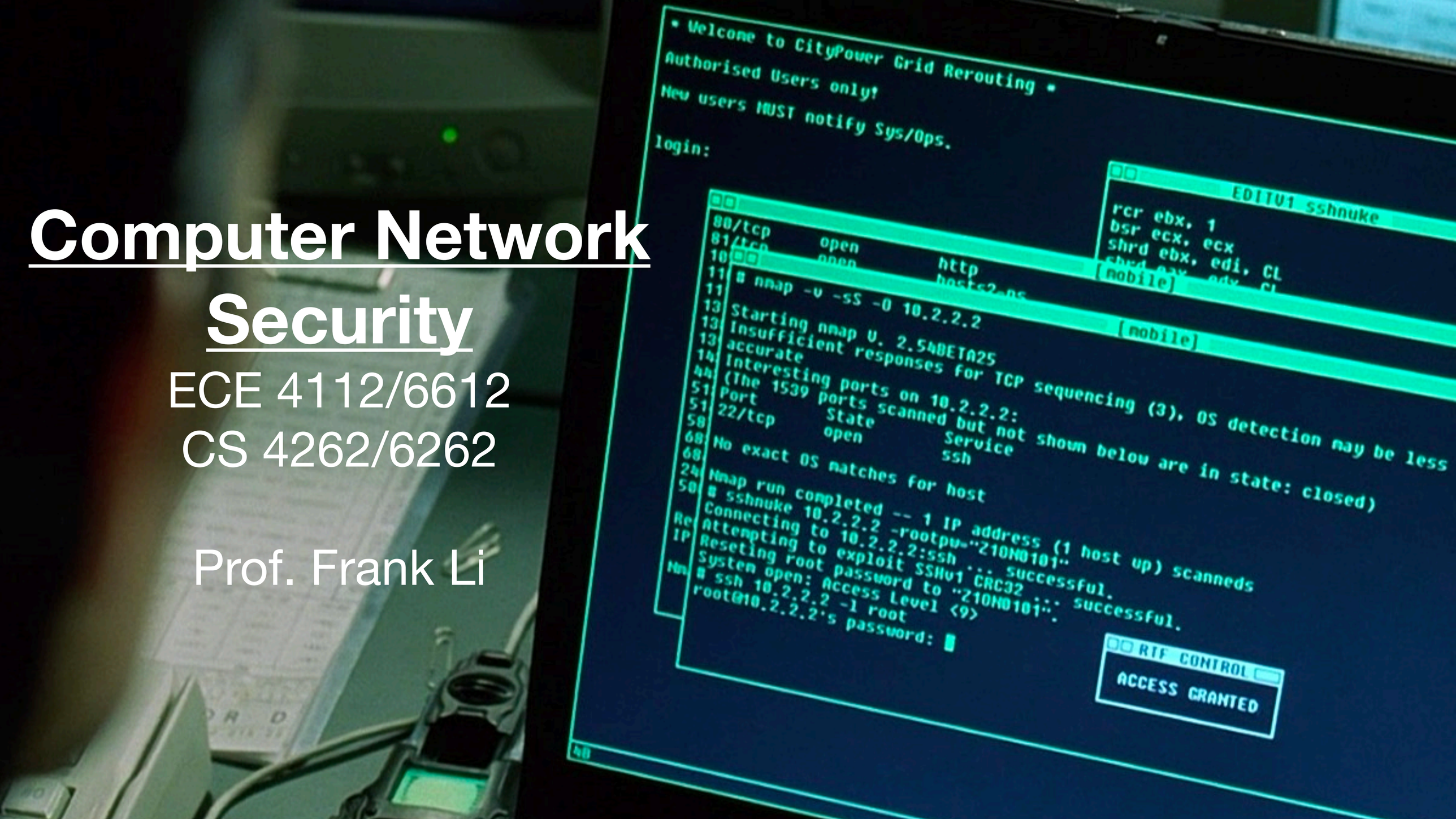
Computer Network Security

ECE 4112/6612
CS 4262/6262

Prof. Frank Li



Logistics

HW3 on web + email security due Monday Nov 13.

Quiz 2 regrades open until Thurs, Nov 9.

Quiz 3 next Thursday (Nov 16):
- web security up to this Thursday's lecture (malware)

Work on your projects!

Tue, Oct 10	**No Class (Fall Break)**
Thu, Oct 12	Web security Part 1: Web attacks and defenses
Tue, Oct 17	Web security Part 2: Web attacks and defenses
Thu, Oct 19	Web security Part 3: Web attacks and defenses
Tue, Oct 24*	**Quiz 2**
Thu, Oct 26*	Authentication
Tue, Oct 31	Email Security (Spam, Phishing)
Thu, Nov 2	Network Access Control
Tue, Nov 7	DoS attacks and defenses
Thu, Nov 9	Malware, Botnet
Tue, Nov 14	Last lecture: Censorship and Anonymous Communication
Thu, Nov 16	**Quiz 3**
Tue, Nov 21	**No Class (Early Thanksgiving Break)**
Thu, Nov 23	**No Class (Thanksgiving Break)**
Tue, Nov 28*	Project: Final Project Presentations
Thu, Nov 30*	Project: Final Project Presentations
Tue, Dec 5	Final Class: Final Project Presentations
Thu, Dec 7	**Final Exam: 2:40 - 5:30 PM (Undergraduate Sections Only)**

Last Lecture:
Network Access Control

Why Have Firewalls Been Successful?

- *Central control – easy administration and update*
  - Single point of control: update one config to change security policies
  - Potentially allows rapid response
- *Easy to deploy – transparent to end users*
  - Easy incremental/total deployment to protect 1,000's
- *Addresses an important problem*
  - Security vulnerabilities in network services are rampant
  - Easier to use firewall than to directly secure code …

Firewall Disadvantages?

- *Functionality loss* – *less connectivity, less risk*
  - May reduce network's usefulness
  - Some applications don't work with firewalls
    - Two peer-to-peer users behind different firewalls
- *The malicious insider problem*
  - Deployment assumes insiders are trusted
    - Malicious insider (or *anyone gaining control of internal machine*) can wreak havoc
- Firewalls establish a *security perimeter*
  - Threat from travelers with laptops, cell phones, …

Getting Around Firewalls

Subverting Firewalls

- Along with possible bugs, packet filters have a fundamentally limited semantic model
  - May only handle certain network layers or applications

- How can a local user who wants to get around their site's firewall exploit this? (**Note**: we're not talking about how an external attacker can escape a firewall's restrictions)

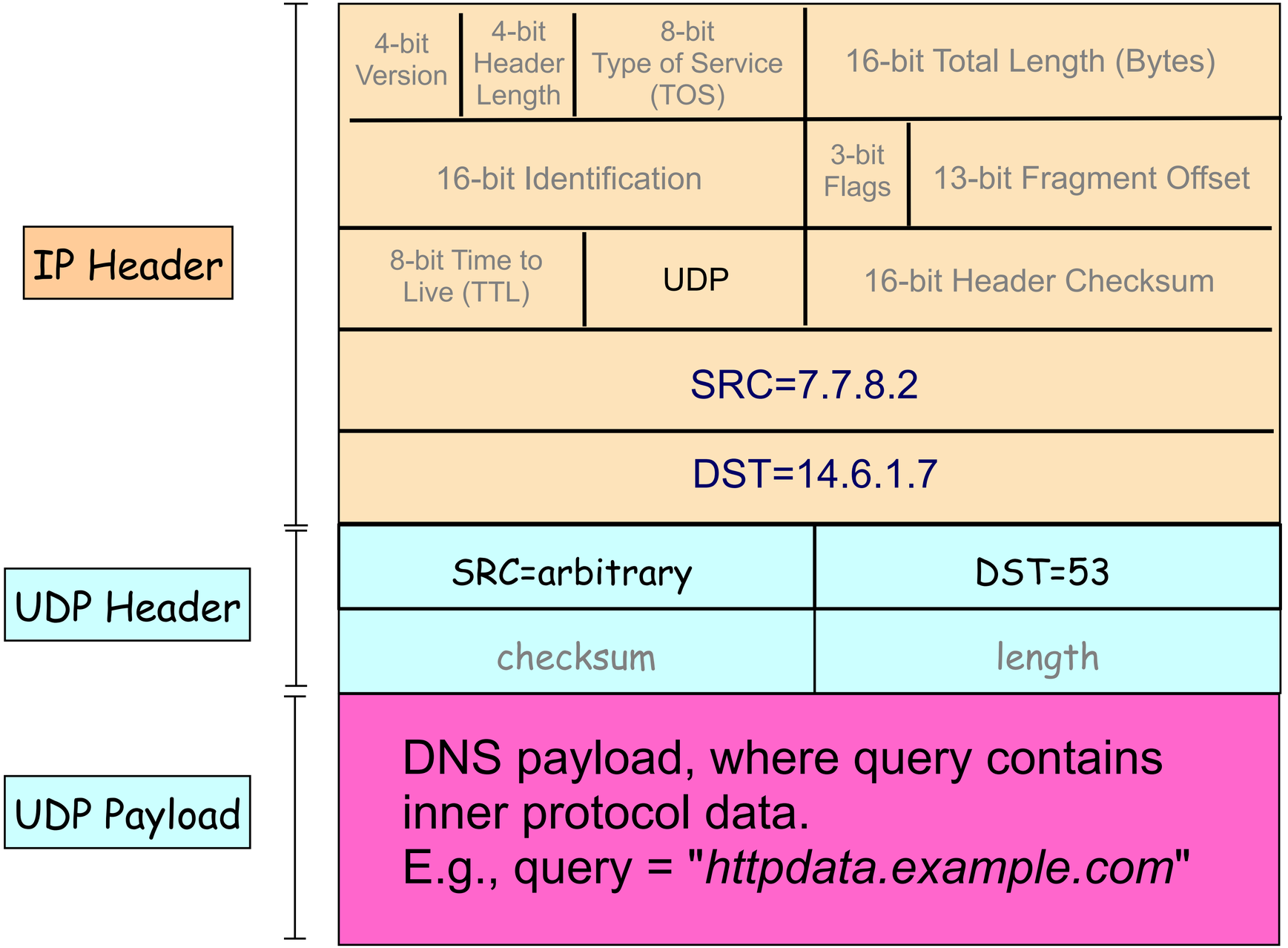
Hiding on Other Ports

- Method #1: use port allocated to another service
  - Who says that e.g. port 53/udp = DNS?
    - Why couldn't it be say Skype or BitTorrent? Just requires that client & server agree on application protocol

- Method #2: **tunneling**
  - Encapsulate one protocol inside another
  - Receiver of "outer" protocol *decapsulates* interior tunneled protocol to recover it
  - Pretty much **any** protocol can be tunneled over another (with enough effort)

- Detecting these methods?
  - **Deep Packet Inspection (DPI):** process packet data to check for correct protocols and apply policies.

Tunnel HTTP GET Request through DNS

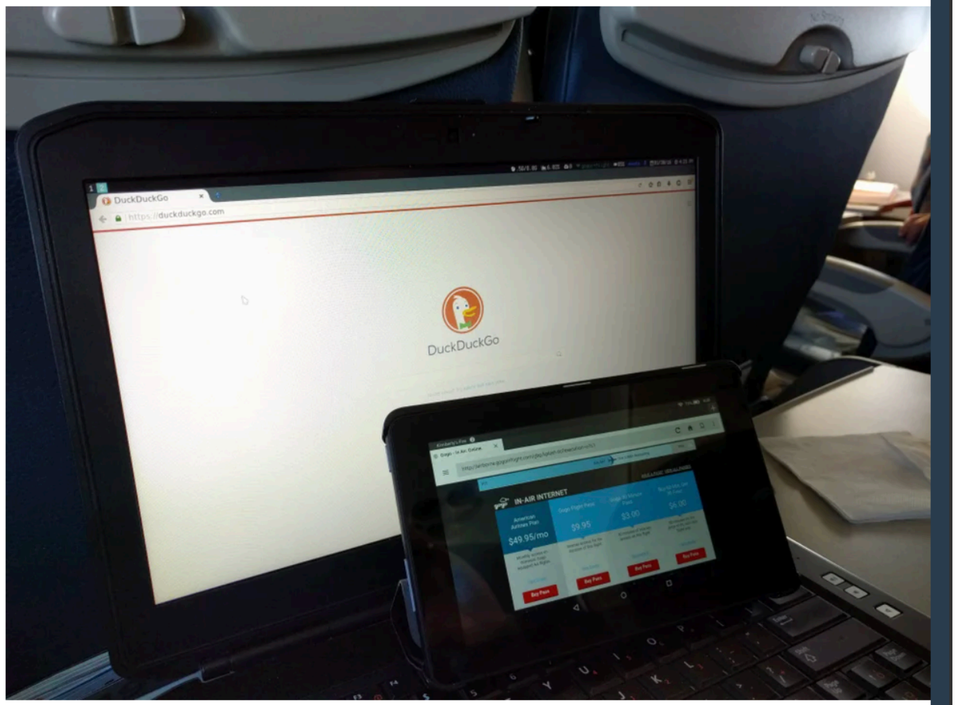


HTTP over DNS tunnel in practice!

- Iodine: Tunnel IPv4 traffic through DNS

All These Other Plebs Are Paying for in-flight Wi-Fi: Linux with Iodine DNS Tunneling

i.imgur.com/3vYC6V...



74 Comments Share Save Hide Report 98% Upvoted

iodine

The latest code is on github

Latest relase (from 2014-06-16): **0.7.0**
Download source / binaries: win32/64, android
Older downloads available below.

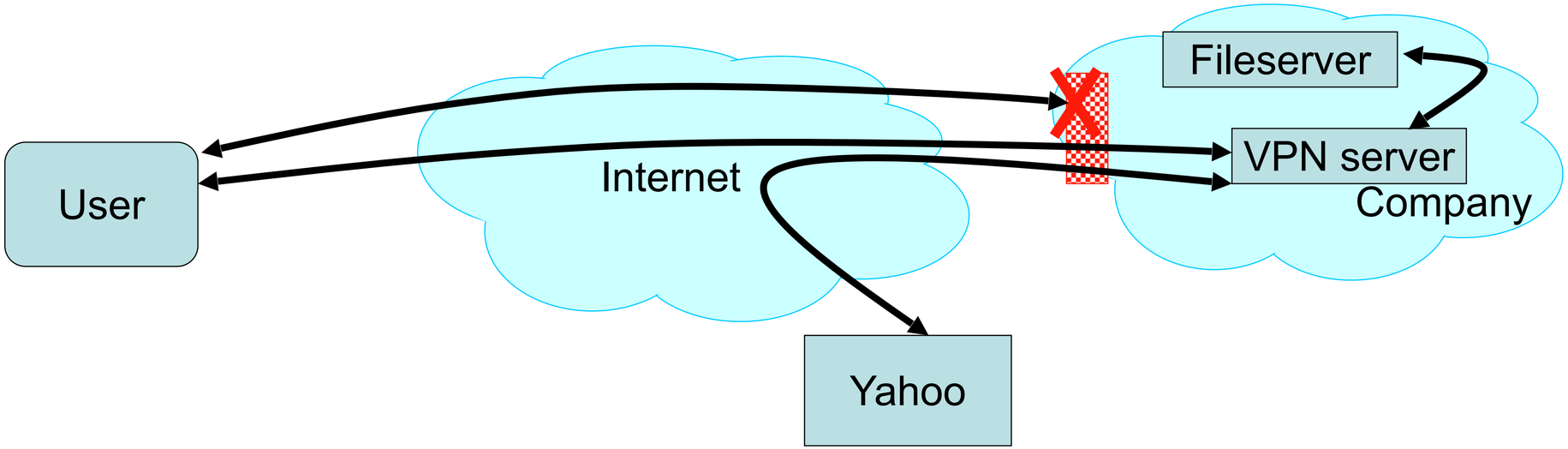
iodine lets you tunnel IPv4 data through a DNS server. This can be usable in different situations where internet access is firewalled, but DNS queries are allowed.

It runs on Linux, Mac OS X, FreeBSD, NetBSD, OpenBSD and Windows and needs a TUN/TAP device. The bandwidth is asymmetrical with limited upstream and up to 1 Mbit/s downstream.

Network Control & Tunneling

- *Tunneling* = embedding one protocol inside another
  - Sender and receiver at each side of the tunnel both cooperate (so it's not useful for initial attacks)
- Traffic takes on properties of outer protocol
  - Including for firewall inspection, which generally can't analyze inner protocol (due to complexity, unless implementing DPI)
- Tunneling has legitimate uses
  - E.g., Virtual Private Networks (VPNs)
    - Remote client establishes a tunnel to a VPN server, which relays/ forwards the client's packets
    - Makes remote client look like it's local to the VPN server's network
    - Tunnel encrypts traffic for privacy & to prevent meddling

Secure External Access to Inside Machines

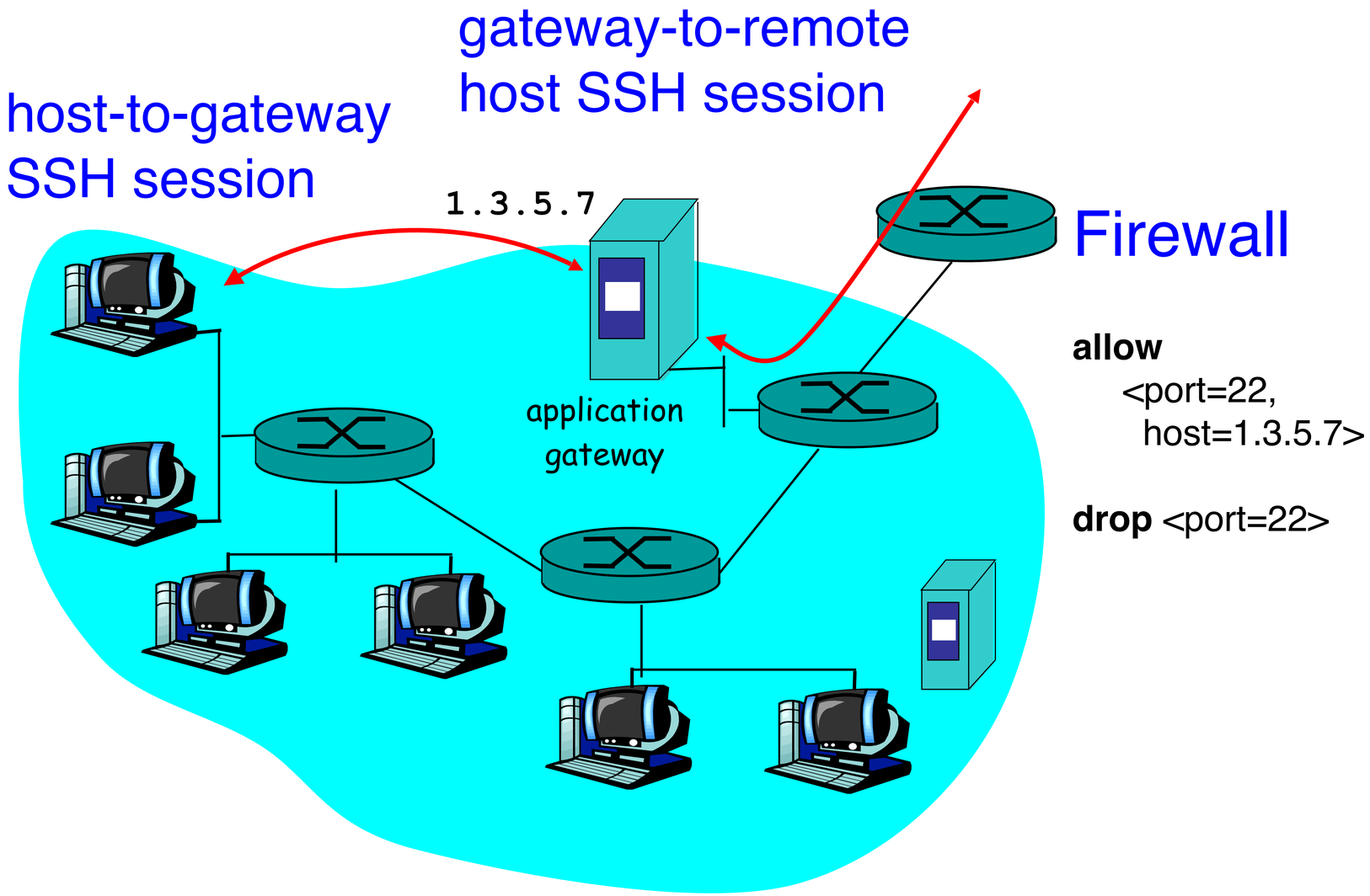


- Often need to provide secure remote access to a network protected by a firewall
  - Remote access, telecommuting, branch offices, …
- Create secure channel (*Virtual Private Network*, or **VPN**) to tunnel traffic from outside host/network to inside network
  - Provides Authentication, Confidentiality, Integrity
    - Requires some form of key management to set up
  - However, also raises *perimeter issues*

Application Proxies

- Can more directly control applications by requiring them to go through a proxy for external access
  - Proxy doesn't simply forward, but acts as an application-level middleman
- Example: SSH gateway
  - Require all SSH in/out of site to go through gateway
  - Gateway logs authentication, inspects decrypted text
  - Site's firewall configured to *prohibit any other* SSH access

SSH Gateway Example



Application Proxies

- Can more directly control applications by requiring them to go through a proxy for external access
  - Proxy doesn't simply forward, but acts as an application-level middleman
- Example: SSH gateway
  - Require all SSH in/out of site to go through gateway
  - Gateway logs authentication, inspects decrypted text
  - Site's firewall configured to prohibit any other SSH access
- Provides a powerful degree of monitoring/control
- Costs?
  - Need to run extra server(s) per app (possible *bottleneck*)
  - Each server requires careful hardening

Today: Network Denial of Service

General Communication Security Goals: CIA

- Confidentiality
  - No one can *read* our data / communication unless we want them to
- Integrity
  - No one can *manipulate* our data / processing / communication unless we want them to
- Authentication
  - We can determine who created a given message / data

General Communication Security Goals: CIAA

- Confidentiality
  - No one can *read* our data / communication unless we want them to
- Integrity
  - No one can *manipulate* our data / processing / communication unless we want them to
- Authentication
  - We can determine who created a given message / data
- Availability
  - We can *access* our data / conduct our processing / use our communication capabilities when we want to

Network Attacks on Availability

- Network Denial-of-Service (DoS, or "*doss*"): *keeping someone from using a network service*
- How broad is this sort of threat?
  - *Very*: **huge** attack surface
- We do though need to consider our threat model …
  - What might motivate a network-based DoS attack?

Motivations for DoS??

- ?

Motivations for DoS

Botnets Beat Spartan Laser on *Halo 3*

By Kevin Poulsen ✉ February 4, 2009 | 12:13 pm | Categories: Cybarmageddon!

What's the most powerful weapon you can wield when playing *Halo 3* online?

I know. You can control the entire map with a battle rifle and a couple of sticky grenades. But that teeny-bopper you just pwned has you beat with the tiny botnet he leased with his allowance money.

Extortion via DDoS on the rise

By Denise Pappalardo and Ellen Messmer, Network World, 05/16/05

Criminals are increasingly targeting corporations with distributed denial-of-service attacks designed not to disrupt business networks but to extort thousands of dollars from the companies.

Ivan Maksakov, Alexander Petrov and Denis Stepanov were accused of receiving $4 million from firms that they threatened with cyberattacks.

The trio concentrated on U.K. Internet gambling sites, according to the prosecution. One bookmaker, which refused to pay a demand for $10,000, was attacked and brought offline--which reportedly cost it more than $200,000 a day in lost business.

November 17th, 2008

Anti fraud site hit by a DDoS attack

Posted by Dancho Danchev @ 4:01 pm

Categories: Botnets, Denial of Service (DoS), Hackers, Malware, Pen testing...

Tags: Security, Cybercrime, DDoS, Fraud, Bobbear...

The popular British anti-fraud site **Bobbear.co.uk** is currently under a DDoS attack (distributed denial of service attack) , originally launched last Wednesday, and is continuing to hit the site with 3/4 million hits daily from hundreds of thousands of malware infected hosts mostly based in Asia and Eastern Europe, according to the site's owner. Targeted DDoS attacks against anti-fraud and volunteer cybercrime fighting communities clearly indicate the impact these communities have on the revenue stream of scammers, and with Bobbear attracting such a high profile underground attention, the site is indeed doing a very good job.

December 8, 2010, 4:18 PM

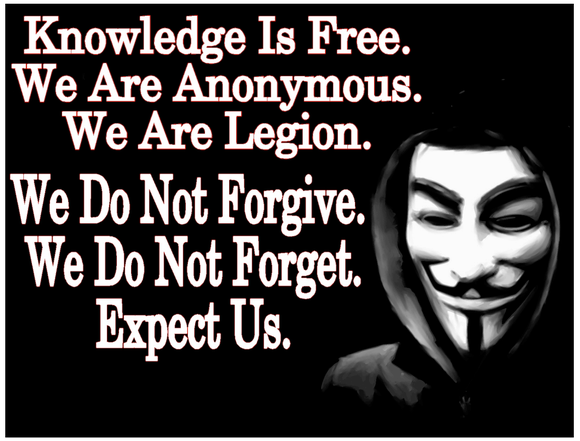
‘Operation Payback’ Attacks Fell Visa.com

By ROBERT MACKEY

A message posted on Twitter by a group of Internet activists announcing the start of an attack on Visa’s Web site, in retaliation for the company’s actions against WikiLeaks.

Last Updated | 6:54 p.m. A group of Internet activists took credit for crashing the Visa.com Web site on Wednesday afternoon, hours after they launched a similar attack on MasterCard. The cyber attacks, by activists who call themselves Anonymous, are aimed at punishing companies that have acted to stop the flow of donations to WikiLeaks in recent days.

The group explained that its distributed denial of service attacks — in which they essentially flood Web sites site with traffic to slow them down or knock them offline — were part of a broader effort called Operation Payback, which



Russia accused of unleashing cyberwar to disable Estonia

· Parliament, ministries, banks, media targeted
· Nato experts sent in to strengthen defences

Ian Traynor in Brussels
The Guardian, Thursday 17 May 2007
Article history

Bronze Soldier, the Soviet war memorial removed from Tallinn. Nisametdinov/AP

A three-week wave of massive cyber-attacks on the small Baltic country of Estonia, the first known incidence of such an assault on a state, is causing alarm across the western alliance, with Nato urgently examining the offensive and its implications.

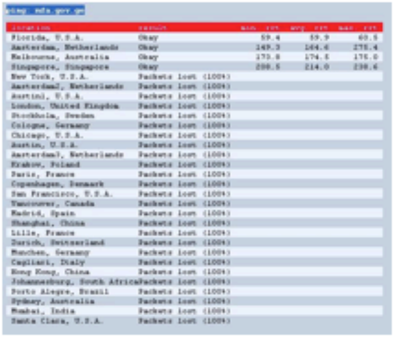
August 11th, 2008

Coordinated Russia vs Georgia cyber attack in progress

Posted by Dancho Danchev @ 4:23 pm

Categories: Black Hat, Botnets, Denial of Service (DoS), Governments, Hackers...
Tags: Security, Cyber Warfare, DDoS, Georgia, South Osetia...

62 TalkBacks ADD YOUR OPINION | SHARE | PRINT | E-MAIL | WORTHWHILE? +18 24 VOTES



In the wake of the Russian-Georgian conflict, a week worth of speculations around Russian Internet forums have finally materialized into a coordinated cyber attack against Georgia's Internet infrastructure. The attacks have already managed to compromise several government web sites, with continuing DDoS attacks against numerous other Georgian government sites, prompting the government to switch to hosting locations to the U.S, with Georgia's Ministry of Foreign Affairs undertaking a desperate step in order to disseminate real-time

Network Attacks on Availability

- Network Denial-of-Service (DoS, or "*doss*"): *keeping someone from using a network service*
- How broad is this sort of threat?
  - *Very*: **huge** attack surface
- We do though need to consider our threat model …
  - What might motivate a network-based DoS attack?
- Two basic approaches available to an attacker:
  - Deny service via a **program flaw**
    - E.g., supply an input that crashes a server
    - E.g., fool a system into shutting down
  - Deny service via **resource exhaustion**
    - E.g., consume CPU, memory, disk, network

DoS Defense in General Terms

- Defending against program flaws requires:
  - Careful coding/testing/review
  - Careful *authentication*
    - Don't obey shut-down orders from imposters
  - Consideration of *behavior of defense mechanisms*
    - E.g. buffer overflow detector that when triggered halts execution to prevent code injection ⇒ denial-of-service
- Defending resources from exhaustion can be **really** hard. Requires:
  - *Isolation mechanisms*
    - Keep adversary's consumption from affecting others
  - *Reliable identification* of different users
    - Know who the adversary is in the first place!

DoS & Networks

- How could you DoS a target's Internet access?
  - Send a zillion packets at them
  - Internet *lacks isolation* between traffic of different users!
- What resources does attacker need to pull this off?
  - At least as much sending capacity ("bandwidth") as the bottleneck link of the target's Internet connection
    - Attacker sends maximum-sized packets
  - **Or**: overwhelm the rate at which the bottleneck router can process packets
    - Attacker sends minimum-sized packets!
      - (in order to maximize the packet arrival rate)

Defending Against Network DoS

- Suppose an attacker has access to a beefy system with high-speed Internet access (a “big pipe”).
- They pump out packets towards the target at a very high rate.
- What might the target do to defend against the onslaught?
  - Install a network filter to discard any packets that arrive with attacker's IP address as their source
    - E.g., `drop * 66.31.1.37:* -> *:*`
    - Or it can leverage *any other packet pattern* in the flooding traffic that’s not in benign traffic
  - Filter = *isolation mechanism*
  - Attacker’s IP address = means of *identifying* misbehaving user

Filtering Sounds Pretty Easy …

- … but it's not. What steps can the attacker take to defeat the filtering?
  - Make traffic appear as though it's from many hosts
    - Spoof the source address so it can't be used to filter
      - Just pick a random 32-bit source IP address for each packet sent
    - How does a defender filter this?
      - They don't! (Unless the traffic has some sort of identifying quirk)
      - Best they can hope for is that operators around the world implement anti-spoofing mechanisms like ingress + egress filtering

Filtering Sounds Pretty Easy …

- … but it's not. What steps can the attacker take to defeat the filtering?
  - Make traffic appear as though it's from many hosts
    - Spoof the source address so it can't be used to filter
      - Just pick a random 32-bit source IP address for each packet sent
    - How does a defender filter this?
      - They don't! (Unless the traffic has some sort of identifying quirk)
      - Best they can hope for is that operators around the world implement anti-spoofing mechanisms like ingress + egress filtering
  - Use many hosts to send traffic rather than just one
    - Distributed Denial-of-Service = **DDoS** ("dee-doss")
    - Requires defender to install complex filters
    - How many hosts are "enough" for the attacker?
      - Today they are very cheap to acquire …botnets :-(

It’s Not A “Level Playing Field”

- When defending resources from exhaustion, need to beware of asymmetries, where attackers can consume victim resources with little comparable effort
  - Makes DoS easier to launch
  - Defense costs much more than attack

Leveling the Playing Field: TCP SYN Cookie

- Each TCP connection requires server to create a small amount of state
  - To consume a lot of memory, attacker needs lots of connections, so lots of source IPs. Easy solution is to spoof source IPs.

- SYN cookies: don't keep state for each TCP connection!
  - Encode the state within the TCP SYN-ACK seq #, and don't store any state (*isolation*)
  - For legit/unspoofed connections, the TCP ACK will contain this state within the ack #
  - For spoofed addresses, attacker can't send TCP ACK (*identification*)
  - Balances the playing field: attackers need to control many IPs to launch this attack

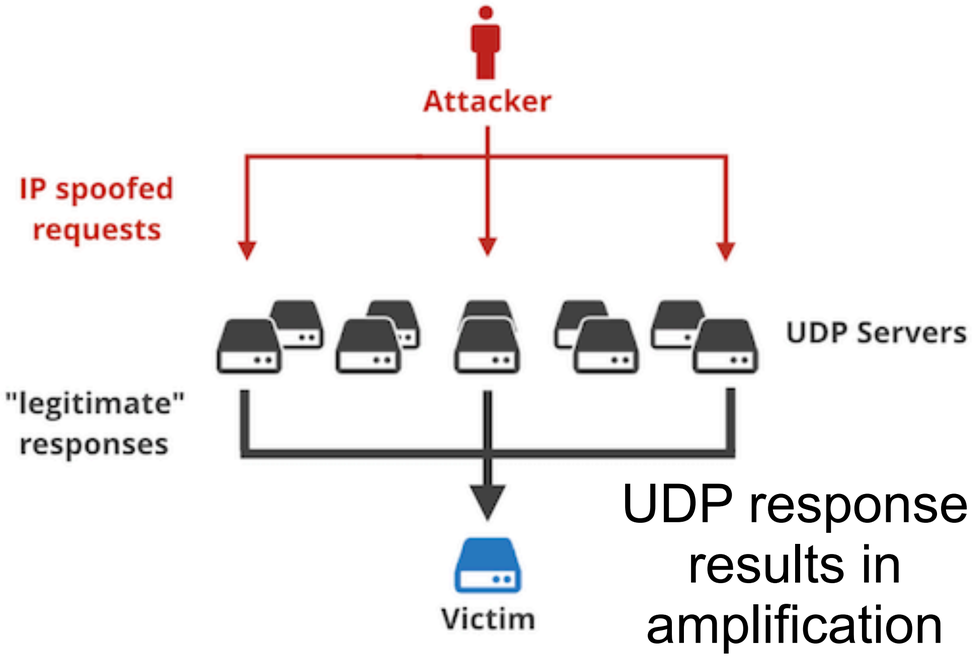
It's Not A "Level Playing Field"

- When defending resources from exhaustion, need to beware of asymmetries, where attackers can consume victim resources with little comparable effort
  - Makes DoS easier to launch
  - Defense costs much more than attack

- Particularly dangerous form of asymmetry: **amplification**
  - Attacker leverages system's own structure to pump up the load they induce on a resource

Amplification Vector: DNS / UDP

- Consider DNS lookups:
  - *Reply is generally much bigger than request*
  - ⇒ Attacker spoofs DNS request to **many** DNS resolvers seemingly from the victim/target
    - Small attacker packet yields large flooding packet
    - May not increase # of packets, but **total byte volume**
  - Works for other request/response protocols too
- Note #1: attacks involve blind spoofing
  - Generally only works for UDP protocols (can't establish TCP conn.)
- Note #2: victim doesn't see attacker address
  - Addresses are those of intermediary systems



Amplification

Cat	Protocol	Port(s)	Description
Network Svc	SNMP v2	161	Monitoring network-attached devices
	NTP	123	Time synchronization
	DNS	53	(Primarily) Domain name resolution
	NetBios	137	Name service protocol of NetBios API
	SSDP	1900	Discovery of UPnP-enabled hosts
Leg.	CharGen	19	Legacy character generation protocol
	QOTD	17	Legacy "Quote-of-the-day" protocol
P2P	BitTorrent	any	BitTorrent's Kademlia DHT impl.
	Kad	any	eMule's Kademlia DHT impl.
Gam	Quake 3	27960	Games using the Quake 3 engine
	Steam	27015	Games using the Steam protocol
Bots	ZAv2	164XY	P2P-based rootkit
	Sality	any	P2P-based malware dropper
	Gameover	any	P2P-based banking trojan

TABLE I: Overview of the analyzed network protocols grouped into five categories: network services, legacy protocols, P2P file sharing, multiplayer games and P2P-based botnets. The *Port(s)* column describes the typical UDP listening port for the service or specifies *any* if ports are chosen randomly.

Christian Rossow. "Amplification Hell: Revisiting Network Protocols for DDoS Abuse." NDSS 2014

Amplification

Protocol	Amplifiers	Tech.	t_{1k}	t_{100k}
SNMP v2	4,832,000	Scan	1.5s	148.9s
NTP	1,451,000	Scan	2.0s	195.1s
DNS_{NS}	255,819	Crawl	35.3s	3530.0s
DNS_{OR}	7,782,000	Scan	0.9s	92.5s
NetBios	2,108,000	Scan	3.4s	341.5s
SSDP	3,704,000	Scan	1.9s	193.5s
CharGen	89,000	Scan	80.6s	n/a
QOTD	32,000	Scan	228.2s	n/a
BitTorrent	5,066,635	Crawl	0.9s	63.6s
Kad	232,012	Crawl	0.9s	108.0s
Quake 3	1,059	Master	0.6s	n/a
Steam	167,886	Master	1.3s	137.1s
ZAv2	27,939	Crawl	1.5s	n/a
Sality	12,714	Crawl	4.7s	n/a
Gameover	2,023	Crawl	168.5s	n/a

TABLE II: Number of amplifiers per protocol, the technique we used to obtain the amplifiers, and the time it took to identify 1000 (t_{1k}) and 100,000 (t_{100k}) amplifiers, respectively.

Christian Rossow. "Amplification Hell: Revisiting Network Protocols for DDoS Abuse." NDSS 2014

Amplification

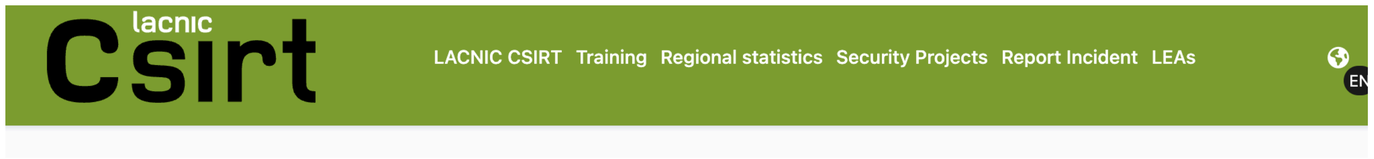
Protocol	BAF *all*	BAF 50%	BAF 10%	PAF *all*	Scenario
SNMP v2	6.3	8.6	11.3	1.00	*GetBulk* request
NTP	556.9	1083.2	4670.0	3.84	Request client statistics
DNS_{NS}	54.6	76.7	98.3	2.08	ANY lookup at author. NS
DNS_{OR}	28.7	41.2	64.1	1.32	ANY lookup at open resolv.
NetBios	3.8	4.5	4.9	1.00	Name resolution
SSDP	30.8	40.4	75.9	9.92	*SEARCH* request
CharGen	358.8	n/a	n/a	1.00	Character generation request
QOTD	140.3	n/a	n/a	1.00	Quote request
BitTorrent	3.8	5.3	10.3	1.58	File search
Kad	16.3	21.5	22.7	1.00	Peer list exchange
Quake 3	63.9	74.9	82.8	1.01	Server info exchange
Steam	5.5	6.9	14.7	1.12	Server info exchange
ZAv2	36.0	36.6	41.1	1.02	Peer list and cmd exchange
Sality	37.3	37.9	38.4	1.00	URL list exchange
Gameover	45.4	45.9	46.2	5.39	Peer and proxy exchange

TABLE III: Bandwidth amplifier factors per protocols. *all* shows the average BAF of all amplifiers, 50% and 10% show the average BAF when using the worst 50% or 10% of the amplifiers, respectively.

Christian Rossow. "Amplification Hell: Revisiting Network Protocols for DDoS Abuse." NDSS 2014

Efforts to Change Amplification Landscape

Community efforts to get amplifiers offline and disable the worst amplification features



DNS Open Resolvers on IPv4

LACNIC CSIRT is working together with CSIRT CEDIA on the "DNS Open Resolvers on IPv4" project that seeks to understand the current status of the region, identify open resolvers, and proactively alert and recommend potential corrections to how this service is configured.

Home > What We Do > Network Reporting > Open CharGen Report

Open CharGen Report

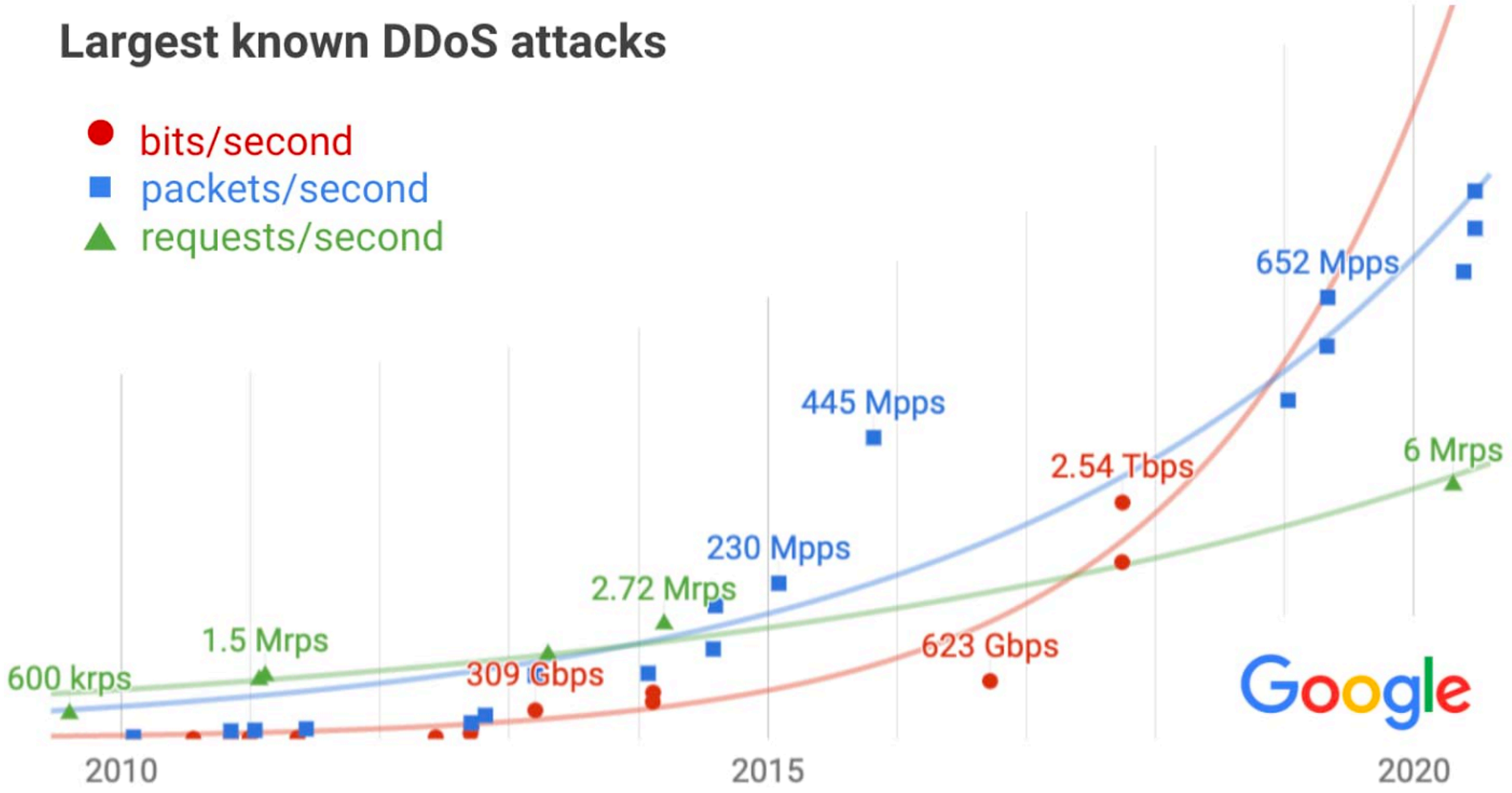
LAST UPDATED: 2023-02-06

This report identifies hosts that have the CharGen service running and accessible on the Internet.

Defenses Beyond Filtering

- So if filtering isn't fully effective...what else can we do?

- In practice, one of the main defenses today is relying on a cloud/CDN provider to "scale up" service as a defense (e.g., Cloudflare, Google, Akamai, Fastly)
  - In the benign/non-attack state, CDN wouldn't serve too many requests
  - If an attack is detected (or the site is under heavy load, potentially legitimately), the CDN would scale up to serve more requests.



Source: Google Cloud

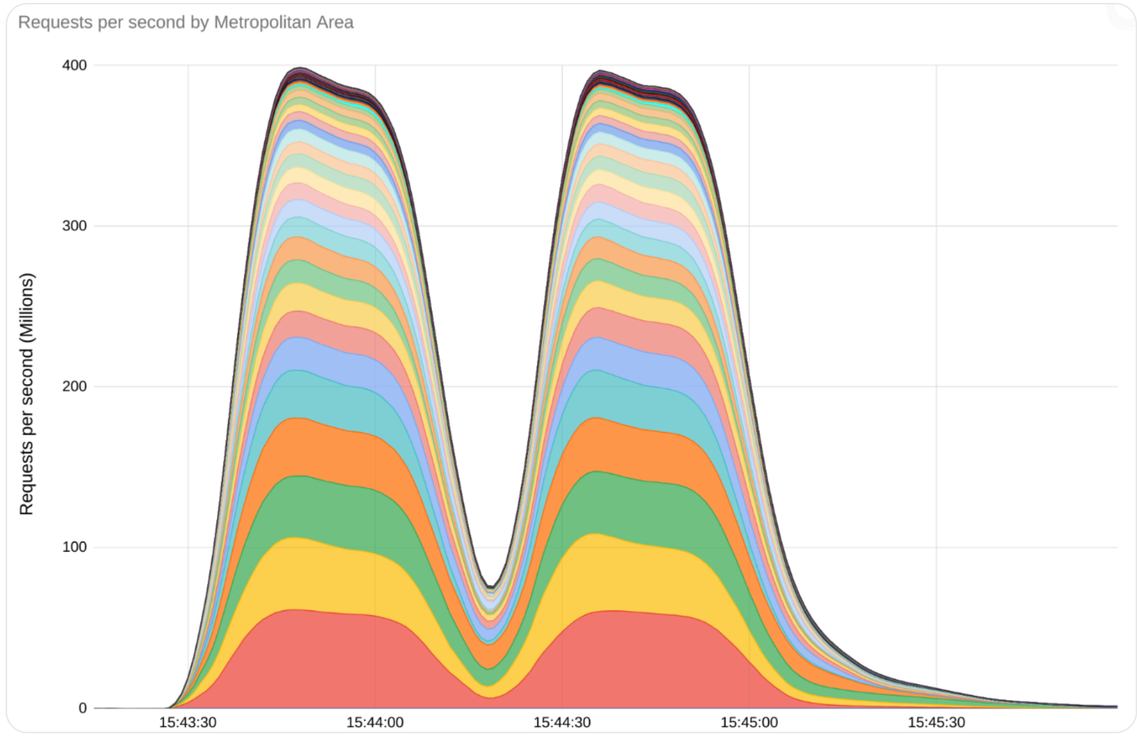
Google Cloud Blog

Contact sales | Get started for fre

Security & Identity

Google mitigated the largest DDoS attack to date, peaking above 398 million rps

October 10, 2023



Google mitigated a DDoS attack which peaked at 398 million requests per second

Source: Google Cloud

Detecting DDoS Attacks across the Internet

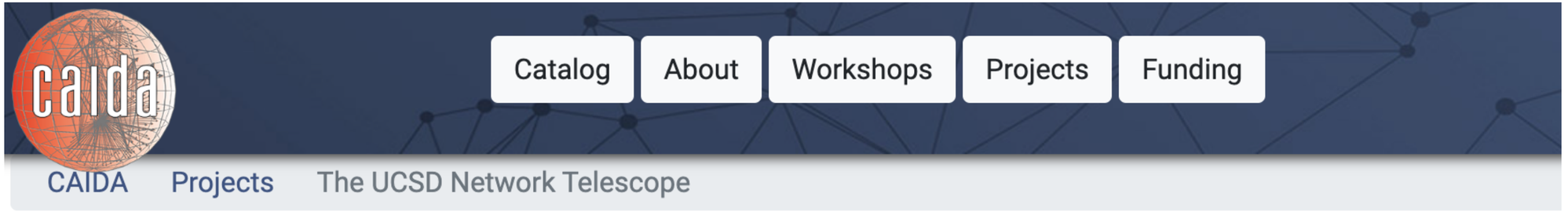
- How can we understand DDoS attacks broadly (especially if we're not a victim/target)?
  - To circumvent filtering, many DDoS attacks will spoof the IP address, most likely picking a random address.
  - That means that in many cases, victim servers will send a response to the spoofed address.

- Network Telescope: inactive range of IP addresses without real hosts or network services
  - Such telescope addresses will receive spoofed responses from victims. This allows one to study both victims and the nature of attacks (assuming random address spoofing)

Network Telescopes

- Example Analysis (for IPv4):
  - Say we have a /8 subnet as a telescope.
  - Given the IPv4 address space is 2^32, this /8 subnet is 2^24 / 2^32 = 1/2^8 = 1/256 ≈ 0.39% of the IPv4 Internet.
  - Let's say we can identify which packets received by the telescope are from an attack (i.e., a signature for the attack, such as the victim/target IP address).
  - Then, we can estimate attack volume (assuming IID randomly spoofed IP source addresses).
    - » Example: Telescope receives 10K PPS attack traffic. Then we can estimate total attack volume is 10K*(256) = 2.56M PPS.

Network Telescopes



On this page

The UCSD Network Telescope

The UCSD Network Telescope is a passive traffic monitoring system built on a globally routed, but lightly utilized /9 and /10 network. Under CAIDA stewardship, this unique resource provides valuable data for network security researchers.

Network Telescopes

Target Nameserver		A	B	C
December 2020 Attack	Observed Packer Rate (PPM)	21.8K	3.8K	2.9K
	Inferred Traffic Volume	1.4 Gbps	247 Mbps	188 Mbps
	Attacker IP Count	5.79M	1.57M	1.33M
March 2021 Attack	Observed Packer Rate (PPM)	125K	123K	13K
	Inferred Traffic Volume	8 Gbps	7.8 Gbps	845 Mbps
	Attacker IP Count	7M	6.19M	823K

Table 2: Attack metrics for two DDoS attacks on TransIP. The first attack targeted nameserver A more intensely; the second targeted all three similarly.

R. Sommese et al. "Investigating the impact of DDoS attacks on DNS infrastructure." IMC 2022.

Network DoS Summary

- Network-based DoS often relies on overwhelming the network link or network processing of a system, making them unavailable. **Hard to defend against!**
  - Try and detect/filter out attack traffic
  - Try and reduce state/computation
  - "Outmuscle" the attacker by having more computational resources than them (i.e., business model for Akamai + Cloudflare)