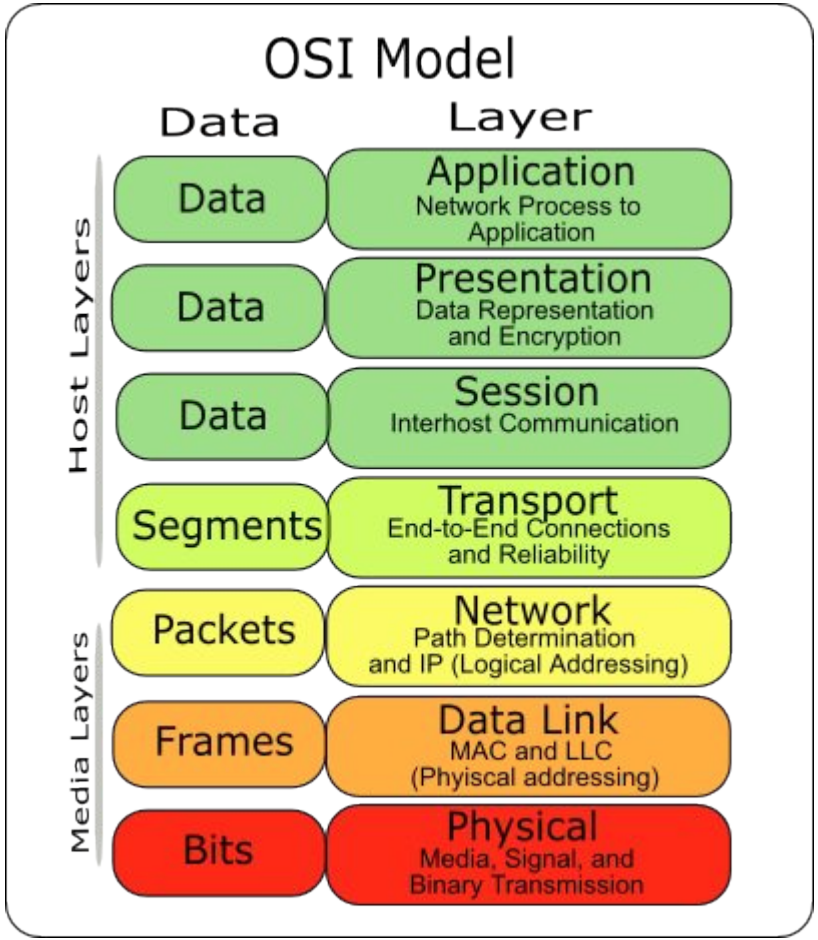
DDoS vs DOS

Distributed Denial of Service (DDoS)	Denial of Service (DoS)
• Multiple systems target a single system with a DoS attack. • The targeted network is then bombarded with packets from multiple locations.	• A single computer is used to flood a server with TCP and UDP packets
• Harder to detect	• Easier to detect
• Use botnets or multiple devices	• Use a single script or tool
• Examples: 1) Buffer overflow 2) SYN attack	• Black Friday Sales • ICMP Flood





Layers of DDoS Attacks



- Layer 7 is usually the most common with HTTP/GET requests. Hard to detect as they deal with applications, easily eat up resources.
- DNS amplification attacks happen at Layer 3.
- SYN, TCP, UDP attacks happen at Layer 4.

Website DDoS - Web Services - Layer 7

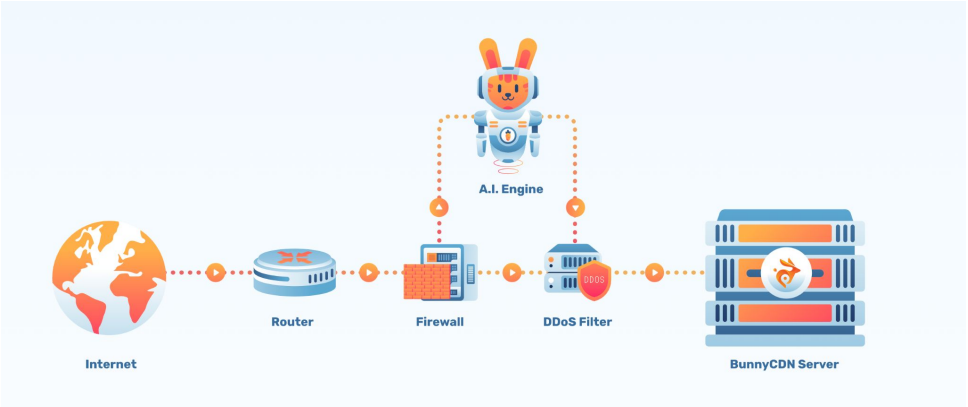
Application DDoS - Layer 4 - Gaming, VOIP, etc.

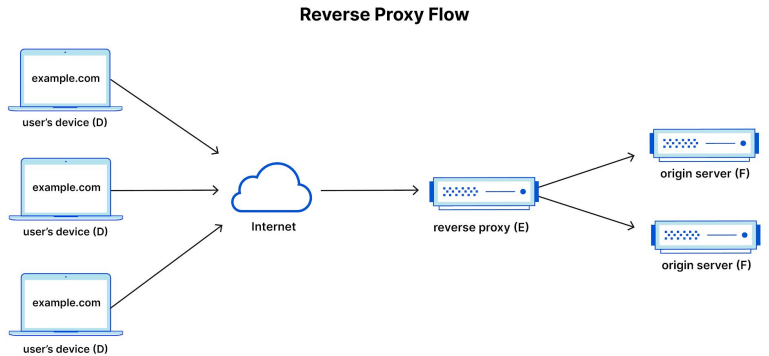
Network DDoS - Layer 3 - for on-premise, cloud, & hybrid networks. Combine DDoS protection, traffic acceleration, & more.

DDoS Mitigation!!

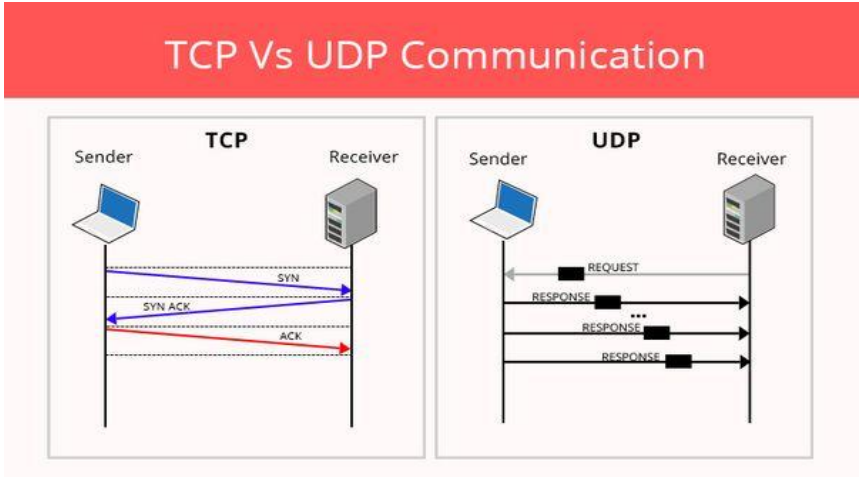
- Different providers provide DDoS Mitigation methods eg Cloudflare, Akamai, OVH.
- Common methods:

  - Using AI to survey traffic and deploy firewall.
  - Using reverse proxy to redirect data to different server with higher bandwidth, than to intended server.
  - Blocking access to ALL traffic
  - Pray no one messes up with you





TCP vs UDP



TCP	UDP
Connection Oriented	Message Oriented
Delivery of message is verified	No verification of delivered message
Ensures message is delivered	Possibility that your data will be lost
Slower	Faster

Introduction to the real problem

SAMP (San Andreas Multiplayer)

- Massive multiplayer mod for GTA San Andreas.
- Allowed seamless multiplayer up to 1500 real-time players in one server.
- Entire mod had several servers, and on average the mod had 90k-100k players.
- Developed by one person, maintained by a close team.
- Variety of gamemodes.

```
313             return 1;
314
315
316     public OnPlayerDeath(playerid, killerid, reason) // Thanks to SiaReyes.
317
318         PlayerInfo[playerid][pDeaths]++;
319
320         if(GetPlayerMoney(playerid) < 500)
321
322           SendClientMessage(playerid, CRED, "[SERVER]: You didn't have $500 to fix your wounds, the server paid in your place!");
323
324         else if(GetPlayerMoney(playerid) > 500)
325
326           SendClientMessage(playerid, CRED, "[SERVER]: You got killed/self-death and paid $500 to fix your wounds.");
327           GivePlayerMoney(playerid, -500);
328           PlayerInfo[playerid][pCash] -= 500;
329
330
331         if(killerid != INVALID_PLAYER_ID)
332
333             PlayerInfo[killerid][pKills]++;
334             new pname[MAX_PLAYER_NAME], ename[MAX_PLAYER_NAME], string[128];
335             GetPlayerName(playerid, pname, sizeof(pname));
336             GetPlayerName(killerid, ename, sizeof(ename));
337             format(string, sizeof(string), "[SERVER]: You killed %s and looted $1,000 from him (+ 1 Score).", pname);
338             SendClientMessage(killerid, CLIME, string);
339             format(string, sizeof(string), "[SERVER]: You have been killed by %s!", ename);
340             SendClientMessage(playerid, CRED, string);
341             GivePlayerMoney(killerid, 1000);
342             PlayerInfo[killerid][pCash] += 1000;
343             SetPlayerScore(killerid, GetPlayerScore(killerid)+1);
344
345
346         SendDeathMessage(killerid, playerid, reason);
347
348         return 1;
349
350
351     public OnPlayerText(playerid, text[])
352
353         ChatLog(playerid, text);
354         if(PlayerInfo[playerid][pMuted] == 1) return SendClientMessage(playerid, CRED, "[SERVER]: You're muted, you can't talk on the chat!");
355         else if(PlayerInfo[playerid][pMuted] == 0) return 1;
356             return 1;
357
358
359     public OnPlayerCommandText(playerid, cmdtext[])
360
361             if(IsSpawned == 0) return SendClientMessage(playerid, CRED, "[SERVER]: You must be spawned to use commands!");
362             return 0;
363
364     public OnPlayerStateChange(playerid, newstate, oldstate)
365
366         return 1;
367
368
```

Can you guess which protocol SAMP uses? UDP or TCP?

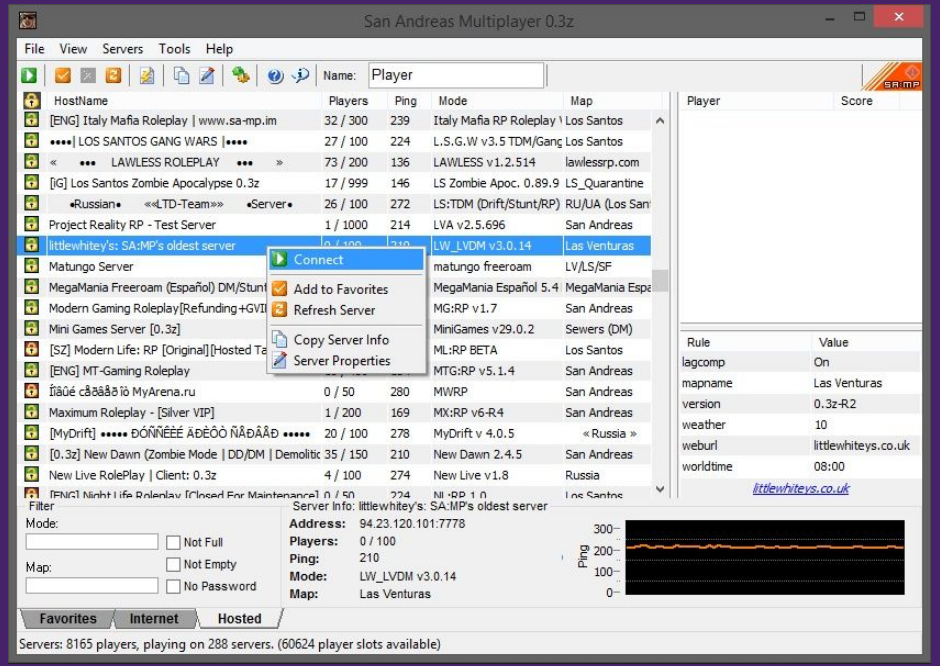
UDP

Why?

In multiplayer games like SAMP, low-latency is critical.

UDP ensures low-latency, high speed

TCP first ensures data is delivered which takes time



Problem starts now..

Pre-Start of Incident

- User “SankeyKing” breaks server rules.
- User punished for breaking rules.
- User threatened to ‘bring down the server’. Nobody believed, and user was punished.

Start of Incident

- Minutes after, server chat starts lagging. Users being to time-out.
- Server brought down, users unable to connect.
- Happened multiple times as days went by.

Some lil detective work

- User tricked to connect to TeamSpeak3, tracked IP to lead location to Pakistan.
- User revealed his FaceBook ID as well, only 16/17 year old.
- DDoS was of high magnitude, expensive.

Mitigating the attack

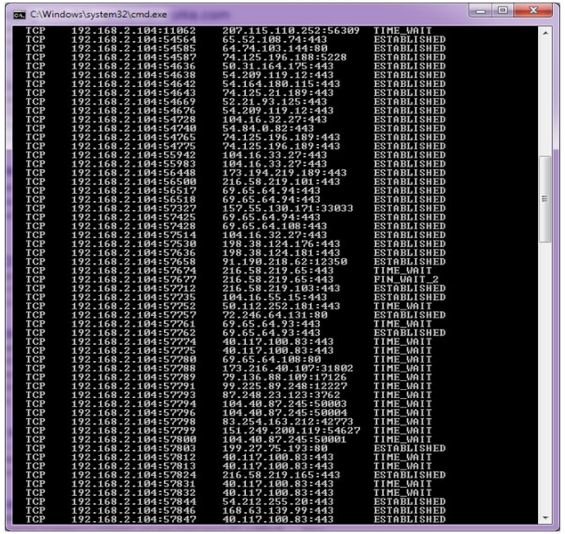
CHANGE VPS

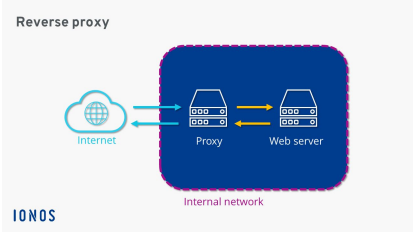
- Tried changing from Cheap VPS to OVH.
- Helped with nothing.
- Costs issue.

REVERSE PROXY

1) Have two servers with same DNS name.

2) Have a new server with high bandwidth taking the traffic, route it to main server.





Manual DDoS protection

1) This protection would be manually enabled/disabled.

2) When enabled, it would filter all the incoming traffic. Verify, and then accept connection.

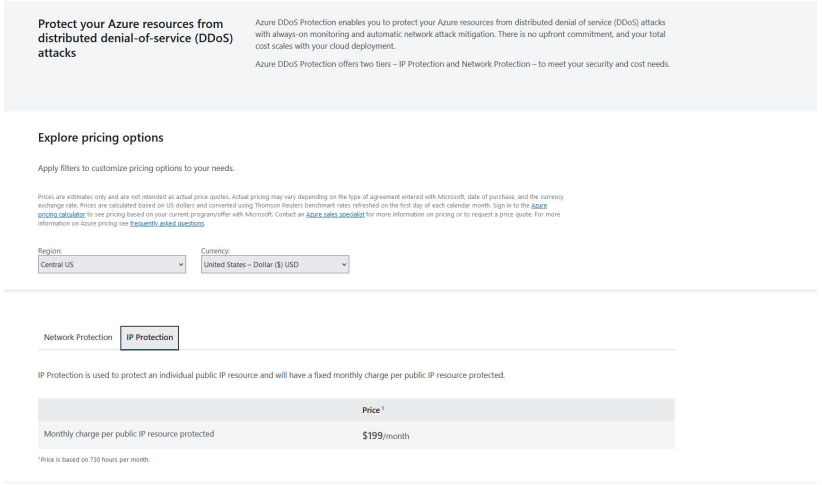
3) Trust issues, connection issues

RANGE BAN

- Tried to ban all IPs from hacker's region.
- All bots IPs were from around the world.
- No help with the issue.

Effects of the DDoS attacks

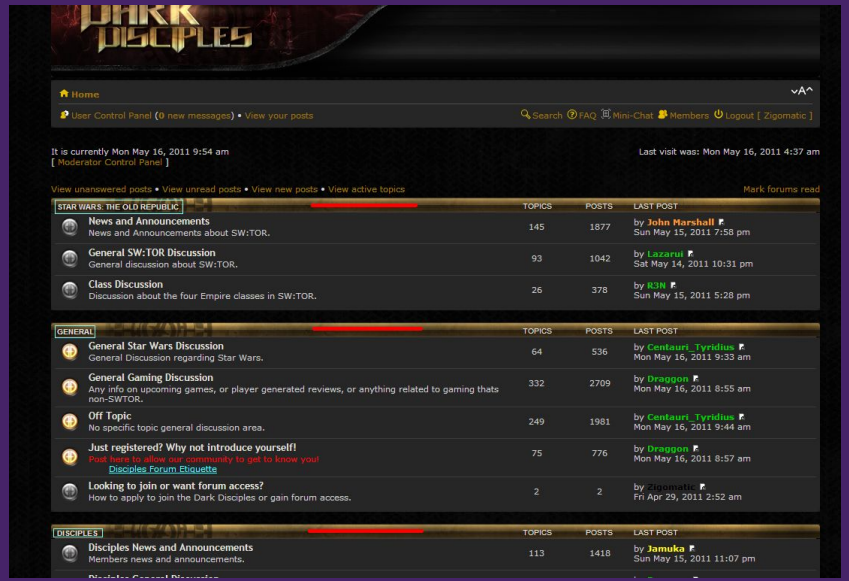
- Could not afford expensive DDoS Protection
- Had to listen in to hacker's demands.
- Players got irritated with constant Connections issues.
- Playerbase went down from an average of 120 players at peak time daily to 40, even 20 at times.



Root-Causing

Finding Clues

- TCP would have been safer, having less effect.
- Finding Root Cause began now.



1. Game server, websites and forums all shared the same DNS name. Different providers.
2. During the attack, website and forums were slow but not down.
3. Websites/forums used TCP but with even minimal DDoS protection.
4. **TS3 used same IP address as game-server but wasn't affected.**

(KINDA) SOLUTION TIME!

Port lead it to all!

192.168.1.1:**10011** - TeamSpeak3 Port (TCP) (Required) (Open)

192.168.1.1:**7777** - Game Server Port (UDP) (Required) (Open) (DDoS protected)

192.168.1.1:**7778** - Game Server Port (UDP) (Not Required) **(Open) !!!!**

- Attacks were directed at UDP ports, TCP was unaffected.
- DDoS protection was offered at IP level & **7777** port by VPS providers, but no firewall was set from host to block traffic from other UDP ports.
- DDoS attack utilized these non-existing firewall to attack on open UDP port with no traffic monitoring.

SOLUTION

Instead of selectively blocking traffic, implement a whitelist.

Rule #1: Allow UDP traffic on port 7777
Rule #2: Drop any traffic on any port

```
num   pkts bytes target     prot opt in     out     source               destination
1     423K  113M ACCEPT     all  --  lo     any     anywhere             anywhere
2     151M   14G ACCEPT     all  --  any    any     anywhere             anywhere             ctstate RELATED,ESTABLISHED
3    2489K 5650M ACCEPT     udp  --  any    any     anywhere             anywhere             udp dpt:7777
4     1098 65262 ACCEPT     tcp  --  any    any     anywhere             anywhere             tcp dpt:http
5      240 10492 ACCEPT     tcp  --  any    any     anywhere             anywhere             tcp dpt:https
6     182K   11M ACCEPT     tcp  --  any    any     anywhere             anywhere             tcp dpt:ssh
7      229  9296 ACCEPT     tcp  --  any    any     anywhere             anywhere             tcp dpt:mysql
8    2630K 5785M DROP       all  --  any    any     anywhere             anywhere
```

Thank you