





Surfing, Snooping, and Spreadsheets: Three Levels of Security Attacks

Owen Barnett, Oliver Calder, Skyler Kessenich, Peter McCrea,
Nick Pandelakis, John Witte



Charger Surfing

Stealing Phone Passwords

From Charging Cables

What are we Replicating?

- A 2020 Usenix paper titled: “Charger Surfing: Exploiting a Power Line Side-Channel for Smartphone Information Leakage.”
- They propose a general attack on charging cables that:
 - Recognizes a target phone’s model.
 - Obtains that phone’s passcode in real time.

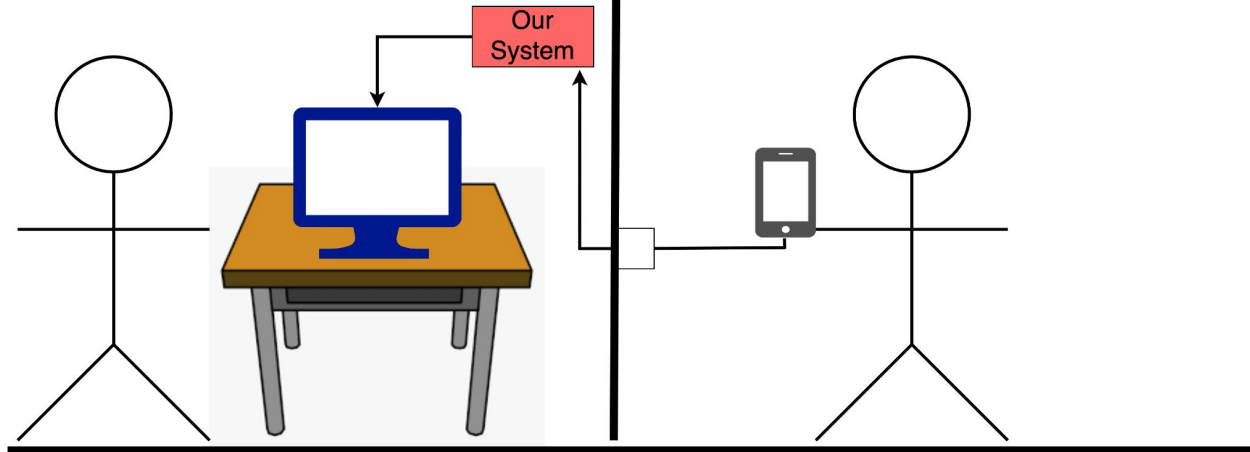
What is a Side-Channel Attack?

- Side-channel attacks exploit the implementation of a computer system.
- These attacks often leverage variabilities in the transfer of data to gain information.
- Examples include:
 - Timing attacks: Guessing passwords using fluctuations in timing.
 - Power-analysis attacks: Using fluctuations in power to gain information (Charger Surfing!).

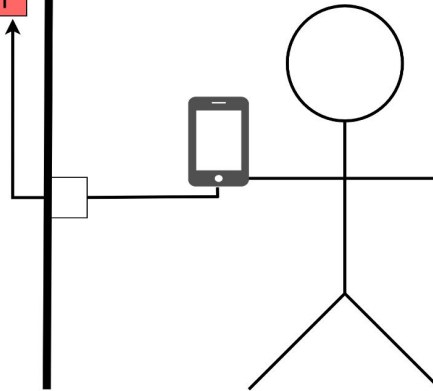
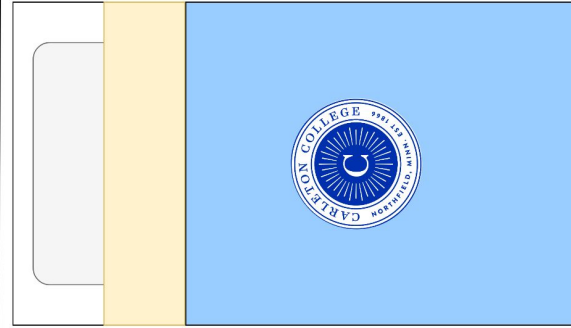
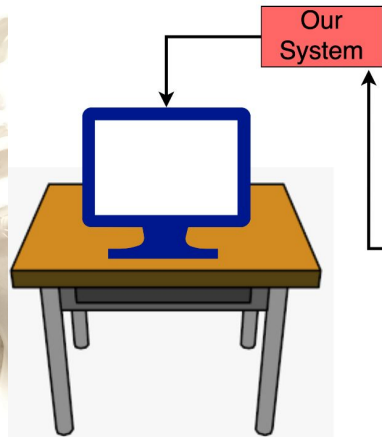
What are we Doing?

- Restricting our analysis to just iPhones.
- Our target: Someone using a phone plugged into a compromised system.
- Our Goals:
 - Identify *when* a button is being pressed on the lock-screen.
 - Identify *which* button is being pressed on the lock-screen.

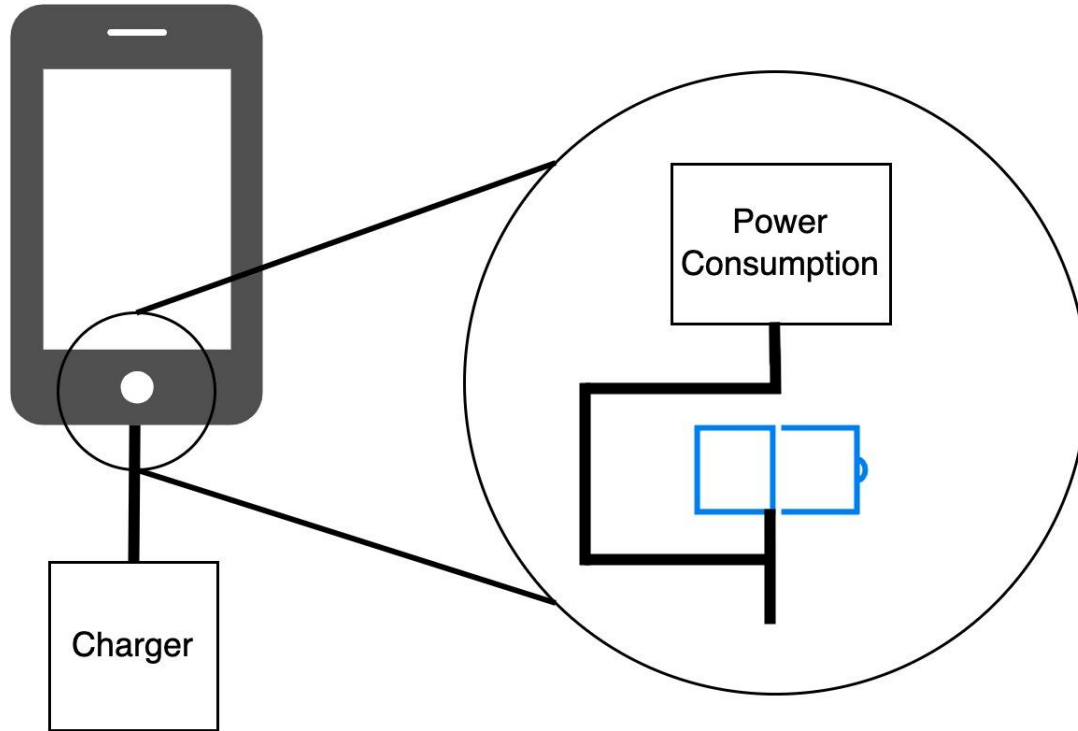
Attack Scenario



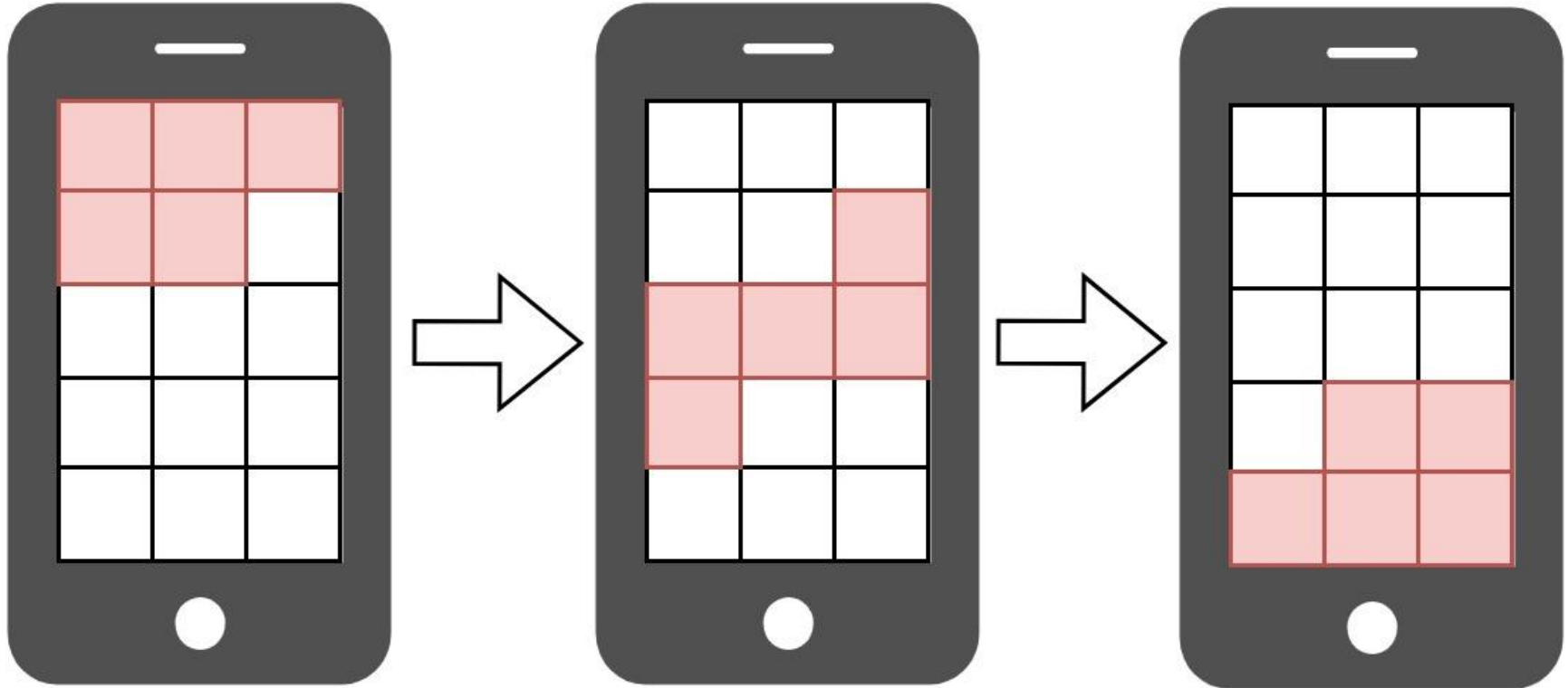
This could be you!



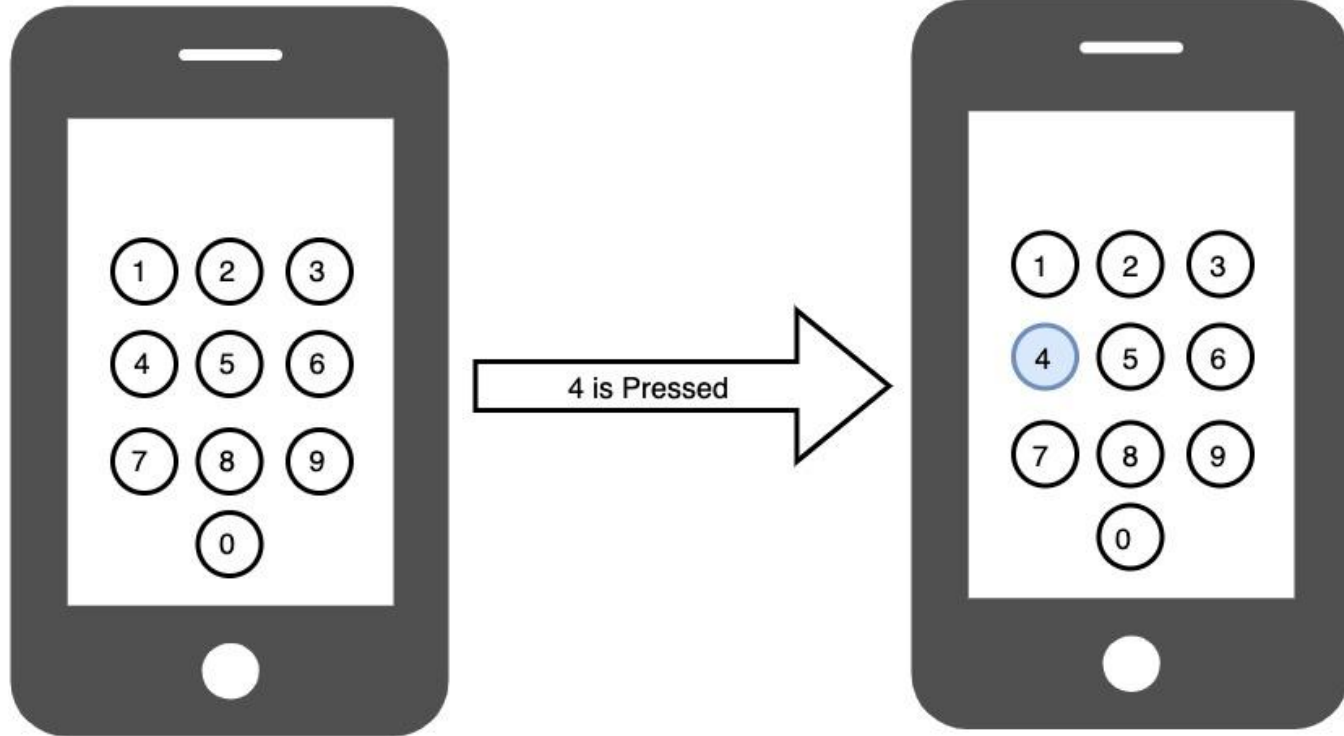
Pass Through Charging



Screen Refresh

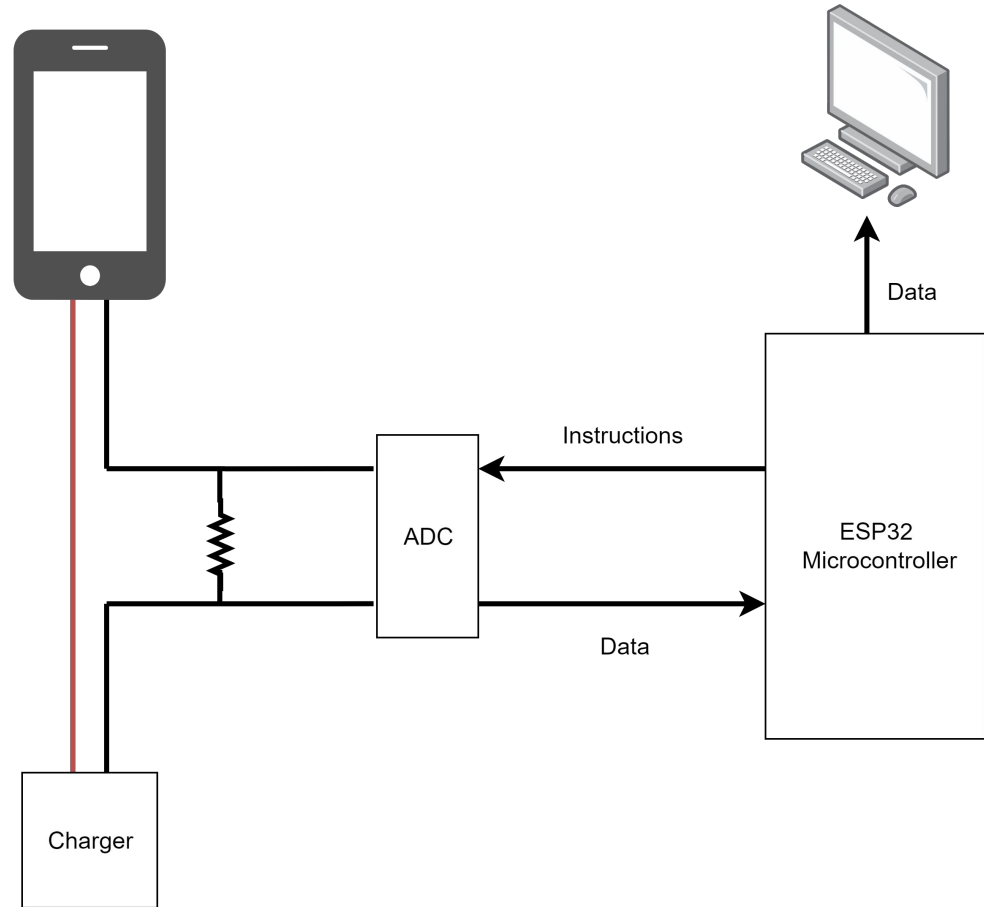


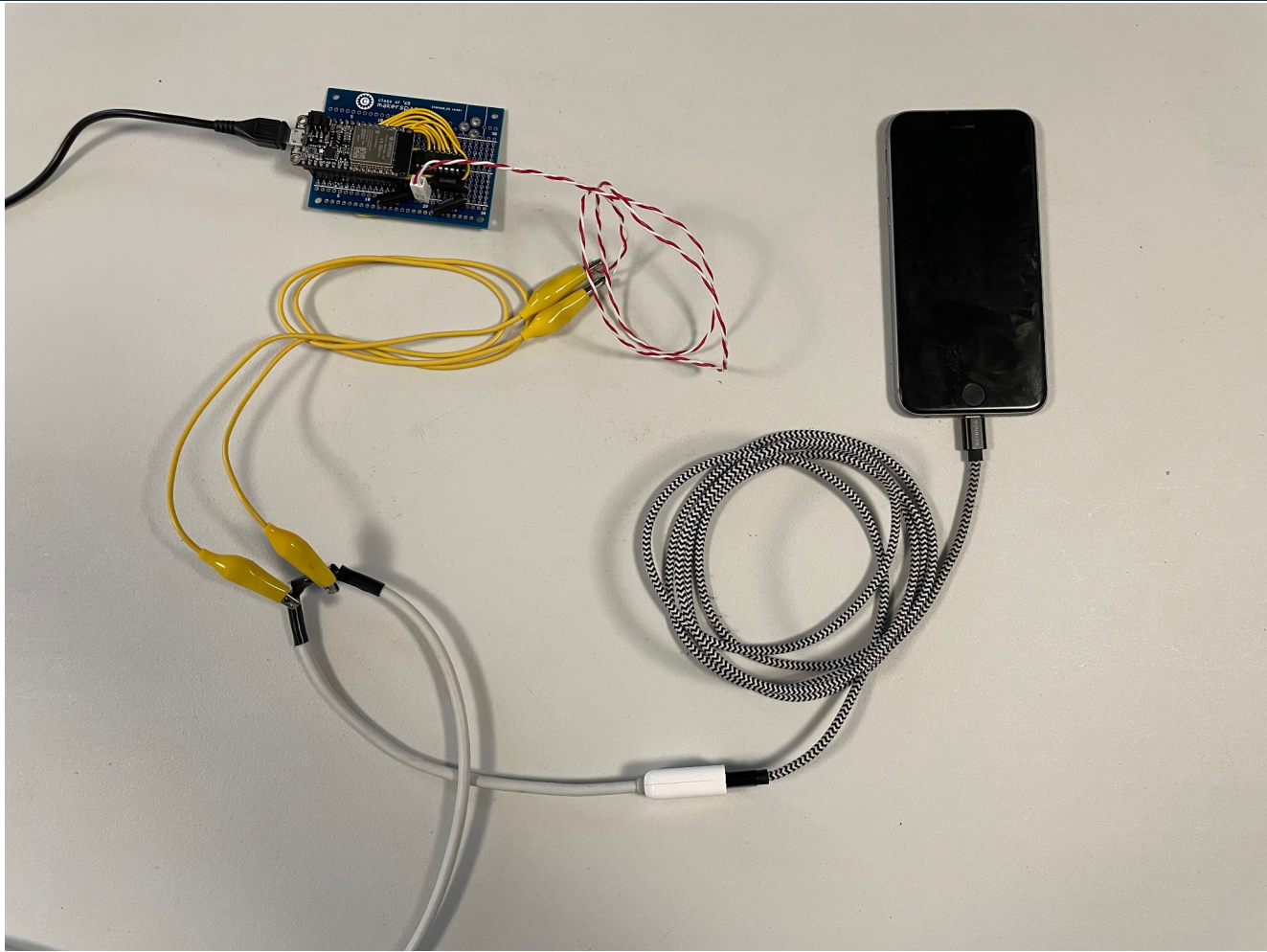
Button Press Animation



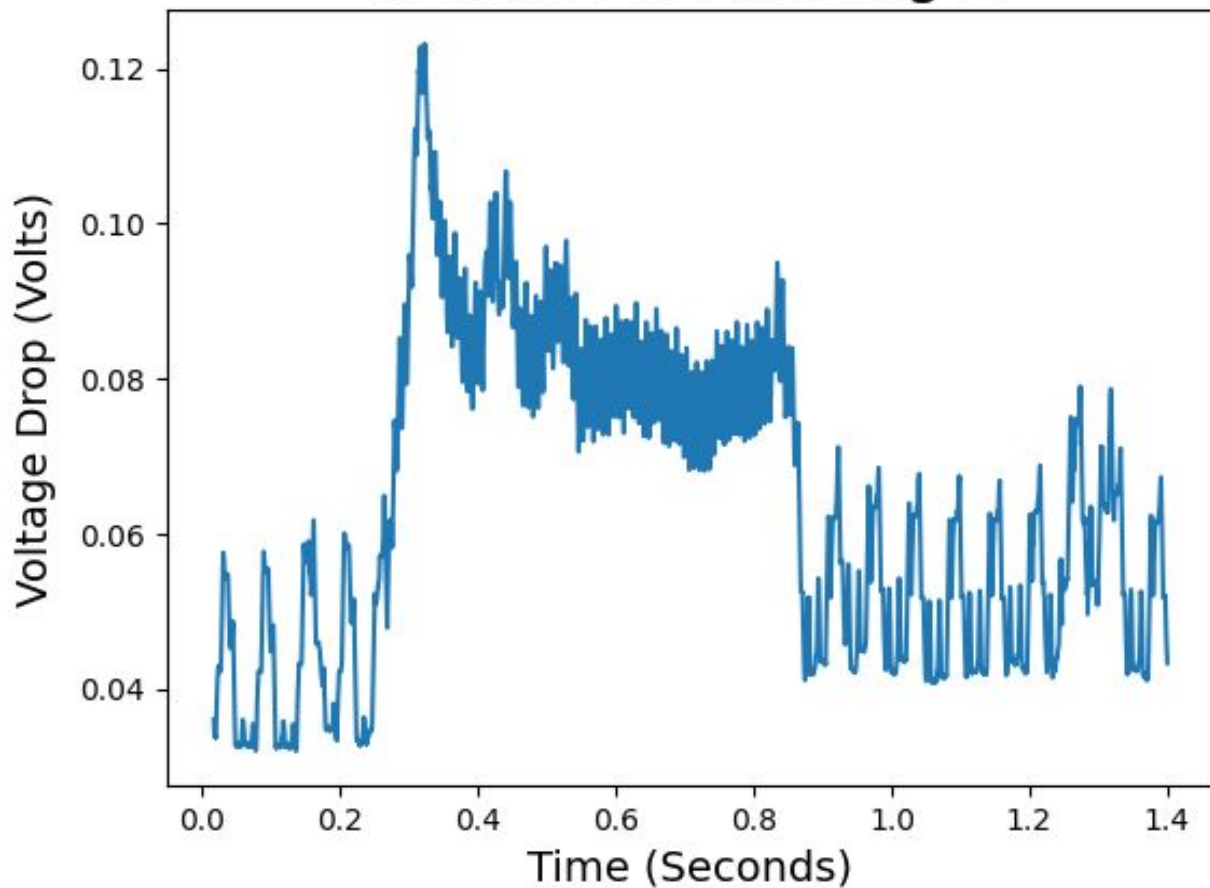
Our Design

- Use an analog-to-digital converter (ADC) to read voltage drops across a small resistor.
- Collect data on an external device using an ESP32 microcontroller.

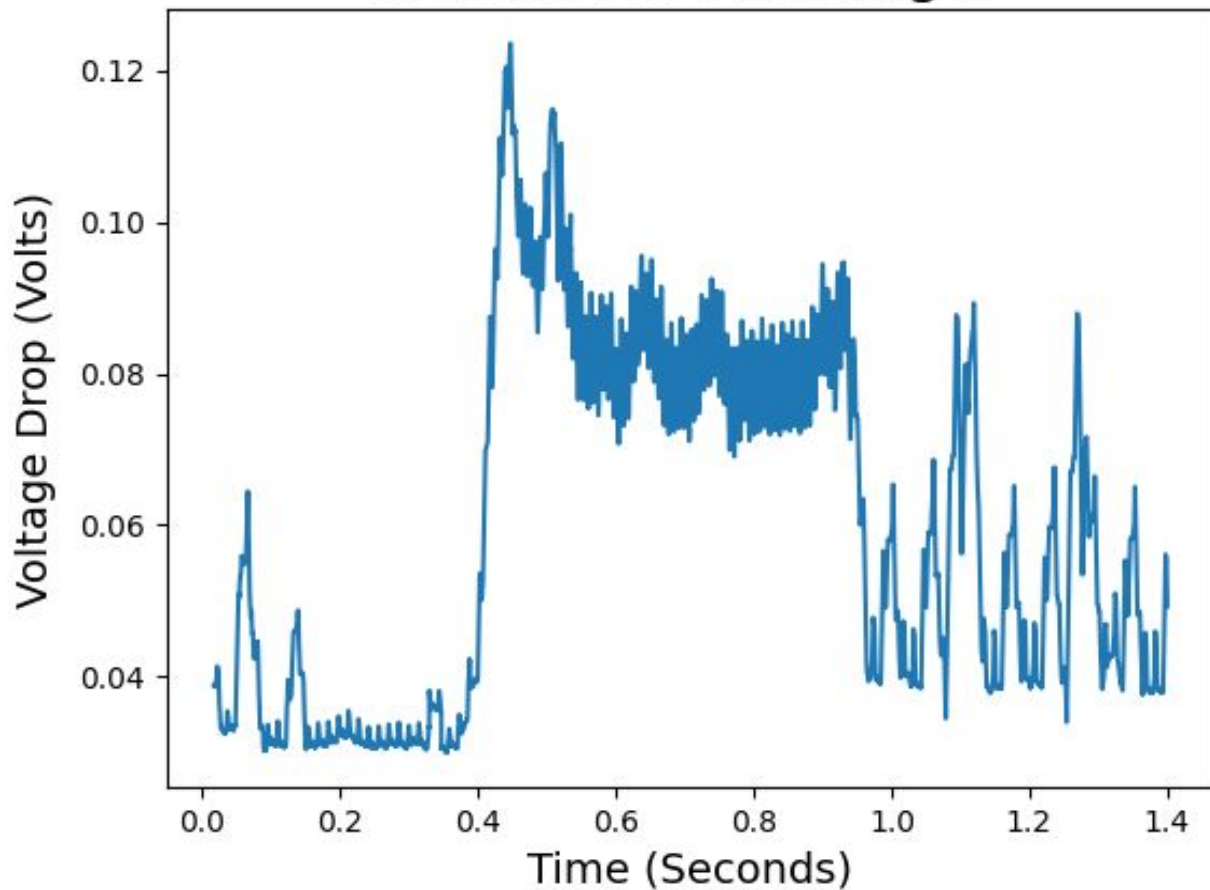




iPhone Xs Pressing 1

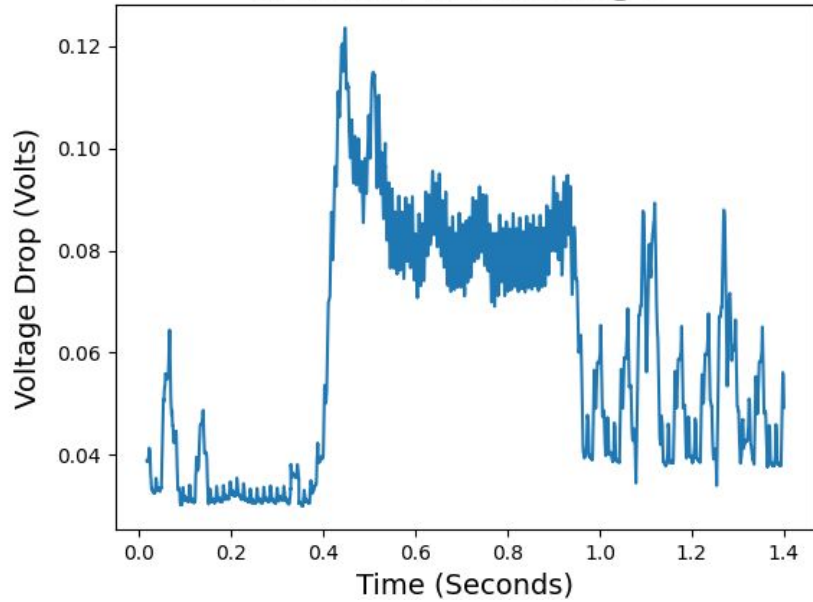


iPhone Xs Pressing 9

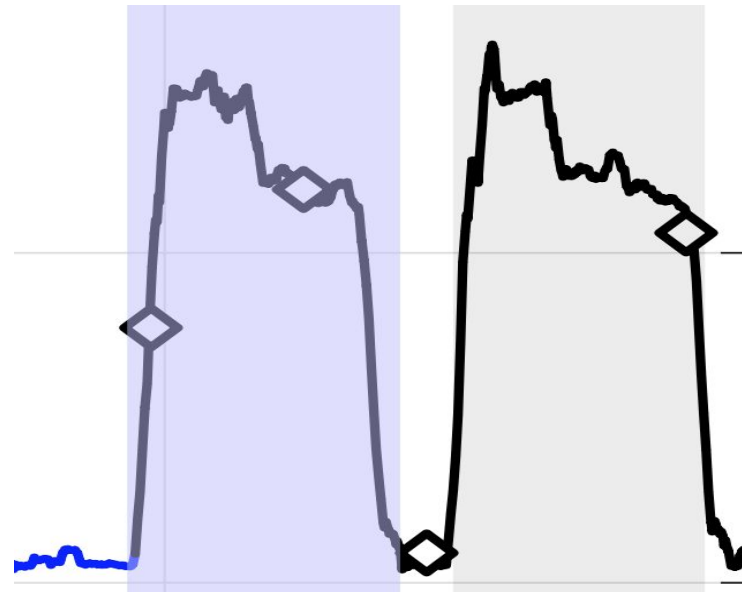


Our Data

iPhone Xs Pressing 9

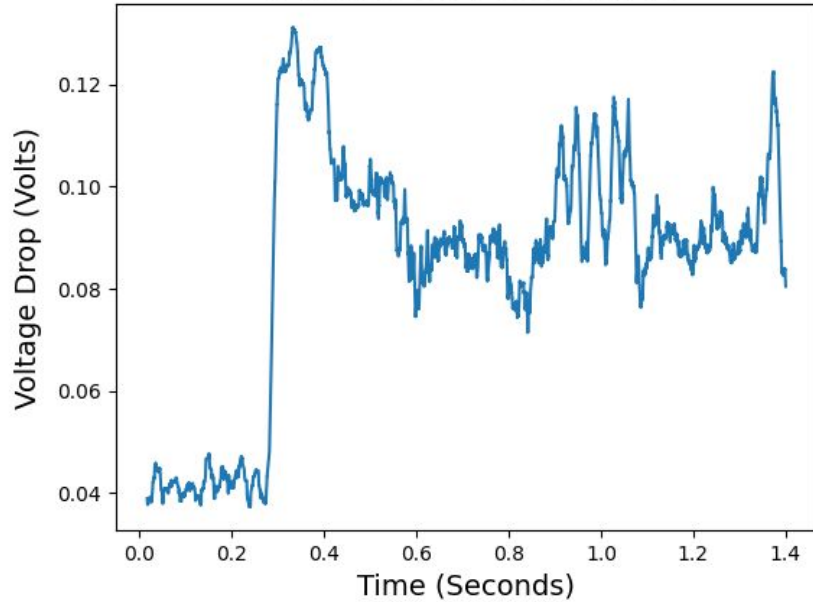


Paper's Data

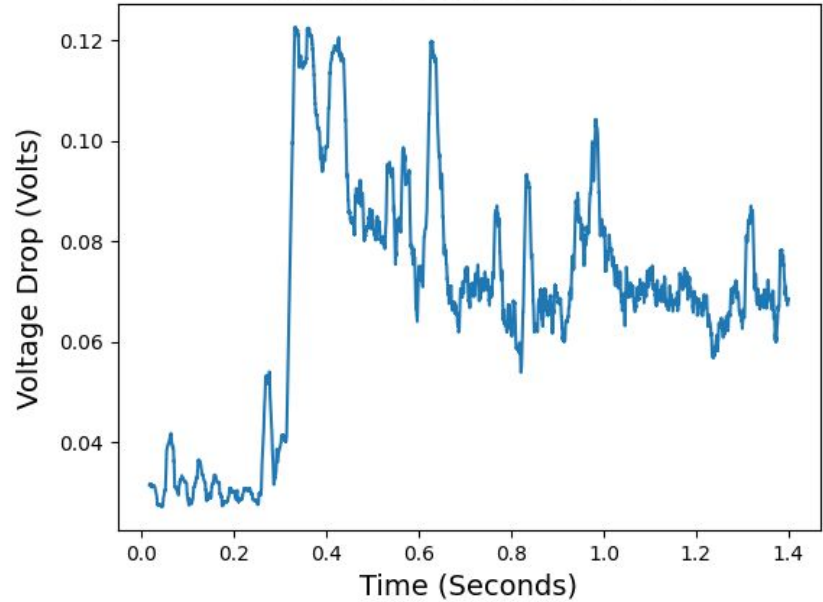


Comparing 1 and 9 on iPhone 6s

iPhone 6s Pressing 1

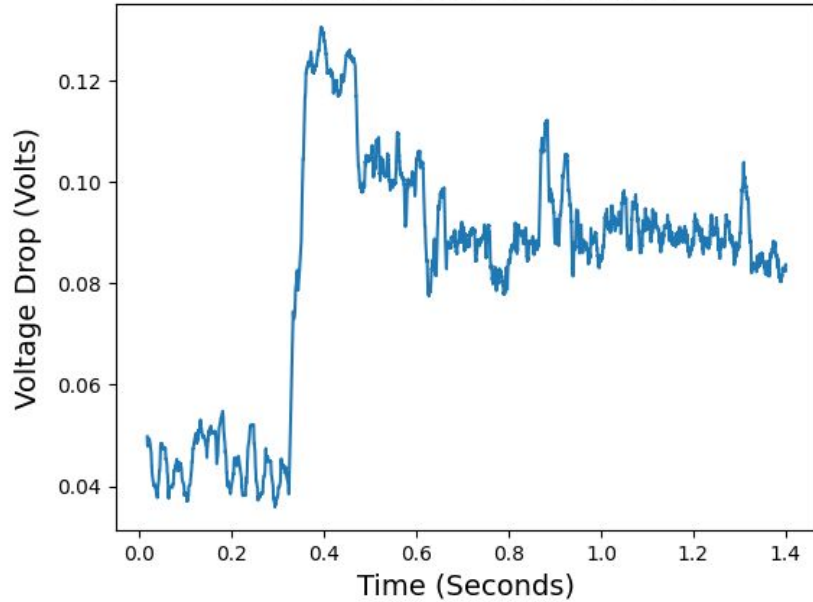


iPhone 6s Pressing 9

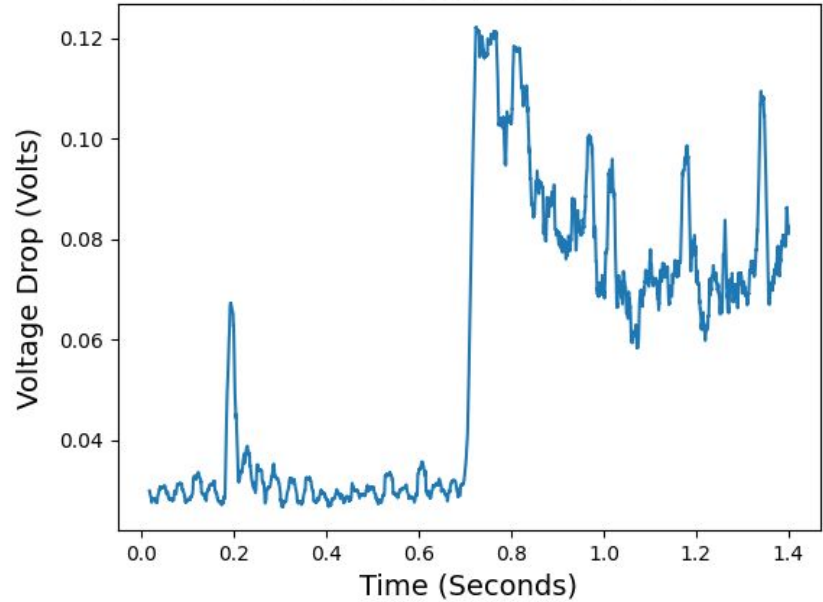


Comparing 1 and 9 on iPhone 6s

iPhone 6s Pressing 1

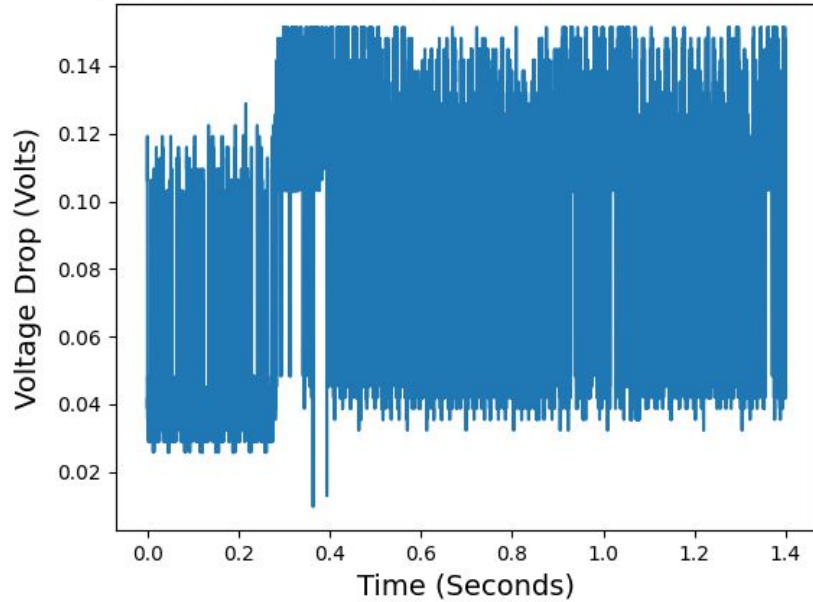


iPhone 6s Pressing 9

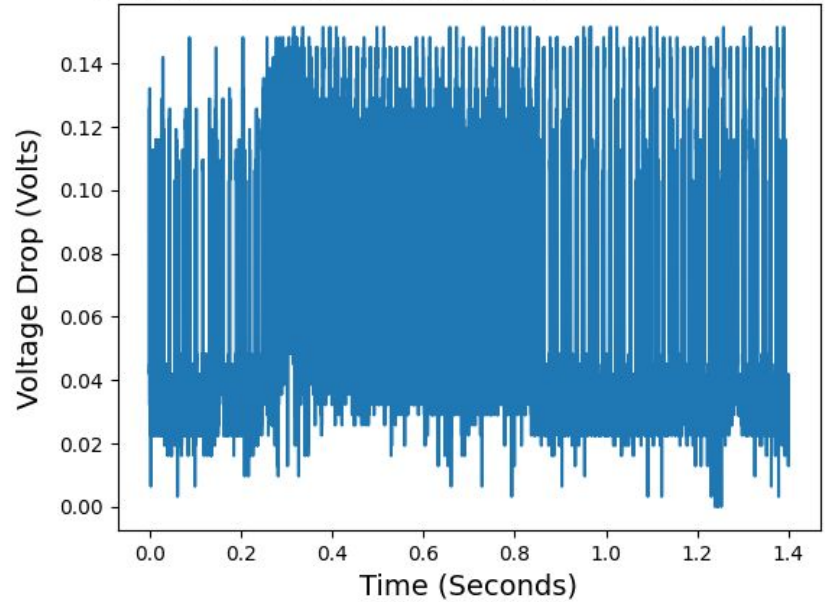


Graphs of Raw Data

iPhone 6s Pressing 1 no Rolling Mean



iPhone Xs Pressing 1 no Rolling Mean



Recognizing Buttons

- We used a convolutional neural network (CNN) to classify voltage data.
- CNNs are well equipped to analyze time-series data.
- CNNs are fairly noise resistant.

Results

- We collected 22 samples of each lock-screen button.
- Using 5-fold cross-validation, we were able to correctly classify buttons with an average of 45% accuracy!
- This is much better than just guessing.

Analysis of Results

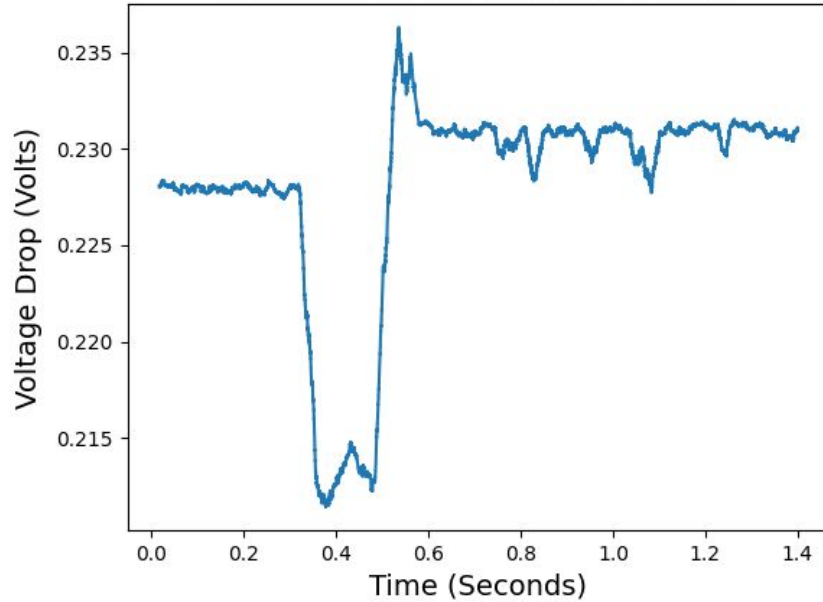
- Our results are *not* robust:
 - This is limited to samples collected on just one day.
 - Our neural net struggled to classify data across days.
 - This limits the real-world effectiveness of this attack.

How Worried Should you be?

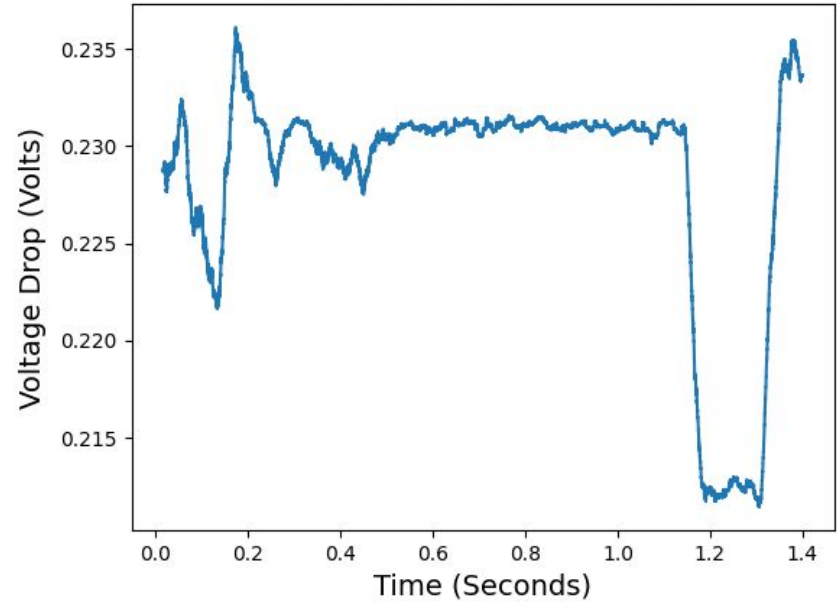
- It is clear that phone charging cables effectively are two-way connections. Power is sent to your phone and information about your phone's processes is sent back.
- Making sense of that data, however, requires precision and sophistication.
- The attack is only useful if:
 - Your phone pin is not a unique pin you use in your life OR
 - An attacker will at some point be able to acquire your phone.

How Worried Should you be?

iPhone 12 no Button Press



iPhone 12 Pressing 1



References

- Cronin, Patrick, et al. "Charger-Surfing: Exploiting a Power Line Side-Channel for Smartphone Information Leakage." *USENIX*, <https://www.usenix.org/conference/usenixsecurity21/presentation/cronin>.

DHT Crawling

Oliver Calder and Peter McCrea



THIS WEBSITE HAS BEEN SEIZED

This domain has been seized by the Federal Bureau of Investigation pursuant to a seizure warrant issued by the United States District Court for the Central District of California under the authority of 18 U.S.C. §1030(i)(1)(A) as part of coordinated law enforcement action taken against illegal DDoS-for-hire services.

This action has been taken in coordination with the United States Attorney's Office of the District of Alaska, the Department of Justice Computer Crime and Intellectual Property Section, and



[Oliver]

Anonymous slide "THIS WEBSITE HAS BEEN SEIZED"...yeah, maybe

Outline

- Peter jumped in to describe the last three bullets. Weird. Switch at a slide boundary.

Traditional Download Model

- Nice diagram

- Problems: slow, hard on the server, maybe say "single point of failure"

Solution: BitTorrent

- Nice

BitTorrent Basics

- Clients/peers

- Central servers: peer-finding, search engine

Anatomy of a torrent

- Metadata (size, title, file names, seeds, etc.)

- Infohash (SHA-1 of metadata, unique ID)

- This slide was perhaps a little slow.

- Better: metadata=title etc., infohash=unique id derived from the metadata, done

- Don't bog down in defining hashes. You can say "for those of you familiar w/ cryptographic hashes, this id is computed as a SHA-1 of the metadata"

Side Effects

- internal data sharing (e.g. Amazon)

- great for illegal file sharing

- central servers still vulnerable

[Peter]

Central Server: Point of Failure

- No initialization w/o server

Solution: Distributed Hash Table

- I missed a little bit of this. Is it clear whether the whole table is busted into pieces? What's in a particular hash table (key, value) pair?

Distr

This section needs more clarity

[Oliver]

DHT Protocol

- ping: "are you there?" "yes I'm here"

- find_node

- "8 closest nodes" (closest to what?)

- get_peers

- announce_peer

- who do you announce to?

- how often?

- is this a request? or a network maintenance action? or...?

[Peter]

- This slide needs time-dependence. X happens, then Y happens, then Z. Help me understand what's going on in this diagram.

DHT Key Concepts

- I'm finding this explanation less clear

The plan

- OK

Now we wait

etc.

"Who is downloading it" -- what info do you get that you're calling "who"?

Being bad with the data

- Your only example is law enforcement hand-over

- What about assembling profiles of individual IP addresses and the things they're downloading? (e.g. this is a movie stealer, this is a legal software person, this is somebody who specializes in porn, etc.)

OVERALL

- This is quite clear

- Slides are hitting the clarity & simplicity goals

- Too long! Hit the key ideas, discard unnecessary stuff, don't repeat

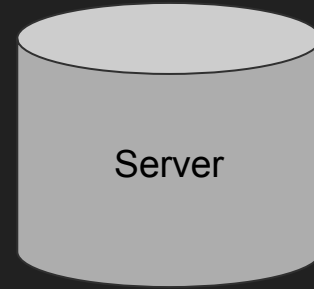
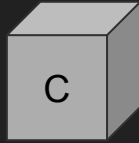
- Tighter early part, longer "what we did"

Outline

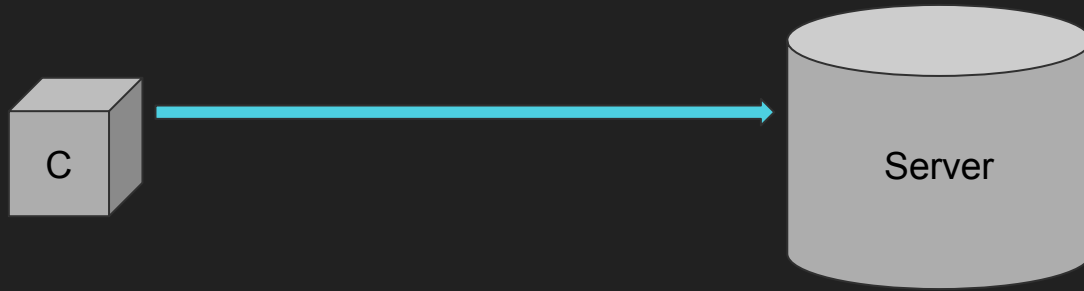
- Traditional download model
- Introduction to Torrenting
- What is the DHT protocol
- Benefits and side-effects
- Our plan
- Results and Analysis

Traditional Downloads

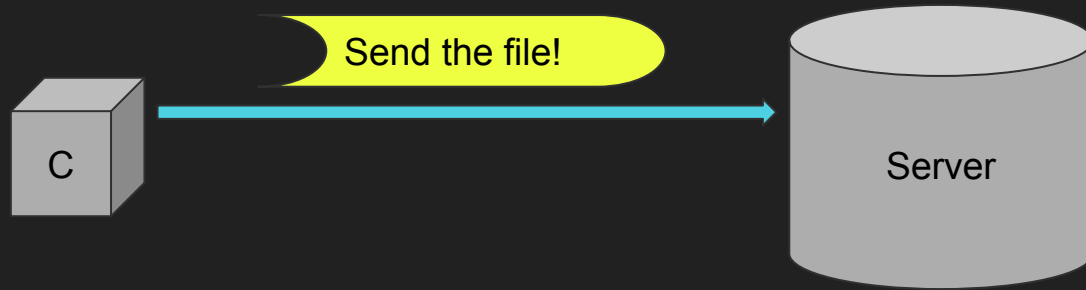
Traditional Download Model



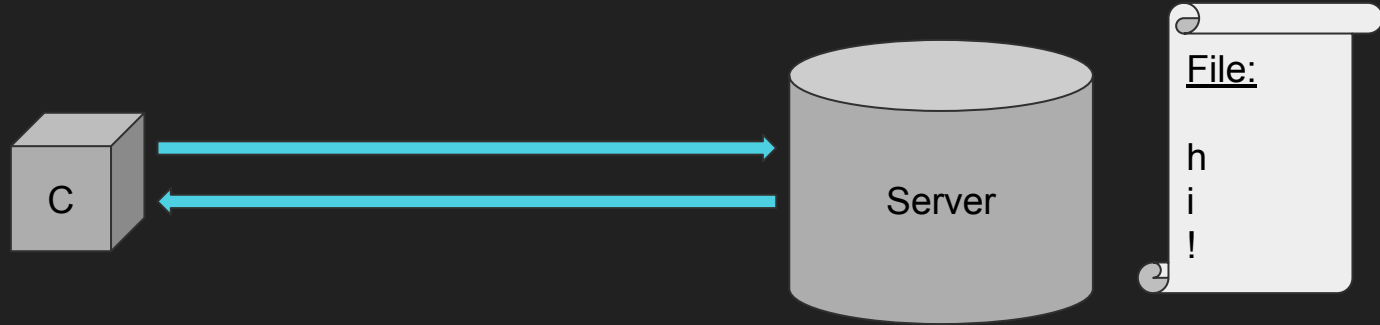
Traditional Download Model



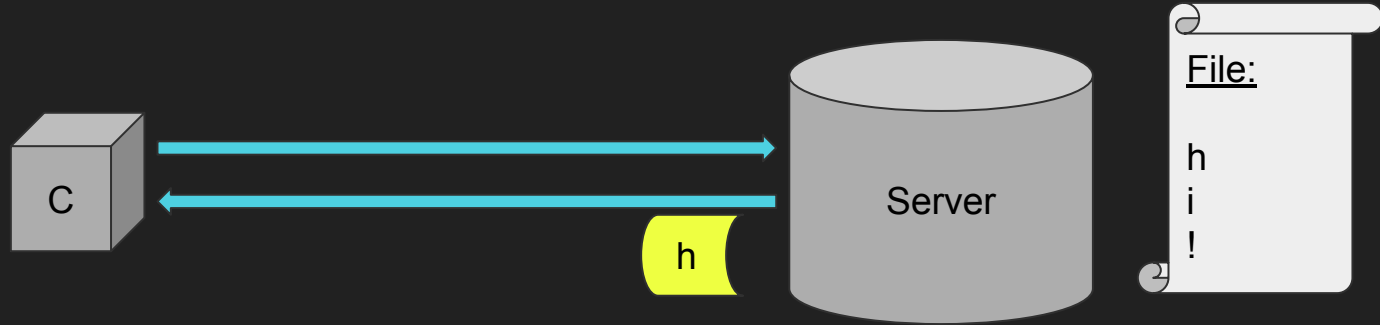
Traditional Download Model



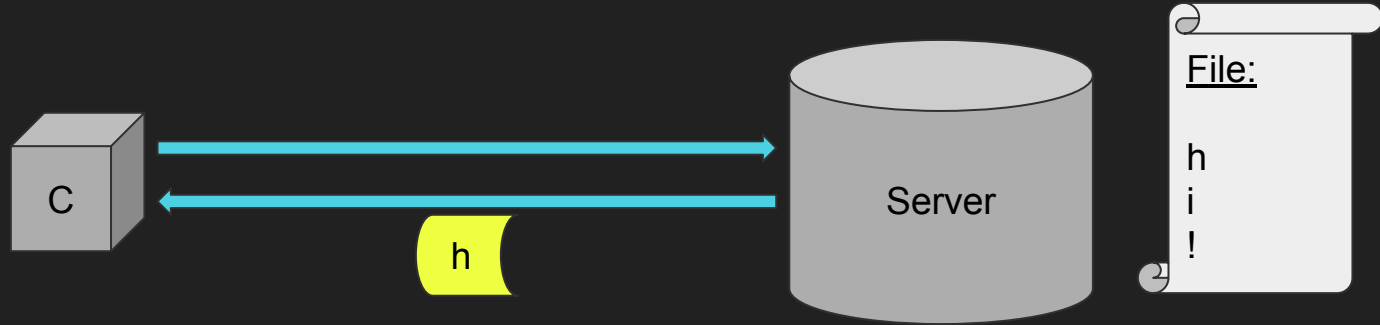
Traditional Download Model



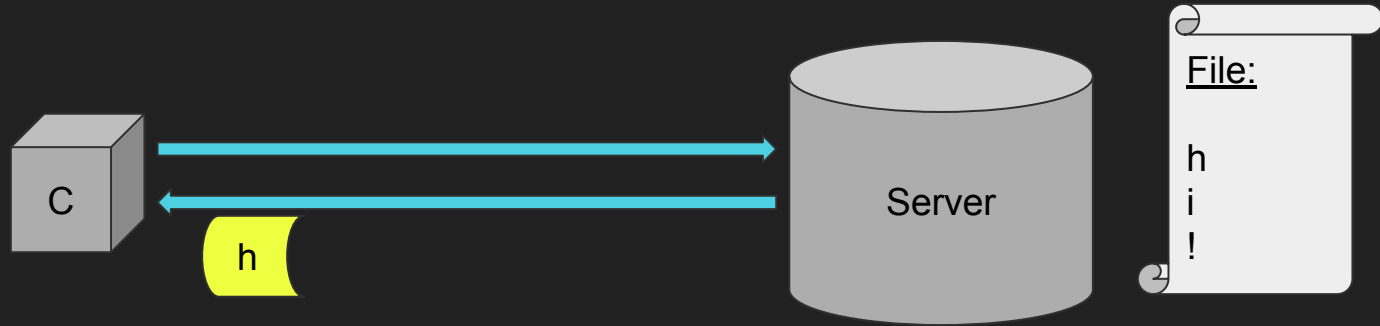
Traditional Download Model



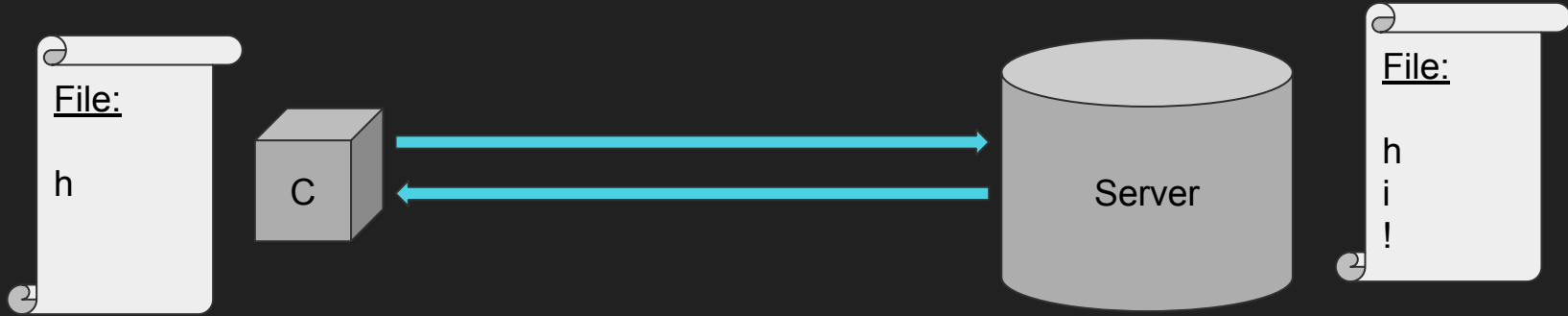
Traditional Download Model



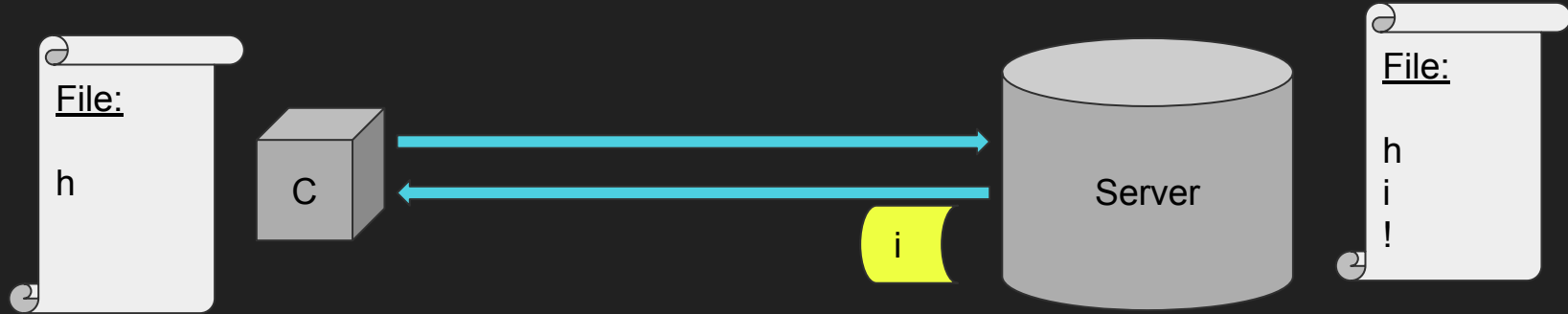
Traditional Download Model



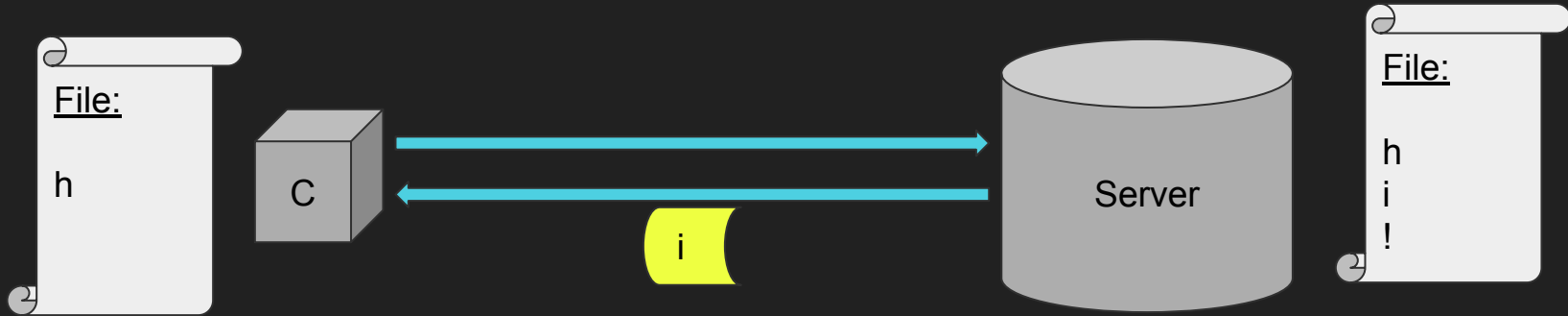
Traditional Download Model



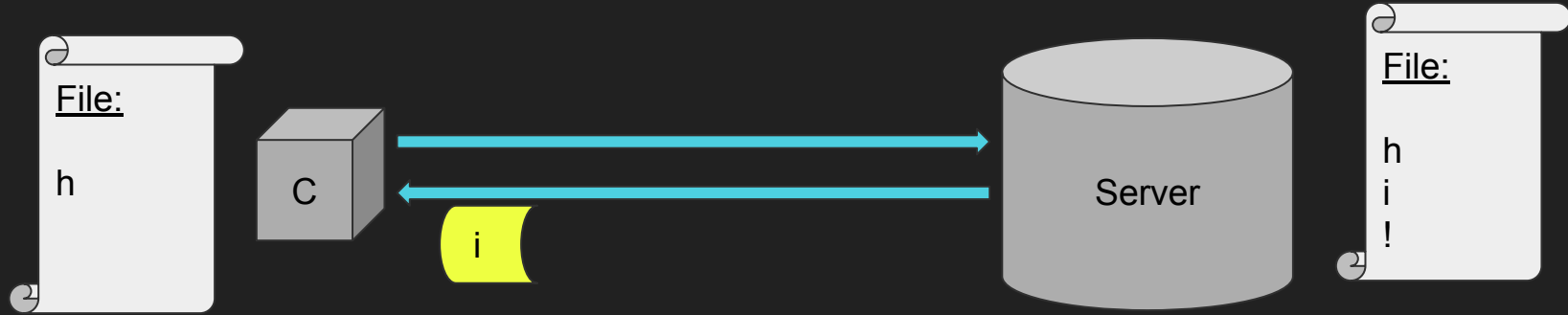
Traditional Download Model



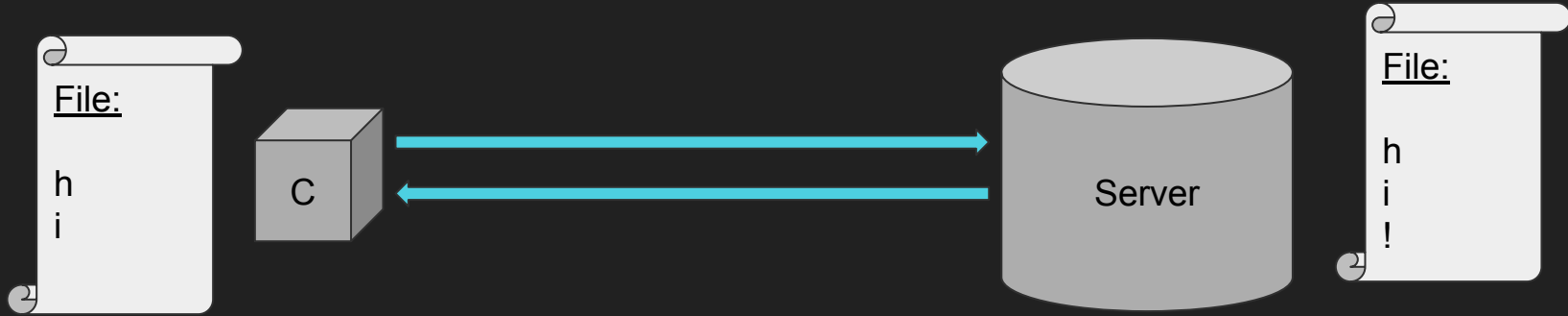
Traditional Download Model



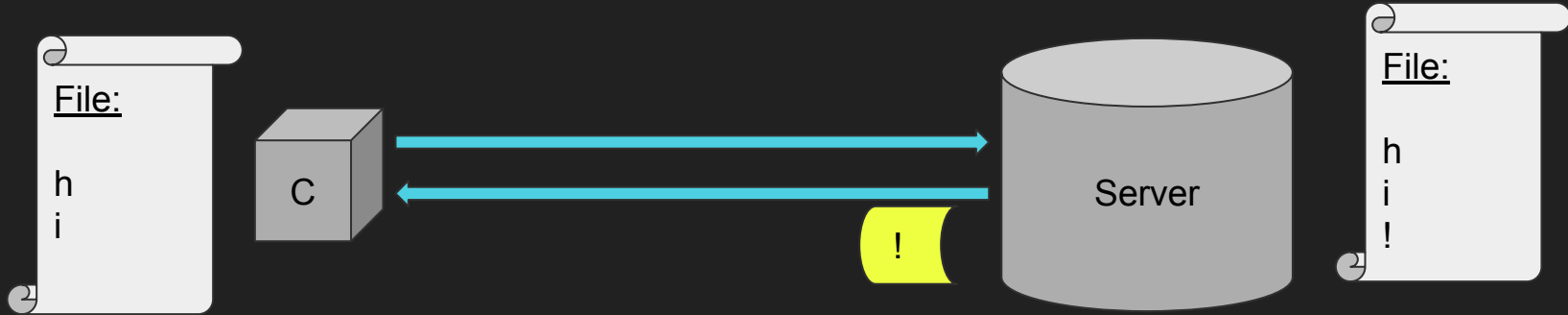
Traditional Download Model



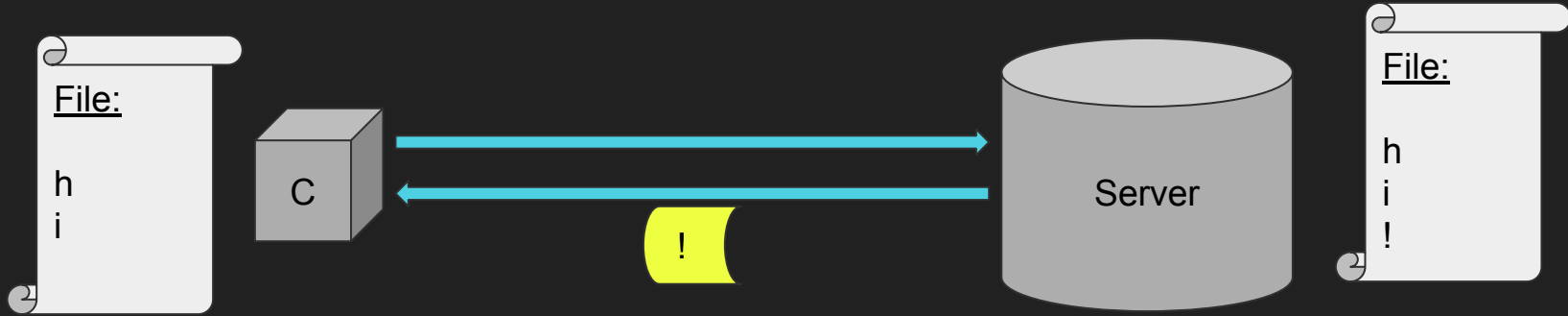
Traditional Download Model



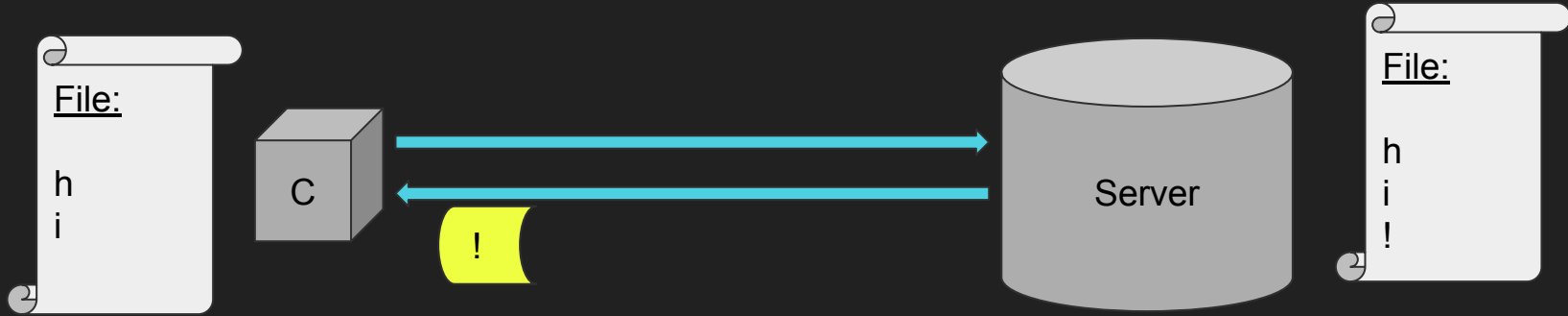
Traditional Download Model



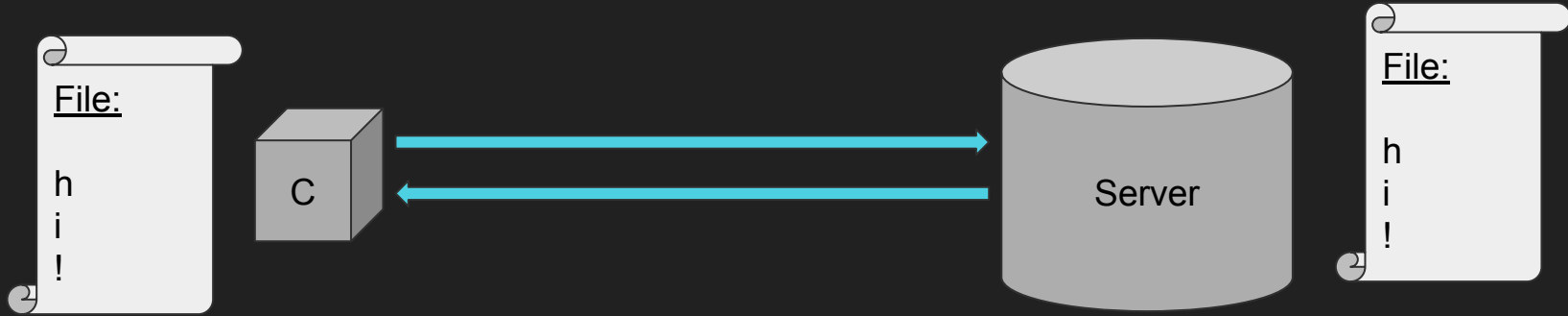
Traditional Download Model



Traditional Download Model



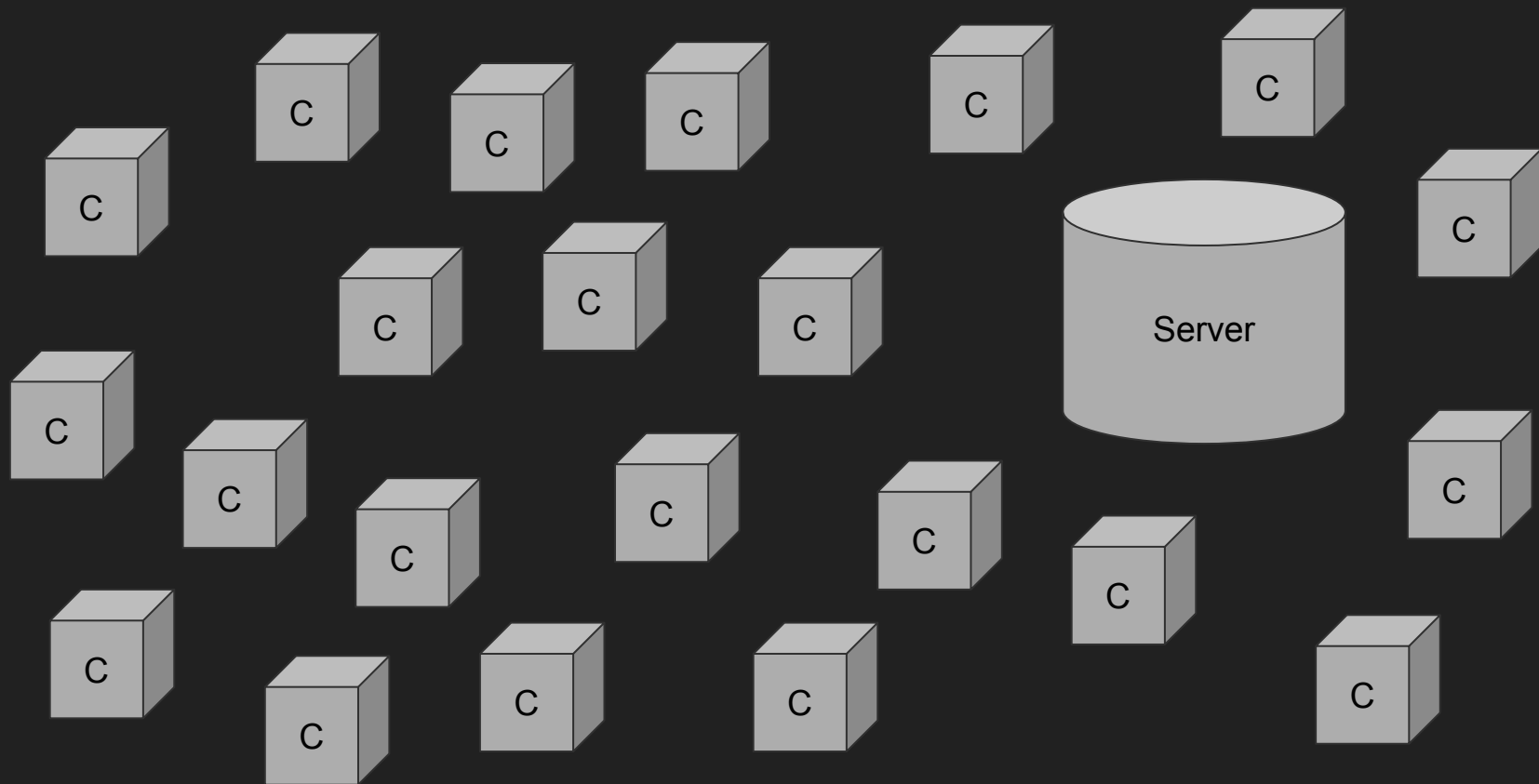
Traditional Download Model



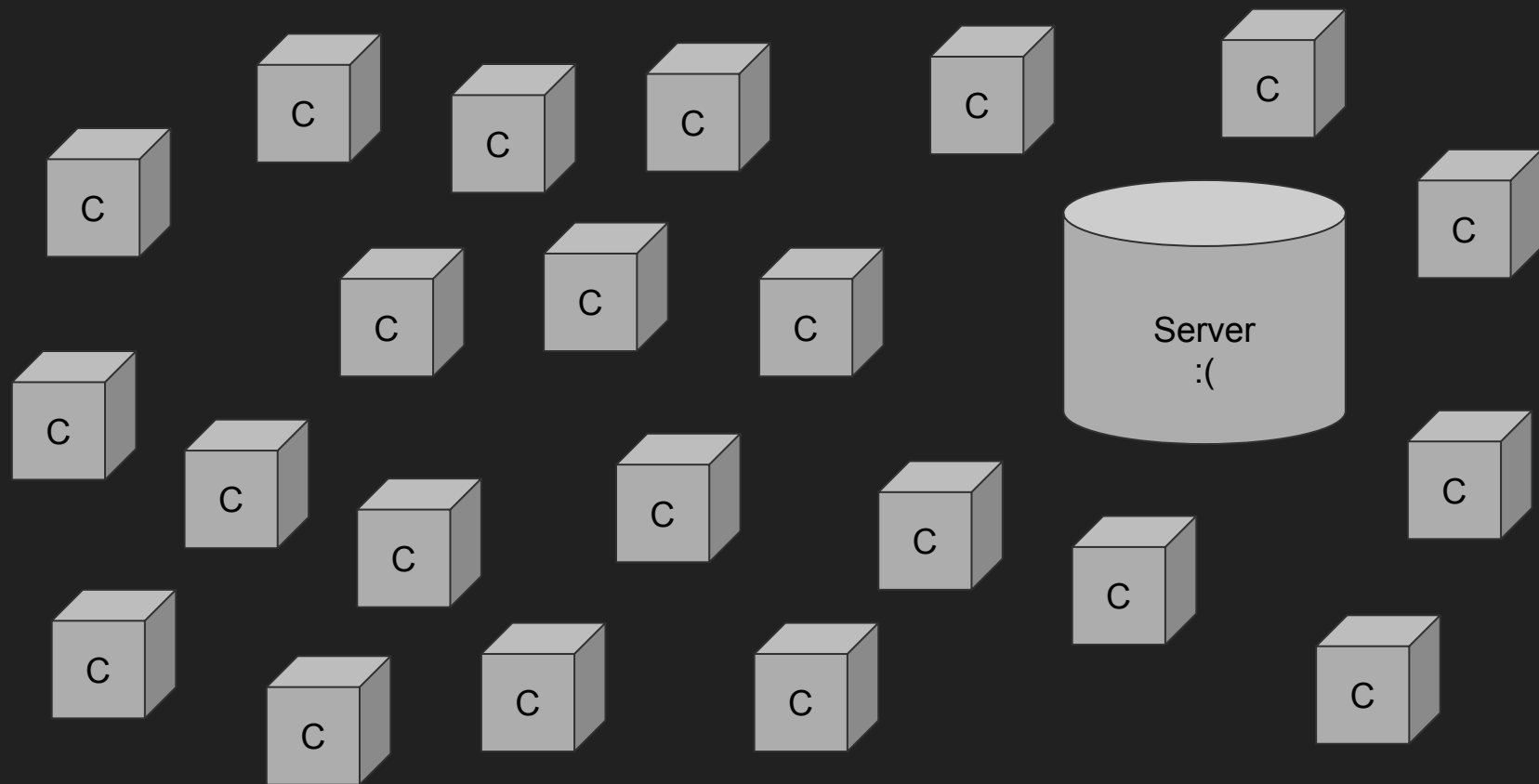
So far so good...

But what if we have 1000 clients?

Traditional Download Model



Traditional Download Model... becomes rather terrifying



Immediate problems

Traditional download model
begins to collapse

- Slow!

Immediate problems

Traditional download model
begins to collapse

- Slow!
- Hard on the server

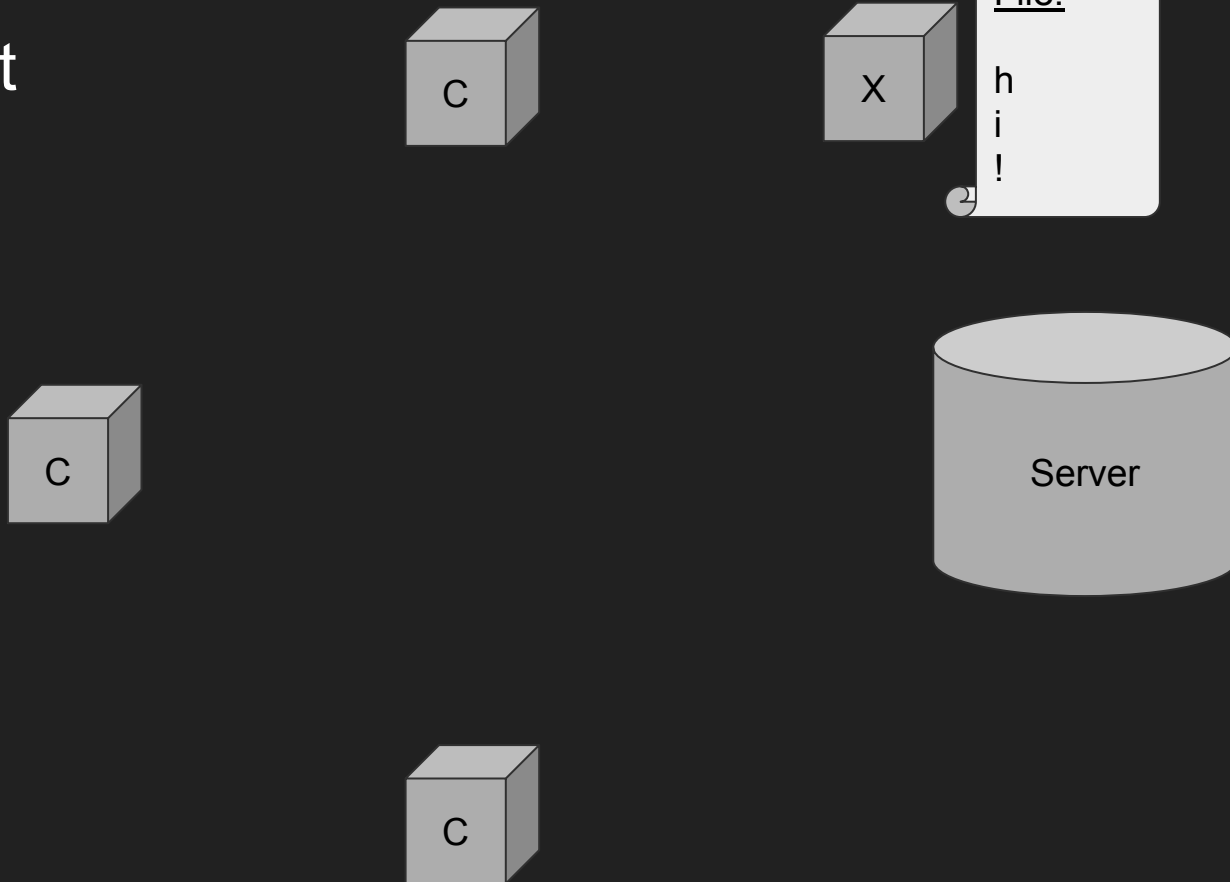
Immediate problems

Traditional download model
begins to collapse

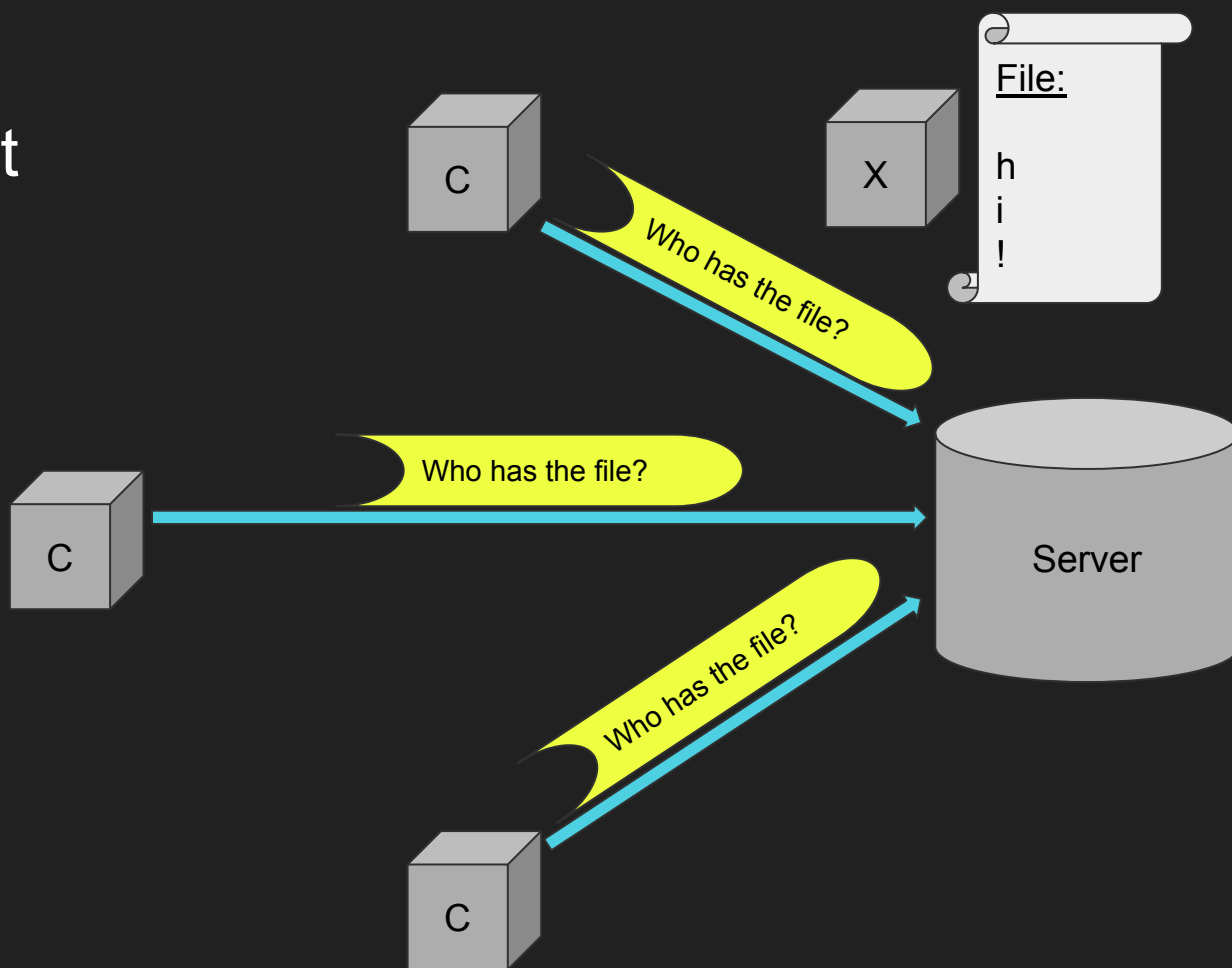
- Slow!
- Hard on the server
- Single point of failure

Solution: BitTorrent

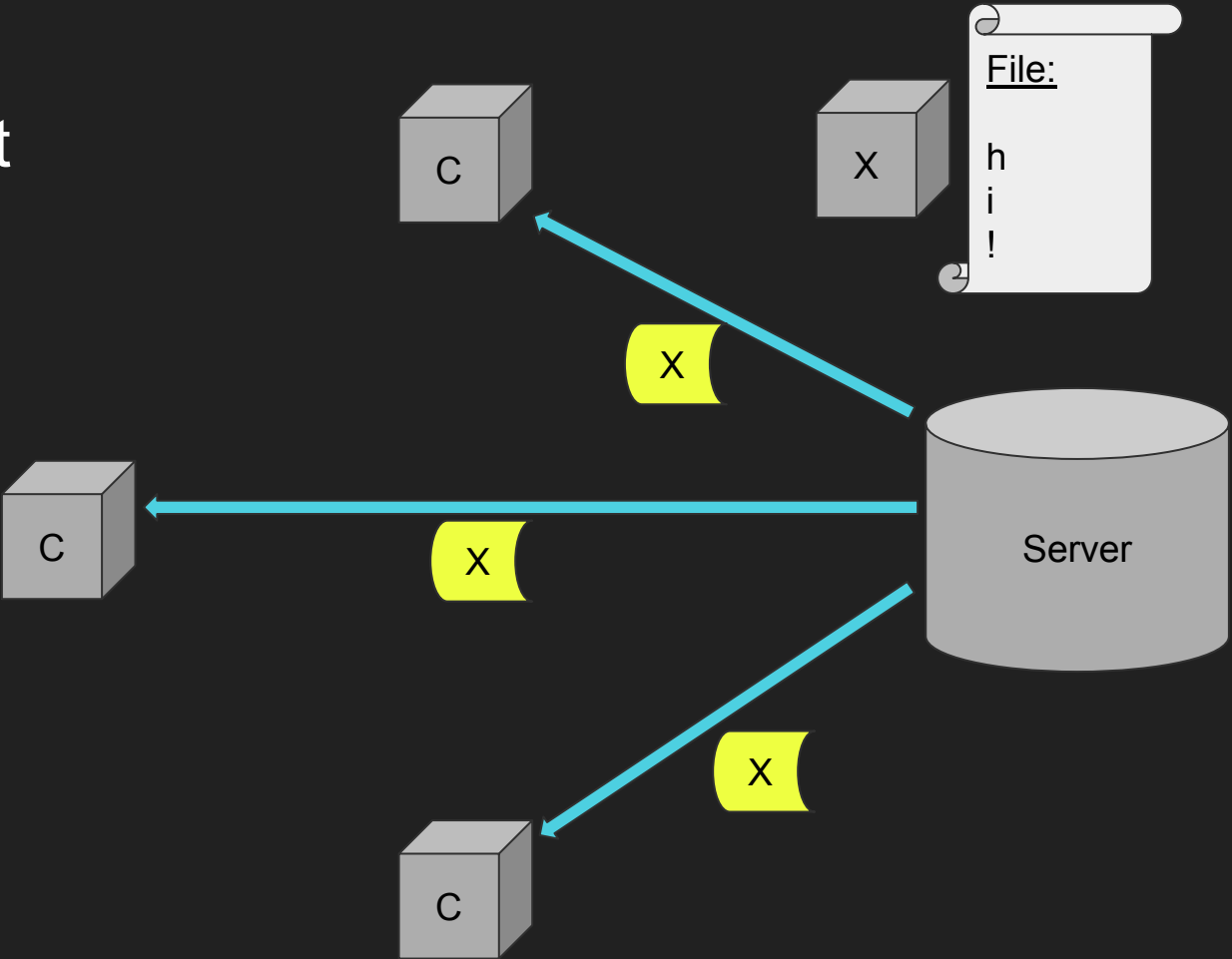
BitTorrent



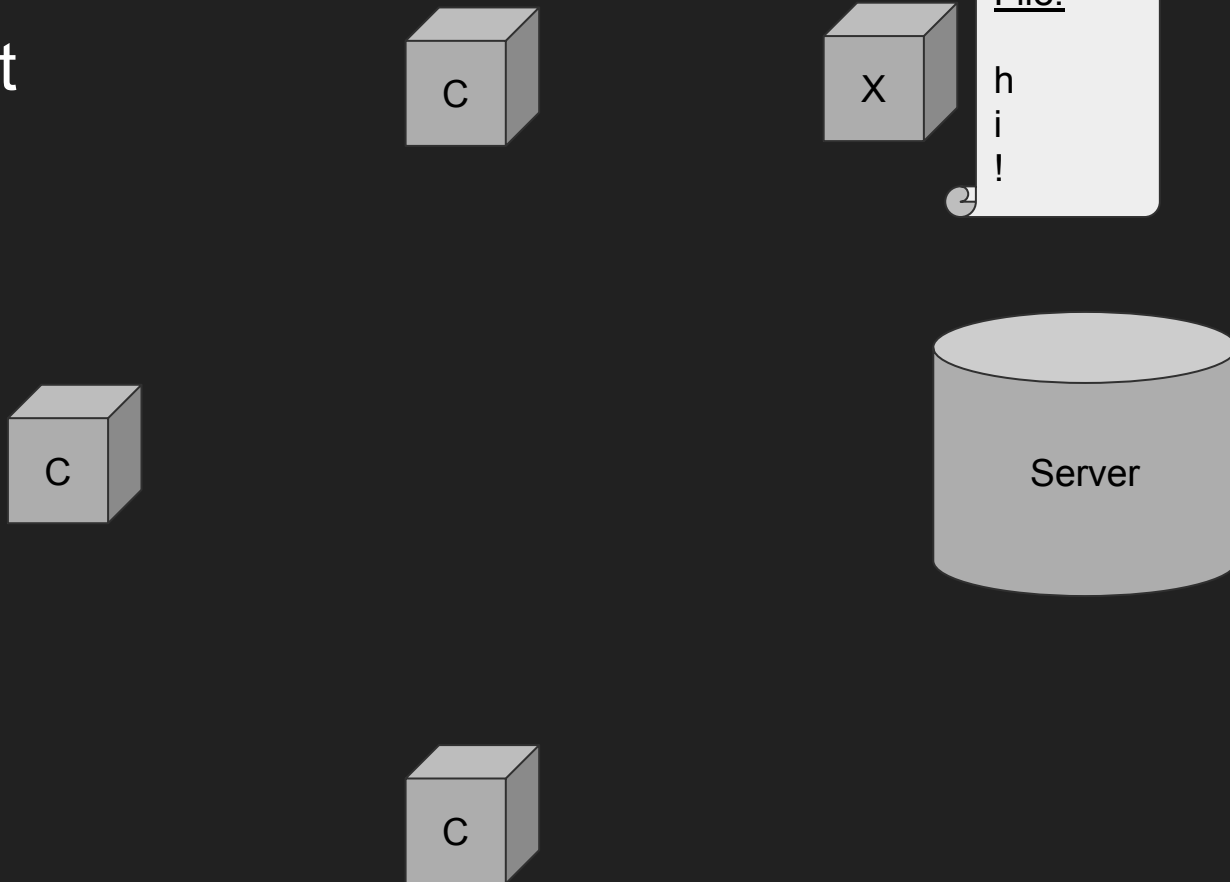
BitTorrent



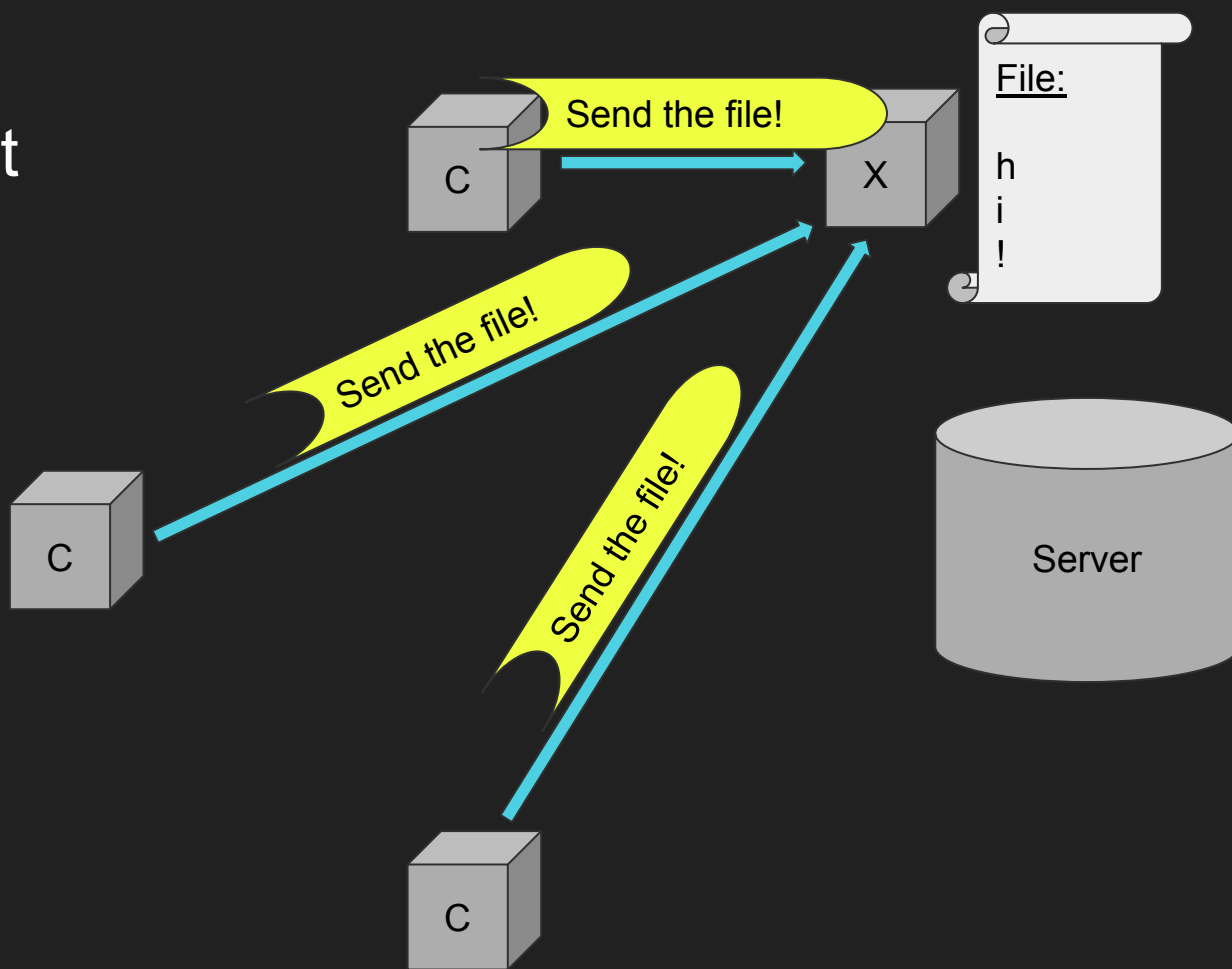
BitTorrent



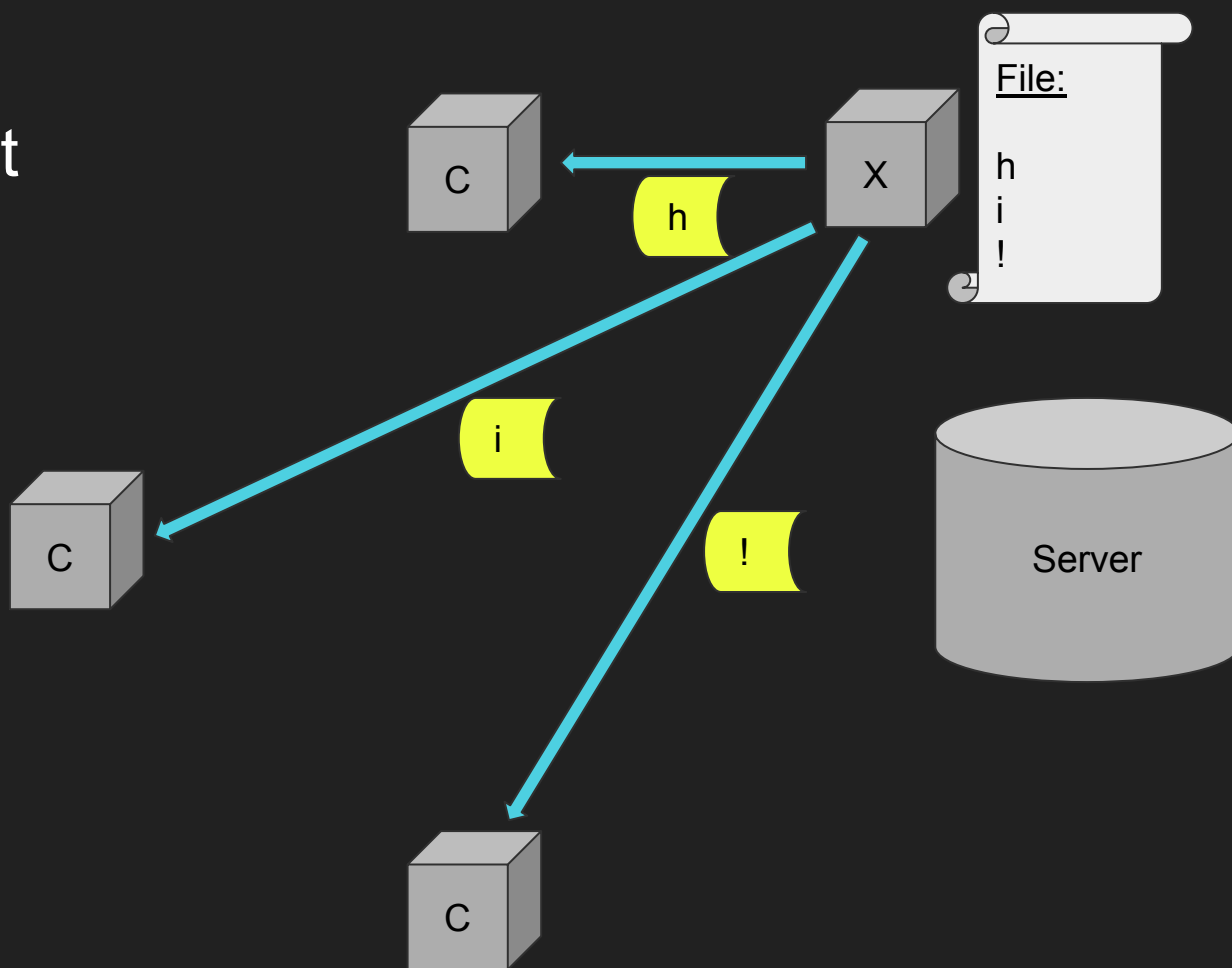
BitTorrent



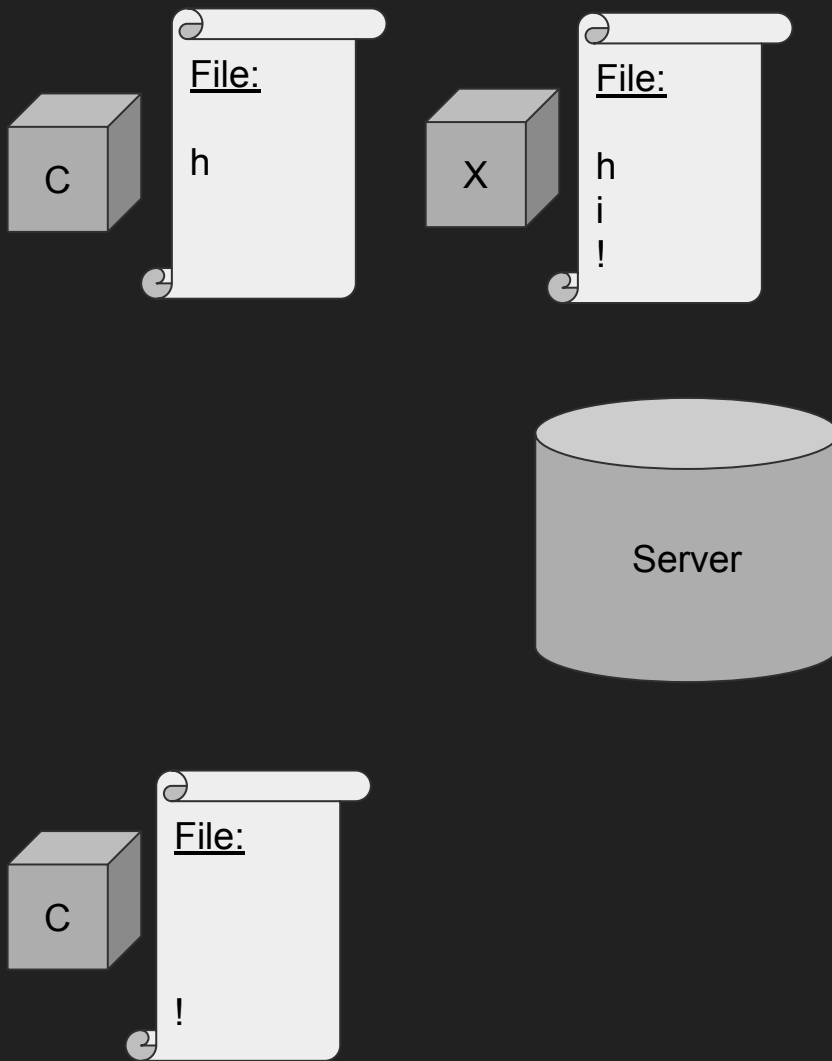
BitTorrent



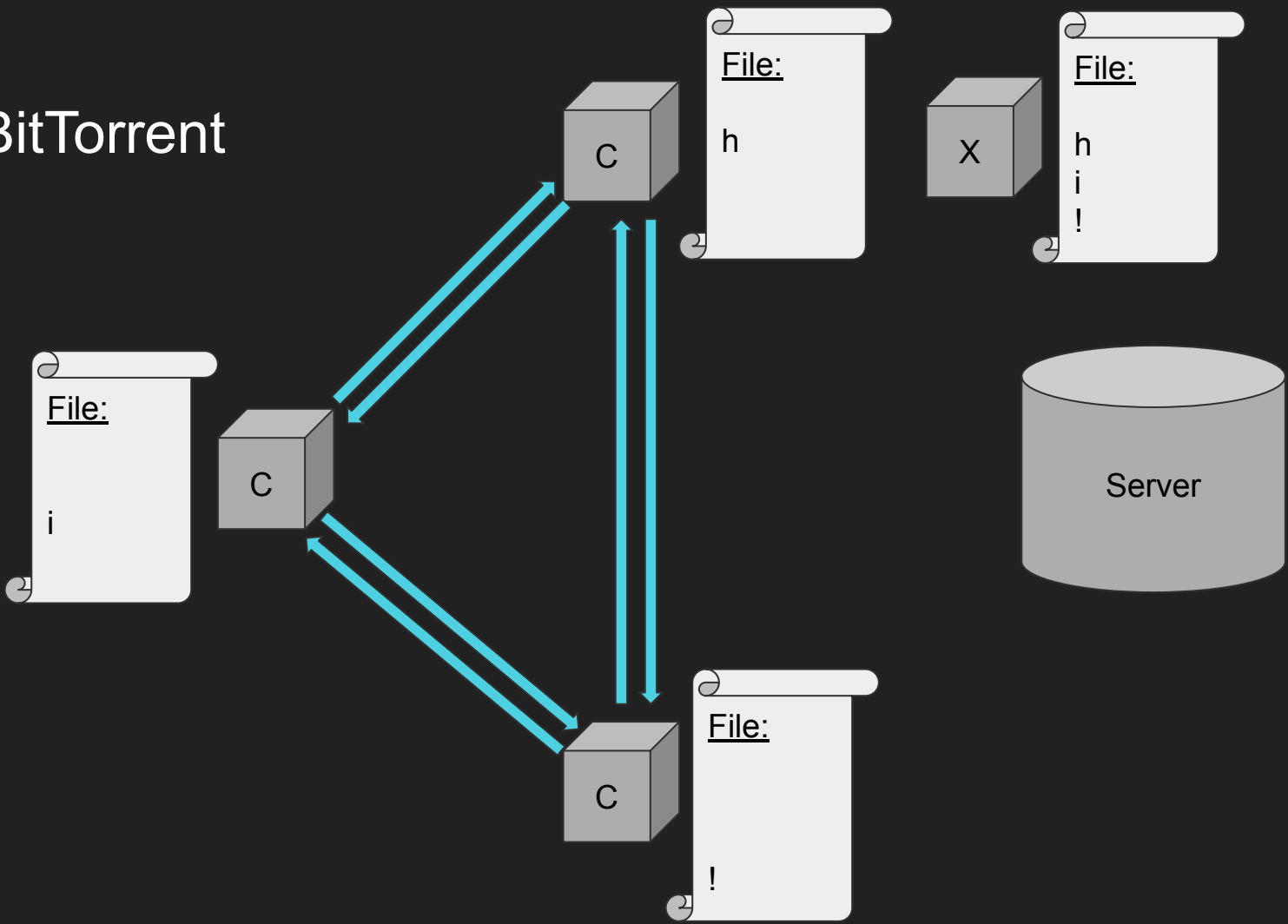
BitTorrent



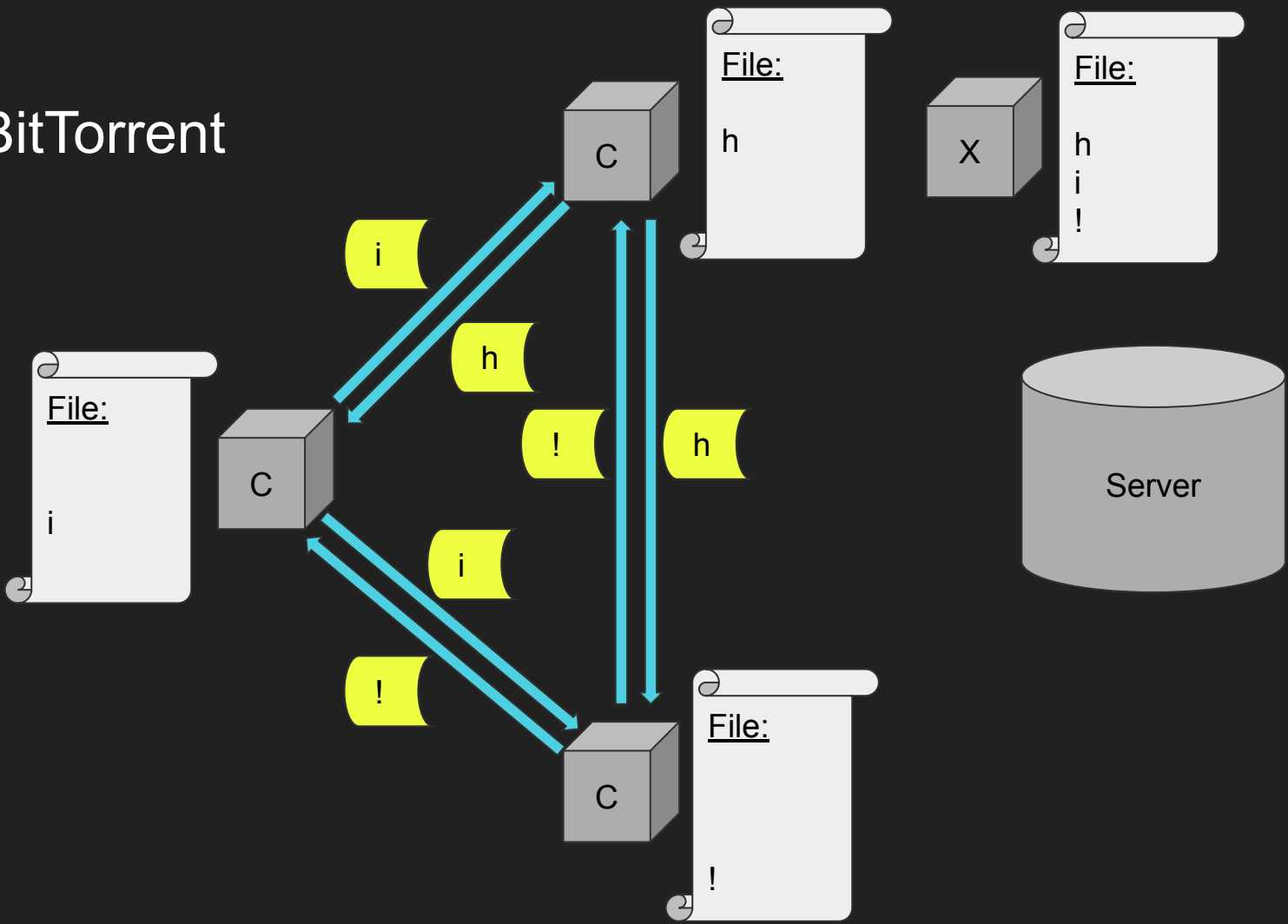
BitTorrent



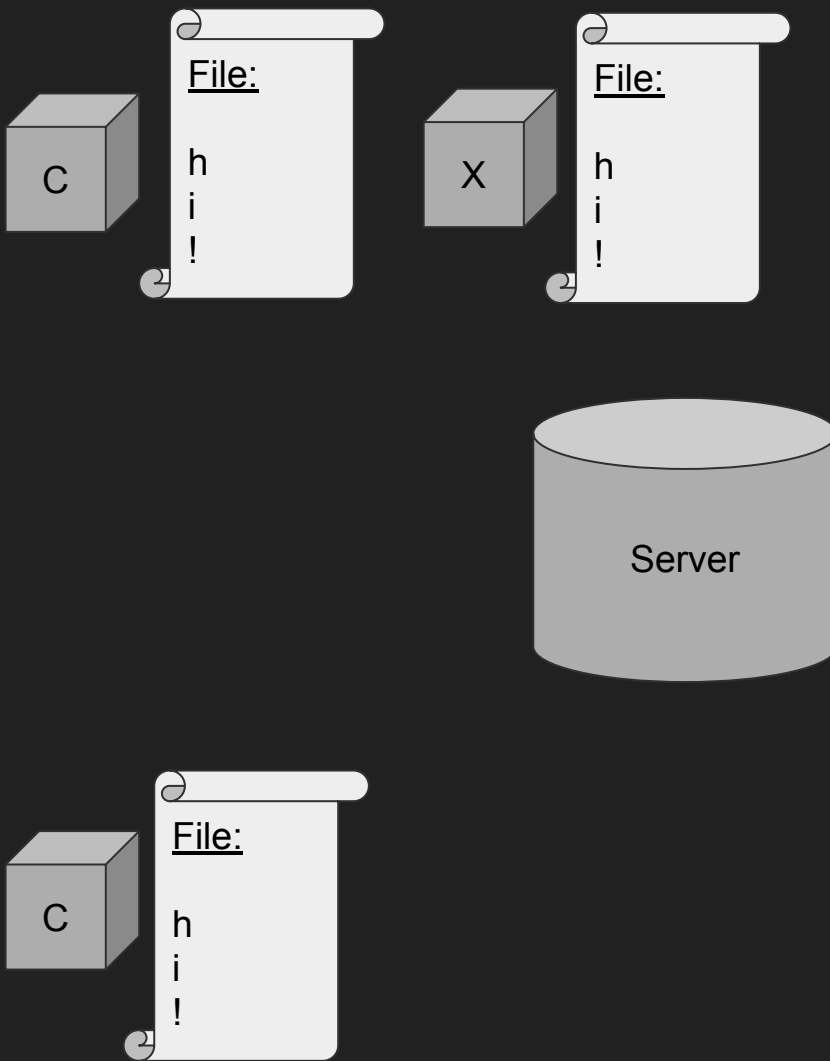
BitTorrent



BitTorrent



BitTorrent



BitTorrent Basics

- Clients (peers):
 - Download pieces of files from each other
- Central server(s):
 - Facilitate finding peers for given file
 - Store metadata
 - Torrent “search engines”

BitTorrent Benefits

- Much faster downloads
- Less strain on individual server
- Resilient to failure of peers

BitTorrent Side Effects

- Great for internal data sharing (Amazon, etc.)

BitTorrent Side Effects

- Great for internal data sharing (Amazon, etc.)
- ... also great for sharing illegal/copyrighted media

BitTorrent Side Effects

- Great for internal data sharing (Amazon, etc.)
- ... also great for sharing illegal/copyrighted media
- Torrent websites/servers are easy targets

Central Server: Point of Failure

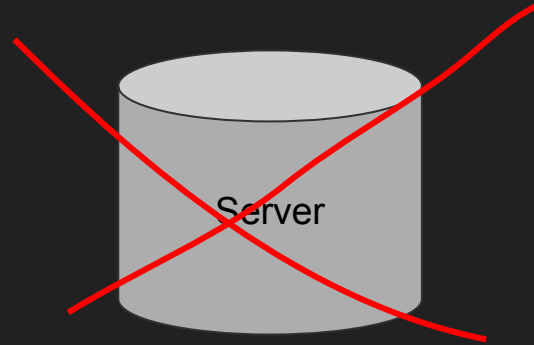
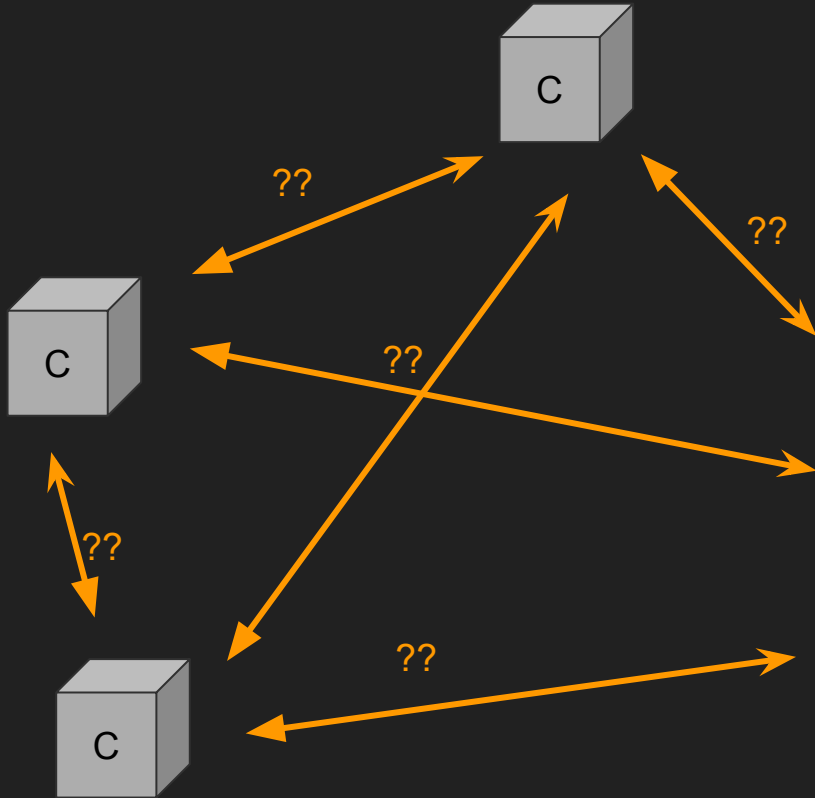
THIS WEBSITE HAS BEEN SEIZED

This domain has been seized by the Federal Bureau of Investigation pursuant to a seizure warrant issued by the United States District Court for the Central District of California under the authority of 18 U.S.C. §1030(i)(1)(A) as part of coordinated law enforcement action taken against illegal DDoS-for-hire services.

This action has been taken in coordination with the United States Attorney's Office of the District of Alaska, the Department of Justice Computer Crime and Intellectual Property Section, and



Central Server: Point of Failure



THIS WEBSITE HAS BEEN SEIZED

This domain has been seized by the Federal Bureau of Investigation pursuant to a seizure warrant issued by the United States District Court for the Central District of California under the authority of 18 U.S.C. §1030(i)(1)(A) as part of coordinated law enforcement action taken against illegal DDoS-for-hire services.

This action has been taken in coordination with the United States Attorney's Office of the District of Alaska, the Department of Justice Computer Crime and Intellectual Property Section, and

NCA
National Crime Agency

POLITIE

The image shows the logos of three law enforcement agencies: the National Crime Agency (NCA), the Federal Bureau of Investigation (FBI), and the Department of Justice Computer Crime and Intellectual Property Section (CCIPS). The FBI logo is the largest and most prominent, featuring the eagle and shield. The NCA logo is to its left, and the CCIPS logo is to its right.

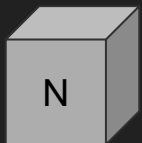
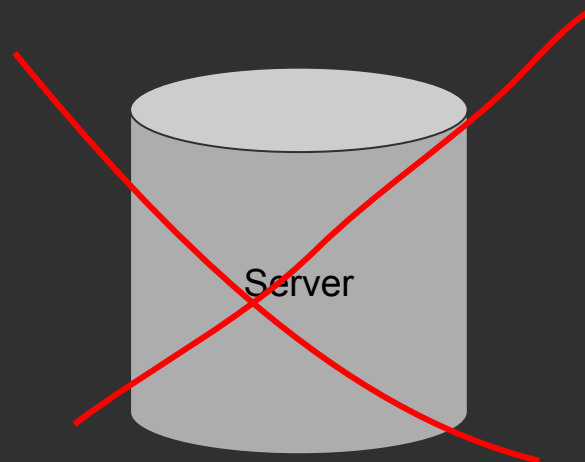
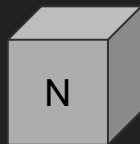
Solution: Distributed Hash Table (DHT)

```
>>> dict = {"a": 1, "b": 2, "c": 3, "d": 4}
>>> print(dict)
{'a': 1, 'b': 2, 'c': 3, 'd': 4}
```

Distributed Hash Table

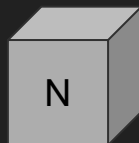
- Each node/server has ID
- Many servers share information about peers
 - Mappings from infohash to IP:port
 - Mappings from peer IDs to IP:port
- All you need is an infohash to download

```
node1 = {  
  "Avengers Endgame": "121.235.14.180:6881",  
  "Hannah Montana.mp3": "33.117.74.62:6881",  
  "Adobe Photoshop": "97.253.196.104:6881"  
}
```

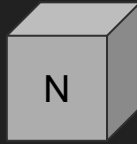


```
node2 = {  
  "textbook.pdf": "121.235.14.180:6881",  
  "XXX.mp4": "8.44.194.128:6881",  
  "hiJeff.txt": "54.94.141.91:6881"  
}
```

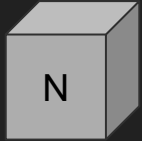
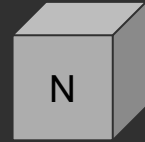
```
node3 = {  
  "Windows 7 cracked (free)": "56.70.234.227: 6881",  
  "music.mp3": "146.231.124.90: 6881",  
  "Arch Linux.iso": "117.31.176.98: 6881"  
}
```



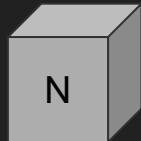
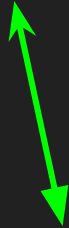
```
node1 = {  
  "Avengers Endgame": "121.235.14.180:6881",  
  "Hannah Montana.mp3": "33.117.74.62:6881",  
  "Adobe Photoshop": "97.253.196.104:6881"  
}
```



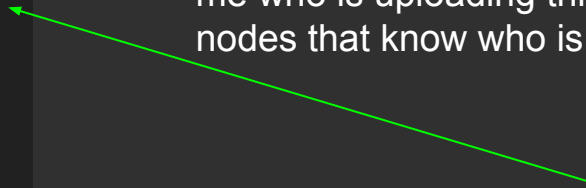
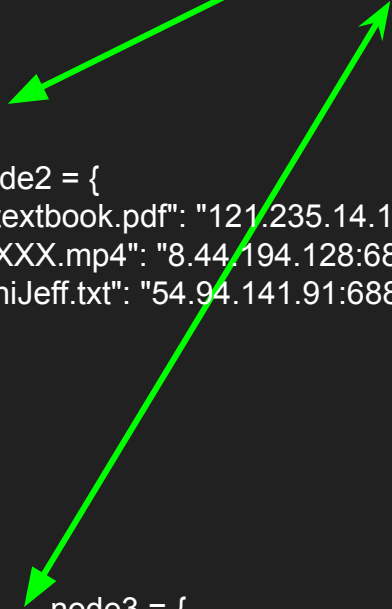
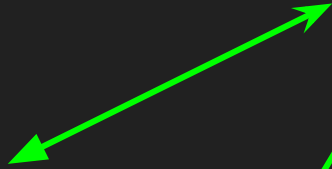
"I want to download Arch Linux. My IP address is 152.197.106.171, please tell me who is uploading this file or find other nodes that know who is uploading this file"



```
node2 = {  
  "textbook.pdf": "121.235.14.180:6881",  
  "XXX.mp4": "8.44.194.128:6881",  
  "hiJeff.txt": "54.94.141.91:6881"  
}
```



```
node3 = {  
  "Windows 7 cracked (free)": "56.70.234.227: 6881",  
  "music.mp3": "146.231.124.90: 6881",  
  "Arch Linux.iso": "117.31.176.98: 6881"  
}
```



DHT Protocol

4 query types

DHT Protocol

4 query types
ping

- ping:
 - Send your ID
 - Get your ID back

DHT Protocol

4 query types

ping

find_node

- find_node:
 - Send:
 - Your ID
 - Target node ID
 - Receive:
 - IP:port of target, if known
 - Otherwise, IP:port of 8 closest nodes

Aside: “closeness”

Optimizing search across the
network

- Infohashes and client IDs are *both* 160 bits
- Distance between an infohash and an ID is equal to the integer value of their XOR
- Store infohashes on nodes with IDs “closest” to the infohash
 - Search time is greatly reduced

DHT Protocol

4 query types

ping

find_node

get_peers

- get_peers:
 - Send:
 - Your ID
 - Infohash of file
 - Receive:
 - Token
 - IP:port of known peers
 - OR IP:port of 16 “closest” nodes

DHT Protocol

4 query types

ping

find_node

get_peers

announce_peer

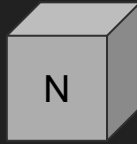
- announce_peer:
 - Send:
 - Your ID
 - Your port
 - Infohash
 - Token
 - Receive
 - Queried node ID

DHT Key Concepts

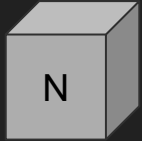
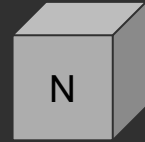
A brief summary

- No central server
- Peers are nodes
- “Closeness”
 - Put data on nodes with closest IDs
- How downloading works:
 - `get_peers`
 - `announce_peer`

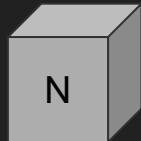
```
node1 = {  
  "Avengers Endgame": "121.235.14.180:6881",  
  "Hannah Montana.mp3": "33.117.74.62:6881",  
  "Adobe Photoshop": "97.253.196.104:6881"  
}
```



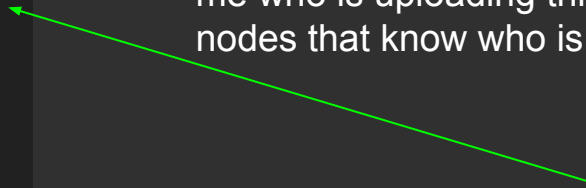
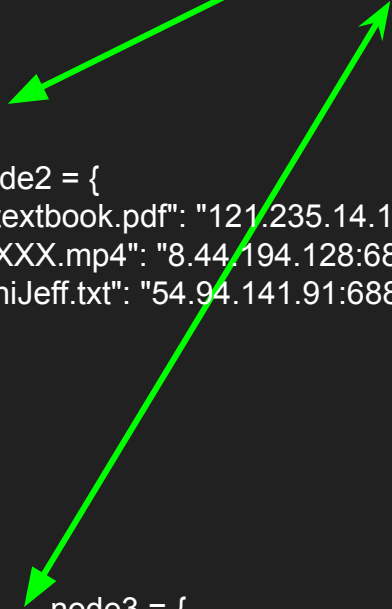
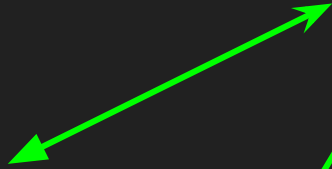
"I want to download Arch Linux. My IP address is 152.197.106.171, please tell me who is uploading this file or find other nodes that know who is uploading this file"



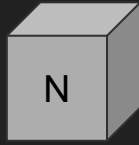
```
node2 = {  
  "textbook.pdf": "121.235.14.180:6881",  
  "XXX.mp4": "8.44.194.128:6881",  
  "hiJeff.txt": "54.94.141.91:6881"  
}
```



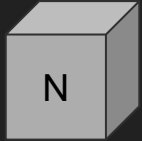
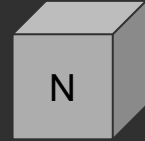
```
node3 = {  
  "Windows 7 cracked (free)": "56.70.234.227: 6881",  
  "music.mp3": "146.231.124.90: 6881",  
  "Arch Linux.iso": "117.31.176.98: 6881"  
}
```



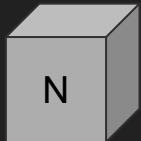
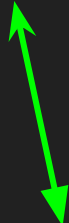

```
node1 = {  
  "Avengers Endgame": "121.235.14.180:6881",  
  "Hannah Montana.mp3": "33.117.74.62:6881",  
  "Adobe Photoshop": "97.253.196.104:6881"  
}
```



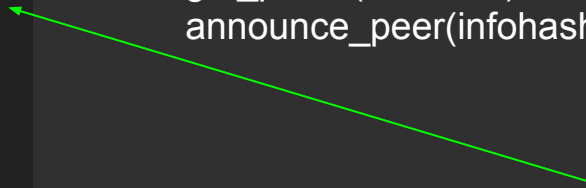
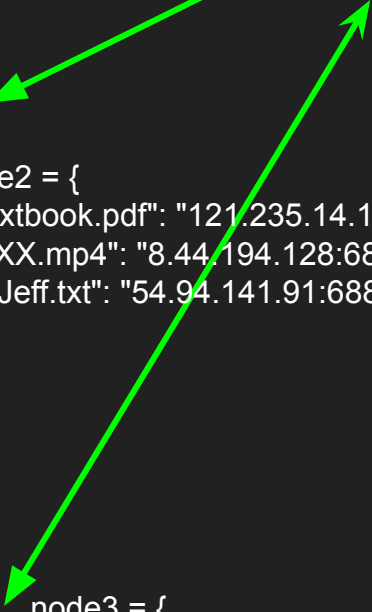
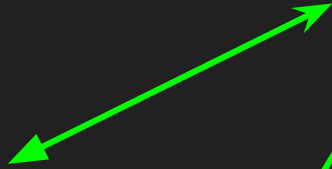
get_peers(infohash)
announce_peer(infohash)



```
node2 = {  
  "textbook.pdf": "121.235.14.180:6881",  
  "XXX.mp4": "8.44.194.128:6881",  
  "hiJeff.txt": "54.94.141.91:6881"  
}
```



```
node3 = {  
  "Windows 7 cracked (free)": "56.70.234.227: 6881",  
  "music.mp3": "146.231.124.90: 6881",  
  "Arch Linux.iso": "117.31.176.98: 6881"  
}
```



DHT

Side Effects

- Peers hold lots of necessary information:
 - Infohashes → IP:port
 - Node IDs → IP:port

DHT

Side Effects

- Peers hold lots of *sensitive* information:
 - Infohashes → IP:port
 - Node IDs → IP:port

DHT Side Effects

- In order to torrent, you need to broadcast your IP address and what files you have/want
- All of this is sent to DHT nodes publicly and unencrypted....

The plan

The plan

Phase 1:

- Host lots of DHT clients on a machine we control
- Nodes in the DHT network contact us looking for torrent information
- Record each request using network analysis tools
 - Who asked (IP)?
 - What are they looking for (infohash)?

Now we wait...

Real requests and IP addresses sent to our DHT clients:

```
BitTorrent DHT Protocol
ip: 137.22.255.13
Key: ip
IP: 137.22.255.13
Port: 39222
Response values: Dictionary...
Key: r
Value: Dictionary...
id: c8e1bdf94c614c613e3f145d9ea973ab111ef94f
Key: id
Value: c8e1bdf94c614c613e3f145d9ea973ab111ef94f
nodes: 8
Key: nodes
Value: 8 nodes
Node 1 (id: c82eefd6ae529049f1f1bbe9ebb3a6db3c870ce1, IPv4/Port: 109.86.139.1:20912)
ID: c82eefd6ae529049f1f1bbe9ebb3a6db3c870ce1
IP: 109.86.139.1
Port: 20912
Node 2 (id: c83bb4bb9760c6f47c3c070cf8db18b7f2c389, IPv4/Port: 136.243.144.151:6882)
ID: c83bb4bb9760c6f47c3c070cf8db18b7f2c389
IP: 136.243.144.151
Port: 6882
Node 3 (id: c80425de7be24a85981516debdc8bb7df5a036af, IPv4/Port: 212.107.142.55:6881)
ID: c80425de7be24a85981516debdc8bb7df5a036af
IP: 212.107.142.55
Port: 6881
Node 4 (id: c84644429a8c877b59ea34d143a60ae0d802709d, IPv4/Port: 185.203.56.20:57498)
ID: c84644429a8c877b59ea34d143a60ae0d802709d
IP: 185.203.56.20
Port: 57498
Node 5 (id: c813f140c66d22f866be327274a381303b1a4906, IPv4/Port: 90.142.47.133:6881)
ID: c813f140c66d22f866be327274a381303b1a4906
IP: 90.142.47.133
Port: 6881
Node 6 (id: c813d528f4a7e78f43afd3502f47f84788a19170, IPv4/Port: 90.243.37.198:7193)
ID: c813d528f4a7e78f43afd3502f47f84788a19170
IP: 90.243.37.198
Port: 7193
Node 7 (id: c86216d6ae529049f1f1bbe9ebb3a6db3c870ce1, IPv4/Port: 73.155.101.235:16537)
ID: c86216d6ae529049f1f1bbe9ebb3a6db3c870ce1
IP: 73.155.101.235
Port: 16537
Node 8 (id: c8441ad6ae529049f1f1bbe9ebb3a6db3c870ce1, IPv4/Port: 31.131.194.107:2921)
ID: c8441ad6ae529049f1f1bbe9ebb3a6db3c870ce1
IP: 31.131.194.107
Port: 2921
```

```
BitTorrent DHT Protocol
Response values: Dictionary...
Key: r
Value: Dictionary...
id: 8b22356b48e66dc4f0cac18ec38217fe641eeefe6
Key: id
Value: 8b22356b48e66dc4f0cac18ec38217fe641eeefe6
nodes: 8
Key: nodes
Value: 8 nodes
Node 1 (id: be23559c605a641587804945cf278f492255dc41, IPv4/Port: 5.189.188.23:46914)
ID: be23559c605a641587804945cf278f492255dc41
IP: 5.189.188.23
Port: 46914
Node 2 (id: b817936526f74cf9a9bb1ef07b73146edaf88387, IPv4/Port: 135.181.210.22:31550)
ID: b817936526f74cf9a9bb1ef07b73146edaf88387
IP: 135.181.210.22
Port: 31550
Node 3 (id: ab4fb5a2aed395d6bd58b1f4f01164a3515b809, IPv4/Port: 173.212.202.248:51421)
ID: ab4fb5a2aed395d6bd58b1f4f01164a3515b809
IP: 173.212.202.248
Port: 51421
Node 4 (id: aedae0ad4bf101c5743d801f1714047720d36ab2, IPv4/Port: 213.136.79.205:49891)
ID: aedae0ad4bf101c5743d801f1714047720d36ab2
IP: 213.136.79.205
Port: 49891
Node 5 (id: a708c123d814e7d41c440518d5e4e1a1a9e00dcf, IPv4/Port: 46.232.210.48:13259)
ID: a708c123d814e7d41c440518d5e4e1a1a9e00dcf
IP: 46.232.210.48
Port: 13259
Node 6 (id: b6819001fbddb66df959acd0f7b50c06f03bf694, IPv4/Port: 213.136.79.238:33498)
ID: b6819001fbddb66df959acd0f7b50c06f03bf694
IP: 213.136.79.238
Port: 33498
Node 7 (id: a35452325d4ab44b4397c51d1d49592442fddee9c, IPv4/Port: 46.242.10.224:7172)
ID: a35452325d4ab44b4397c51d1d49592442fddee9c
IP: 46.242.10.224
Port: 7172
Node 8 (id: b45a9637af1461ceb28db2b8221a4df1f83a9b33, IPv4/Port: 80.201.196.144:6881)
ID: b45a9637af1461ceb28db2b8221a4df1f83a9b33
IP: 80.201.196.144
Port: 6881
```


Save the infohashes we are asked for...

```
** (tshark:58411) 03:38:19.735470 [Main MESSAGE] -- Capture started.  
** (tshark:58411) 03:38:19.739640 [Main MESSAGE] -- File: "/var/folders/  
131  
d3581f006b6a32e412820594c425677f2cd06273  
  
d3581f006b6a32e4128205943f60801984fb11f2  
  
d3581f006b6a32e412820594ed11cbc00b73cae2  
  
d3581f006b6a32e412820594db5b45718cc9bd02  
  
d3581f006b6a32e41282059478e9539284e0ace8  
  
8ba950c191e14363d713a6c3d0f4114c5e116346  
  
8ba950c191e14363d713a6c336fb4c5ad4161731  
  
8ba950c191e14363d713a6c34f98bbe5253446b6  
  
8ba950c191e14363d713a6c39d0e56138ebcfbcb  
  
8ba950c191e14363d713a6c3501b04d567bce478  
  
8ba950c191e14363d713a6c3e15666d057e630ec  
  
8ba950c191e14363d713a6c3b6be6ab341b08403  
  
8ba950c191e14363d713a6c3a86515e645fcfab5  
  
8ba950c191e14363d713a6c330cb7d43c0fb24b1  
190  
1f9b6a691c2fe108e990df3504b8aa7cc0abe24c  
  
8ba950c191e14363d713a6c300b985a29494aeda  
4089  
8ba950c191e14363d713a6c3018a5969e4137f68  
  
8ba950c191e14363d713a6c38c8a0f59b6632efa  
  
8ba950c191e14363d713a6c3d5ce8ef72ba8ad15  
  
8ba950c191e14363d713a6c3a95111fa41160dac  
4123  
8ba950c191e14363d713a6c3e9e90024c27fd6d1  
  
8ba950c191e14363d713a6c36faa7cbe26fb7b7f  
  
8ba950c191e14363d713a6c36063d3913ab976e7
```

The plan

Phase 1:

- Host lots of DHT clients on a machine we control
- Nodes in the DHT network contact us looking for torrent information
- Record each request using network analysis tools
 - Who asked (IP)?
 - What are they looking for (infohash)?

The plan

Phase 1:

- Host lots of DHT clients on a machine we control
- Nodes in the DHT network contact us looking for torrent information
- Record each request using network analysis tools
 - Who asked (IP)?
 - What are they looking for (infohash)?

Phase 2:

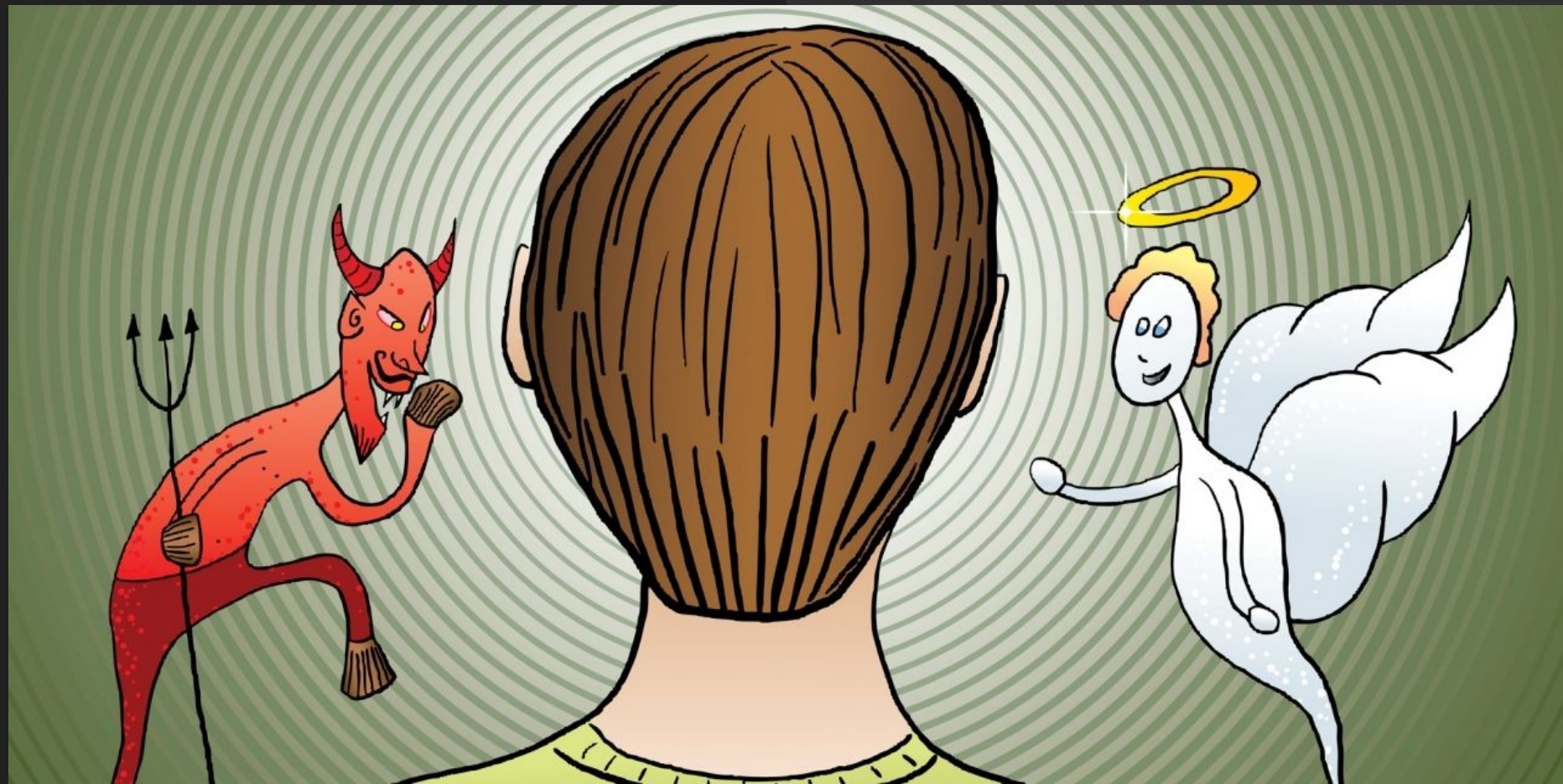
- For each infohash we've recorded:
- Search for peers by recursively making `get_peers()` requests using a burner `client_ID`

59,553 unique detected infohashes and 42,047 IPs later...

InfoHash / Torrent	IP addresses of Downloaders
f1fcdc1462d36530f526c1d9402eec9100b7ba18	['1.162.71.37:21430', '2.82.176.136:39572', '2.85.36.170:39207', '2.135.65.18:43008', '2.247.250.103:6881', '5.9.62.124:49253', '5.101.171.196:58502', '5.101.171.196:61319', '5.227.190.8:10481', '5.245.181.128:46540', '8.18.113.101:33696', '27.44.241.252:7152', '27.187.65.159:37375', '31.41.91.61:56085', '31.179.73.26:10520', '34.219.79.186:20419', '35.138.153.66:6881', '37.120.141.134:32287', '37.120.202.6:16183', '37.120.204.173:4578', '37.138.16.46:46732', '37.194.193.5:19975', '41.86.7.38:22335', '41.182.180.122:29397', '2.85.36.170:39207', '5.9.62.124:1', '5.189.140.45:59213', '5.206.45.141:1', '23.111.141.3:19790', '34.218.147.178:1', '37.6.0.143:1', '37.120.244.53:1', '37.204.54.198:1', '41.82.164.198:1', '45.129.56.202:22133', '45.164.136.4:43826', '46.32.144.19:46430', '46.160.252.33:62837', '47.217.47.43:1', '49.184.234.48:52336', '49.184.234.48:1', '50.92.14.193:41309', '50.92.14.193:1', '59.124.220.85:1', '60.216.139.208:19219', '61.125.146.76:54271', '62.16.252.26:1', '68.203.120.152:55899', '68.203.120.152:40329', '1.40.43.82:12715', '2.36.89.151:5923', '23.131.160.210:6881', '23.237.0.42:49311', '24.53.242.91:37375', '24.130.32.136:6881', '37.59.55.104:49264', '42.3.167.37:23667', '71.231.104.61:26741', '73.88.14.40:6999', '73.114.46.101:5099', '73.189.55.156:8999', '73.244.23.129:42626', '75.6.4.17:55454', '75.6.4.17:55423', '75.6.4.17:55391', '75.6.4.17:55464', '75.60.228.35:51200', '75.172.174.23:15267', '75.179.38.247:59464', '79.121.42.19:23584', '80.111.142.213:65400', '82.138.47.5:38246', '83.46.15.34:59574', '2.135.65.18:43008', '2.247.250.103:6881', '24.41.23.98:63111', '24.41.23.98:1', '24.41.23.98:59384', '36.44.171.24:18899', '37.204.11.214:1', '37.215.31.182:6321', '45.14.23.139:22560', '46.166.191.20:45952', '49.49.218.93:36454', '63.153.143.244:15654', '73.193.44.244:6881', '76.8.147.133:2329', '78.107.195.78:50004', '81.96.50.25:21123', '82.64.96.33:49154', '82.64.204.122:14332', '82.65.57.48:55686', '82.65.240.27:12958', '82.66.57.220:49594', '83.136.109.86:1900', '83.159.33.34:61322', '84.17.60.122:52086', '1.204.111.164:1', '2.80.178.214:1', '5.39.76.190:6942', '5.59.188.97:6889', '46.28.110.167:6890', '46.146.116.193:1', '47.203.66.216:1', '48.254.102.26:1', '61.92.139.226:1', '66.135.71.144:1', '68.237.89.130:1', '70.119.46.4:1', '71.161.203.51:1', '75.6.4.17:1', '77.81.142.124:1', '82.47.219.216:6881', '83.136.109.86:1900', '85.67.226.223:1', '85.222.36.228:1', '85.238.76.45:59932', '88.163.68.162:1', '91.204.58.197:1', '92.33.253.155:1', '92.220.51.214:1', '93.170.116.88:1', '93.184.9.50:1', '95.161.217.69:1', '24.2.90.90:9399', '24.41.23.98:52557', '24.41.23.98:51983', '24.41.23.98:52809', '24.41.23.98:17745', '24.41.23.98:60008', '24.137.42.85:45384', '45.15.16.206:49164', '45.133.182.148:20791', '45.248.151.169:30633', '46.142.124.33:6881', '46.146.216.43:47298', '58.224.217.200:50909', '63.153.143.244:15654', '68.93.134.147:46921', '68.93.134.147:46814', '70.26.9.112:52785', '70.39.113.221:6881', '70.121.32.165:6983', '75.189.4.28:11668', '78.199.99.34:33822', '79.53.90.93:1071', '84.173.48.90:8999', '5.189.140.45:59213', '24.3.26.185:34065', '36.226.215.173:16385', '37.165.43.28:18281', '42.188.48.220:47807', '61.125.146.76:54271', '79.133.18.232:26461', '82.64.96.33:49154', '82.65.197.218:63810', '82.65.240.27:12958', '84.17.60.132:49796', '90.153.44.225:24662', '91.167.183.111:13068', '120.244.62.212:25109', '124.120.251.222:63481', '124.120.251.222:1027', '162.201.165.38:31484', '178.121.2.124:1', '180.164.181.206:5977', '180.183.119.136:38502', '189.147.188.57:6881', '194.37.1.255:58505', '24.3.147.58:21017', '27.4.30.60:42252', '37.41.60.101:62611', '42.188.197.39:34198', '45.89.174.116:6569', '49.36.125.119:6881', '79.117.24.173:44630', '92.97.92.121:13593', '94.10.83.212:51220', '103.115.135.140:45537', '103.206.228.46:9796', '174.207.97.21:38740', '182.64.180.16:6881', '194.233.98.47:28750', '195.206.105.116:56078', '200.74.168.190:46647']
cab507494d02ebb1178b38f2e9d7be299c86b862	['50.92.14.193:41309', '50.92.14.193:1', '59.124.220.85:1', '60.216.139.208:19219', '61.125.146.76:54271', '62.16.252.26:1', '68.203.120.152:55899', '68.203.120.152:40329', '1.40.43.82:12715', '2.36.89.151:5923', '23.131.160.210:6881', '23.237.0.42:49311', '24.53.242.91:37375', '24.130.32.136:6881', '37.59.55.104:49264', '42.3.167.37:23667', '71.231.104.61:26741', '73.88.14.40:6999', '73.114.46.101:5099', '73.189.55.156:8999', '73.244.23.129:42626', '75.6.4.17:55454', '75.6.4.17:55423', '75.6.4.17:55391', '75.6.4.17:55464', '75.60.228.35:51200', '75.172.174.23:15267', '75.179.38.247:59464', '79.121.42.19:23584', '80.111.142.213:65400', '82.138.47.5:38246', '83.46.15.34:59574', '2.135.65.18:43008', '2.247.250.103:6881', '24.41.23.98:63111', '24.41.23.98:1', '24.41.23.98:59384', '36.44.171.24:18899', '37.204.11.214:1', '37.215.31.182:6321', '45.14.23.139:22560', '46.166.191.20:45952', '49.49.218.93:36454', '63.153.143.244:15654', '73.193.44.244:6881', '76.8.147.133:2329', '78.107.195.78:50004', '81.96.50.25:21123', '82.64.96.33:49154', '82.64.204.122:14332', '82.65.57.48:55686', '82.65.240.27:12958', '82.66.57.220:49594', '83.136.109.86:1900', '83.159.33.34:61322', '84.17.60.122:52086', '1.204.111.164:1', '2.80.178.214:1', '5.39.76.190:6942', '5.59.188.97:6889', '46.28.110.167:6890', '46.146.116.193:1', '47.203.66.216:1', '48.254.102.26:1', '61.92.139.226:1', '66.135.71.144:1', '68.237.89.130:1', '70.119.46.4:1', '71.161.203.51:1', '75.6.4.17:1', '77.81.142.124:1', '82.47.219.216:6881', '83.136.109.86:1900', '85.67.226.223:1', '85.222.36.228:1', '85.238.76.45:59932', '88.163.68.162:1', '91.204.58.197:1', '92.33.253.155:1', '92.220.51.214:1', '93.170.116.88:1', '93.184.9.50:1', '95.161.217.69:1', '24.2.90.90:9399', '24.41.23.98:52557', '24.41.23.98:51983', '24.41.23.98:52809', '24.41.23.98:17745', '24.41.23.98:60008', '24.137.42.85:45384', '45.15.16.206:49164', '45.133.182.148:20791', '45.248.151.169:30633', '46.142.124.33:6881', '46.146.216.43:47298', '58.224.217.200:50909', '63.153.143.244:15654', '68.93.134.147:46921', '68.93.134.147:46814', '70.26.9.112:52785', '70.39.113.221:6881', '70.121.32.165:6983', '75.189.4.28:11668', '78.199.99.34:33822', '79.53.90.93:1071', '84.173.48.90:8999', '5.189.140.45:59213', '24.3.26.185:34065', '36.226.215.173:16385', '37.165.43.28:18281', '42.188.48.220:47807', '61.125.146.76:54271', '79.133.18.232:26461', '82.64.96.33:49154', '82.65.197.218:63810', '82.65.240.27:12958', '84.17.60.132:49796', '90.153.44.225:24662', '91.167.183.111:13068', '120.244.62.212:25109', '124.120.251.222:63481', '124.120.251.222:1027', '162.201.165.38:31484', '178.121.2.124:1', '180.164.181.206:5977', '180.183.119.136:38502', '189.147.188.57:6881', '194.37.1.255:58505', '24.3.147.58:21017', '27.4.30.60:42252', '37.41.60.101:62611', '42.188.197.39:34198', '45.89.174.116:6569', '49.36.125.119:6881', '79.117.24.173:44630', '92.97.92.121:13593', '94.10.83.212:51220', '103.115.135.140:45537', '103.206.228.46:9796', '174.207.97.21:38740', '182.64.180.16:6881', '194.233.98.47:28750', '195.206.105.116:56078', '200.74.168.190:46647']
e84213a794f3ccd890382a54a64ca68b7e925433	['75.6.4.17:55464', '75.60.228.35:51200', '75.172.174.23:15267', '75.179.38.247:59464', '79.121.42.19:23584', '80.111.142.213:65400', '82.138.47.5:38246', '83.46.15.34:59574', '2.135.65.18:43008', '2.247.250.103:6881', '24.41.23.98:63111', '24.41.23.98:1', '24.41.23.98:59384', '36.44.171.24:18899', '37.204.11.214:1', '37.215.31.182:6321', '45.14.23.139:22560', '46.166.191.20:45952', '49.49.218.93:36454', '63.153.143.244:15654', '73.193.44.244:6881', '76.8.147.133:2329', '78.107.195.78:50004', '81.96.50.25:21123', '82.64.96.33:49154', '82.64.204.122:14332', '82.65.57.48:55686', '82.65.240.27:12958', '82.66.57.220:49594', '83.136.109.86:1900', '83.159.33.34:61322', '84.17.60.122:52086', '1.204.111.164:1', '2.80.178.214:1', '5.39.76.190:6942', '5.59.188.97:6889', '46.28.110.167:6890', '46.146.116.193:1', '47.203.66.216:1', '48.254.102.26:1', '61.92.139.226:1', '66.135.71.144:1', '68.237.89.130:1', '70.119.46.4:1', '71.161.203.51:1', '75.6.4.17:1', '77.81.142.124:1', '82.47.219.216:6881', '83.136.109.86:1900', '85.67.226.223:1', '85.222.36.228:1', '85.238.76.45:59932', '88.163.68.162:1', '91.204.58.197:1', '92.33.253.155:1', '92.220.51.214:1', '93.170.116.88:1', '93.184.9.50:1', '95.161.217.69:1', '24.2.90.90:9399', '24.41.23.98:52557', '24.41.23.98:51983', '24.41.23.98:52809', '24.41.23.98:17745', '24.41.23.98:60008', '24.137.42.85:45384', '45.15.16.206:49164', '45.133.182.148:20791', '45.248.151.169:30633', '46.142.124.33:6881', '46.146.216.43:47298', '58.224.217.200:50909', '63.153.143.244:15654', '68.93.134.147:46921', '68.93.134.147:46814', '70.26.9.112:52785', '70.39.113.221:6881', '70.121.32.165:6983', '75.189.4.28:11668', '78.199.99.34:33822', '79.53.90.93:1071', '84.173.48.90:8999', '5.189.140.45:59213', '24.3.26.185:34065', '36.226.215.173:16385', '37.165.43.28:18281', '42.188.48.220:47807', '61.125.146.76:54271', '79.133.18.232:26461', '82.64.96.33:49154', '82.65.197.218:63810', '82.65.240.27:12958', '84.17.60.132:49796', '90.153.44.225:24662', '91.167.183.111:13068', '120.244.62.212:25109', '124.120.251.222:63481', '124.120.251.222:1027', '162.201.165.38:31484', '178.121.2.124:1', '180.164.181.206:5977', '180.183.119.136:38502', '189.147.188.57:6881', '194.37.1.255:58505', '24.3.147.58:21017', '27.4.30.60:42252', '37.41.60.101:62611', '42.188.197.39:34198', '45.89.174.116:6569', '49.36.125.119:6881', '79.117.24.173:44630', '92.97.92.121:13593', '94.10.83.212:51220', '103.115.135.140:45537', '103.206.228.46:9796', '174.207.97.21:38740', '182.64.180.16:6881', '194.233.98.47:28750', '195.206.105.116:56078', '200.74.168.190:46647']
b26c81363ac1a236765385a702aec107a49581b5	['82.64.96.33:49154', '82.64.204.122:14332', '82.65.57.48:55686', '82.65.240.27:12958', '82.66.57.220:49594', '83.136.109.86:1900', '83.159.33.34:61322', '84.17.60.122:52086', '1.204.111.164:1', '2.80.178.214:1', '5.39.76.190:6942', '5.59.188.97:6889', '46.28.110.167:6890', '46.146.116.193:1', '47.203.66.216:1', '48.254.102.26:1', '61.92.139.226:1', '66.135.71.144:1', '68.237.89.130:1', '70.119.46.4:1', '71.161.203.51:1', '75.6.4.17:1', '77.81.142.124:1', '82.47.219.216:6881', '83.136.109.86:1900', '85.67.226.223:1', '85.222.36.228:1', '85.238.76.45:59932', '88.163.68.162:1', '91.204.58.197:1', '92.33.253.155:1', '92.220.51.214:1', '93.170.116.88:1', '93.184.9.50:1', '95.161.217.69:1', '24.2.90.90:9399', '24.41.23.98:52557', '24.41.23.98:51983', '24.41.23.98:52809', '24.41.23.98:17745', '24.41.23.98:60008', '24.137.42.85:45384', '45.15.16.206:49164', '45.133.182.148:20791', '45.248.151.169:30633', '46.142.124.33:6881', '46.146.216.43:47298', '58.224.217.200:50909', '63.153.143.244:15654', '68.93.134.147:46921', '68.93.134.147:46814', '70.26.9.112:52785', '70.39.113.221:6881', '70.121.32.165:6983', '75.189.4.28:11668', '78.199.99.34:33822', '79.53.90.93:1071', '84.173.48.90:8999', '5.189.140.45:59213', '24.3.26.185:34065', '36.226.215.173:16385', '37.165.43.28:18281', '42.188.48.220:47807', '61.125.146.76:54271', '79.133.18.232:26461', '82.64.96.33:49154', '82.65.197.218:63810', '82.65.240.27:12958', '84.17.60.132:49796', '90.153.44.225:24662', '91.167.183.111:13068', '120.244.62.212:25109', '124.120.251.222:63481', '124.120.251.222:1027', '162.201.165.38:31484', '178.121.2.124:1', '180.164.181.206:5977', '180.183.119.136:38502', '189.147.188.57:6881', '194.37.1.255:58505', '24.3.147.58:21017', '27.4.30.60:42252', '37.41.60.101:62611', '42.188.197.39:34198', '45.89.174.116:6569', '49.36.125.119:6881', '79.117.24.173:44630', '92.97.92.121:13593', '94.10.83.212:51220', '103.115.135.140:45537', '103.206.228.46:9796', '174.207.97.21:38740', '182.64.180.16:6881', '194.233.98.47:28750', '195.206.105.116:56078', '200.74.168.190:46647']
e2467cbf021192c241367b892230dc1e05c0580e	['85.222.36.228:1', '85.238.76.45:59932', '88.163.68.162:1', '91.204.58.197:1', '92.33.253.155:1', '92.220.51.214:1', '93.170.116.88:1', '93.184.9.50:1', '95.161.217.69:1', '24.2.90.90:9399', '24.41.23.98:52557', '24.41.23.98:51983', '24.41.23.98:52809', '24.41.23.98:17745', '24.41.23.98:60008', '24.137.42.85:45384', '45.15.16.206:49164', '45.133.182.148:20791', '45.248.151.169:30633', '46.142.124.33:6881', '46.146.216.43:47298', '58.224.217.200:50909', '63.153.143.244:15654', '68.93.134.147:46921', '68.93.134.147:46814', '70.26.9.112:52785', '70.39.113.221:6881', '70.121.32.165:6983', '75.189.4.28:11668', '78.199.99.34:33822', '79.53.90.93:1071', '84.173.48.90:8999', '5.189.140.45:59213', '24.3.26.185:34065', '36.226.215.173:16385', '37.165.43.28:18281', '42.188.48.220:47807', '61.125.146.76:54271', '79.133.18.232:26461', '82.64.96.33:49154', '82.65.197.218:63810', '82.65.240.27:12958', '84.17.60.132:49796', '90.153.44.225:24662', '91.167.183.111:13068', '120.244.62.212:25109', '124.120.251.222:63481', '124.120.251.222:1027', '162.201.165.38:31484', '178.121.2.124:1', '180.164.181.206:5977', '180.183.119.136:38502', '189.147.188.57:6881', '194.37.1.255:58505', '24.3.147.58:21017', '27.4.30.60:42252', '37.41.60.101:62611', '42.188.197.39:34198', '45.89.174.116:6569', '49.36.125.119:6881', '79.117.24.173:44630', '92.97.92.121:13593', '94.10.83.212:51220', '103.115.135.140:45537', '103.206.228.46:9796', '174.207.97.21:38740', '182.64.180.16:6881', '194.233.98.47:28750', '195.206.105.116:5607

Phase 3

What to do with the data...



Phase 3

Being “good”

Phase 3

Being “good”

- Look up metadata for each infohash
- Provide information to other peers on the network

Being “good” with this data:

- Take our infohashes, and download the metadata
- We turn the infohashes and metadata into a torrent search engine!
- Even if many torrent search engines get taken down, any client in the DHT can quickly create a new one

InfoHash	IP addresses and Ports
f1fcdc1462d36530f526c1d9402eec9100b7ba18	['1.162.71.37:21430', '2.82.176.136:39572', '2.85.36.170:39207']

Name: ubuntu-21.10-desktop-amd64.iso
File: info/f1fcdc1462d36530f526c1d9402eec9100b7ba18.torrent

GENERAL

Name: ubuntu-21.10-desktop-amd64.iso
Hash: f1fcdc1462d36530f526c1d9402eec9100b7ba18
Created by:
Created on: Fri Nov 26 23:15:09 2021
Comment: Torrent downloaded from torrent cache at <http://itorrents.org>
Piece Count: 11889
Piece Size: 256.0 KiB
Total Size: 3.12 GB
Privacy: Public torrent

Phase 3

Being “bad”

Phase 3

Being “bad”

- Look up who these IP addresses are
- Get location information
- Who owns these IP addresses (institutions, VPN companies, local ISPs, etc)

Being “bad” with this data:

- Our infohashes and IP addresses are a massive database of tens of thousands of people who have downloaded potentially illegal material...
- Request ISPs to hand over data associating IP addresses to people
- This data can be used by law enforcement / copyright holders

InfoHash f1fcdc1462d36530f526c1d9402eec9100b7ba18 IP addresses and Ports ['1.162.71.37:21430', '2.82.176.136:39572', '2.85.36.170:39207']

```
# whois.arin.net

NetRange:      137.22.192.0 - 137.22.255.255
CIDR:          137.22.192.0/18
NetName:       CARLETON
NetHandle:     NET-137-22-192-0-1
Parent:        NET137 (NET-137-0-0-0-0)
NetType:       Direct Allocation
OriginAS:      Carleton College (CARLET)
RegDate:       1989-10-11
Updated:       2021-12-14
Ref:           https://rdap.arin.net/registry/ip/137.22.192.0

OrgName:       Carleton College
OrgId:         CARLET
Address:       1 North College Street
City:          Northfield
StateProv:    MN
PostalCode:   55057
Country:      US
RegDate:       1989-10-11
Updated:       2021-09-10
Ref:           https://rdap.arin.net/registry/entity/CARLET
```

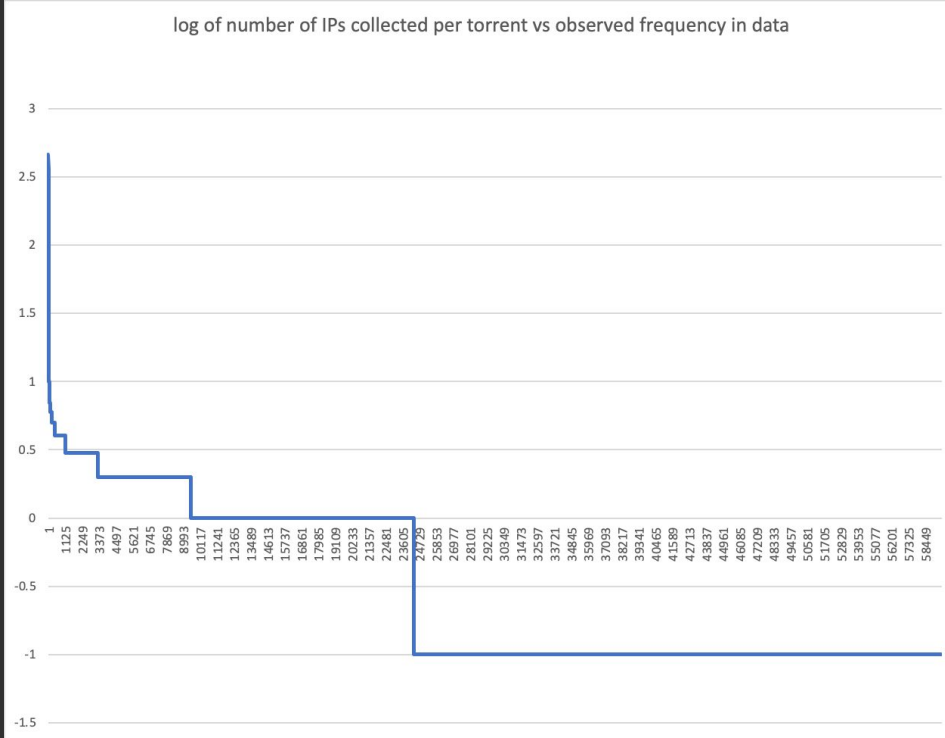


Data:

- 59,553 infohashes saved
- 42,047 IPs collected

Data:

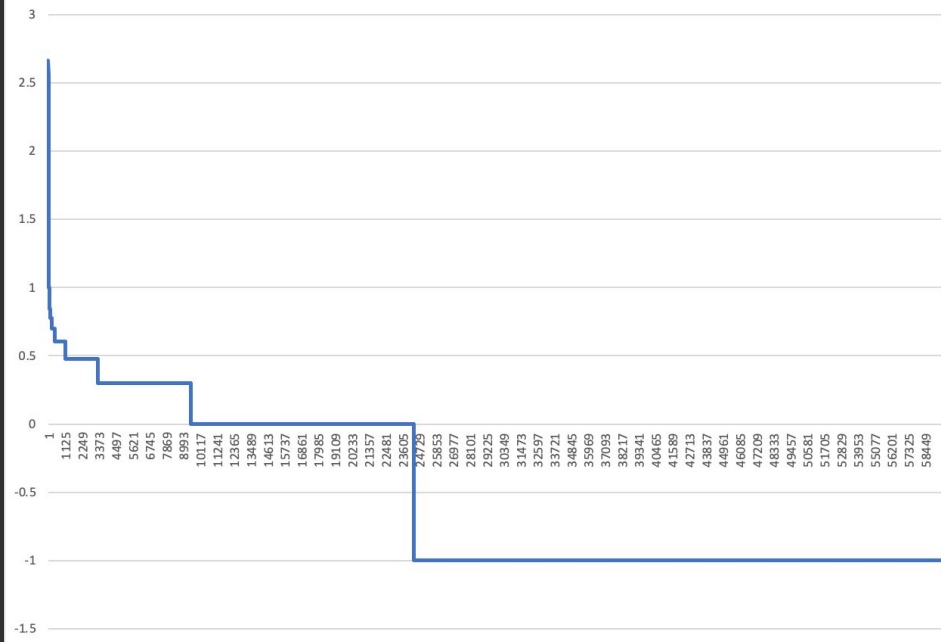
- 59,553 infohashes saved
- 42,047 IPs collected



Data:

- 59,553 infohashes saved
- 42,047 IPs collected
- Popular torrents are *very* popular
 - Most popular torrent had 458 unique IPs collected
- Unpopular torrents are *very not* popular
 - 35,209 torrents with 0 unique IPs collected
 - 14,870 torrents with 1 unique IP collected

log of number of IPs collected per torrent vs observed frequency in data





#all_helpdesk ▾ Helpdesk-wide anno...

80



3



Tuesday, February 15th ▾

A

Adit 11:38 AM

Change Notice: Friday morning 8AM security will be restricting Bittorrent traffic on the Carleton Guest network. No change to the Eduroam network. Please let me know if you have concerns about this change.



A



6 replies Last reply 6 days ago

Thursday, February 17th ▾


References

Wolchok, Scott, and J. Alex Halderman. "Crawling BitTorrent DHTs for Fun and Profit." In 4th USENIX Workshop on Offensive Technologies (WOOT 10). 2010.

Wang, Liang, and Jussi Kangasharju. "Measuring large-scale distributed systems: case of bittorrent mainline dht." In IEEE P2P 2013 Proceedings, pp. 1-10. IEEE, 2013.

BitTorrent protocol and extensions:

- https://www.bittorrent.org/beps/bep_0003.html (BitTorrent Protocol)
- https://www.bittorrent.org/beps/bep_0005.html (DHT Extension)
- https://www.bittorrent.org/beps/bep_0009.html (Metadata Extension)

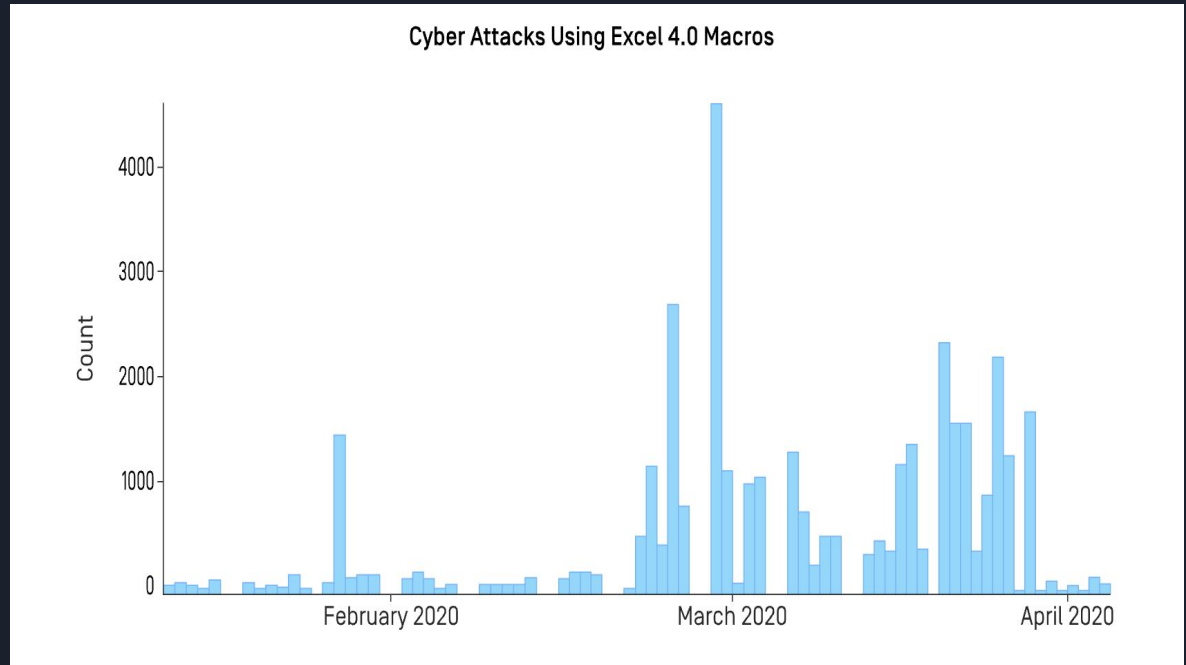


Spreadsheet Security: Exploiting Microsoft Macro Functionality

By John Witte and Skyler Kessenich

Motivation

- ★ Volume of companies reliant on excel for data management
- ★ COVID led to increase in attack usage
 - Originally spiked in 2015 from Team Rocket Kitten
- ★ Increase in phishing attacks at financial companies
- ★ Relevant to our internships



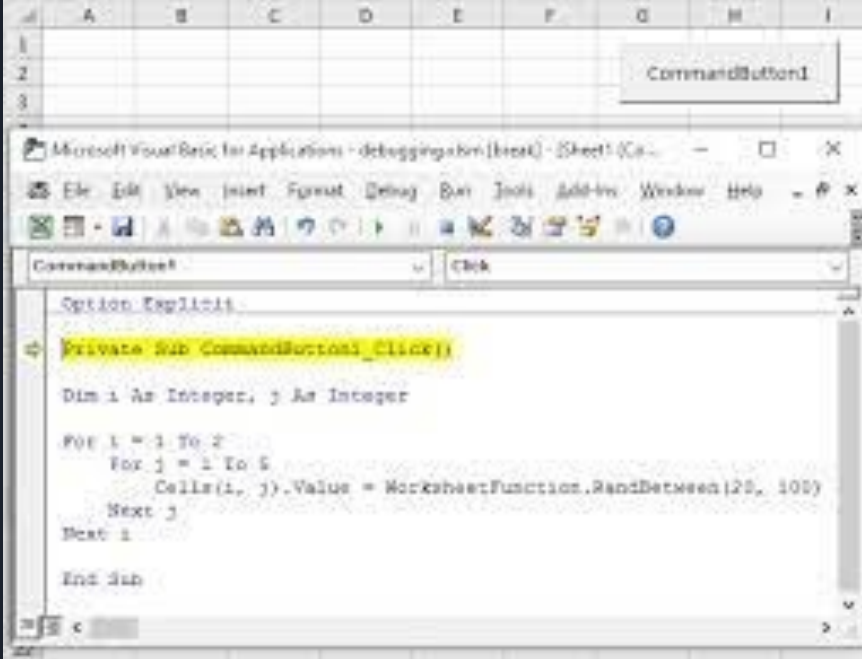


Background: History

- ★ First attack in 1995
- ★ Melissa Attack 1999
- ★ 2000-2014 Dormant years
- ★ 2014 ZeuS, DRIDEX, ROVNIX info stealing macro attacks
- ★ 2015 Banks and Team Rocket Kitten
- ★ 2020 COVID Uptick

Background: What is a Macro

- ★ Functionality allowing users to automate repetitive tasks
 - Helps with data entry at a large scale
- ★ Code written in VBA
- ★ Examples:
 - Check cells for certain features
 - Sort sheets
 - Create alerts
 - Link sheets
 - Open, write and read other documents



```
Option Explicit

Private Sub CommandButton1_Click()

    Dim i As Integer, j As Integer

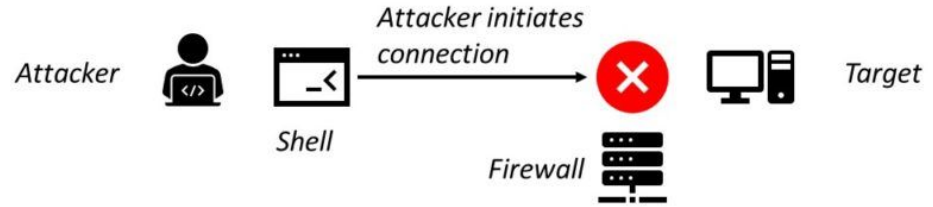
    For i = 1 To 2
        For j = 1 To 5
            Cells(1, j).Value = WorksheetFunction.RandBetween(20, 100)
        Next j
    Next i

End Sub
```

Background: Reverse Payload Attack

- ★ Reverse Shell Attack
 - Reverse TCP
- ★ Metasploit

Without Reverse Shell



With Reverse Shell



How it Works - Step 1: Writing the Payload

- ★ Uses Metasploit's MSFVenom on Kali Linux
- ★ Puts attacking IP and Port in file
- ★ When file is run, reverse shell initiates



How it Works - Step 2: Embedding In an Excel Macro

- ★ Excel's VBA Language
- ★ Excel needs to run the payload
- ★ Uses auto_open command to automatically run the macro



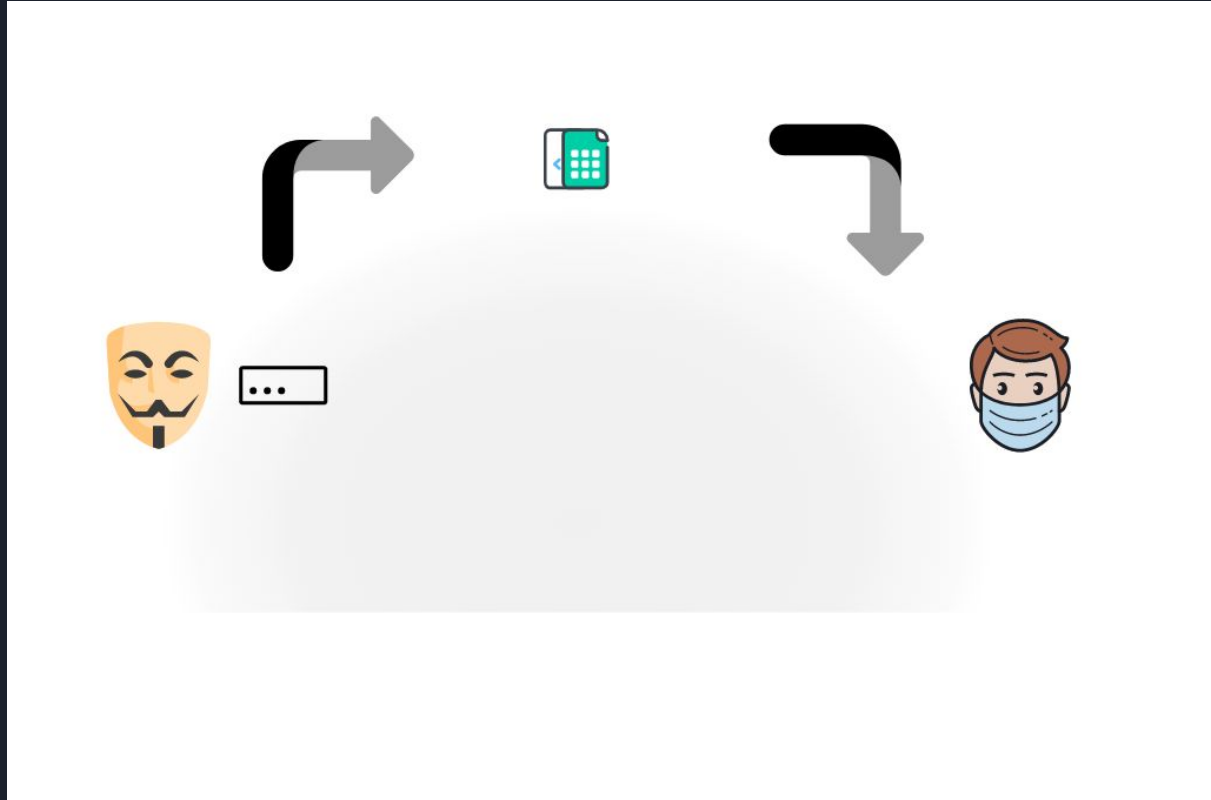
How it Works - Step 3: Set Up Listening Machine

- ★ Uses Metasploit's MSFConsole
- ★ Tells attacking machine:
 - IP address
 - Listening Port
 - Payload
- ★ If connection is made, the machine has root access



How it Works - Step 4: Phishing the Target

- ★ Target must acquire Excel sheet
- ★ Methods:
 - Email
 - Web Download
- ★ Once opened, the attack launches



Our Process - The Payload

- ★ Payload
 - MSFVenom's Reverse TCP Python payload
 - Python file
- ★ Embedding the Macro
 - Not first approach
 - Write to file
 - Execute
- ★ Running the payload







Roadblocks:

- ★ Running the File Directly from Sheet
 - Debugging in VBA
- ★ Where to write file to machine
- ★ Obfuscating malicious code
 - Encoding
 - Replacing
 - Spitting



Defenses

- ★ Turning off Macros
 - Microsoft shutting down macros for windows 10 and later
- ★ Anti Virus software
 - Different techniques
 - Obfuscation makes this hard



Conclusion

- ★ Able to start remote shell on target computer
 - Works on any computer with python and excel
- ★ What we took away from this:
 - Replicating attacks is hard
 - Do not open emails with attached documents unless you are 100% sure it is from known contact
- ★ Applications
 - Presented to our jobs



Works Cited

- ★ “20 Years of Macro Malware: From Harmless Concept to Targeted Attacks.” *Security News*, <https://www.trendmicro.com/vinfo/us/security/news/cybercrime-and-digital-threats/20-years-of-macro-malware-from-harmless-concept-to-targeted-attacks>.
- ★ Bansal, -- By Sumit, et al. “24 Useful Excel Macro Examples for VBA Beginners (Ready-to-Use).” *Trump Excel*, 20 Dec. 2021, <https://trumpexcel.com/excel-macro-examples/>.
- ★ Brook, Chris. “What Is Macro Malware?” *Digital Guardian*, 5 Dec. 2018, <https://digitalguardian.com/blog/what-macro-malware>.
- ★ Grinberg, Shiran. “Office Macro Attacks.” *Cynet*, <https://www.cynet.com/attack-techniques-hands-on/office-macro-attacks/>.
- ★ Lam, Vinh. “Excel 4.0 Macro: Old Feature, New Attack Technique.” *OPSWAT*, 26 July 2021, <https://www.opswat.com/blog/excel-4-0-macro-old-feature-new-attack-technique#:~:text=The%20First%20Excel%204.0%20Macro%20Attack&text=It%20involves%20an%20infected%20sheet,target%20into%20opening%20the%20file>.
- ★ “MSFVenom Cheat Sheet.” *HackTricks*, <https://book.hacktricks.xyz/shells/shells/msfvenom>.
- ★ Newman, Lily Hay. “Microsoft's Small Step to Disable Macros Is a Huge Win for Security.” *Wired*, Conde Nast, 11 Feb. 2022, <https://www.wired.com/story/microsoft-disables-macros-default-security-phishing/>.
- ★ “Rocket Kitten Showing Its Claws: Operation Woolen-Goldfish and the Ghole Campaign.” *Security News*, <https://www.trendmicro.com/vinfo/us/security/news/cyber-attacks/operation-woolen-goldfish-when-kittens-go-phishing>.



Acknowledgments

- Thomas Baraniak
- Aaron Heidgerken-Greene
- Mike Tie
- Jeff Ondich