FOX IT

Detecting Quantum Insert

Using Bro-IDS

5 August - BroCon 2015

Yun Zheng Hu
Fox-IT Security Research Team

$ whoami

Yun Zheng Hu
Principal Security Expert

www.fox-it.com

github.com/fox-it

@YunZhengHu

Fox-IT
Delft, Netherlands

North Sea
THE NETHERLANDS
U.K.
AMSTERDAM
DELFT
BELGIUM
GERMANY
FRANCE

Past contributions to Bro

- BIT-968: `bytestring_to_count()`
- BIT-969: `reverse()`

Agenda

- What is QUANTUM INSERT?
- How to perform QUANTUM INSERT?
- Detection
- Demo
- Injections we detected in the wild

What is QUANTUMINSERT?

What is QUANTUMINSERT?

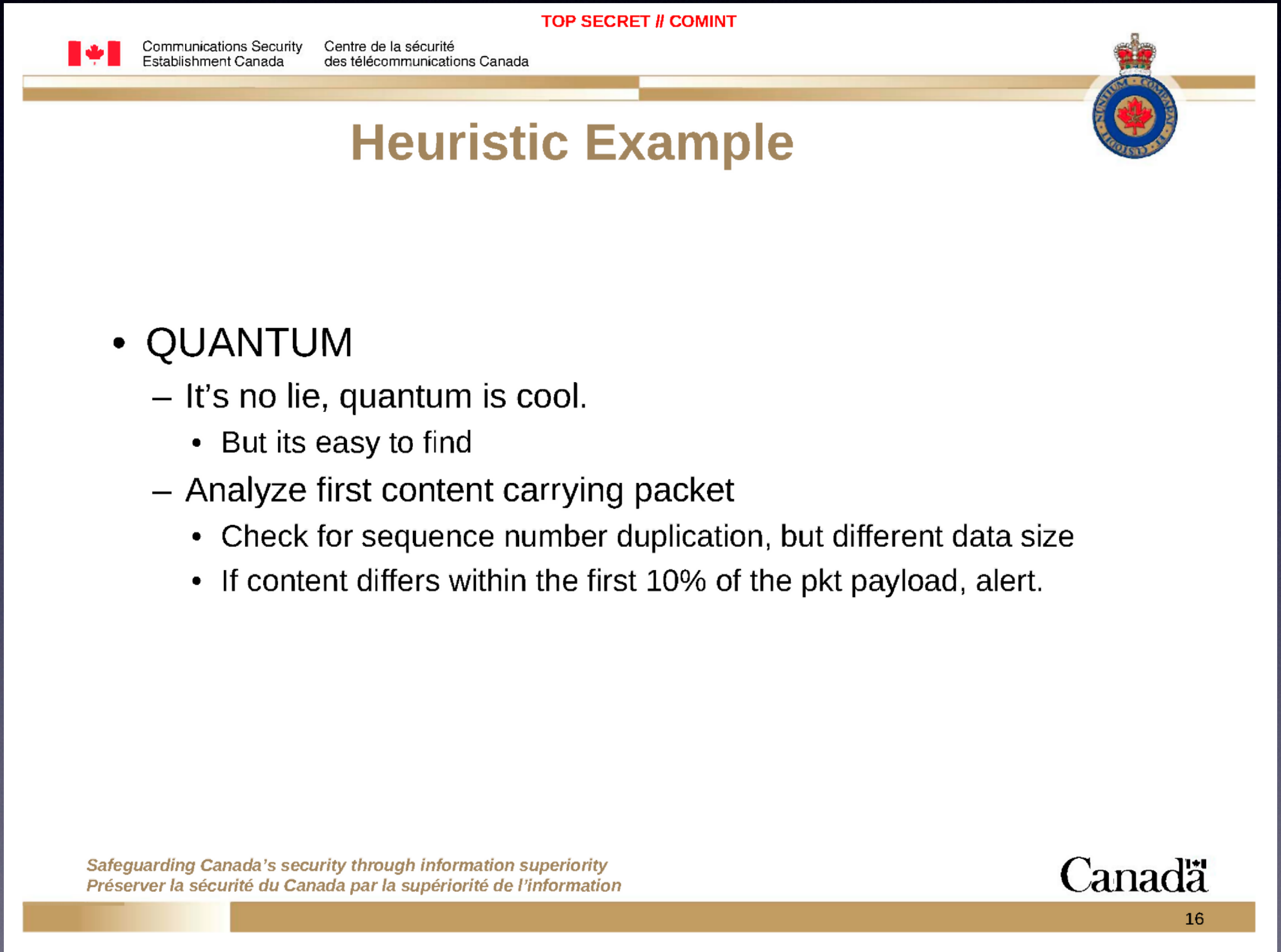
- Snowden leaks
- Codename for TCP hijacking
    - Specifically targeting HTTP
    - More injection than hijacking
- React faster than other servers
    - Win race condition

Other QUANTUM attacks

Name	Description
QUANTUMDNS	DNS Injection/Redirection of A records
QUANTUMBOT	Hijacking idle IRC bots and c&c communication from bots.
QUANTUMSKY	Deny access to webpage by injecting/spoofing RST packets
QUANTUMBISCUIT	Enhance QI behind large proxies

source: https://firstlook.org/theintercept/document/2014/03/12/one-way-quantum/

Slide that started it all



TOP SECRET // COMINT

Communications Security Establishment Canada Centre de la sécurité des télécommunications Canada

Heuristic Example

- QUANTUM
  - It's no lie, quantum is cool.
    - But its easy to find
  - Analyze first content carrying packet
    - Check for sequence number duplication, but different data size
    - If content differs within the first 10% of the pkt payload, alert.

Safeguarding Canada's security through information superiority
Préserver la sécurité du Canada par la supériorité de l'information

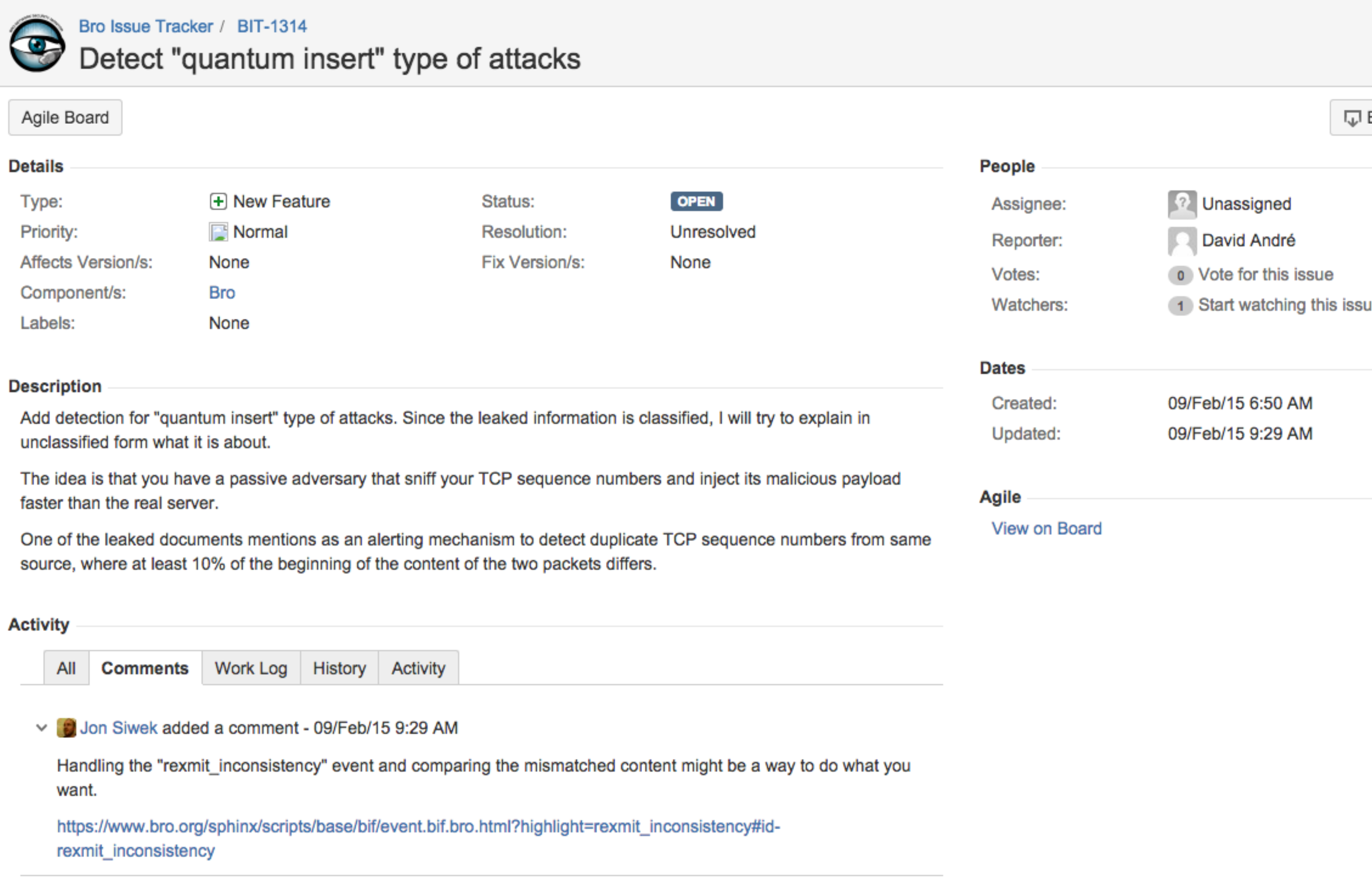
Canada

16

source: https://www.eff.org/files/2015/01/23/20150117[...]network_based_anomaly_.pdf

Security Research Team

- How does it really work?
  - Perform a successful Quantum Insert
  - Capture a PCAP (or it didn’t happen)
  - Check existing IDS software for detection



Bro Issue Tracker / BIT-1314

Detect "quantum insert" type of attacks

Agile Board

Details

Type:	New Feature	Status:	OPEN
Priority:	Normal	Resolution:	Unresolved
Affects Version/s:	None	Fix Version/s:	None
Component/s:	Bro		
Labels:	None		

People

Assignee:	Unassigned
Reporter:	David André
Votes:	0 Vote for this issue
Watchers:	1 Start watching this issu

Dates

Created:	09/Feb/15 6:50 AM
Updated:	09/Feb/15 9:29 AM

Agile

View on Board

Description

Add detection for "quantum insert" type of attacks. Since the leaked information is classified, I will try to explain in unclassified form what it is about.

The idea is that you have a passive adversary that sniff your TCP sequence numbers and inject its malicious payload faster than the real server.

One of the leaked documents mentions as an alerting mechanism to detect duplicate TCP sequence numbers from same source, where at least 10% of the beginning of the content of the two packets differs.

Activity

Jon Siwek added a comment - 09/Feb/15 9:29 AM

Handling the "rexmit_inconsistency" event and comparing the mismatched content might be a way to do what you want.

https://www.bro.org/sphinx/scripts/base/bif/event.bif.bro.html?highlight=rexmit_inconsistency#id-rexmit_inconsistency

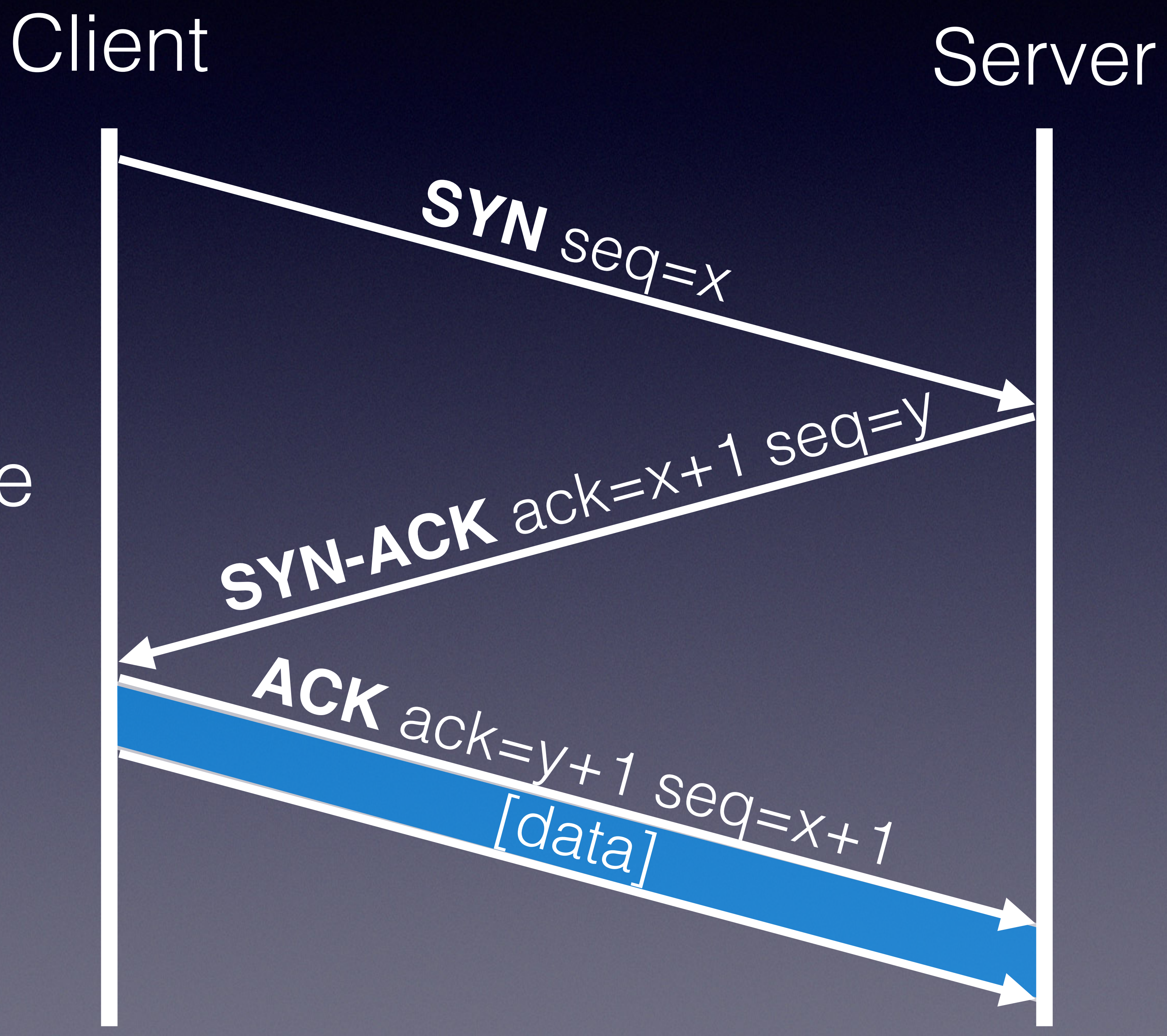
Initial IDS Coverage

- Bro should detect it using `rexmit_inconsistency`, but it didn’t work
- Snort protocol decoders did not trigger anything
- Suricata did not trigger anything, needed:
  - `stream-event:reassembly_overlap_different_data`

Howto QUANTUM

TCP 3-way Handshake

1. SYN
2. SYN/ACK response
3. ACK



TCP Hijacking

FREE KEVIN

- Kevin Mitnick
  - Successfully hijacked a remote TCP session
  - Predicted the TCP sequence numbers
- Nowadays, TCP sequence numbers are random
  - Have to sniff and leak the information

QI vs TCP Injection

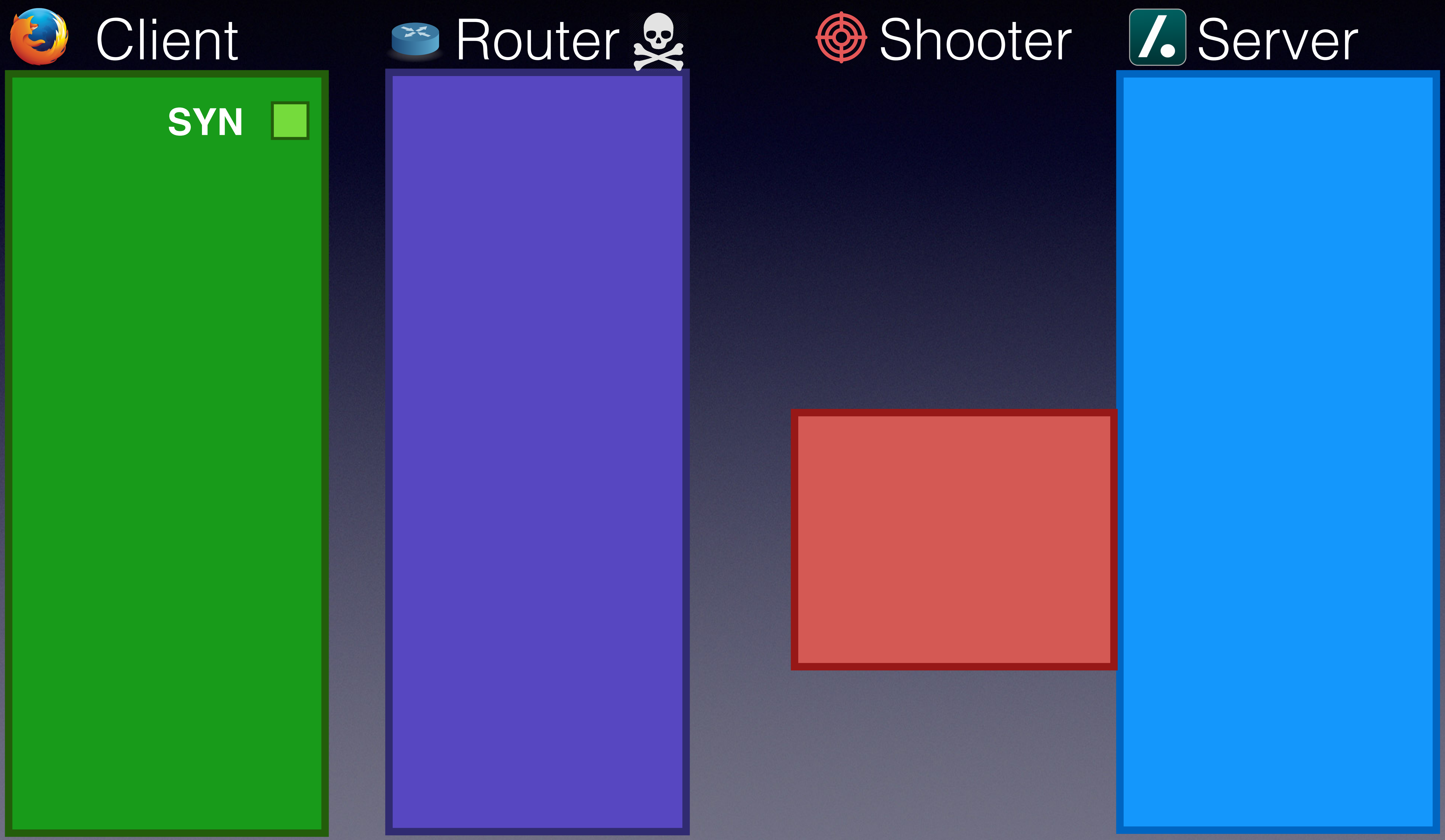
- Quantum Insert is TCP packet injection
- But specifically against HTTP sessions
- Confirms target by checking tracking Cookies
- Uses a **monitor** to leak the information
- Uses a **shooter** to spoof and insert the packet

FOX IT

Requirements

- Observe & Leak TCP Session information
- Able to spoof packets
- Racing the response (be faster)

TCP Injection



Client

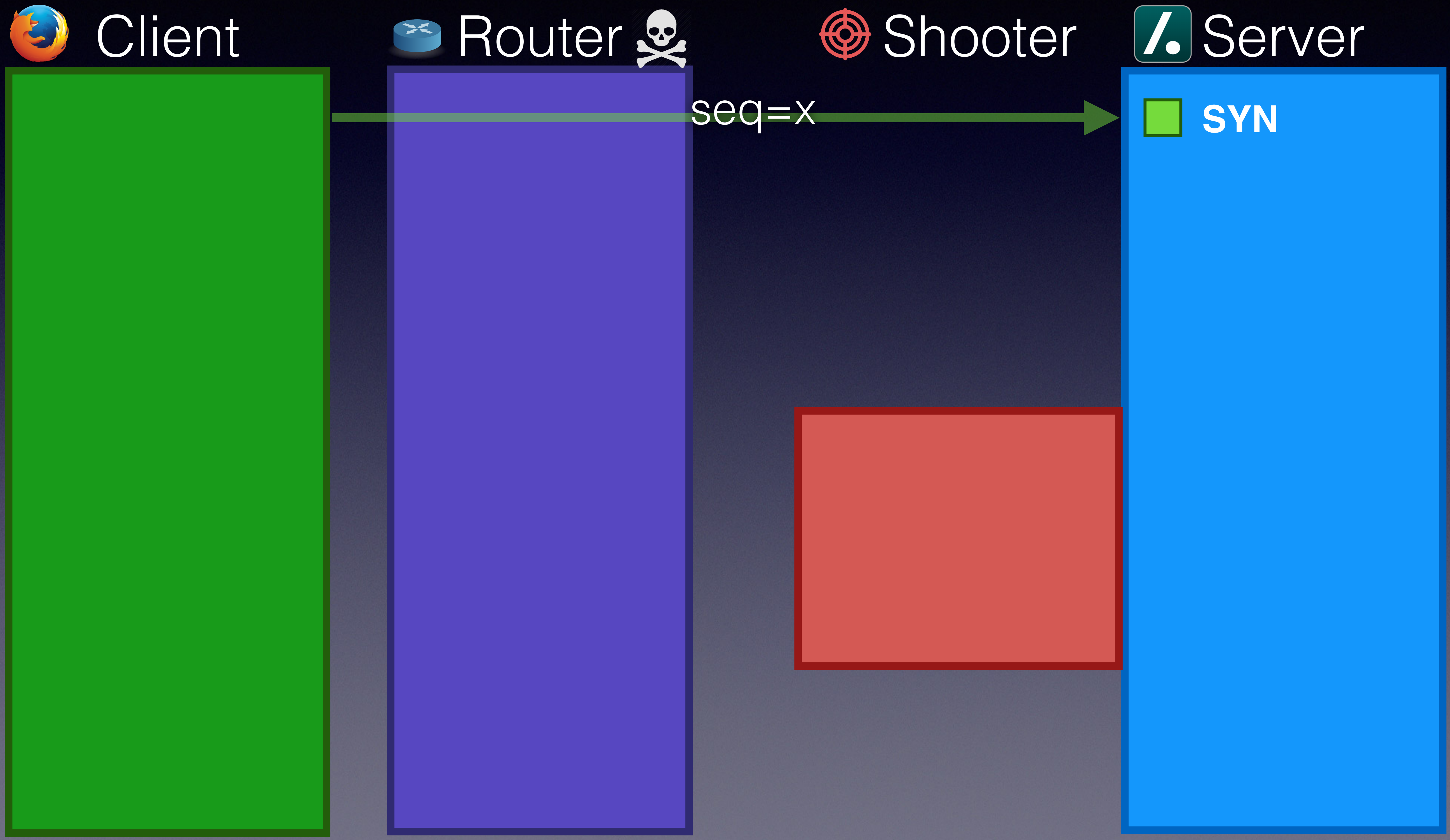
Router ☠

Shooter

Server

SYN 🟩

TCP Injection



Client

Router ☠

Shooter

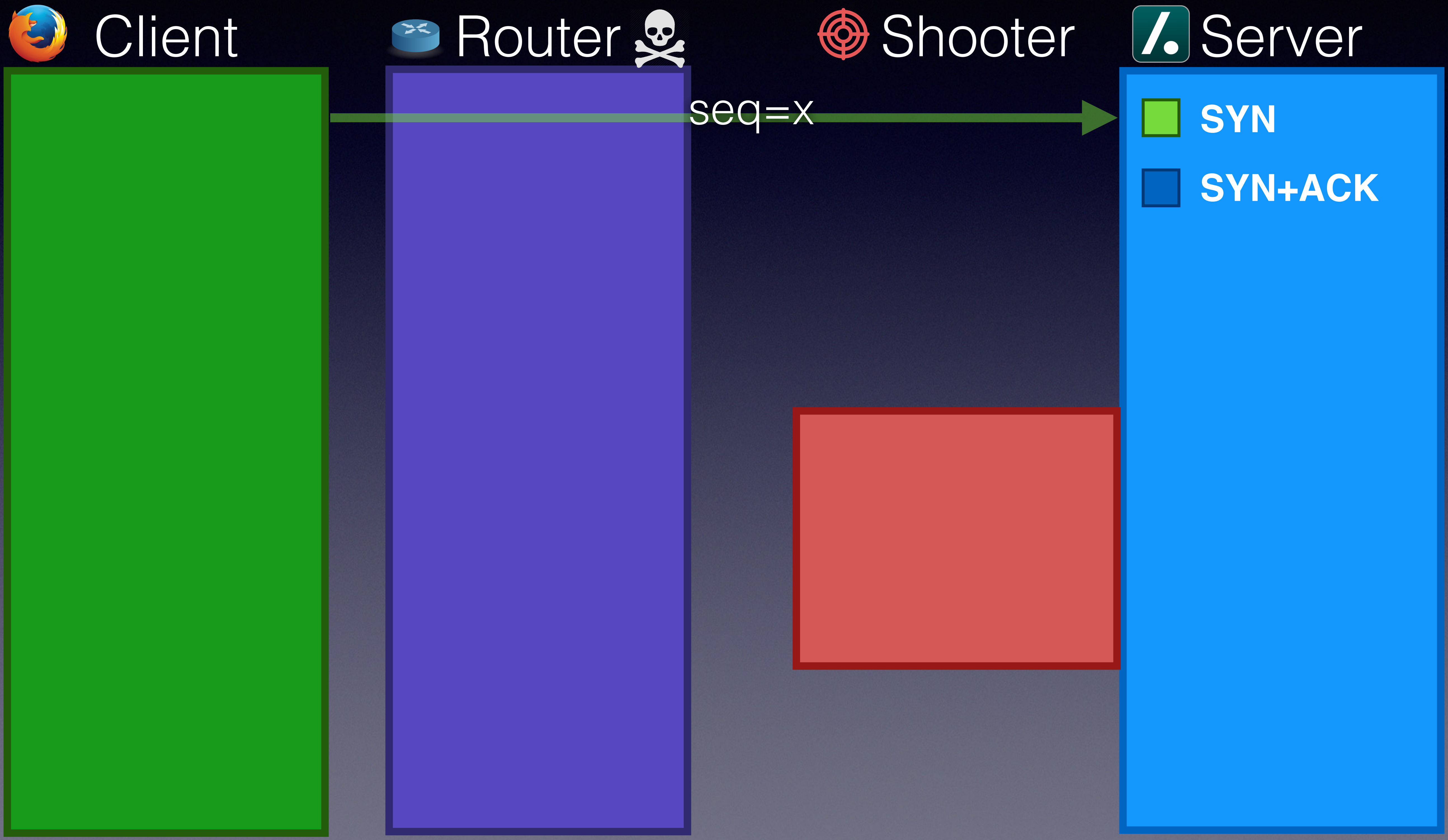
Server

seq=x

🟩 SYN

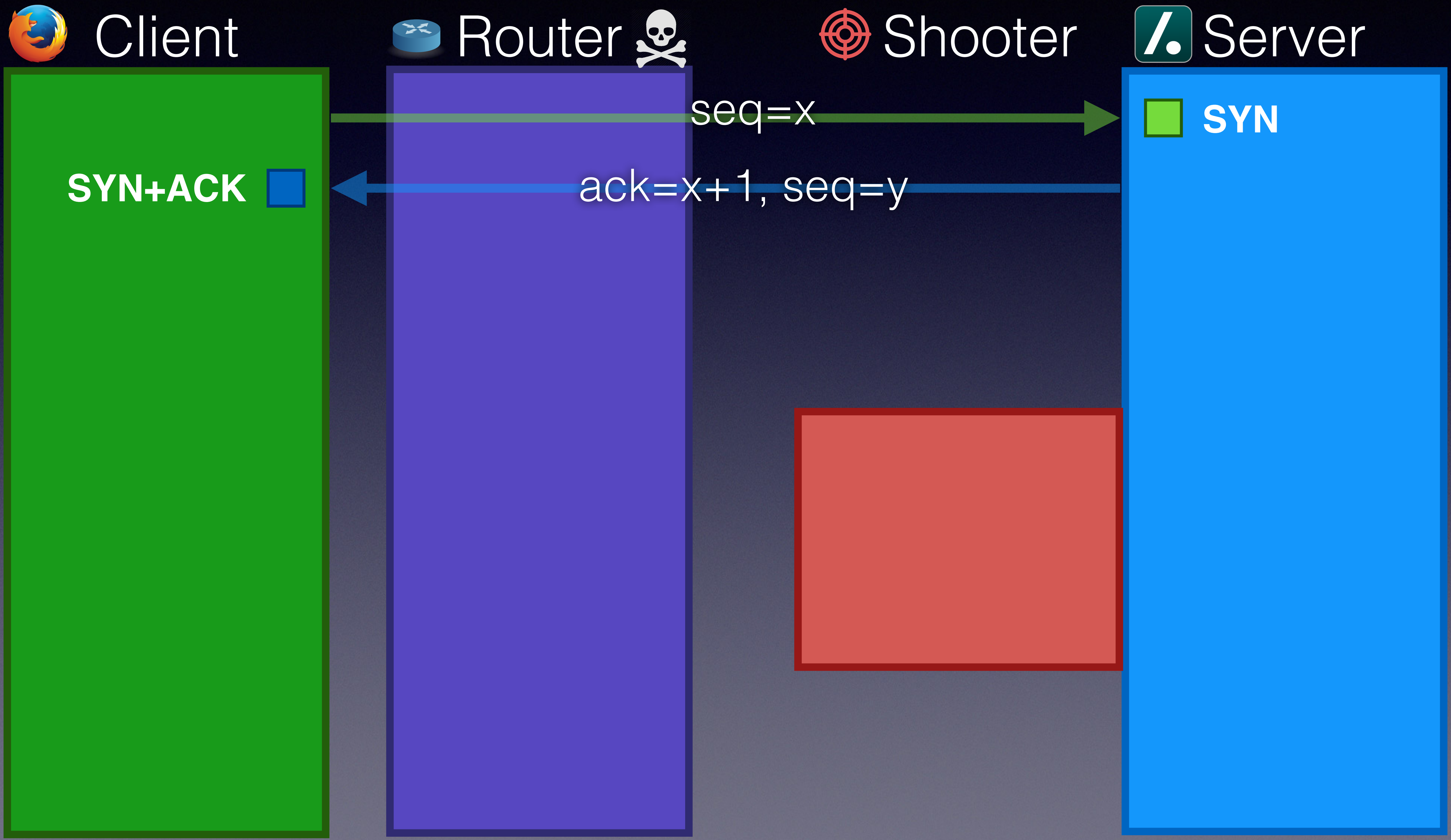
FOX IT

TCP Injection



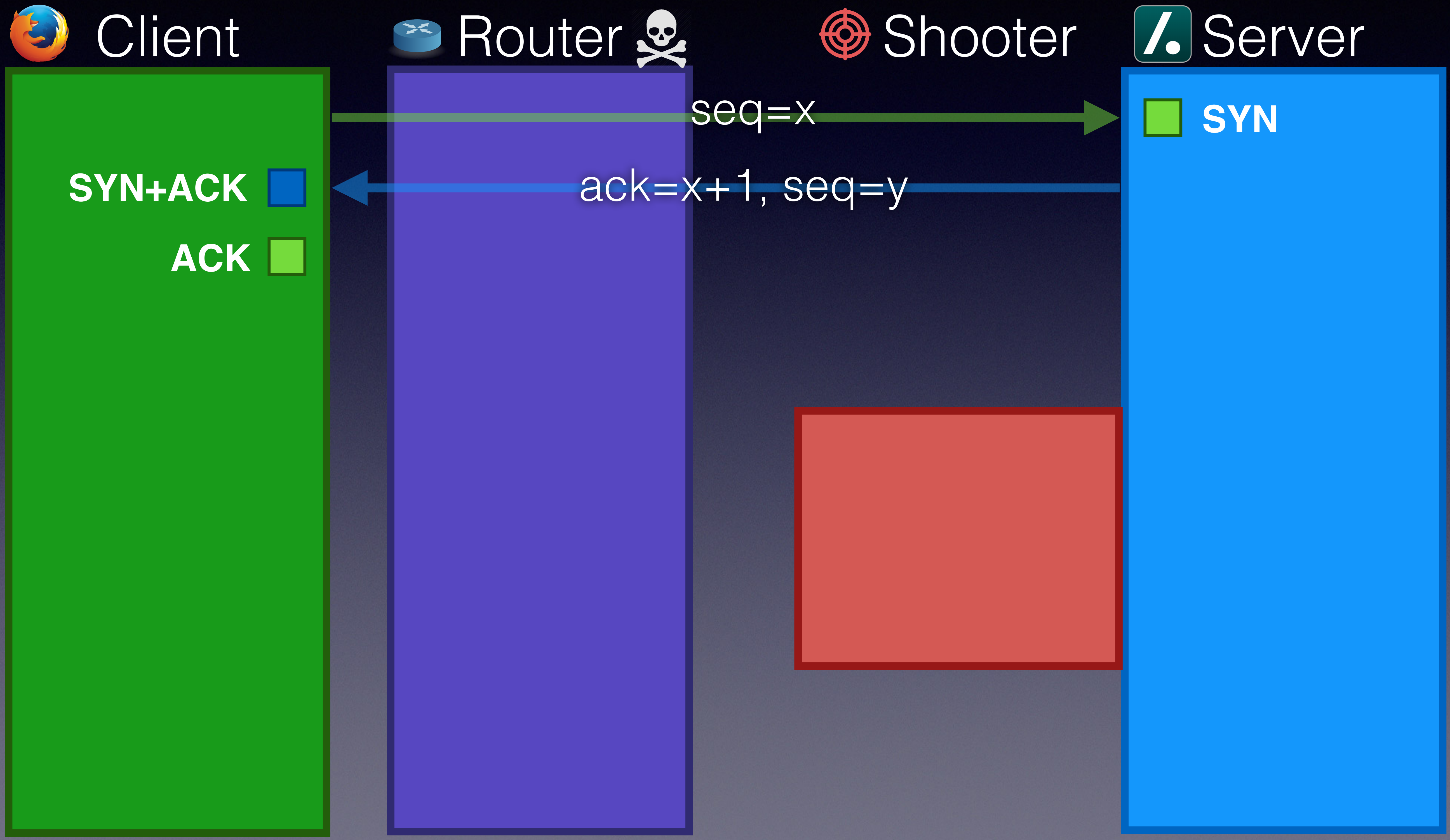
FOX IT

TCP Injection

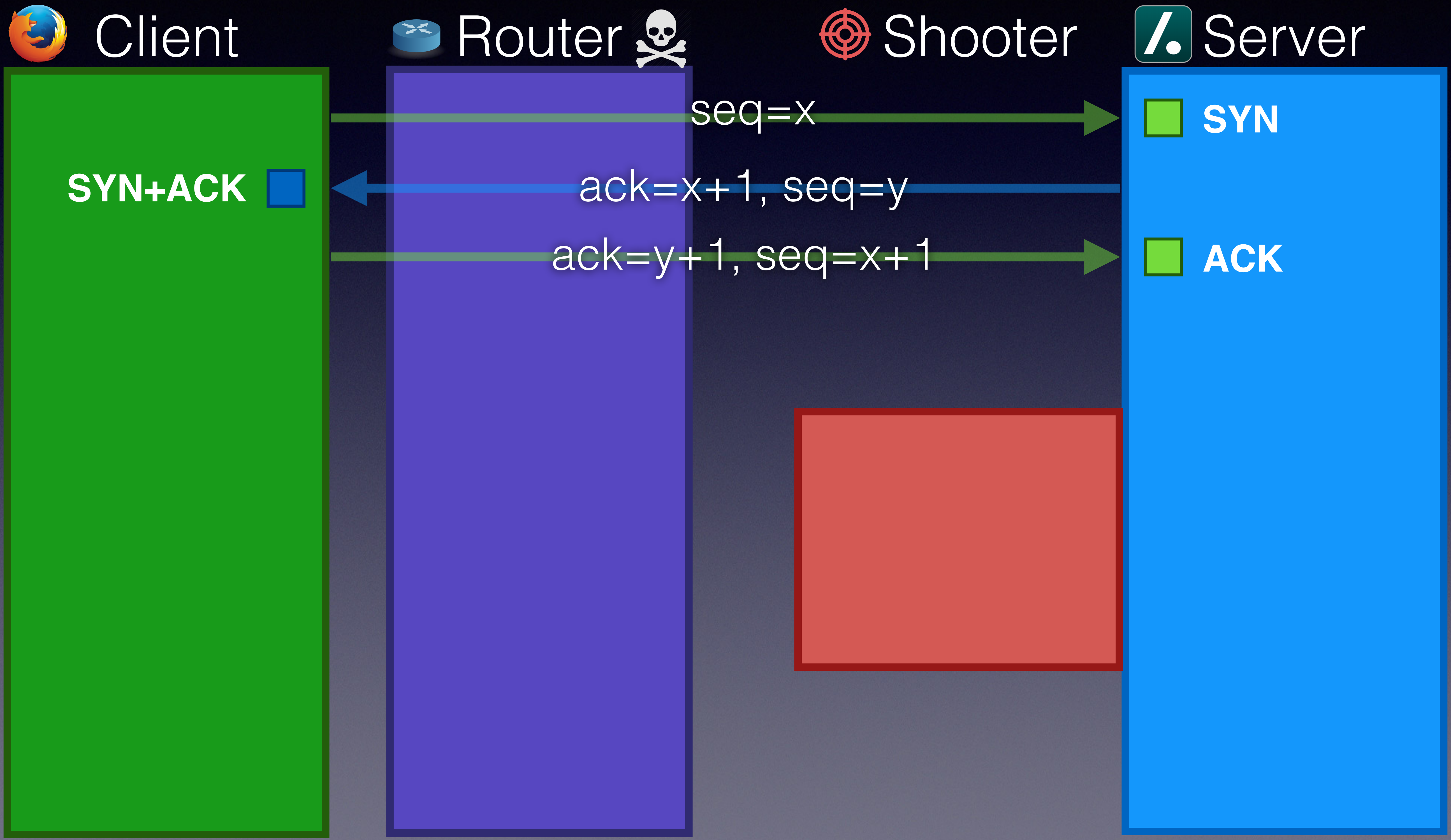


FOX IT

TCP Injection

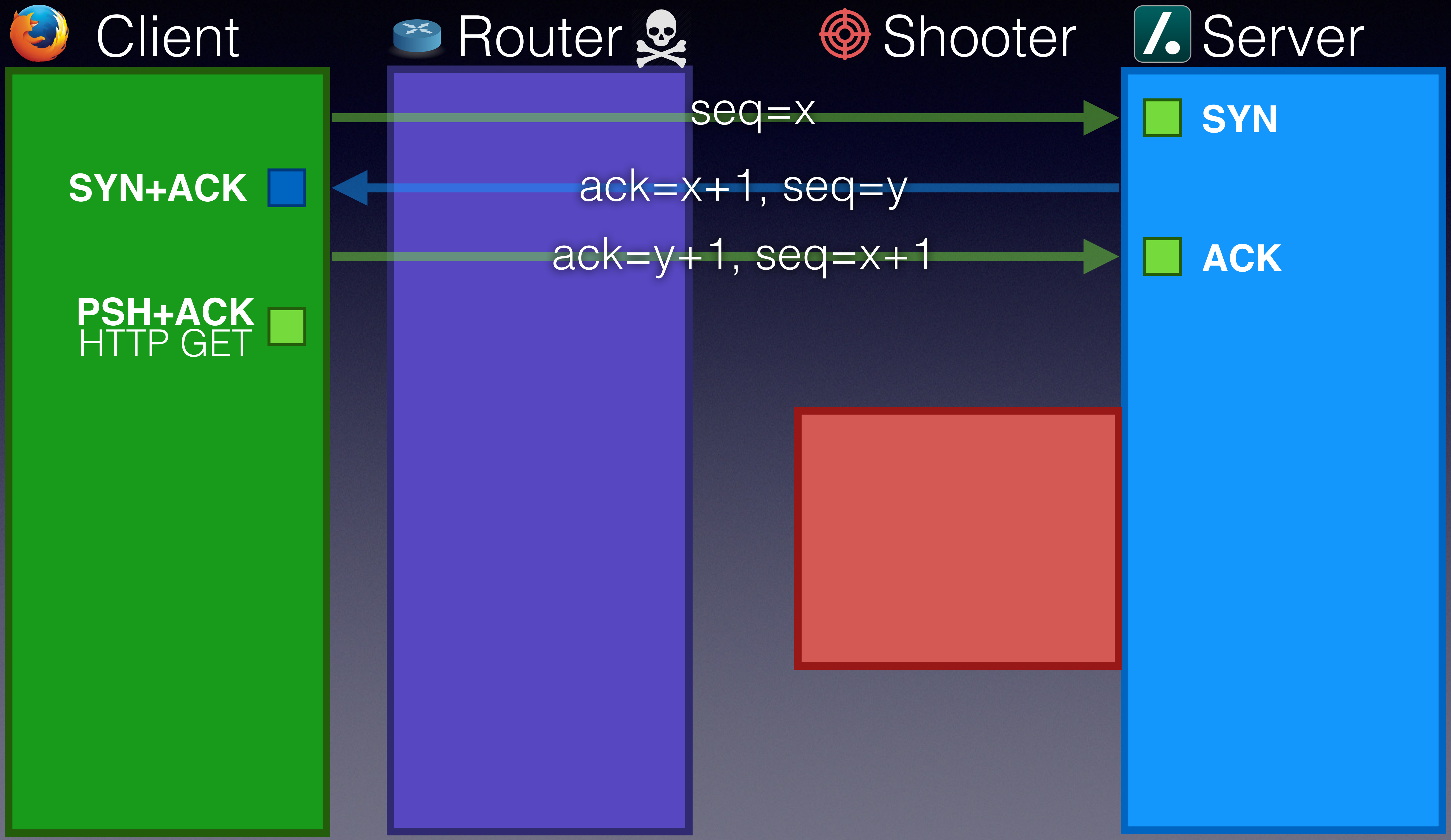


TCP Injection

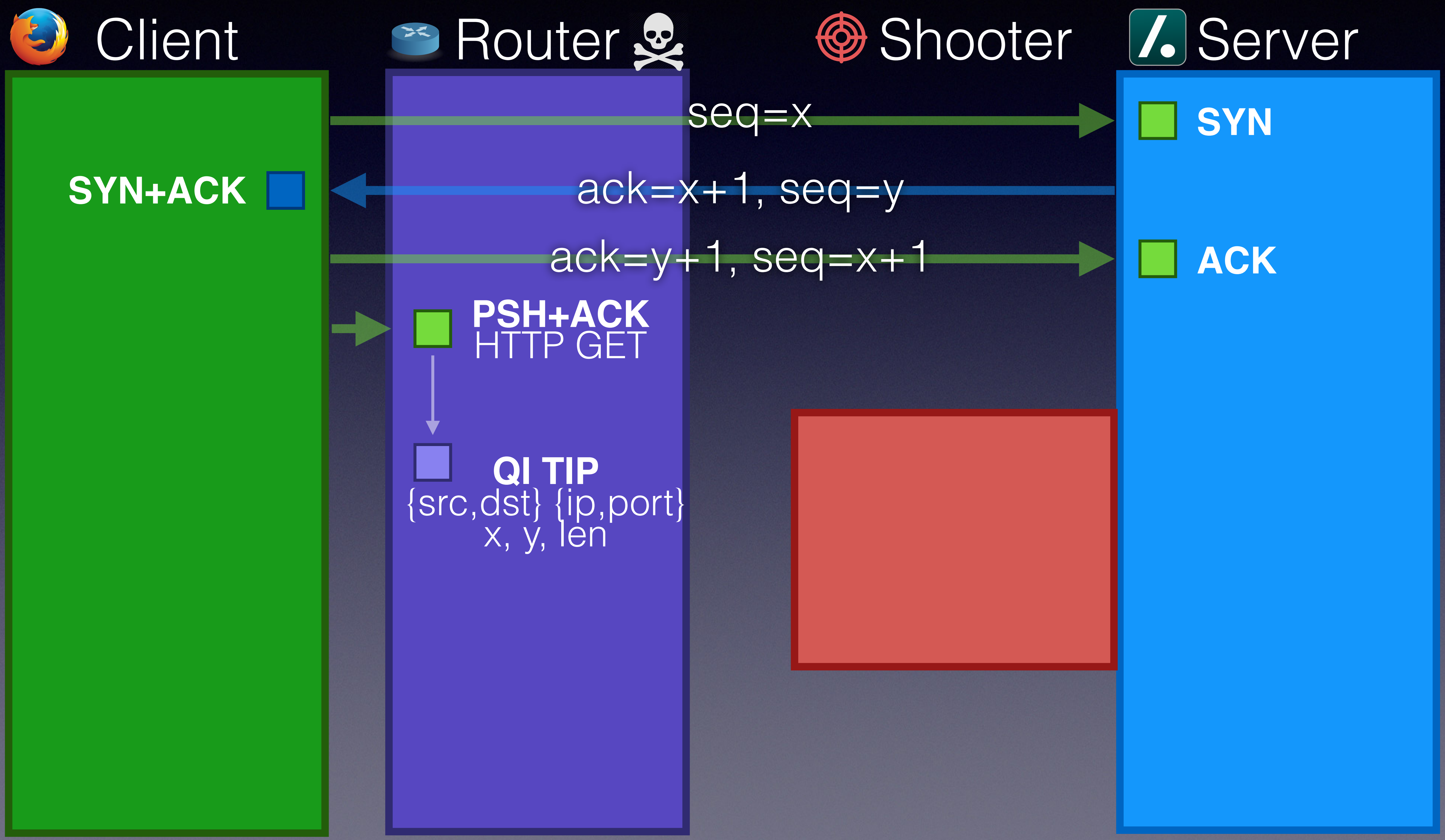


FOX IT

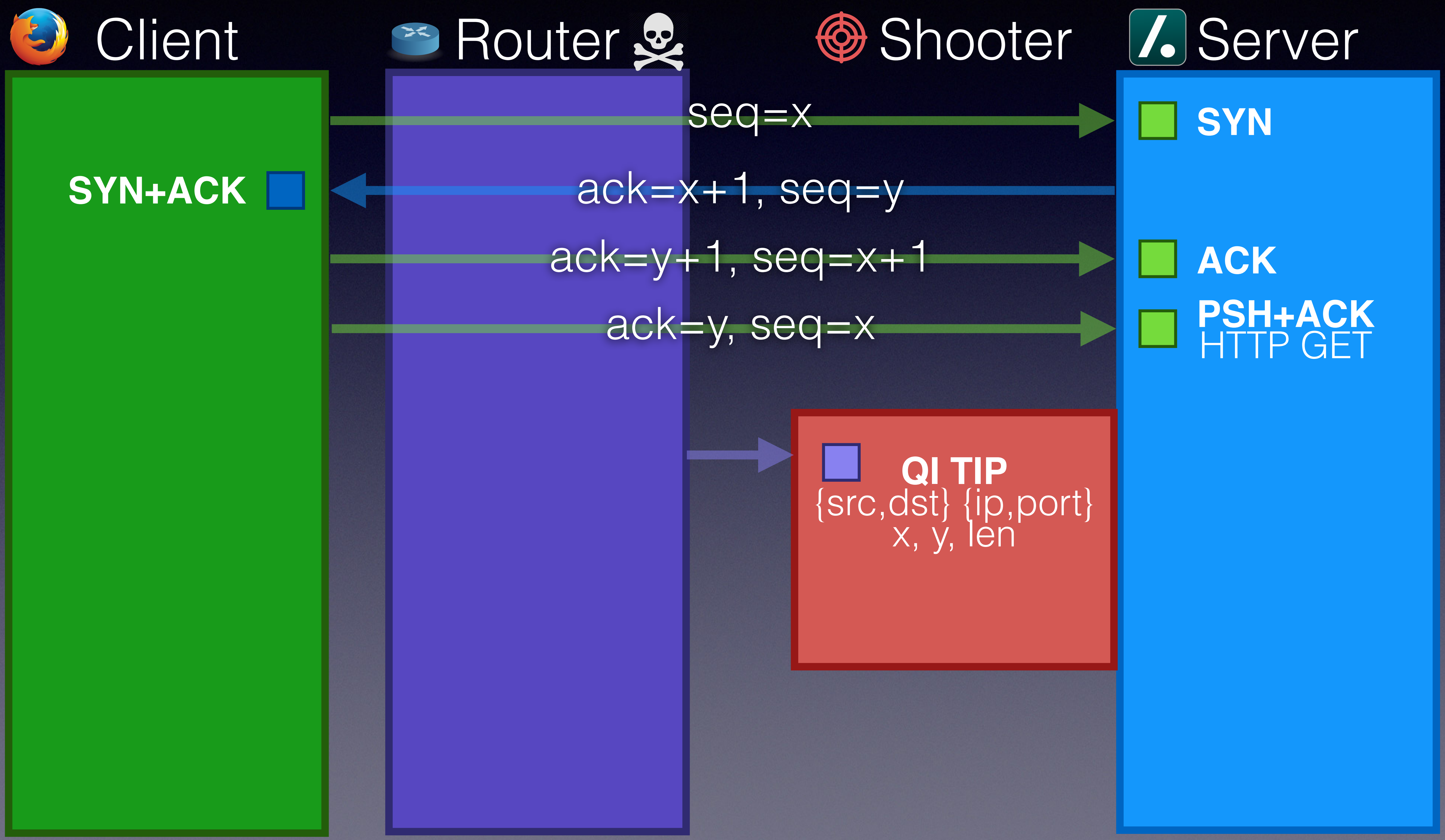
TCP Injection



TCP Injection



TCP Injection



Client

Router ☠

Shooter

Server

seq=x

SYN

SYN+ACK

ack=x+1, seq=y

ack=y+1, seq=x+1

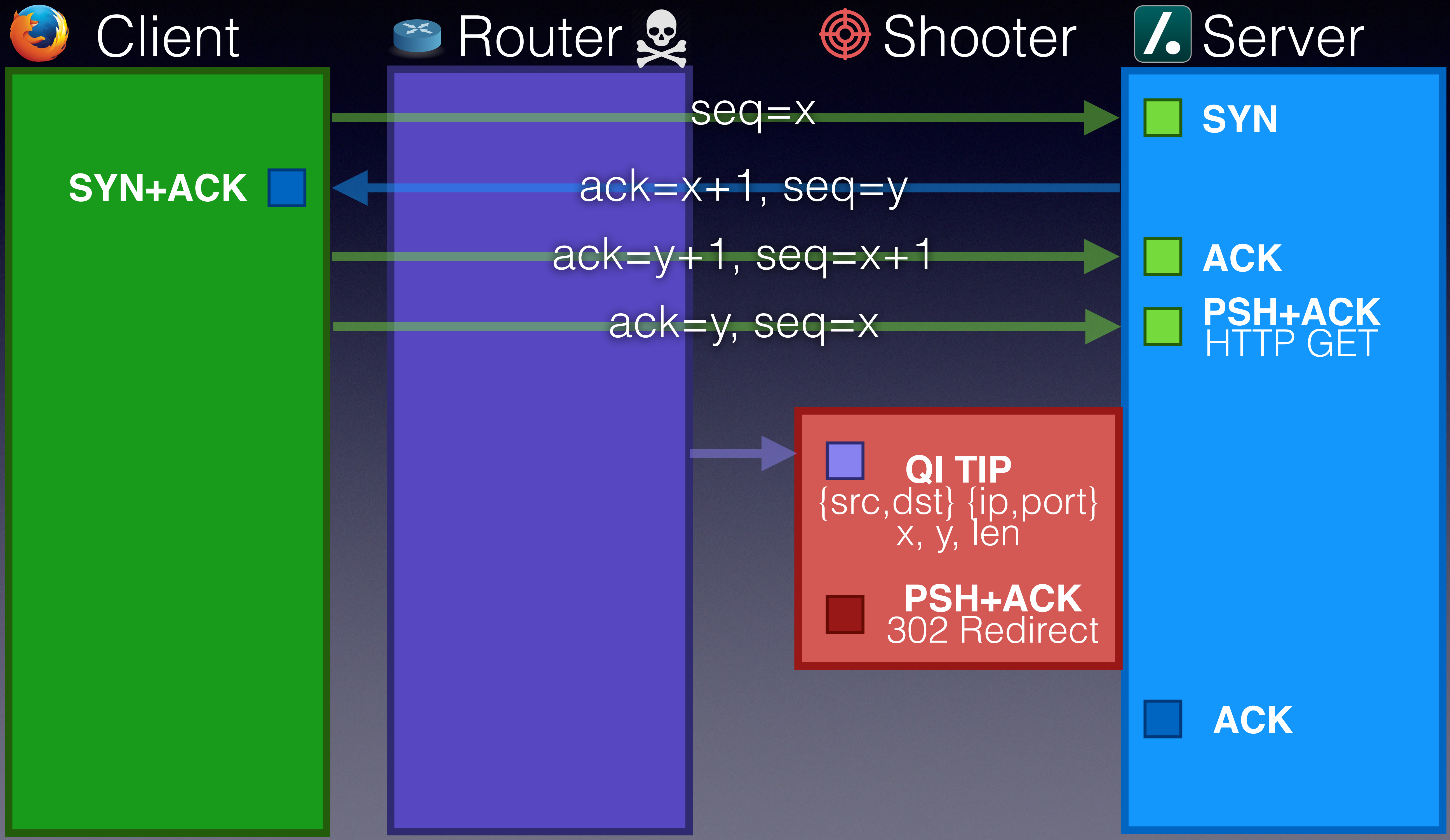
ACK

ack=y, seq=x

PSH+ACK
HTTP GET

QI TIP
{src,dst} {ip,port}
x, y, len

TCP Injection



Client

Router

Shooter

Server

seq=x

SYN

SYN+ACK

ack=x+1, seq=y

ACK

ack=y+1, seq=x+1

PSH+ACK
HTTP GET

ack=y, seq=x

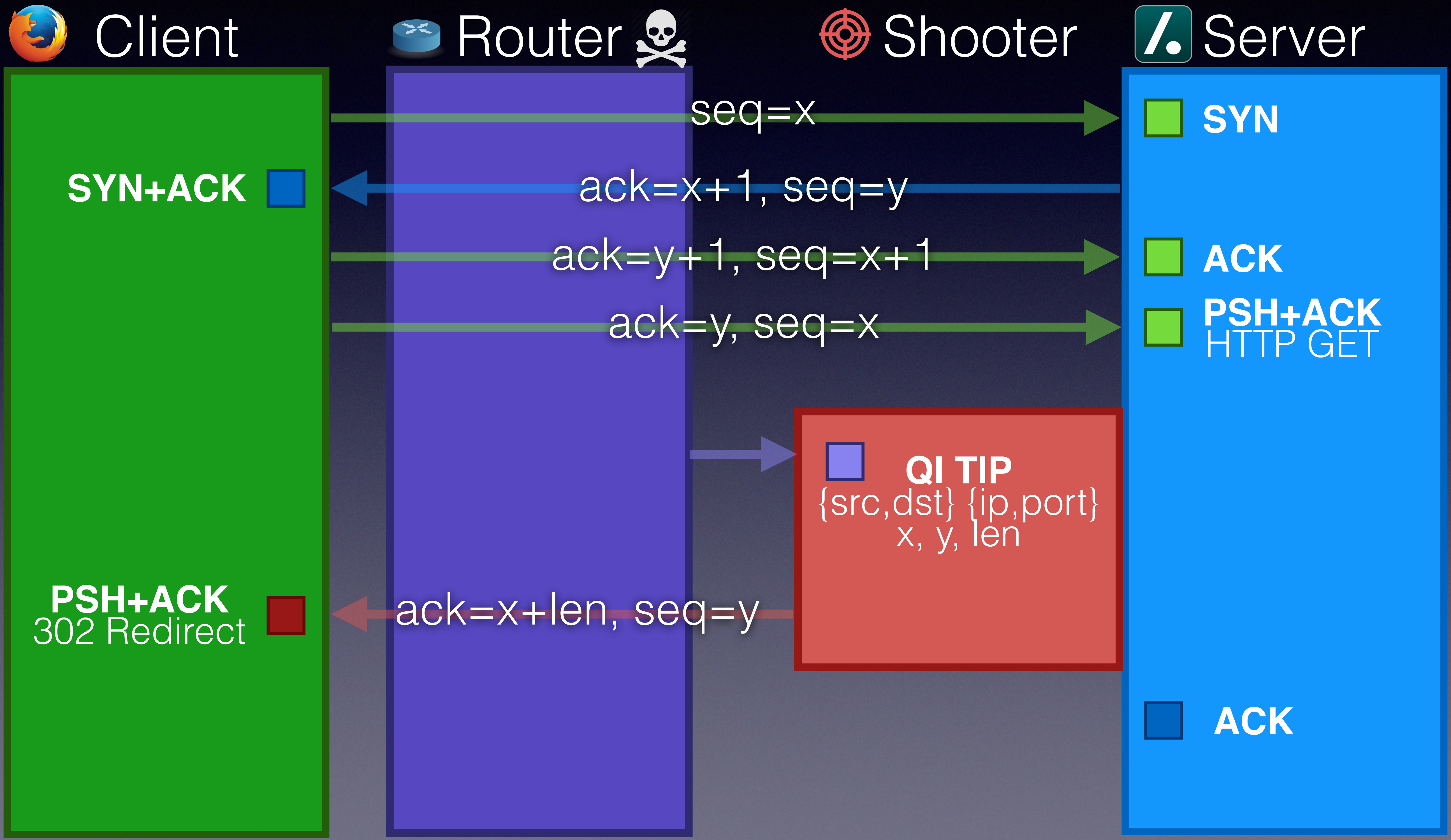
QI TIP
{src,dst} {ip,port}
x, y, len

PSH+ACK
302 Redirect

ACK

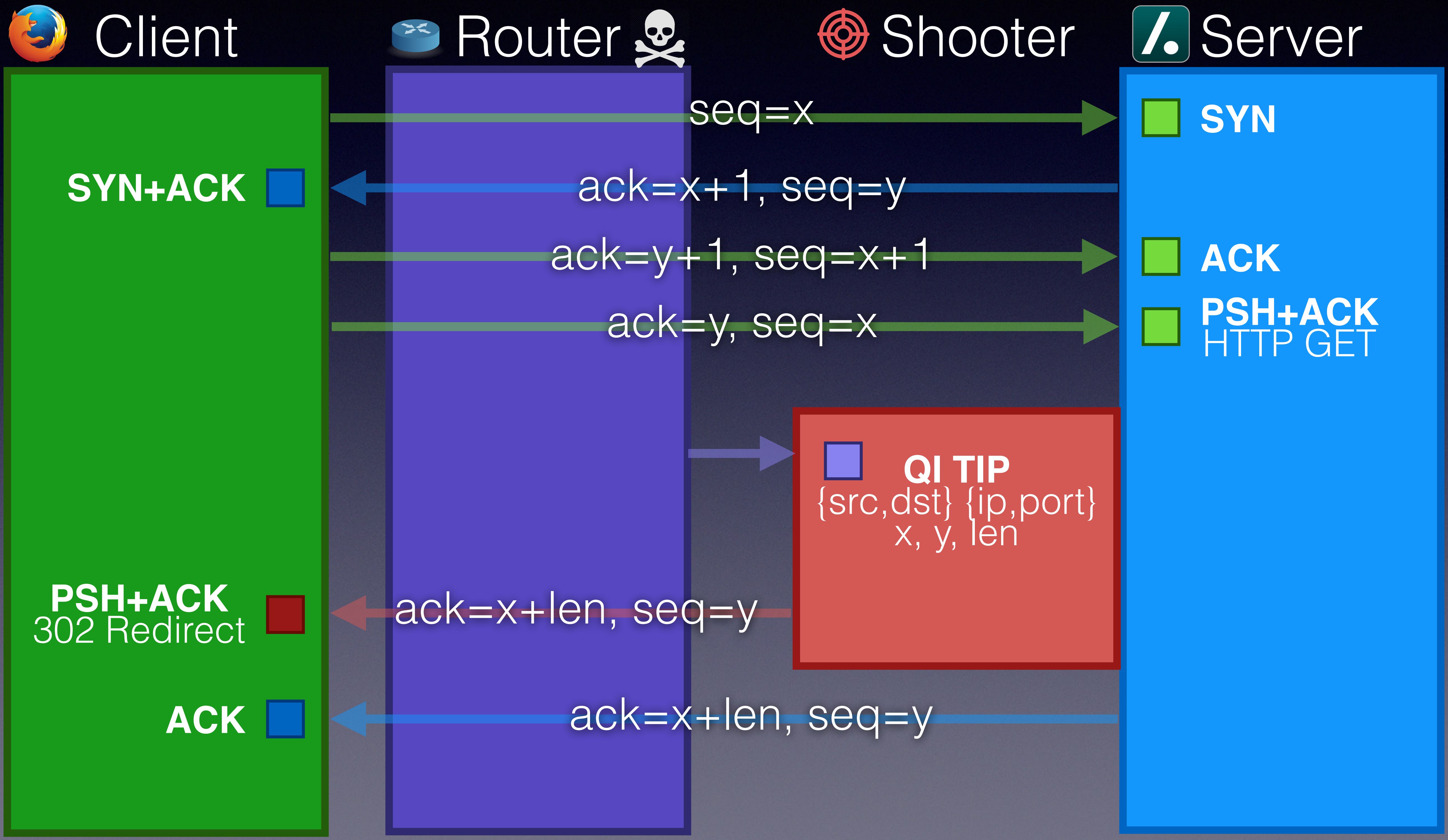
FOX IT

TCP Injection

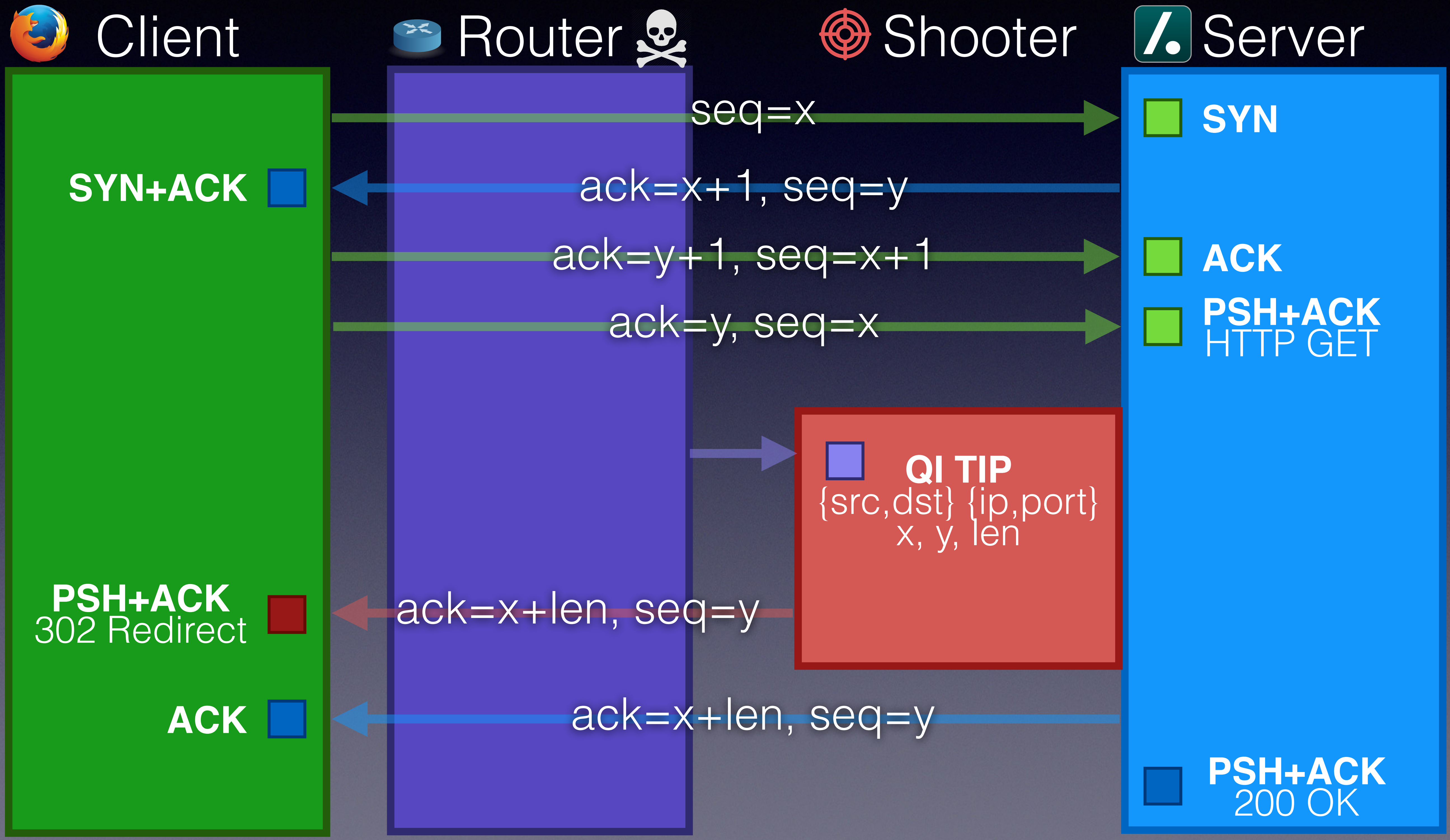


FOX IT

TCP Injection

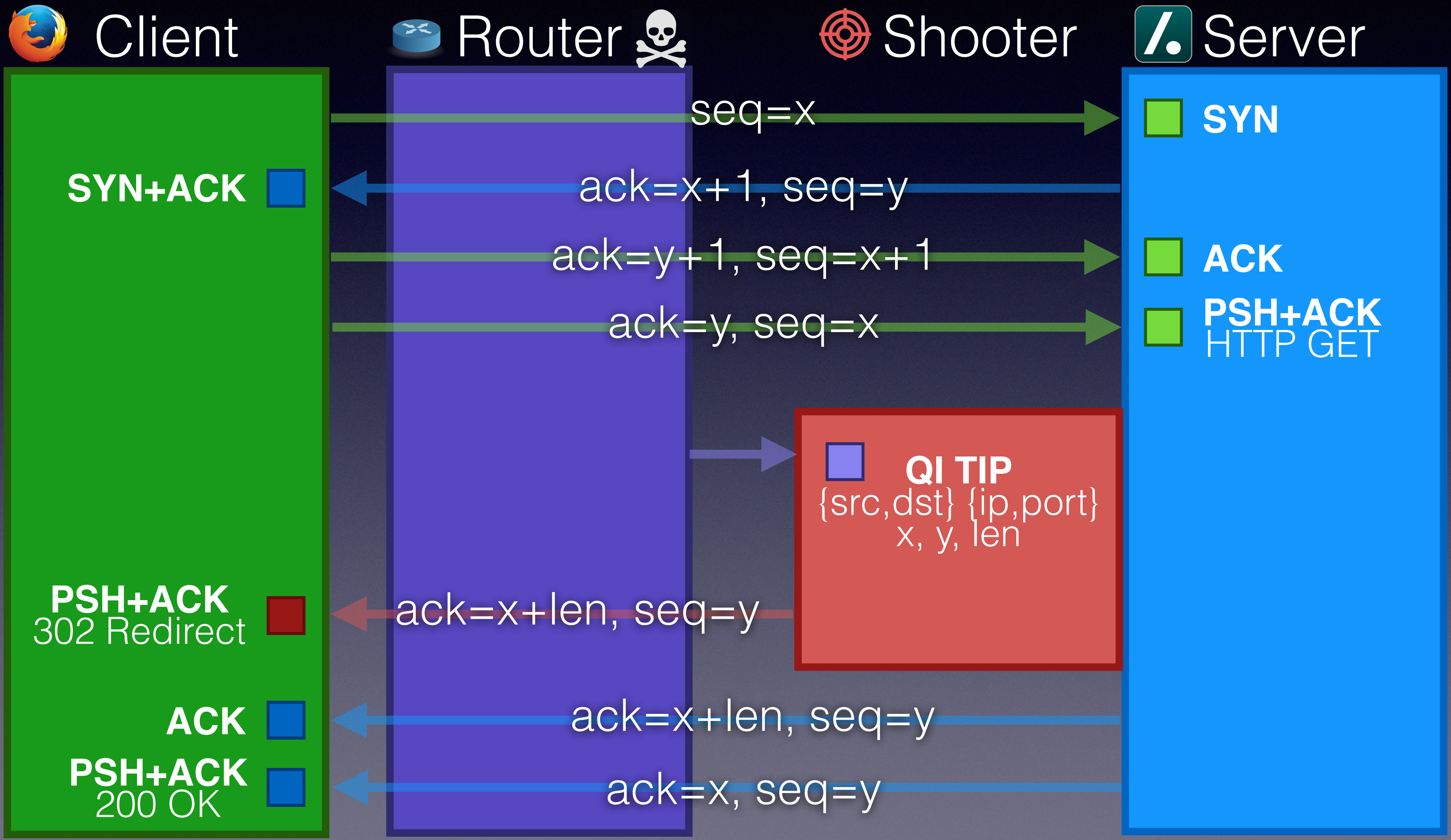


TCP Injection



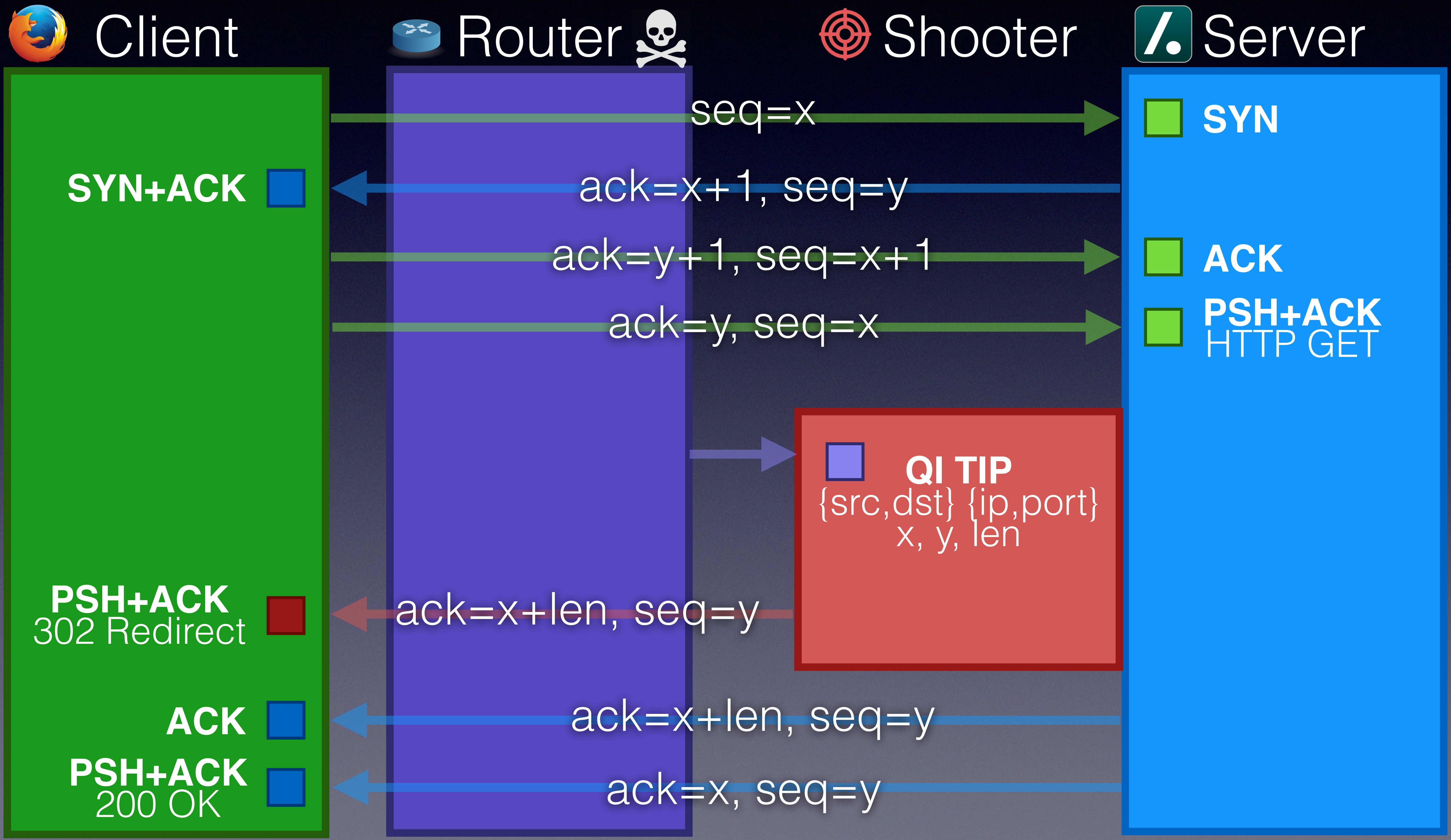
FOX IT

TCP Injection



FOX IT

TCP Injection



TCP segment overlap

- Client receives:
  - Spoofed & Inserted packet
  - Original HTTP response packet
- Attacker can easily solve this, eg by specifying:
  - `Content-Length: 0`

Overlapping TCP segments

```
HTTP/1.1 302 Found
Location: http://fox-it.com/
Content-Length: 0
```

Packet #1 - Sequence 1 (Length 71)

Overlapping TCP segments

```
HTTP/1.1 302 Found
Location: http://fox-it.com/
Content-Length: 0
```

Packet #1 - Sequence 1 (Length 71)

```
Last-Modified: Tue, 21 Apr 2015 19:16:41 GMT
Connection: close
ETag: "5536a219-1caf5"
Accept-Ranges: bytes
Vary: Accept-Encoding, User-Agent
Content-Encoding: gzip
Transfer-Encoding: chunked

6dca …
```

Packet #2 - Sequence 1 - (Length 1448)

Overlapping TCP segments

```
HTTP/1.1 302 Found
Location: http://fox-it.com/
Content-Length: 0

Last-Modified: Tue, 21 Apr 2015 19:16:41 GMT
Connection: close
ETag: "5536a219-1caf5"
Accept-Ranges: bytes
Vary: Accept-Encoding, User-Agent
Content-Encoding: gzip
Transfer-Encoding: chunked

6dca …
```

Reassembled Data

Getting more speed

- Injecting on the first SYN-ACK response from the Server
    - Improved speed
    - But cannot confirm request/victim

FOX IT

Detecting Quantum Insert

How to detect QI

- QI results in duplicate sequence numbers
  - Which means TCP segment overlap
  - Check if overlapping segments are different

Other packet artefacts

- Time to Live usually differs from other packets
- Can give away where in the chain the packets are being injected
- Could have different TCP options

Bro policy

- Uses `tcp_packet` callback
- keeps track of the last sequence number and payload of a connection
- check for duplicate sequence numbers
    - check for payload difference
- Inefficient but works

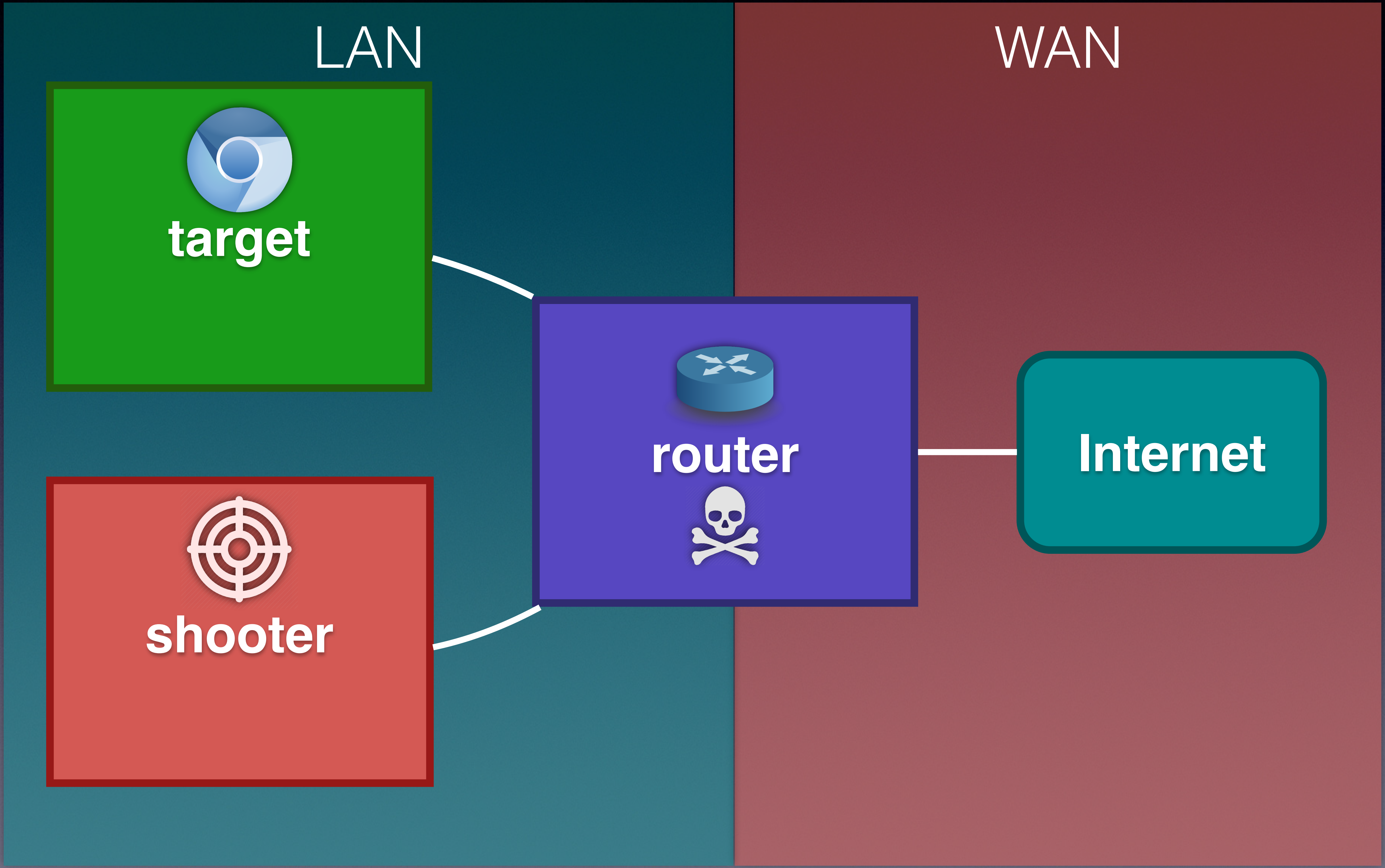
Bro patches

- Integrated in the TCP Reassembly code
- Rolling buffer of old segments, configureable using `tcp_max_old_segments`
- Overlapping segments with different data will trigger the `rexmit_inconsistency` event
- Merged in commit c1f060be on June 28 2015

Demo

FOX IT

Demo Setup



TCP Injections in the wild

Examples of detected QI

- Network Appliances performing TCP injection
  - Blocking content, such as ads
- Some Chinese websites result in TCP injection
  - Mostly for blocking purposes

False positives?

- SSL Traffic
- Window size changes
- Recommendations:
    - Ignore SSL/TLS
    - Limit to HTTP responses

Research

- All the research, pcaps, and tools are published on our GitHub and blog:
  - https://github.com/fox-it/quantuminsert
  - blog.fox-it.com/2015/04/20/deep-dive-into-quantum-insert/

Recommendations

- As a server
    - Use SSL + HTTP Strict Transport Security
    - Resources should be over SSL as well
- As a client
    - Use https directly, don’t rely on redirects
    - Isolated VM for browsing only

Questions?

FOX IT

FOR A MORE SECURE SOCIETY

Bonus Bro policy!

- `meterpreter.bro`
  - Detect Metasploit meterpreter payload transfer
  - Nice for lateral movement detection!
  - Uses sequence numbers to check the size
- Will be available after the talk:
  - https://github.com/fox-it/bro-scripts