Blind In/On-Path Attacks
and Applications to VPNs

William J. Tolley†‡, Beau Kujath†‡, Mohammad Taha Khan§, Narseo Vallina-Rodriguez¶£, and Jedidiah R. Crandall†‡

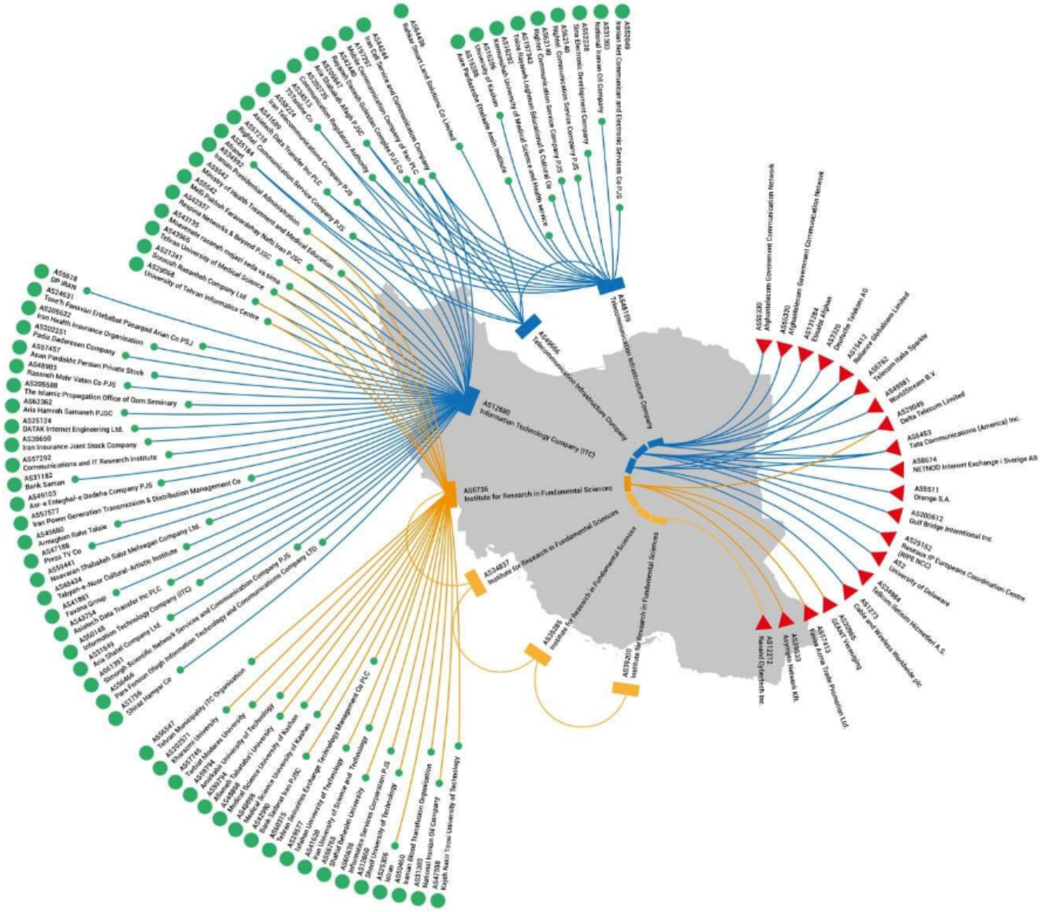
Arizona State University†, Breakpointing Bad‡,
Washington & Lee University§,
IMDEA Networks Institute¶, International Computer Science Institute£

Research question

Do VPNs (and related technologies such as Psiphon, Orbot, *etc.*) protect the connections tunneled through them from inference, interference, and hijacking?

- Public Wifi
- State-controlled cell tower
- In-path state-controlled ISP

In-path state-controlled ISP



Reproduced and cropped from https://www.article19.org/ttn-iran-november-shutdown/

Attacker with *.facebook.com SSL/TLS cert

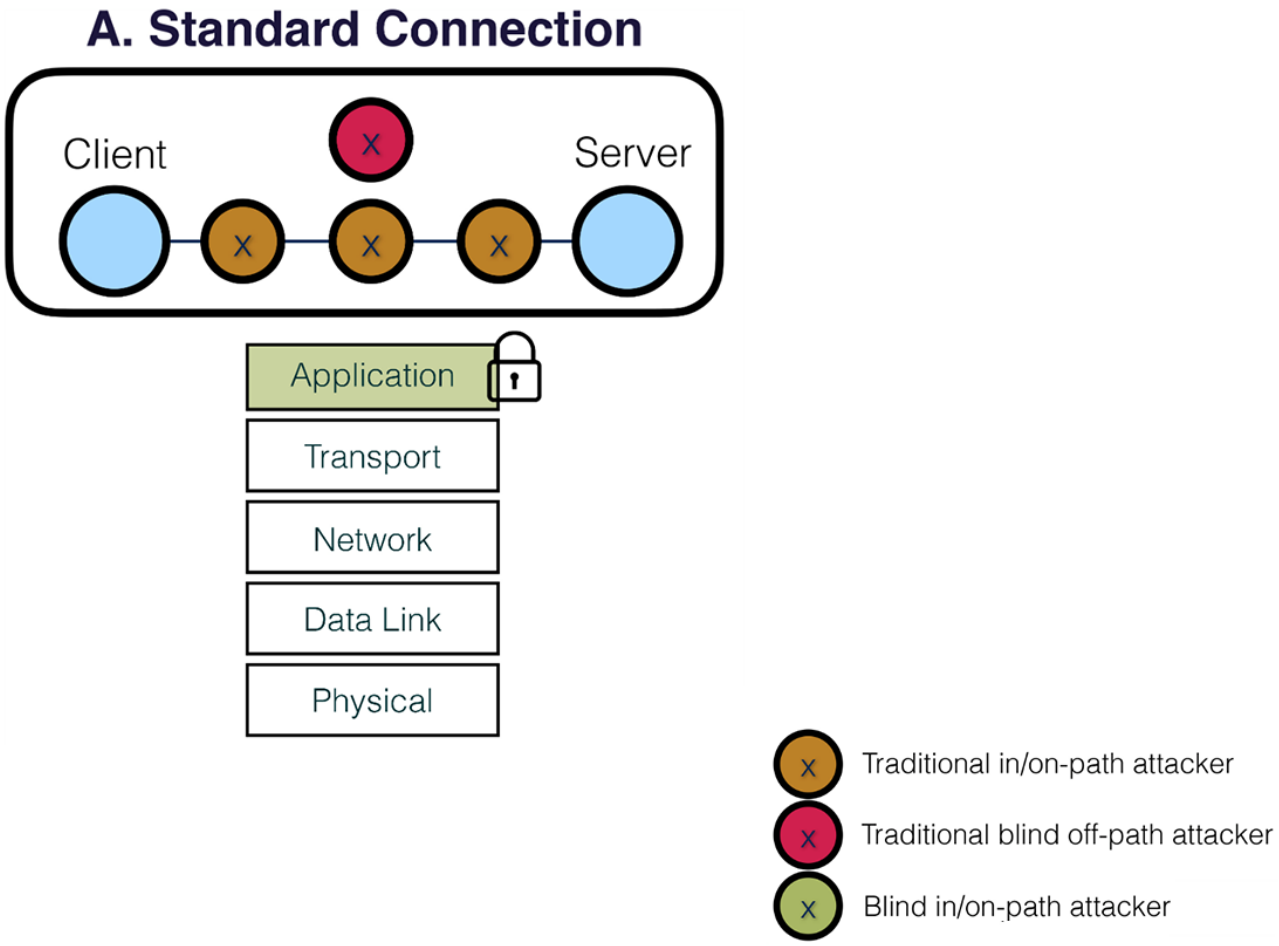
[protected] from Tehran, IRAN, CC BY-SA 2.0 https://creativecommons.org/licenses/by-sa/2.0, via Wikimedia Commons

(https://commons.wikimedia.org/wiki/File:Iran_election_(2).jpg)

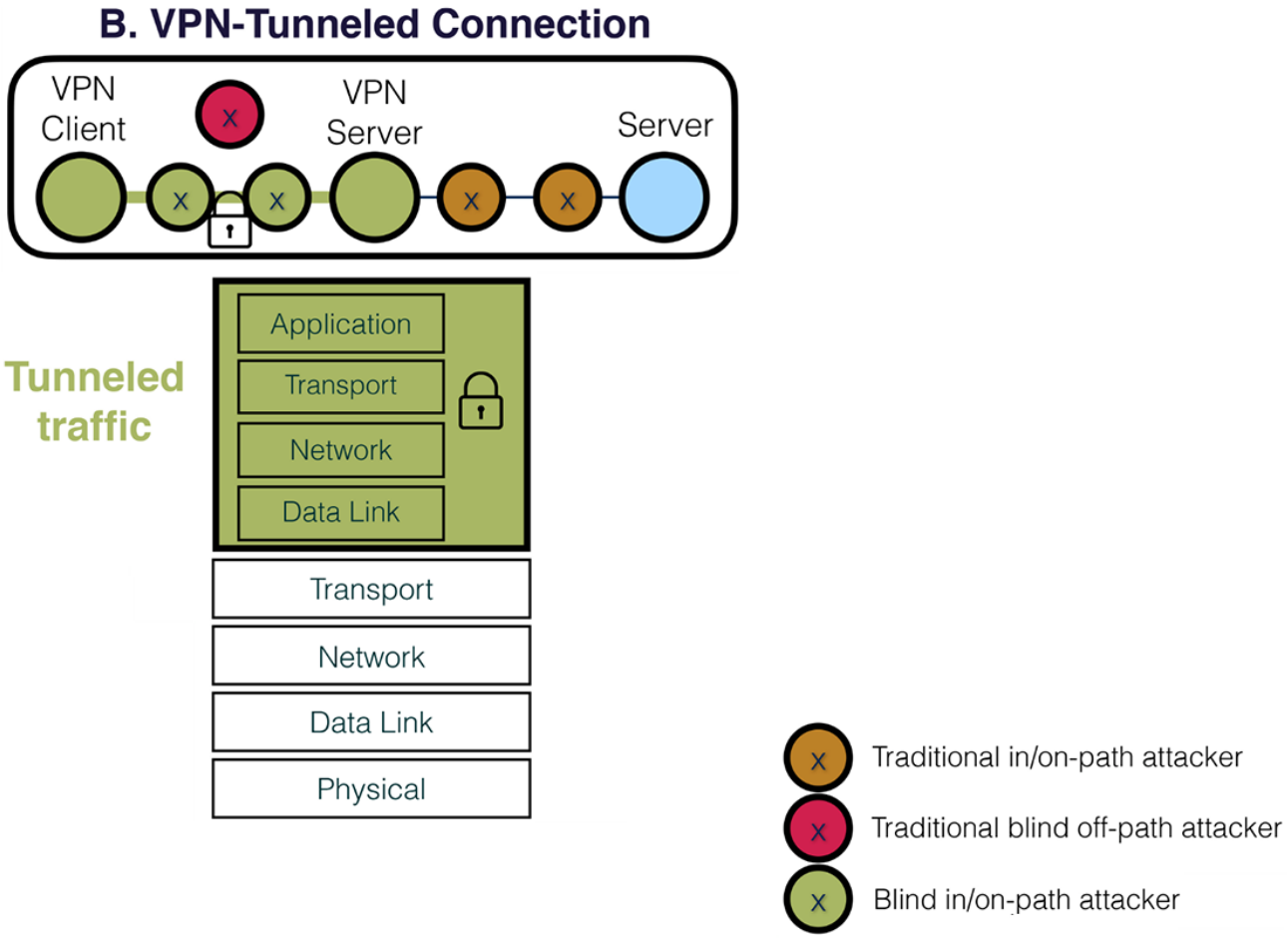
What if the Facebook users in Iran in 2009 had all used TLS and a VPN?

E.g. the latest version of WireGuard from May, 2021

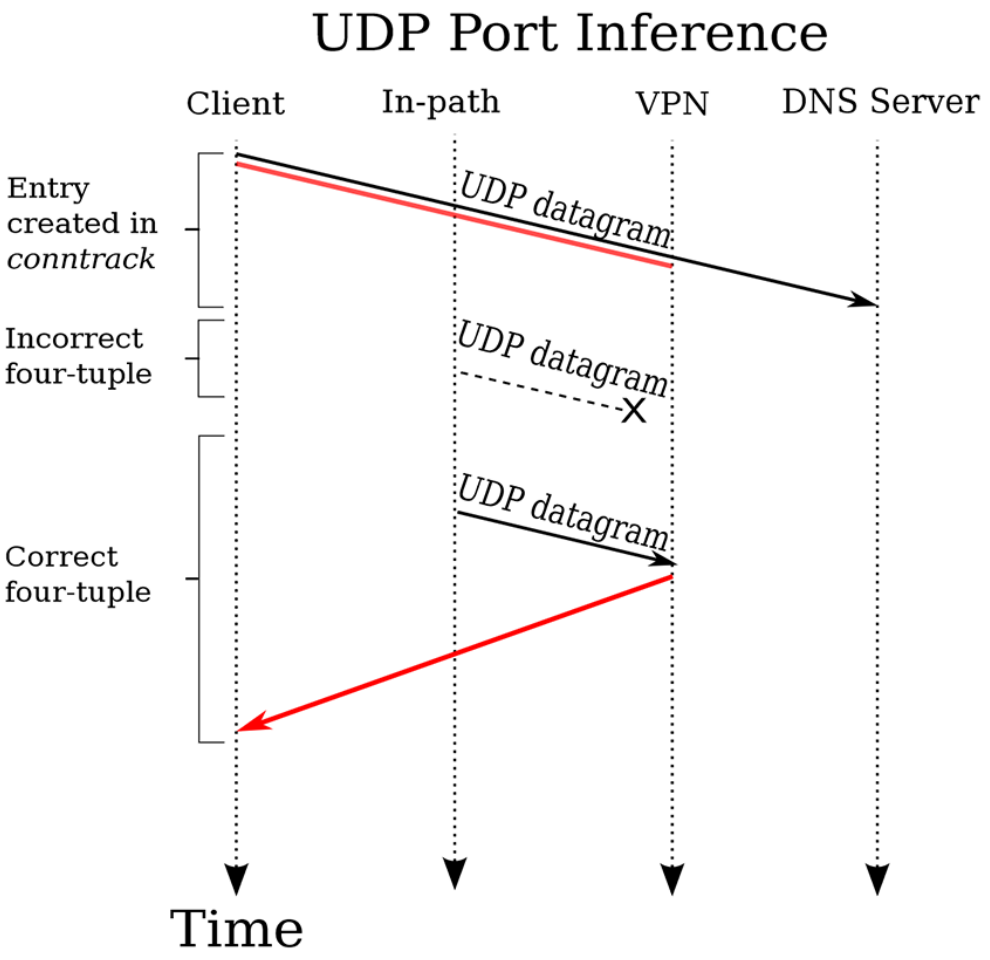
Need for new terminology



New terminology: *Blind In/On-Path Attacker*



Server-side attack on DNS over UDP



IP	UDP			DNS		
...	...	dst port	...	...	TXID	...

- Off-path attacker
  - $2^{16} \times 2^{16} = 2^{32}$, ☹
- In/On-path attacker
  - $2^{16} + 2^{16} = 2^{17}$
  - 32,768× faster than 2^{32} ☺

Is hijacking DNS practical?

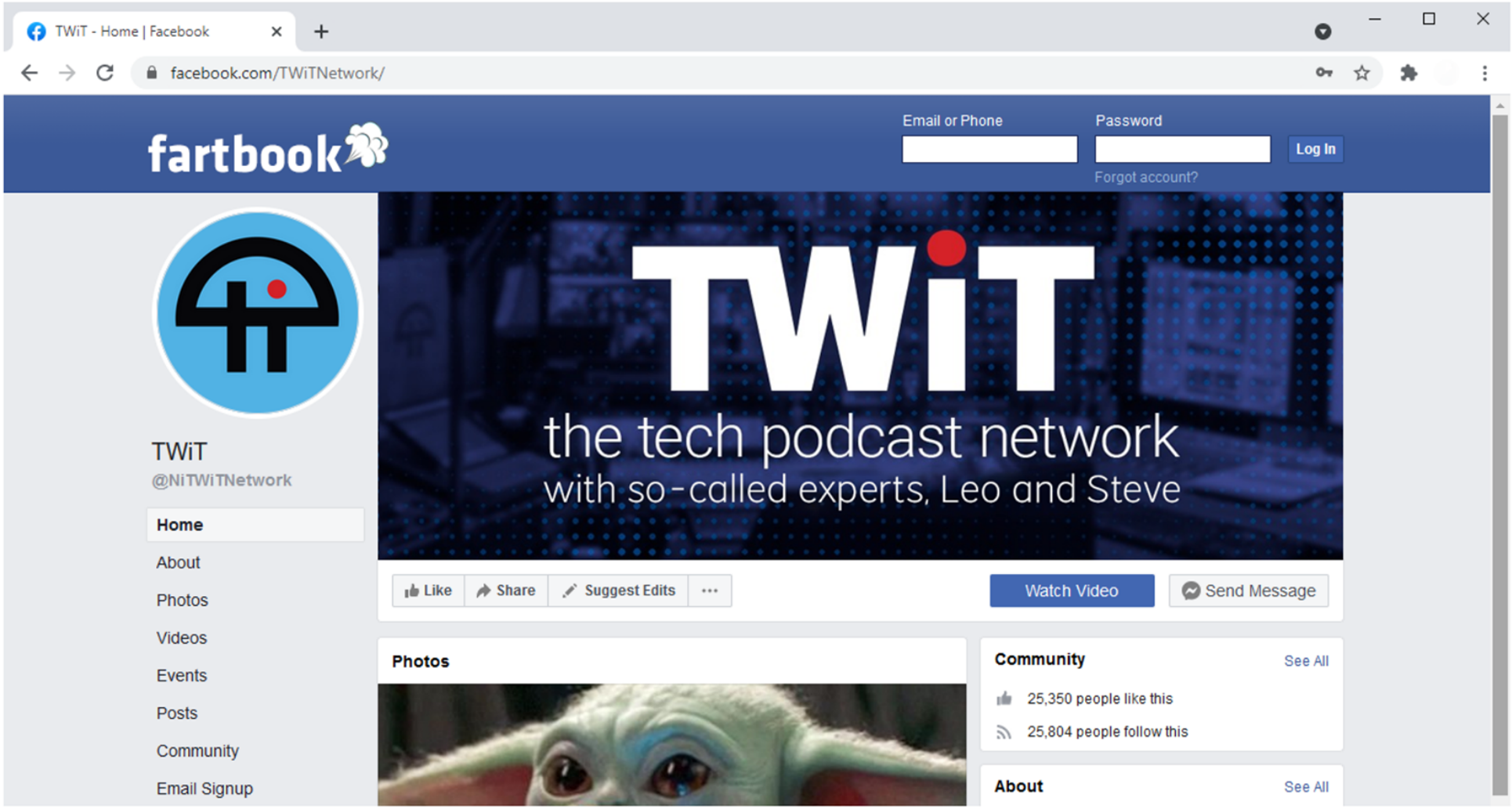
Tested for different DNS timeouts:

- 15 seconds (*e.g.*, Android 11): 75.3% successful
- 10 seconds (*e.g.*, Ubuntu 20.04): 48.1% successful
- 5 seconds (*e.g.*, Firefox 80.0.1): 11.6% successful

The timeout of DNS queries is controlled by applications

Falls back to system's default settings when unspecified

Man-in-the-middle despite TLS and VPN



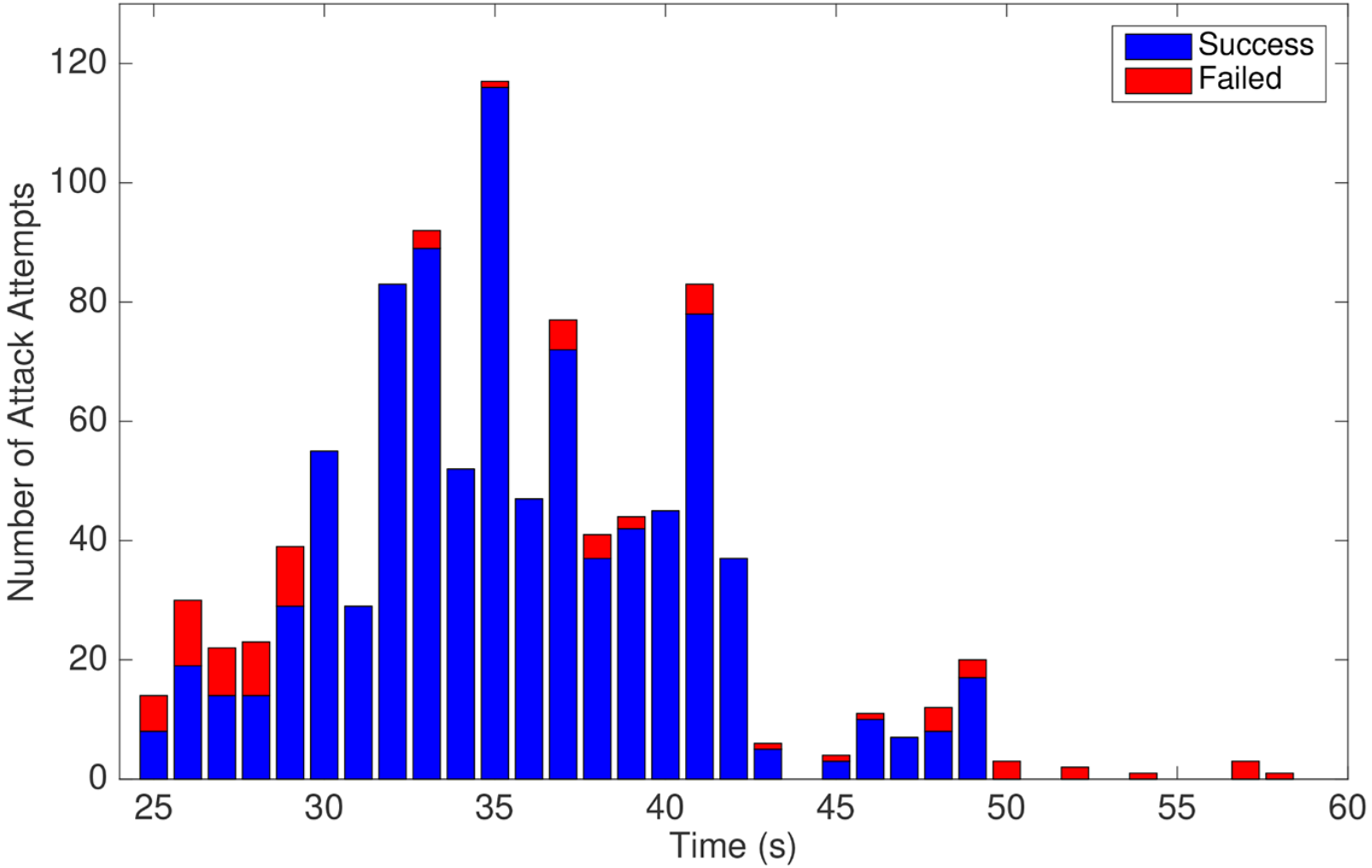
Client- *vs.* server-side attacks

- We also did *client-side attacks*
  - Infer that a client is connected to a VPN
  - Infer the existence of TCP connections in the VPN tunnel
  - Reset or even hijack active TCP connections
- The DNS over UDP attack you just saw is *server-side*
  - Interface and all packet fields are identical for attack *vs.* legitimate traffic
  - It's also possible to do any of our TCP attacks above server-side

Disclosure and mitigation

- Ethical Disclosure
  - CVE-2019-9461
  - CVE-2019-14899
  - Correspondence with Linux kernel developers
- Mitigation
  - *Client-side **mitigated** by many vendors by distinguishing the interface*
  - *Server-side totally **unmitigated** by any vendor despite ethical disclosure*

Client-side results



Future work

- Have client-side attacks actually been mitigated by vendors?
- How practical are server-side attacks for a real ISP?
- Can we detect and prevent server-side attacks?
- What about things like Shadowsocks?
- What about padding, *etc.*?
  - *e.g.*, obsfproxy
- What else can go wrong when you stack layers of abstraction on top of each other and encrypt them?

Conclusion

- You can encrypt your packets, but you can't hide their existence, timing, or size
- Blind in/on-path attackers should be considered when designing any protocols that might be tunneled (*e.g.*, in a VPN)

Thank you!

- Contact: william@breakpointingbad.com
- Artifact: https://git.breakpointingbad.com/Breakpointing-Bad-Public/vpn-attacks

- This material is based upon work supported by the U.S. National Science Foundation under Grant nos. 1518523, 1518878, 1801613, and 2007741, as well as the Open Technology Fund and the Ministry of Science and Innovation (Spain) (PID2019-111429RB-C22).