

Concepts in Crypto

Parker Higgins parker@eff.org @xor

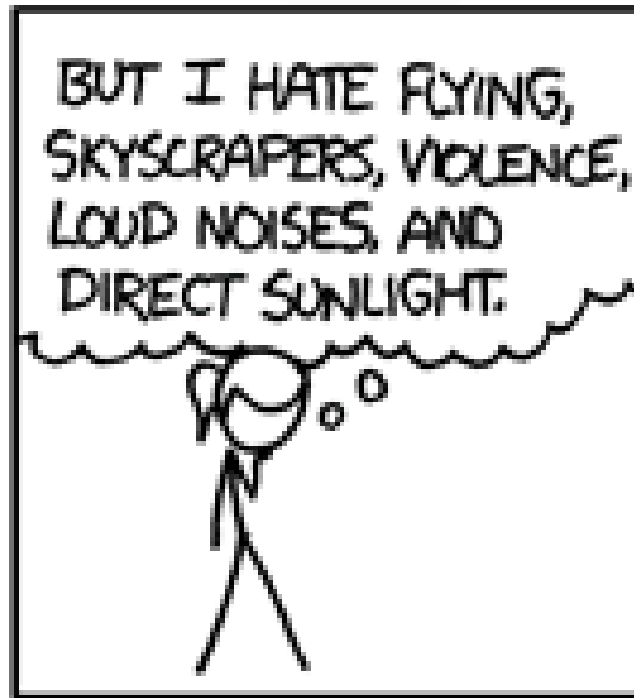
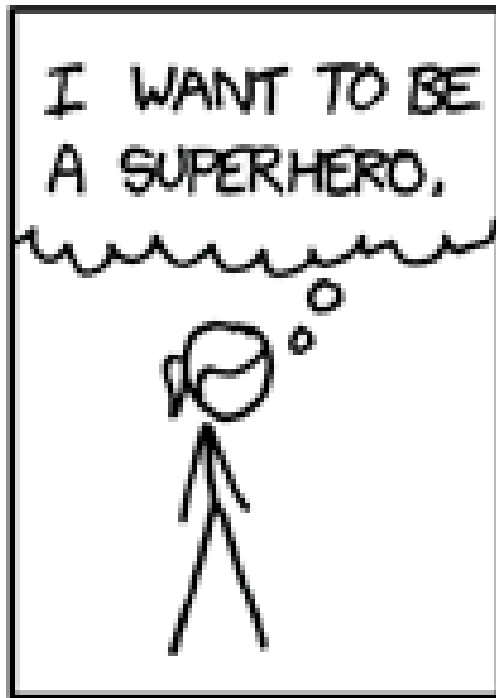
PGP: 4FF3 AA1B D29E 1638 32DE C765 9433 5F88 9A36 7709

Micah Lee micah@eff.org @micahflee

PGP: 5C17 6163 61BD 9F92 422A C08B B4D2 5A1E 9999 9697



Who We Are



Fighting for Crypto Rights

Bernstein v. United States - Wikipedia, the free encyclopedia - Mozilla Firefox

File Edit View History Bookmarks Tools Help

W Bernstein v. United States - Wi...

https://en.wikipedia.org/wiki/Bernstein_v_United_States

Create account Log in

Article Talk Read Edit View history Search

Bernstein v. United States

From Wikipedia, the free encyclopedia

Bernstein v. United States is a set of court cases brought by Daniel J. Bernstein challenging restrictions on the export of cryptography from the United States.

The case was first brought in 1995, when Bernstein was a student at University of California, Berkeley, and wanted to publish a paper and associated source code on his *Snuffle* encryption system. Bernstein was represented by the Electronic Frontier Foundation, who hired outside lawyer *Cindy Cohn*. After four years and one regulatory change, the *Ninth Circuit Court of Appeals* ruled that software source code was speech protected by the *First Amendment* and that the government's regulations preventing its publication were unconstitutional.^[1] Regarding those regulations, the EFF states:

Years before, the government had placed encryption, a method for scrambling messages so they can only be understood by their intended recipients, on the United States Munitions List, alongside bombs

Bernstein I	
Court	United States District Court for the Northern District of California
Full case name	<i>Daniel J. Bernstein et al., v. United States Department of State et al.</i>
Date decided	April 15, 1996
Citation(s)	922 F. Supp. 1426
Judge(s) sitting	Marilyn Hall Patel

Bernstein II	
Court	United States District Court for the Northern District of California
Full case name	<i>Daniel J. Bernstein et al., v. United States Department of State et al.</i>
Date decided	December 9, 1996



Crypto Terminology

- Plaintext
- Key
- Ciphertext
- Public Key Crypto
- Symmetric Crypto



Open Source Crypto

- How your crypto works should not be a secret
- The **only** secret should be the key
- Through these covert partnerships [with tech companies], the agencies [like NSA] have inserted secret vulnerabilities known as backdoors or trapdoors into commercial encryption software.
- The Guardian



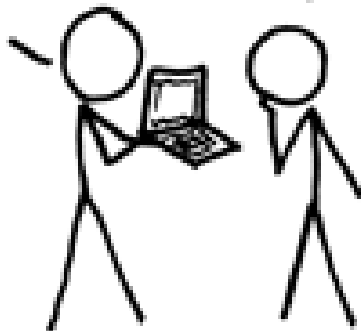
Threat Modeling

A CRYPTO NERD'S
IMAGINATION:

HIS LAPTOP'S ENCRYPTED.
LET'S BUILD A MILLION-DOLLAR
CLUSTER TO CRACK IT.

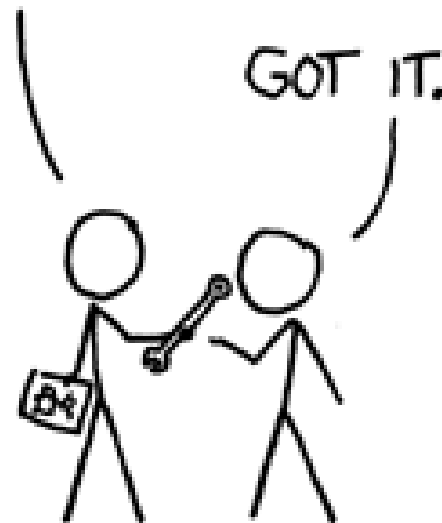
NO GOOD! IT'S
4096-BIT RSA!

BLAST! OUR
EVIL PLAN
IS FOILED!



WHAT WOULD
ACTUALLY HAPPEN:

HIS LAPTOP'S ENCRYPTED.
DRUG HIM AND HIT HIM WITH
THIS \$5 WRENCH UNTIL
HE TELLS US THE PASSWORD.



Types of Encryption

- Transport encryption
 - HTTPS, when connecting to websites
- End to end encryption
 - PGP, Off-the-Record
- Disk encryption
 - TrueCrypt
 - FileVault
 - LUKS



Diffie-Hellman Key Exchange

How is it possible for two people to come up with a shared crypto key when everything is being spied on?



PGP: Pretty Good Privacy

- Originally written by Phil Zimmermann in 1991 for anti-nuclear weapons activists
- Keys are split into two halves:
 - Public key (share it widely)
 - Secret key (keep it secret, keep it safe)
- With a public key you can:
 - Encrypt messages that can only be decrypted with the associated secret key
 - Verify signatures that that were signed with the associated secret key
- With a secret key you can:
 - Decrypt messages that were encrypted with the associated public key
 - Digitally sign messages



PGP in Practice

- GnuPG: open source implementation of OpenPGP (you shouldn't use the proprietary program called PGP)
- Thunderbird: a desktop email client, you can use it to check your email
- Enigmail: Thunderbird addon that adds OpenPGP functionality



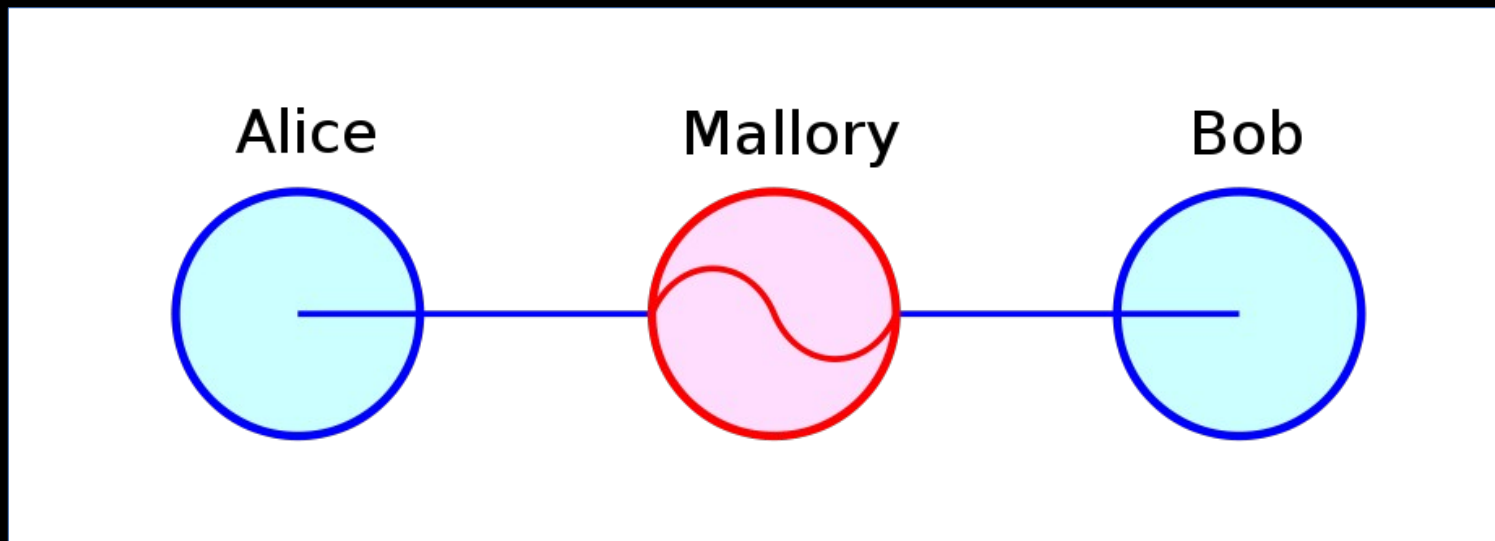
HTTPS

- You already use it every day!
- End-to-end encryption between your browser and the website's server
- Install HTTPS Everywhere!
<https://www.eff.org/https-everywhere>



HTTPS Everywhere

Man in the Middle Attacks (Woman in the Way?)



Certificate Authorities (CAs)

- When you load an HTTPS website it gives you its certificate, which includes its public key
- Your web browser uses this public key to initiate a secure session
- What if there's a MITM attack and you get a malicious public key instead?!
- CAs are companies whose job is to verify that the public key you get is valid



Certificate Authorities (CAs)



This is probably not the site you are looking for!

You attempted to reach www.washingtonpost.com, but instead you actually reached a server identifying itself as **a248.e.akamai.net**. This may be caused by a misconfiguration on the server or by something more serious. An attacker on your network could be trying to get you to visit a fake (and potentially harmful) version of www.washingtonpost.com.

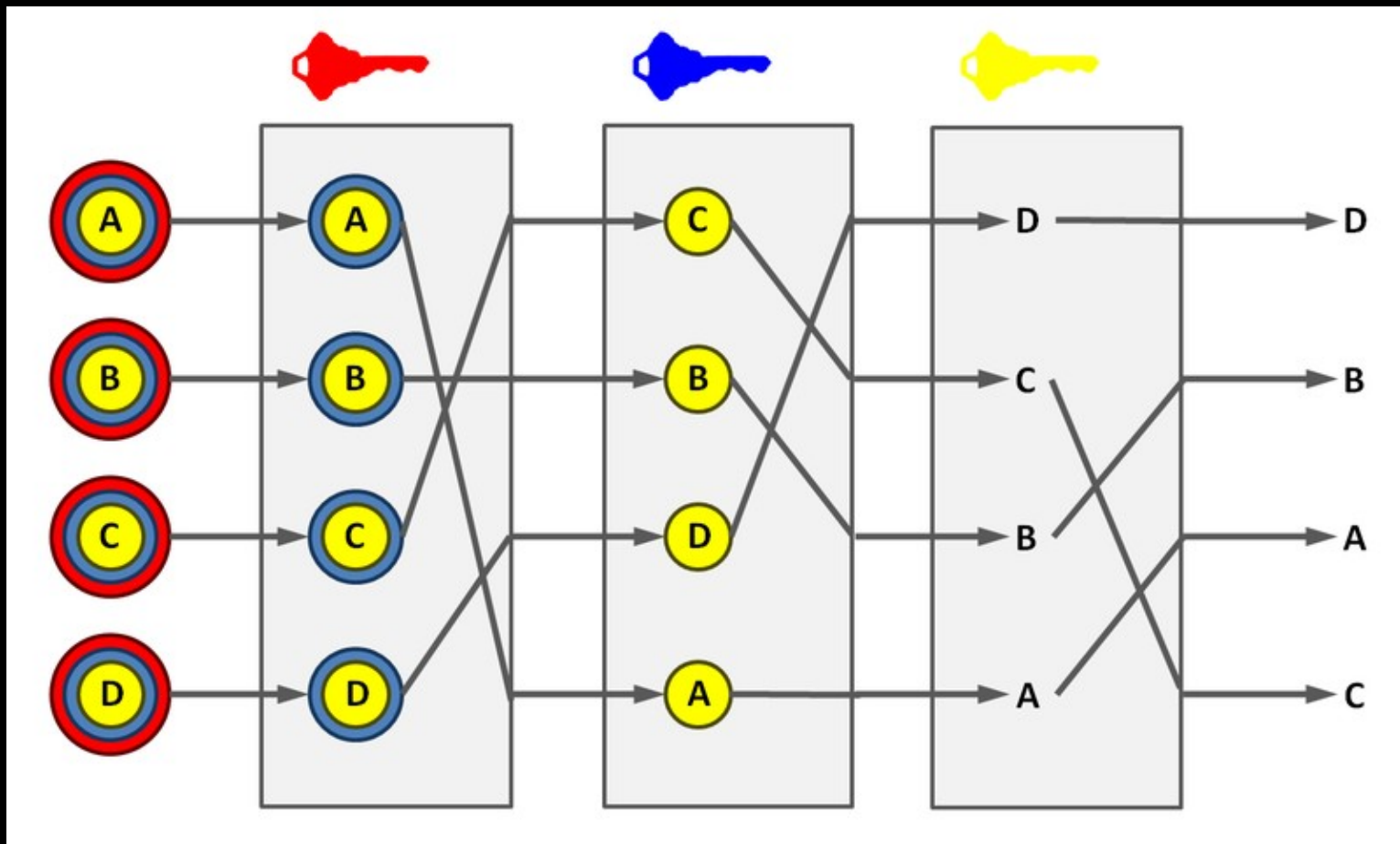
You should not proceed, **especially** if you have never seen this warning before for this site.

▶ [Help me understand](#)

Mix Networks

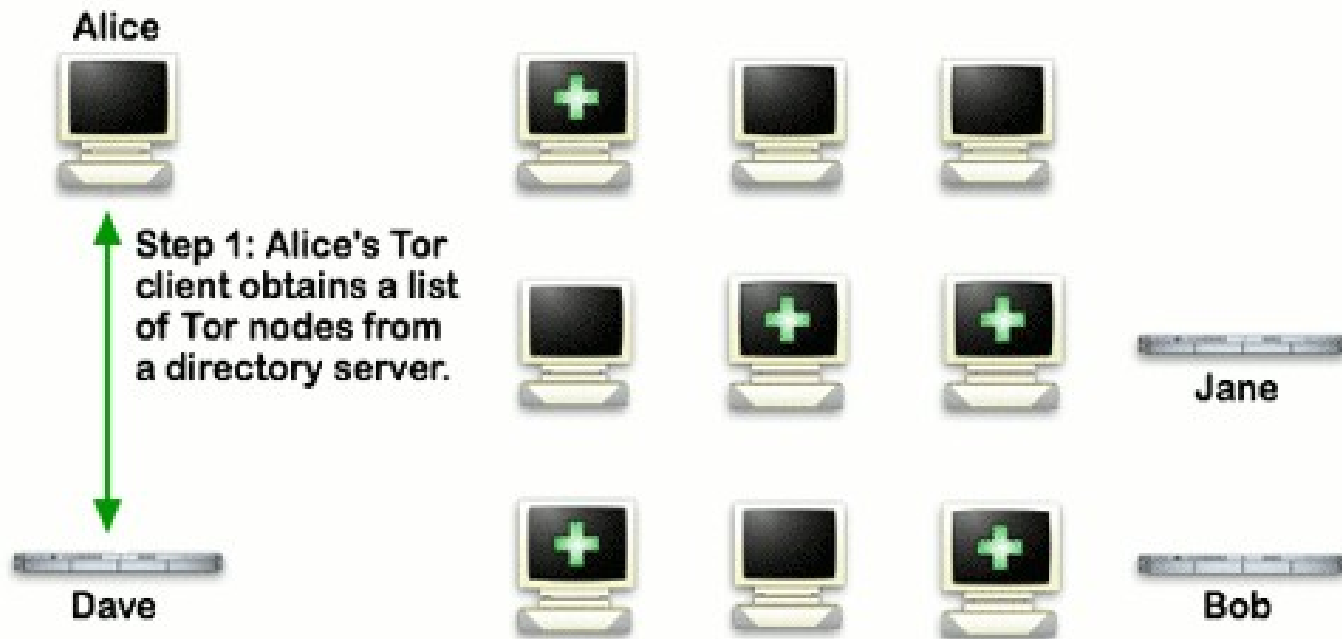


Anonymous Remailers

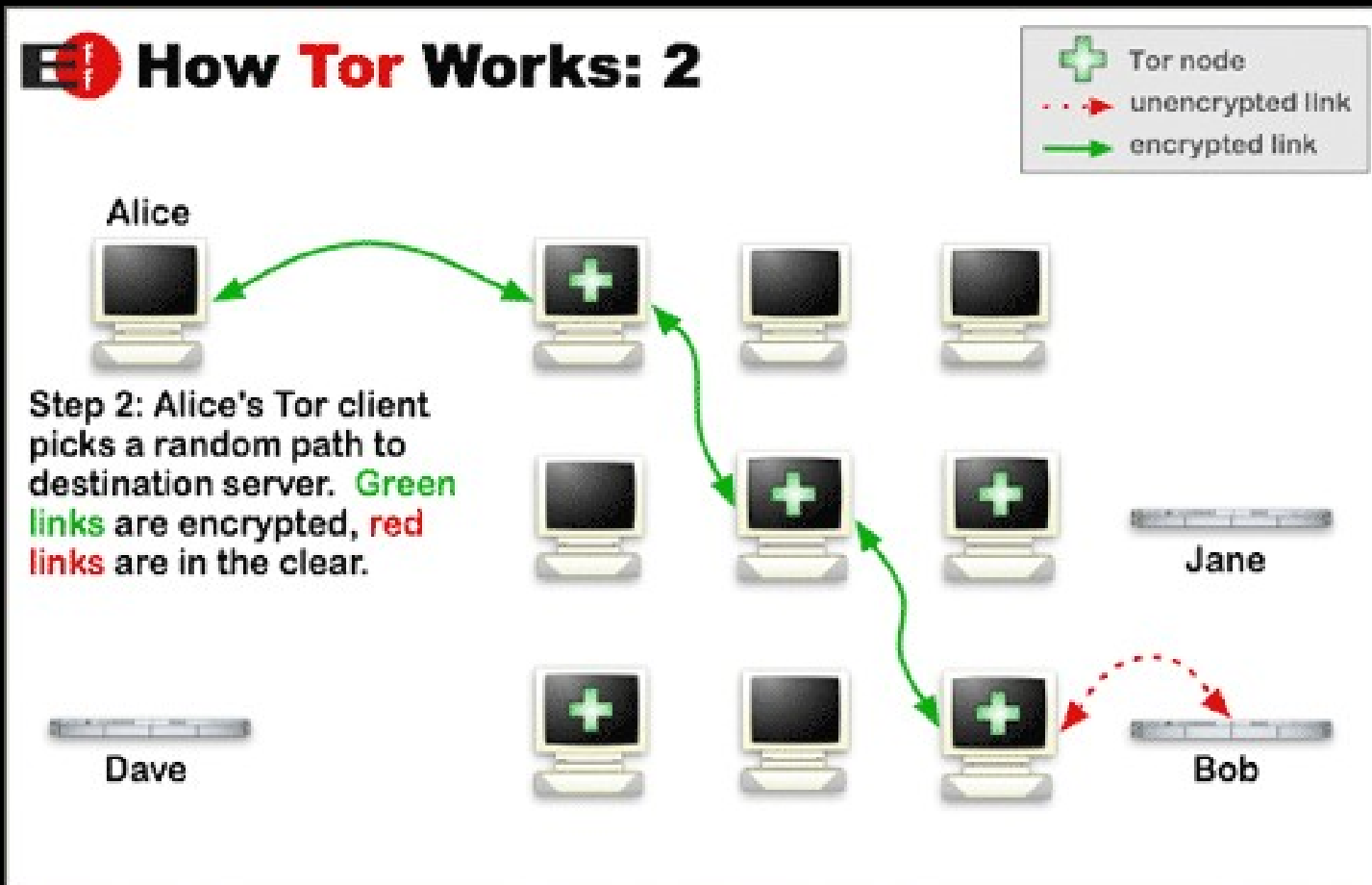


Tor: The Onion Router

How Tor Works: 1

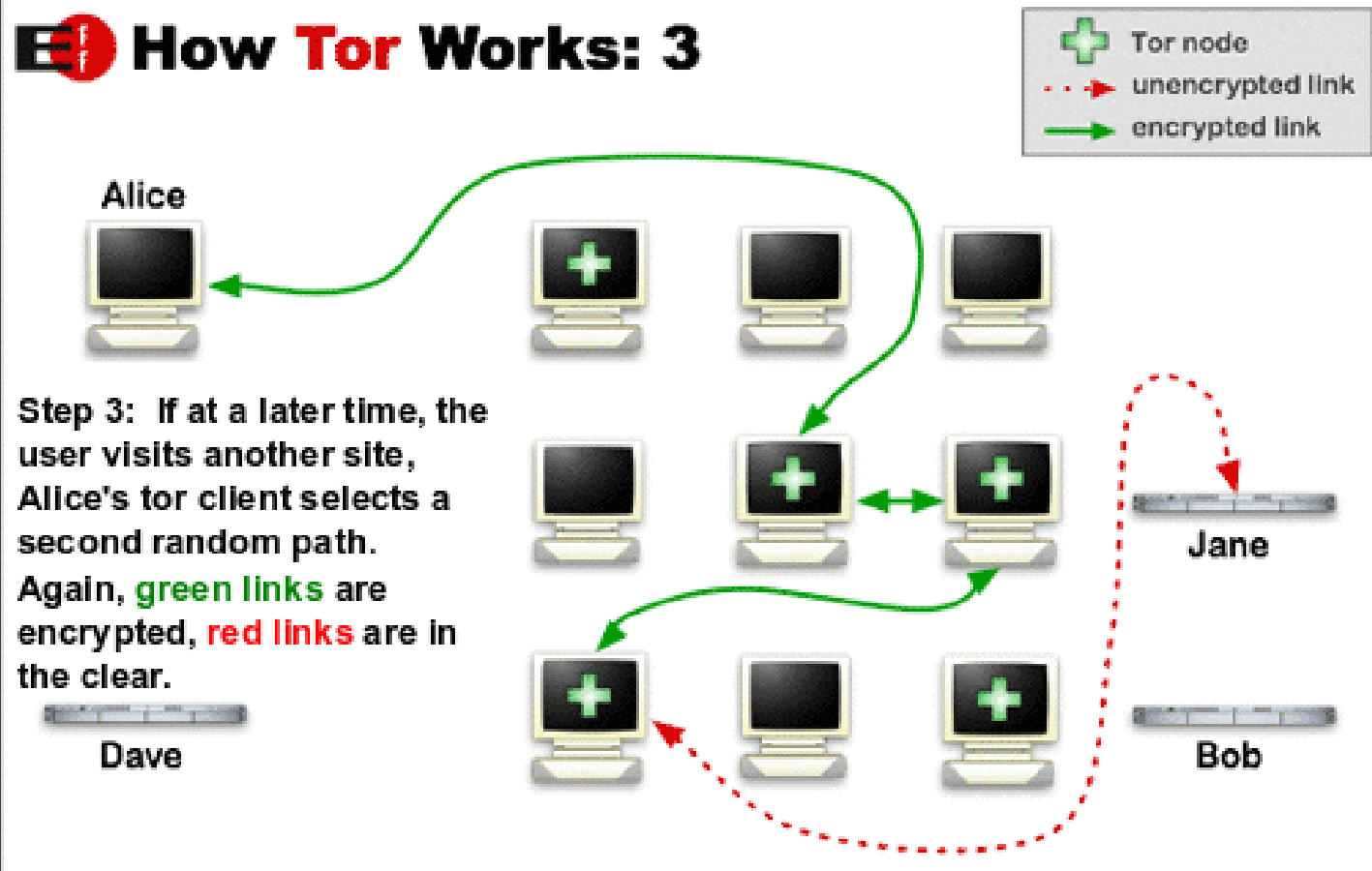


Tor: The Onion Router

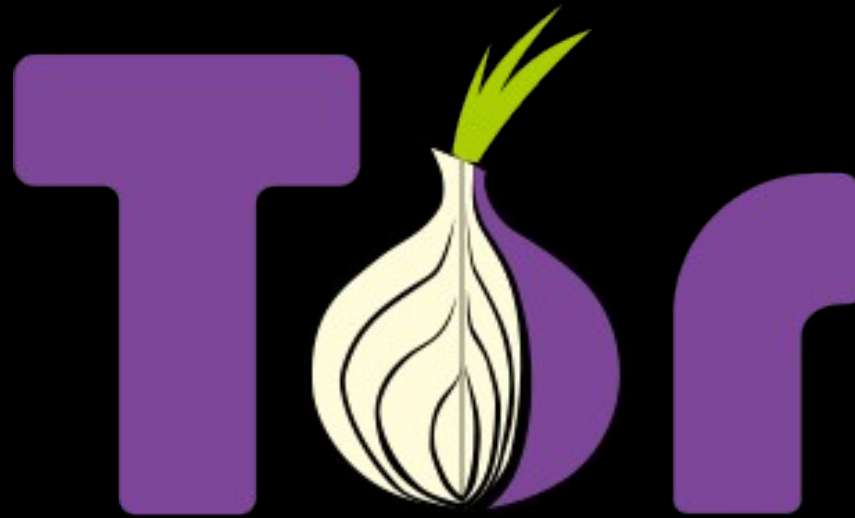


Tor: The Onion Router

How Tor Works: 3



Tor is Easy to Use



Download the Tor Browser from:
<https://www.torproject.org/>



Use Crypto Today

Off-the-Record IM Encryption

- End-to-end encryption chat over any existing service (Google Talk, Facebook, Jabber, AOL, etc.)
- Windows & Linux: Pidgin and OTR plugin
<https://pidgin.im/>
<https://otr.cypherpunks.ca/>
- Mac: Adium
<https://adium.im/>
- iOS, Android: ChatSecure



Use Crypto Today

Full Disk Encryption

- If you leave your laptop on the bus, your can still remain safe!
- Windows: TrueCrypt, BitLocker
<http://www.truecrypt.org/>
- Mac: FileVault (built-in)
- Linux: LUKS (built-in)
- Newer versions of Android (built-in)



Learn More

- EFF's Surveillance Self-Defense Guide:
<https://ssd.eff.org/>
- Security in a Box:
<https://securityinabox.org/>
- Encryption Works:
<https://pressfreedomfoundation.org/encryption-works>



Thank You!

Parker Higgins parker@eff.org @xor

PGP: 4FF3 AA1B D29E 1638 32DE C765 9433 5F88 9A36 7709

Micah Lee micah@eff.org @micahflee

PGP: 5C17 6163 61BD 9F92 422A C08B B4D2 5A1E 9999 9697

