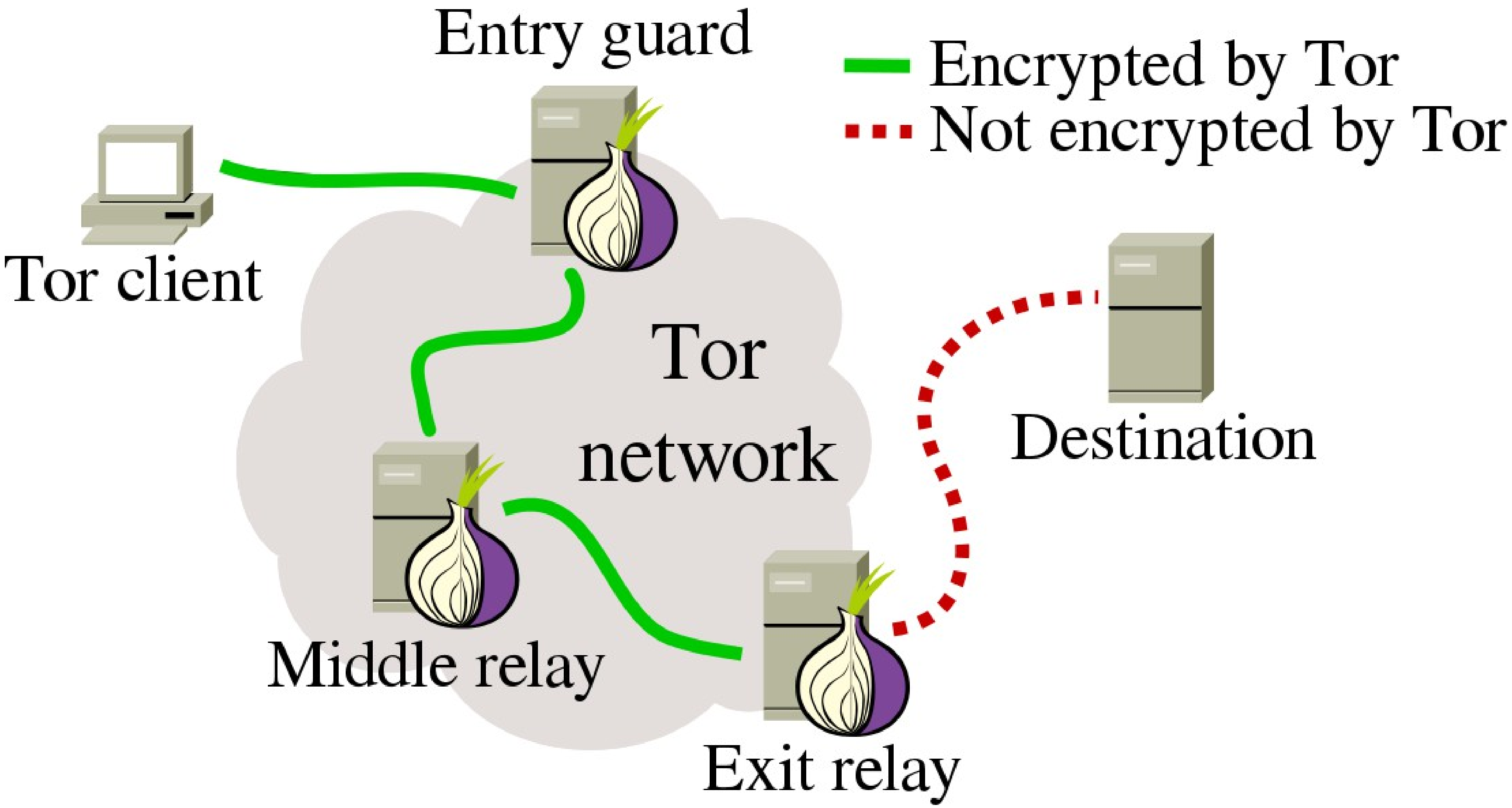
Surfing safely over the Tor anonymity network

TorProject.org

Georg Koppen – gk@torproject.org

Philipp Winter – phw@torproject.org

How does Tor work?



Entry guard

Tor client

Tor network

Middle relay

Exit relay

Destination

🟩 Encrypted by Tor

🟥 Not encrypted by Tor

What are exit relays?

- Currently ~7,000 relays, ~1,000 are exits
- All run by **volunteers**
- Exit relay can be set up in **10 minutes**
- Motivation of operators differs
  - Altruism, research, PR, curiosity, ...
- https://www.eff.org/pages/tor-and-https

What are bad exit relays?

- Good exit relays are like good ISPs
  - **Neutral** to what they relay

- Bad exit relays **manipulate** traffic
  - Misconfigured (AV scanner, OpenDNS, FD limit)
  - Man-in-the-middle attacks
  - Traffic sniffing

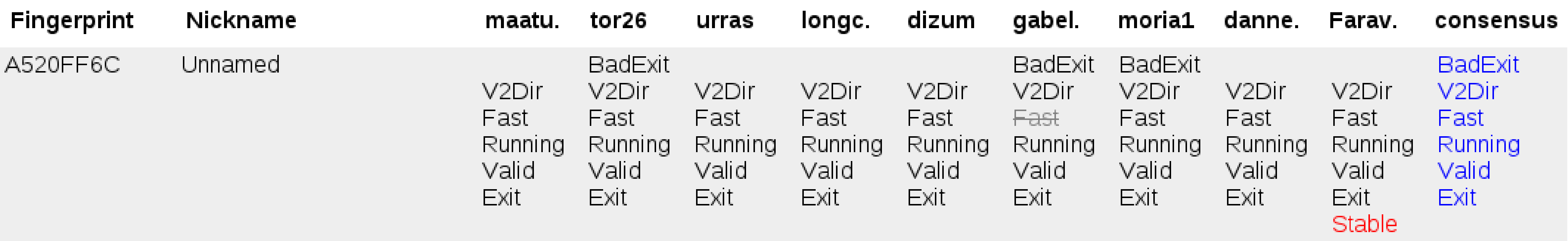
How do we find bad exits?

- Our users often tell us about them
  - Write to bad-relays@lists.torproject.org

- We **systematically scan** the network
  - https://github.com/NullHypothesis/exitmap
  - Looks for common attacks over all exit relays
  - MitM, sslstrip, HTML injection, DNS poisoning, TLS tampering, ...

What happens to bad exits?

- Relays get “**BadExit**” flag
- Clients will no longer select them as exits
- Three out of nine directory authorities “vote”
  - Convince Roger, Sebastian, and Peter
  - **More than 50%** of votes necessary



Fingerprint	Nickname	maatu.	tor26	urras	longc.	dizum	gabel.	moria1	danne.	Farav.	consensus
A520FF6C	Unnamed	V2Dir Fast Running Valid Exit	BadExit V2Dir Fast Running Valid Exit	V2Dir Fast Running Valid Exit	V2Dir Fast Running Valid Exit	V2Dir Fast Running Valid Exit	BadExit V2Dir ~~Fast~~ Running Valid Exit	BadExit V2Dir Fast Running Valid Exit	V2Dir Fast Running Valid Exit	V2Dir Fast Running Valid Exit Stable	BadExit V2Dir Fast Running Valid Exit

Types of attackers

- Mostly **opportunistic** attackers
  - Motivated by curiosity
- Some **targeted** attackers
  - Motivated by financial gain
- Often not clear if attack done by **upstream**

Implications for Tor users

- Probability of encountering a bad exit isn't:

$$\frac{\#\,\textit{bad exits}}{\#\,\textit{good exits}}$$

- Fast relays **more likely** in circuit than slow relays
- Relays **come and go** frequently
- Tor Browser safer than vanilla Firefox

Anecdotes (1/3)

The relay that did HTTPS MitM for Bitcoin sites

Anecdotes (2/3)

The **NSA** has mounted increasingly successful **attacks** to unmask the identities and locations of users of **TOR**.

It has been able to "stain" anonymous traffic as it enters the **TOR** network, enabling the **NSA** to **identify** users as internet exits.

The Washington Post

If you use **TOR** online or even visit their web sites to read about the **TOR** services, there is a good chance your IP address has been collected and stored by the **NSA**...

according to top-secret source code for a program the **NSA** uses to conduct internet **surveillance**.

Wired.com

Anecdotes (3/3)

Chasing a group of Russian relays

The future

- Work on **Sybil attack** detector
  - Helps find “clusters” of similar relays
- Add more **exitmap modules**
  - Any suggestions?
- Better **onion services**
  - If facebook can do it, others can, too

Part 2

Tor Browser

Which browser are we using?

- First only **Torbutton** as Firefox extension
- Tor Browser based on a free browser: Firefox
- Using Chromium is **blocked**

https://trac.torproject.org/projects/tor/wiki/doc/ImportantGoogleChromeBugs

Did you really get Tor Browser?

- Download over **HTTPS**

- GPG-signed bundles

- Certificate authority pinning for updater

- **Deterministic builds** for Windows, OS X, and Linux

Tor Browser: Key features

- Self-contained “portable” app
- **No disk activity records** by default
- Third Party tracking prevention
- Browser **fingerprinting defenses**
- Traffic obfuscation/Censorship circumvention
- Browser security enhancements

Tor Browser: Components

- Firefox ESR
- Tor
- TorLauncher
- Torbutton
- HTTPS-Everywhere
- NoScript
- Pluggable Transports

Tor Browser: Philosophy

- Preserve existing user model
- Favor the implementation mechanism least likely to break sites
- Plugins must be **restricted**
- Minimize Global Privacy Options
- No filters
- Stay **up-to-date**

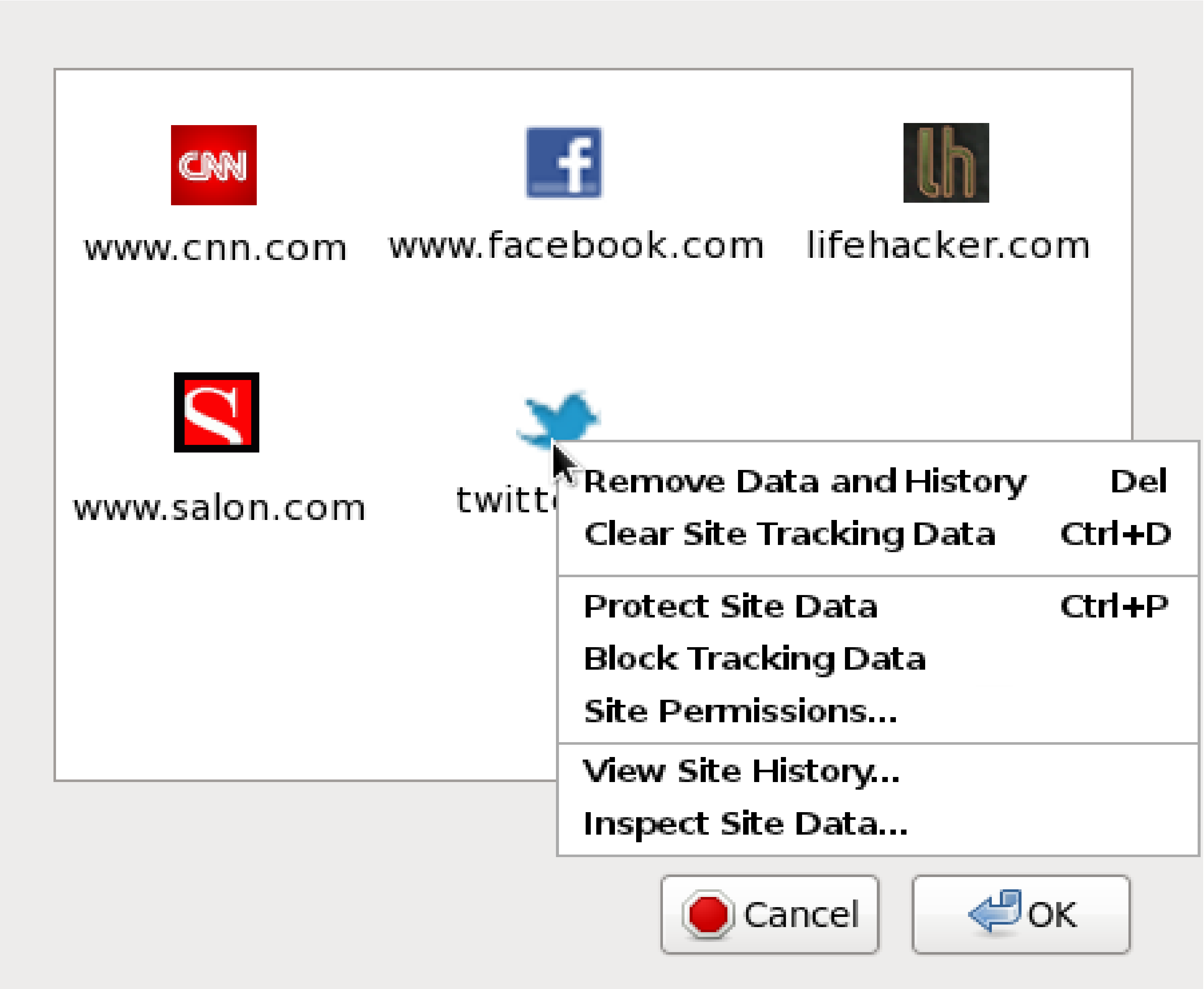
Tracking Protection

Goal: **All identifiers are bound to the URL bar domain**

This means:

- Cache state, {cookies}, DOM Storage, HTTP Authentication, TLS session Ids (+ resumption), {HSTS cookies}... used on foo.com should not be available on bar.com
- If binding to the URL bar domain is not possible (e.g. Flash cookies) we try to disable the feature

Goal: First Party Top-Level Privacy UI



Fingerprinting defenses

Goal: **Make Tor Browser users as uniform as possible**

This means:

- Returning the same values for canvas extraction, User Agent, HTTP headers, Time zone; {using the same fonts}
- Putting users into different buckets (for screen and window sizes e.g.)

Fingerprinting defenses cont.

- Disabling features otherwise, e.g. plugins, GamePad API, NTLM authentication, open TCP port fingerprinting...

Long-term unlinkability

- Clear all linkable identifiers and browser state on request easily
- Thwarts powerful trackers (e.g. search engines)
- Implemented via a “**New Identity**” button in Tor Browser

The future

- Tor circuits bound to the URL bar domain
- Security Slider
- Signed Tor Browser updates verified via the Tor consensus
- Hardened bundles (with ASan, PartitionAlloc, support for Unix Domain Sockets, ...)

Conclusions

- Use Tor Browser in default config
- Problem of bad exits not negligible but also blown out of proportion
- Help needed in many areas

Thanks for coming!

...and don't forget to grab some **stickers**!

gk@torproject.org

35CD 74C2 4A9B 15A1 9E1A
81A1 9437 3AA9 4B7C 3223

phw@torproject.org

2A9F 5FBF 714D 42A9 F82C
0FEB 268C D15D 2D08 1E16