

Abuse

Project 2...

- Recording is paused: Nick's Design
- "Consider a pointer"
- "When in doubt, another layer of indirection"

Abuse: Content Warning

- Serious content warnings ahead
 - Sexual Harassment
 - Domestic Violence
 - Child Abuse
- The Facebook abuse protocol (or variants thereof) may be on the test
 - But only if it ends up in the textbook by Wednesday (it isn't there yet)
- Stalkerware and Tor Hidden Services are not
- Implicit blue background on all slides
 - But I think it is important to be aware of these hard problems

The Facebook Problem: Applied Cryptography in Action

- Facebook Messenger now has an encrypted chat option
 - Limited to their phone application
 - Really can't do this with the web because the server could always provide corrupted JavaScript
- The cryptography in general is very good
 - Used a well regarded asynchronous messenger library (from Signal) with many good properties, including forward secrecy
- When Alice wants to send a message to Bob
 - Queries for Bob's public key from Facebook's server
 - Encrypts message and send it to Facebook
 - Facebook then forwards the message to Bob
- Both Alice and Bob are using encrypted and authenticated channels to Facebook

Facebook's Messenger Problem: Abuse

- Much of Facebook's biggest problem is dealing with abuse...
 - What if either Alice or Bob is a stalker, an a-hole, or otherwise problematic?
- Facebook would expect the other side to complain
 - And then perhaps Facebook would kick off the perpetrator for violating Facebook's Terms of Service
- But fake abuse complaints are also a problem
 - So can't just take them on face value
- And abusers might also want to release info publicly
 - Want sender to be able to ***deny to the public*** but not to Facebook
 - ***Deniability*** is in many ways anti-***authentication***:
Want to make it so you don't have public key signatures

This abuse is often highly gendered

- Most common is male->female harassment
 - Both text and images
 - Lea Kissner @ Twitter describes the images involved as the "Zipper Problem"
- As such lets acknowledge it in describing the protocol
 - Alice **Alex** == Abuser
 - ~~Bob~~ **Bailey** == Recipient
- Any messenger system with a large number of users, especially supporting images, is going to have these issues

Facebook's Problem Quantified

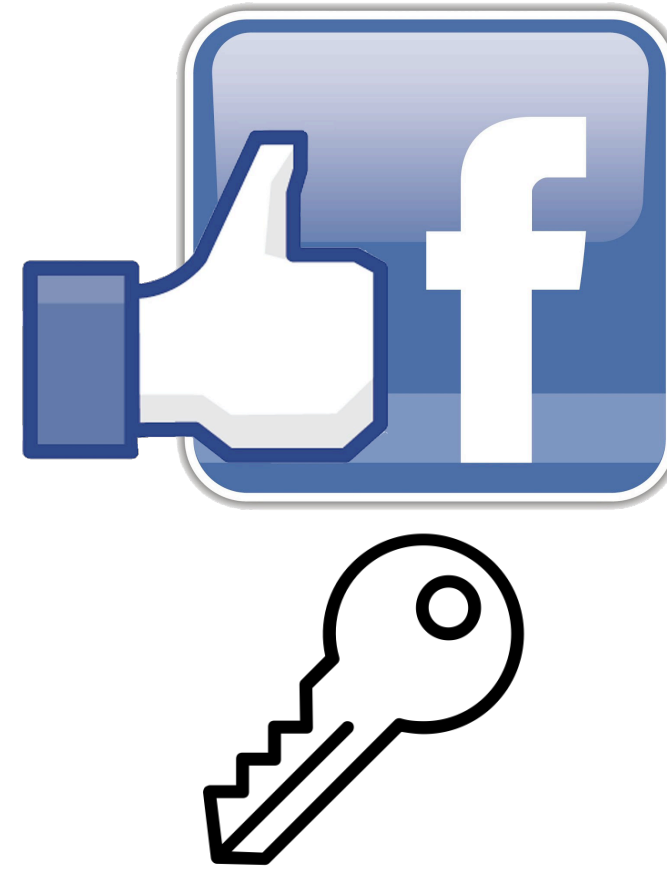
- Unless Bailey forwards the unencrypted message to Facebook
 - Facebook ***must not*** be able to see the contents of the message
- If Bailey does forward the unencrypted message to Facebook
 - Facebook ***must ensure*** that the message is what Alex sent to Bailey
- Nobody ***but*** Facebook should be able to verify this:
No public signatures!
 - Critical to prevent abusive release of messages to the public being verifiable:
Messages are ***deniable*** for everybody but Facebook

The Protocol In Action

Alex



What Is Bailey's Public
Key?



Bailey



Aside: Key Transparency...

- Both Alex and Bailey are trusting Facebook's honesty...
 - What if Facebook gave Alex a different key for Bailey? How would he know?
- Facebook messenger has a ***nearly*** hidden option which allows Alex to see Bailey's key
 - If they ever get together, they can manually verify that Facebook was honest by looking at a series of "safety numbers" or QR code
- The mantra of central key servers: ***Trust but Verify***
 - The simple option is enough to force honesty, as each attempt to lie has some probability of being caught
- This is the biggest weakness of Apple iMessage:
 - iMessage has (fairly) good cryptography but there is no way to verify Apple's honesty

The Protocol In Action



Alex



```
{message=E(Kpub_b,  
  M={"Hey Bailey: Abusive  
    Message",  
      krand}),  
mac=HMAC(krand, M),  
to=Bailey}
```

```
{message=E(Kpub_b,  
  M={"Hey Bailey: Abusive  
    Message",  
      krand}),  
mac=HMAC(krand, M),  
to=Bailey,  
from=Alex,  
time=now,  
fbmac=HMAC(Kfb, {mac, from,  
                  to, time})}
```

Bailey



Some Notes

- Facebook ***can not*** read the message or ***even verify Alex's HMAC***
 - As the key for the HMAC is in the message itself
- Only Facebook knows their HMAC key
 - And its the only information Facebook ***needs*** to retain in this protocol:
Everything else can be discarded
- Bailey upon receipt checks that Alex's HMAC is correct
 - Otherwise Bailey's messenger silently rejects the message
 - Forces Alex's messenger to be honest about the HMAC, ***even thought Facebook never verified it***
- Bailey trusts Facebook when Facebook says the message is from Alex
 - Bailey does ***not verify*** a signature, because there is no signature to verify...
But the Signal protocol uses an ephemeral key agreement so that implicitly verifies Alex as well

Now To Report Abuse



Alex



Bailey



```
{Abuse{
  M={"Hey Bailey: Abusive
    Message",
    k_rand}},
  mac=HMAC(k_rand, M),
  to=Bailey,
  from=Alex,
  time=now,
  fbmac=HMAC(K_fb, {mac, from,
    to, time})}^2
```

Facebook's Verification

- First verify that Bailey correctly reported the message sent
 - Verify $\mathbf{fbmac} = \mathbf{HMAC}(\mathbf{K}_{\mathbf{fb}}, \{\mathbf{mac}, \mathbf{from}, \mathbf{to}, \mathbf{time}\})$
 - Only Facebook can do this verification since they keep $\mathbf{K}_{\mathbf{fb}}$ secret
 - This enables Facebook to confirm that this is the message that it relayed from Alex to Bailey
- Then verify that Bailey didn't tamper with the message
 - Verify $\mathbf{mac} = \mathbf{HMAC}(\mathbf{k}_{\mathbf{rand}}, \{\mathbf{M}, \mathbf{k}_{\mathbf{rand}}\})$
- Now Facebook knows this was sent from Alex to Bailey and can act accordingly
 - But Bailey ***can't prove*** that Alex sent this message to anyone ***other than Facebook***
 - And Bailey ***can't tamper with the message*** because the HMAC is also a hash

Here Be Blue Slides...

- The rest of this lecture is "blue slides"
 - I won't have explicit "**takeaway**" portions either that you are responsible for
- But there are many important real-world takeaways
 - So I'm including them:
There is a serious moral dimension to computing and we neglect them at our peril

Stalkerware:

The Intimate Partner Threat in Action

- The "Intimate Partner Threat" is one of the most powerful adversaries
 - They have physical access to your devices
 - They have intimate knowledge
 - They are integrated into your social circle
 - They are highly motivated: no "bear race" is possible
- IPT is often associated within the larger context of stalking & domestic violence
 - This is an area of computer security where there are lives at stake
 - It is also an insanely hard problem because of the attacker's resources

Example IPT Attack: Compromise Facebook

- I have a colleague who was going through a divorce
 - It was not the friendliest of divorces
 - One day their (ex) partner broke into their facebook account!
- How?
 - Facebook password recovery option: Have 3 friends help out
 - So select three Facebook friends of the target:
 - The family dog
 - A member of their wedding party on the attacker's side
 - The attacker's best friend
 - Attack relied on knowledge of the social circle

The Stalkerware Problem

- Generally refers to surreptitious monitoring software installed on the target's computer
 - <https://stopstalkerware.org>
- Method of installation usually takes advantage of physical access
 - Unlocked computer
 - Unlocked phone
- Once installed it enables surreptitious monitoring without notifying the victim
 - Common features include monitoring all messages and location tracking

The Problem:

Reuse of functionality...

- Cellular phones support "Mobile Device Management"
 - A **business** phone belongs to the business, not the user
 - So the phones have hooks to monitor components
- Family Sharing/Parental Controls
 - EG, Apple allows sharing location with others in the family...
The IPT can surreptitiously enable this
- Android supports "side-loading" of applications with physical access and the password
 - Allows bypassing security checks and vetting that occurs in application stores

Just One Example: Installation of Stalkerware...

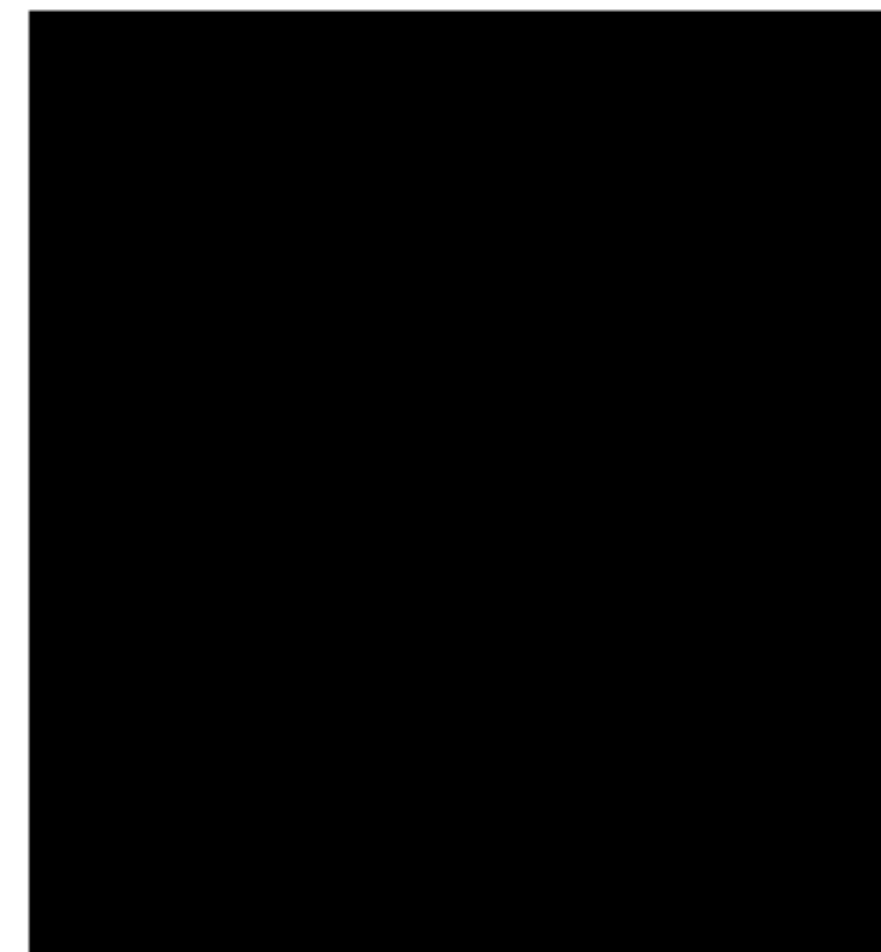
- Abuser does a Google search
 - Google doesn't allow ads on a lot of these searches, but...
 - There is still plenty of SEO optimized content!
- Abuser gets victim's passcode
 - Watches victim input the passcode
 - Just asks
 - Knows victim uses 123456
- And now when the victim is asleep...

How to Catch a cheating spouse using Android and



To catch a cheating spouse you will need to get a hold of their phone. You will need their pass-code too to get into the phone. Using an android spy app like [REDACTED] you can catch your cheating spouse without needing to root the device either.

Here are the exact steps to putting [REDACTED] on an Android phone and catching your spouse cheating. It is how you can use [REDACTED] as an android spy apps for a cheating spouse. Go head and follow along on your own android phone or just use this as a guide when you have a few minutes alone with your spouses phone (*TIP: Do this while they are sleeping.*)



Recovery is hard!

- It can be very hard to recover
 - These programs can be hard to detect:
And on Android it often starts by rooting the phone
 - If you think you are a victim of stalkerware, ***trust your instincts!***
 - A good guide here:
<https://stopstalkerware.org/information-for-survivors/>
- Personal recommendation: Prevention is easier!
 - My phone uses a 5 word random passphrase which ***nobody else knows!***
 - But I also enable fingerprint/face unlock
 - So the passphrase is seldom used but it is necessary for any of the abusive installations or configuration changes

Some More on Tor

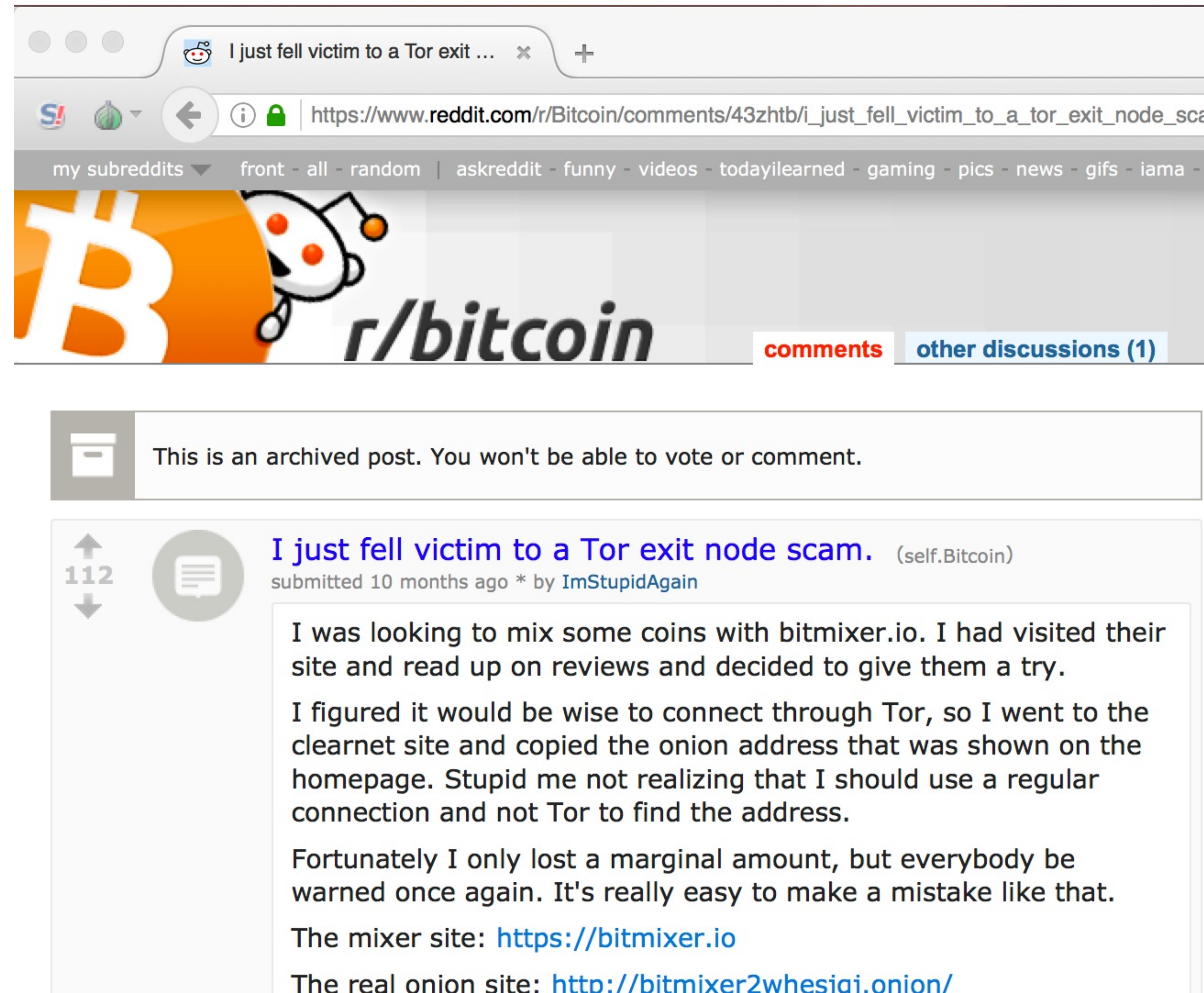
- Picking up from last week...
- Why Tor is painful for clients
- Why Tor is only so-so for counter-censorship
- Why Tor Hidden Services are a plague

In Tor You Are Relying On Honest Exit Nodes...

Computer Science 161

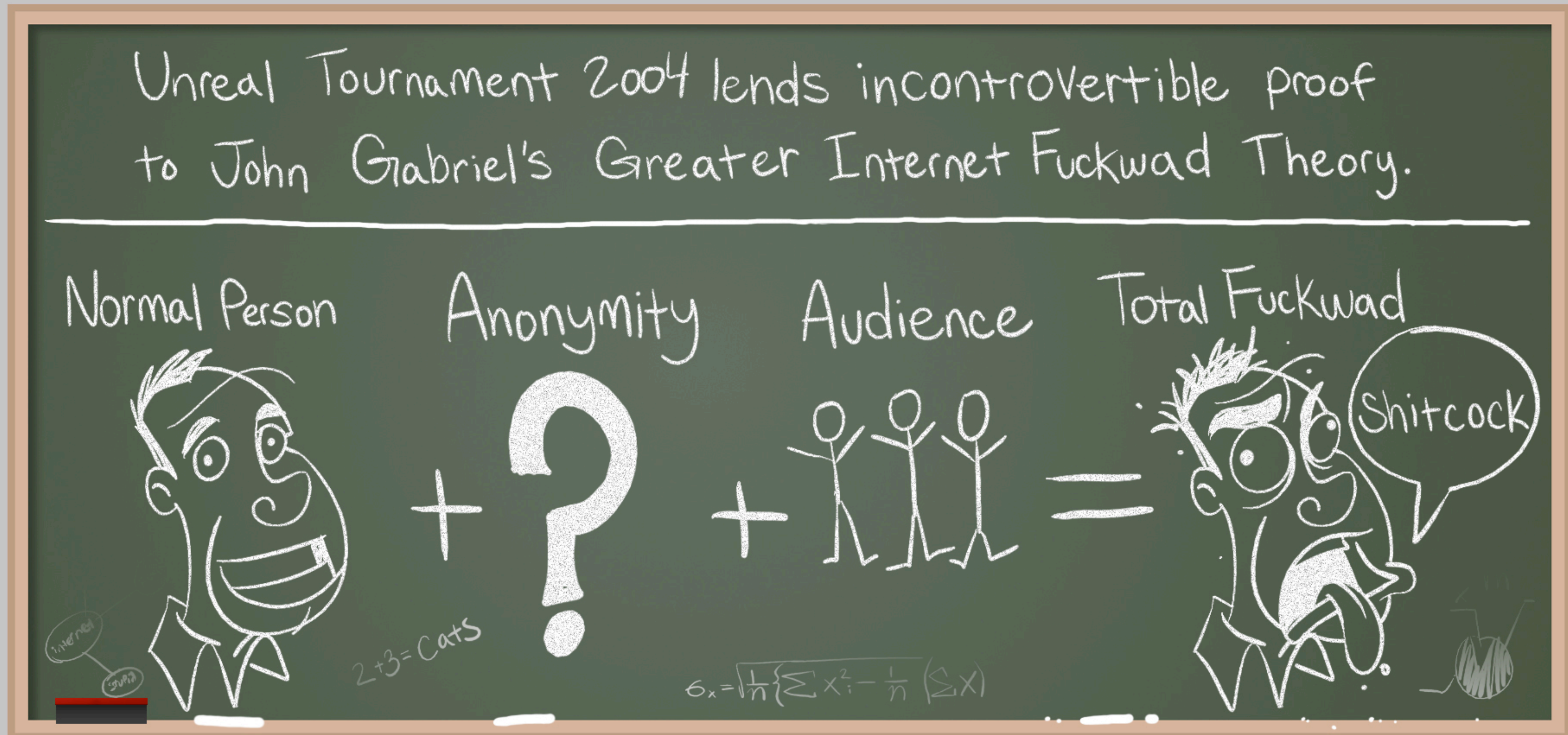
Popa & Weaver

- The exit node, where your traffic goes to the general Internet, is a man-in-the-middle...
- Who can see and modify all non-encrypted traffic
- The exit node also does the DNS lookups
- Exit nodes have not always been honest...



Anonymity Invites Abuse...

(Stolen from Penny Arcade)



This Makes Using Tor Browser Painful...

Due to "Fate Sharing"



And Also Makes Running Exit Nodes Painful...

- If you want to receive abuse complaints...
 - Run a Tor Exit Node
- Assuming your ISP even allows it...
 - Since they don't like complaints either
- Serves as a large limit on Tor in practice:
 - Internal bandwidth is plentiful, but exit node bandwidth is restricted
- Know a colleague who ran an exit node for research...
 - And got a ***visit from the FBI!***

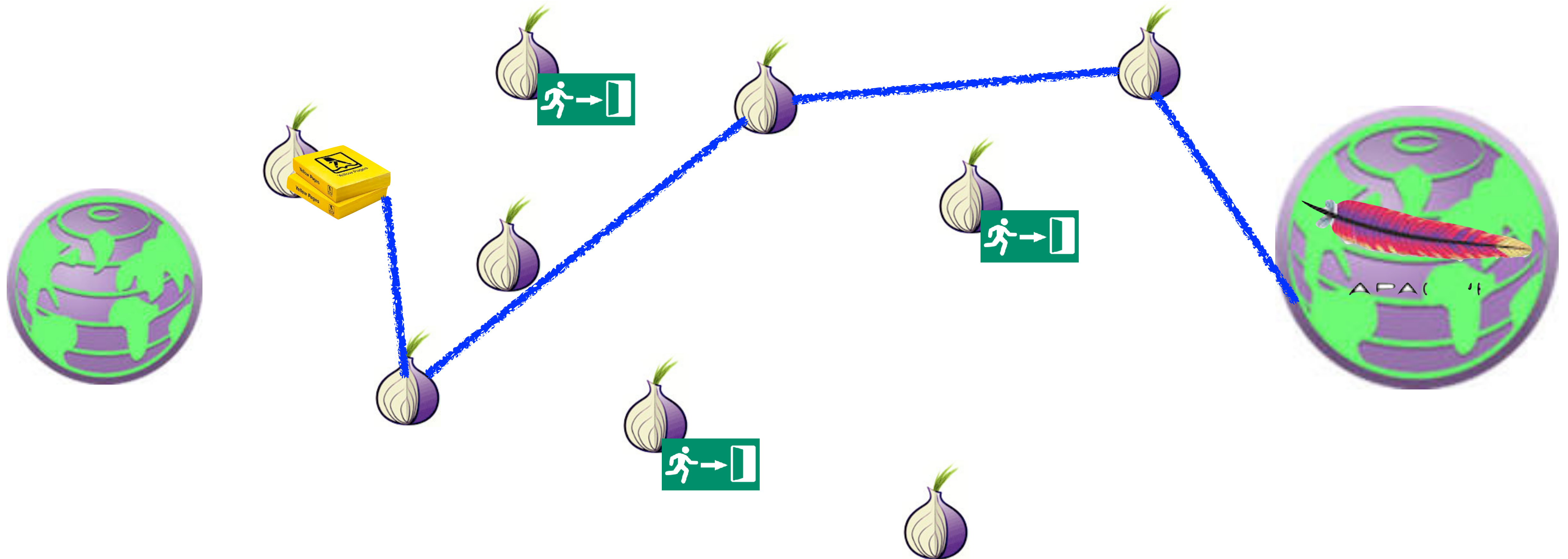
Censorship Evasion...

- Tor is actually really ***bad*** for evading censorship
 - It is trivial to tell that someone on the network is running Tor
- There are ***optional*** pluggable transports that attempt to hide the traffic
 - The problem is you have to learn about these...
Yet if the censor does, it won't work!
- And then the user has all the bad of Tor...
 - Fate sharing with all other users of the exit nodes
 - Significantly worse latency
 - Oh, and Tor Browser's not saving history is not necessarily nice!
- Only good thing is it is "free"
 - Tor project gets paid largely for counter-censorship
 - Users are "paying" by providing traffic for those who want anonymity to hide in

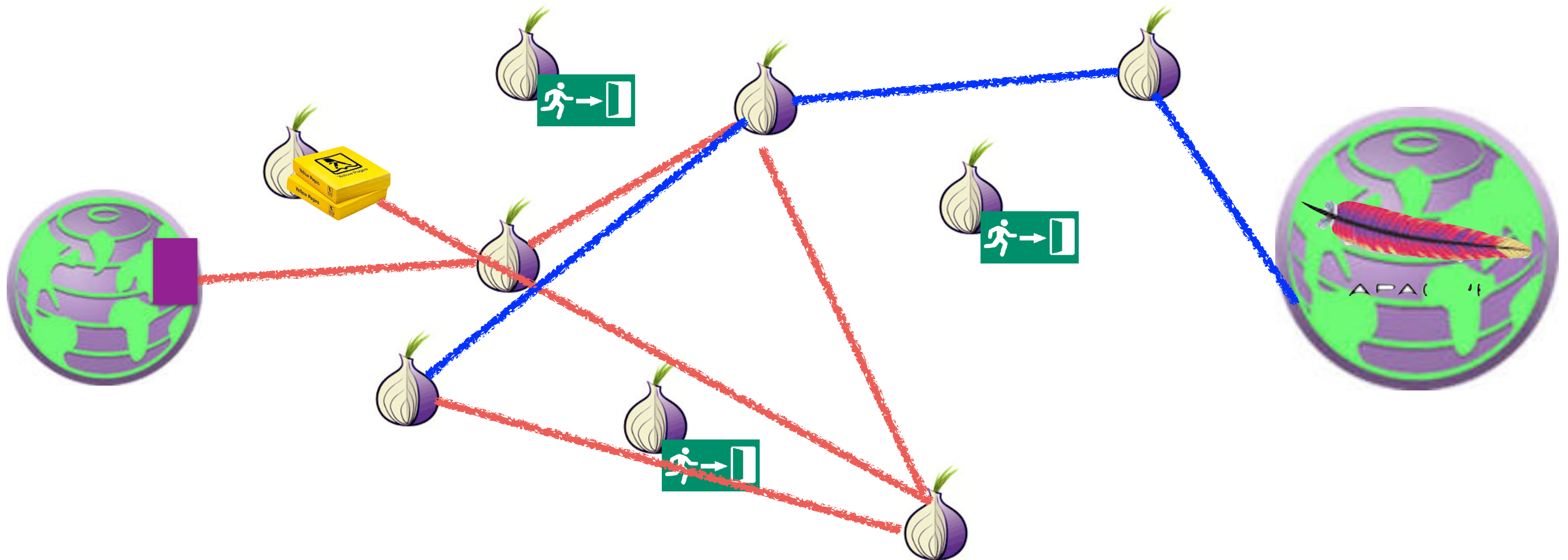
Tor Browser is also used to access Tor Hidden Services aka .onion sites

- Services that only exist in the Tor network (the "Dark Web")
 - So the service, not just the client, has possible anonymity protection
- A hidden service name is a hash of the hidden service's public key
 - Used to be smaller: <https://facebookcorewwi.onion>
 - In this case, Facebook spent a lot of CPU time to create something distinctive
 - Now larger: <https://facebookwkhpilnemxj7asaniu7vnjjbiltxjqh3mhbshg7kx5tfyd.onion>
 - Change was designed to prevent one attack at the cost of forcing records of .onion domains to be unmemorizeable
- Using this key hash, can query to set up a circuit to create a hidden service at a rendezvous point
 - And because it is the hash of the key we have end-to-end security when we finally create a final connection

Tor Hidden Service: Setting Up Introduction Point



Tor Hidden Service: Query for Introduction, Arrange Rendezvous



Home | Alphabay Market

About Tor

!

←

i

pwoah7foa6au2pul.onion/index.php

↻

🔍

Search

☰

a

AlphaBay Market

Logged in as seanbridges

Balance: BTC 0.0000 / XMR 0.0000

Autoshop Logout

▲ USD 573.53

▲ CAD 735.76

▲ EUR 506.38

▲ AUD 753.03

▲ GBP 437.84

HOME

SALES

MESSAGES

ORDERS

LISTINGS

BALANCE

FEEDBACK

FORUMS

API

SUPPORT

🏠

Home

seanbridges

Joined: Aug 30, 2016

Trust level: Level 1

Total sales: USD 0.00

Total orders: USD 0.00


Search:

Search

⚠️

We highly recommend that you disable Javascript when viewing the marketplace for better security.


Featured Listings



[FE 100%]

► FRESH CC/CVV USA

VISA/MASTERCARD /DISCOVER/AMEX (OLD MAGIC QUALITY/VALIDITY) - (New Stock OF CC +10K) - (Delivery Instantly) - (Always Online)

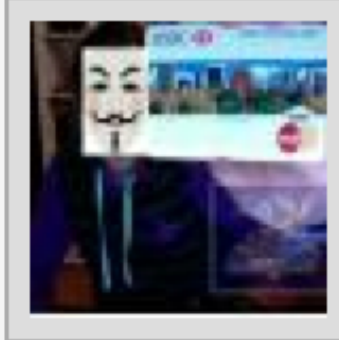


[Bulk] USA HIGH LEVEL CC - VISA

RANDOM CREDIT - BUSINESS/SIGNATUREWORLDWIDE - GET /PLATINUM [AUTO FULFILL ON - DAILY SUPPORT] Browse store for more types and levels CCs!

6329 - CVV & Cards - st0n3d


Buy: USD 8.50



[MS] EDITABLE HQ TEMPLATES OF DOCUMENTS

VERIFIED EVERYWHERE INSTANTLY! - OVER 250 TEMPLATES TO CHOOSE FROM, SAMPLES ON ymhulceusuzrj3i5.onion

51105 - Other



Double Your Bitcoins in ONE Day ! GUARANTEED! (2 in 1) \$7000+ in 20 TWENTY MINUTES (50 + COPIES SOLD 100% POSITIVE FEEDBACK!)

183848 - Other - BitcoinThief

Buy: USD 600.00

CC / ACCOUNT AUTOSHOP

Access the CC autoshop

Access the account autoshop

BROWSE CATEGORIES

► ☐ Fraud25438

► ☐ Drugs & Chemicals136335

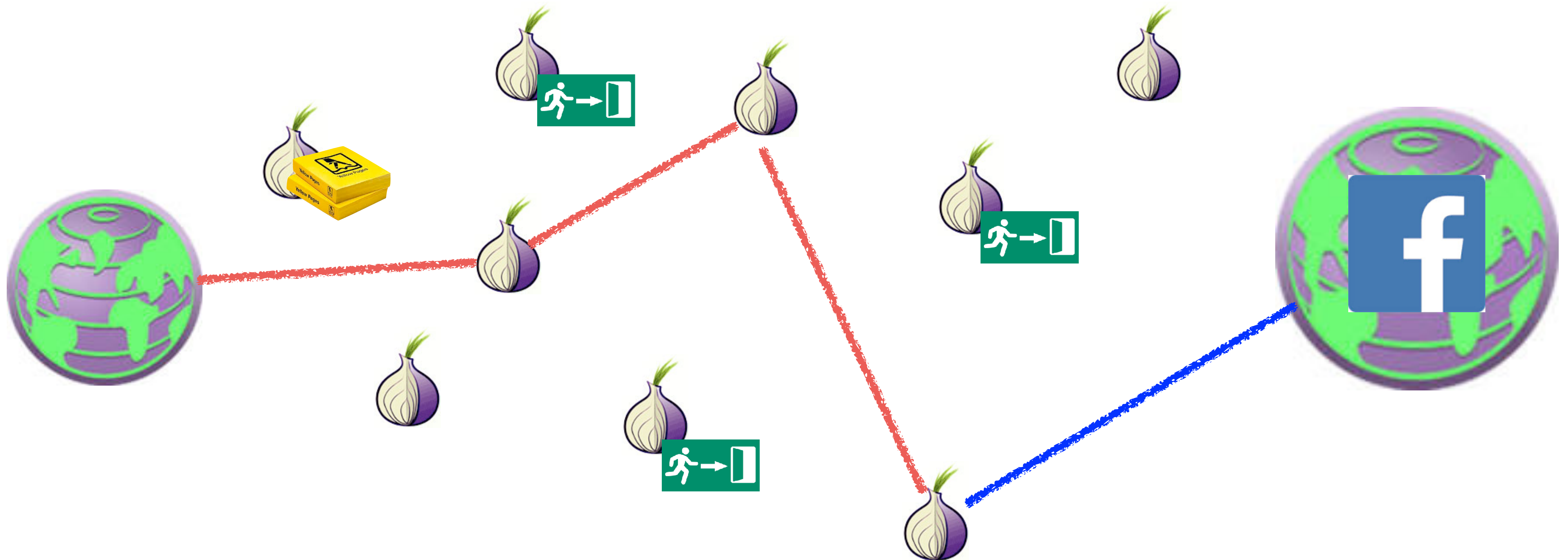
► ☐ Guides & Tutorials10029

31

Remarks...

- A hidden service wants to keep the guard node constant for a long period of time...
 - Since the creation of new circuits is far easier to notice than any other activity
- Want to use a different node for the rendezvous point and introduction
 - Don't want the rendezvous point to know who you are connecting to
- These are ***slow!***
 - Going through 6+ hops in the Tor network!

Non-Hidden Tor Hidden Service: Connect Directly to Rendezvous



Non-Hidden Hidden Services

Improve Performance

- No longer rely on exit nodes being honest
 - No longer rely on exit node bandwidth either
- Reduces the number of hops to be the same as a not hidden service
- Result: Huge performance win!
 - Not slow like a hidden service
 - Not limited by exit node bandwidth
 - Facebook does this
- Any ***legitimate*** site offering a Tor hidden service should use this technique
 - Since legitimate sites don't need to hide!

Real use for *true hidden* hidden services

- "Non-arbitrageable criminal activity"
 - Some crime which is universally attacked and targeted
 - So can't use "bulletproof hosting", CloudFlare, or suitable "foreign" machine rooms
- Dark Markets
 - Marketplaces based on Bitcoin or other alternate currency
- Cybercrime Forums
 - Hoping to protect users/administrators from the fate of earlier markets
- And worse...

The Dark Market Concept

- Four innovations:
- A censorship-resistant payment (Bitcoin)
 - Needed because illegal goods are not supported by Paypal etc
- An eBay-style ratings system with mandatory feedback
 - Vendors gain positive reputation through continued transactions
- An escrow service to handle disputes
 - Result is the user (should) only need to trust the market, not the vendors
 - The market is *shifting* trust, not eliminating it
- Accessable *only* as a Tor hidden service
 - Hiding the market from law enforcement

The Dark Markets: History

- All pretty much follow the template of the original “Silk Road”
 - Founded in 2011, Ross Ulbricht busted in October 2013
- The original Silk Road actually (mostly) lived up to its libertarian ideals
 - Including the libertarian ideal that if someone rips you off you should be able to call up the Hell’s Angels and put a hit on them
 - And the libertarian idea if someone is foolish enough to THINK you are a member of the Hell’s Angels you can rip them off for a large fortune for a fake hit
- Since then, markets come and go...
 - And even information about them is harder:
Reddit no longer supports them, deepdotweb got busted...
Leaving "Dread": Reddit as a Tor Hidden Service

The Dark Markets: Not So Big, and ***Not Growing!***

- Kyle Soska and Nicolas Christin of CMU have crawled the dark markets for years
 - These markets ***deliberately*** leak sales rate information from mandatory reviews
- So simply crawl the markets, see the prices, see the volume, voila...
- Takeaways:
 - Market size has been relatively steady for years, about \$300-500k a day sales
 - Latest peak got close to \$1M a day
 - Dominated by Pot, MDMA, and stimulants, with secondary significance with opioids and psychedelics
 - A few sellers and a few markets dominate the revenue: A fair bit of “Winner take all”
 - But knock down any “winner” and another one takes its place

The Scams...

- You need a reputation for honesty to be a good crook
 - But you can burn that reputation for short-term profit
- The “Exit Scam” (pioneered by Tony76 on Silk Road)
 - Built up a positive reputation
 - Then have a big 4/20 sale
 - Require buyers to “Finalize Early”
 - Bypass escrow because of “problems”
 - Take the money and run!
- Can also do this on an entire *market* basis
 - The “Sheep Marketplace” being the most famous

And Now Once Again a ***SERIOUS*** Content Warning...

- The rest of the lecture is going to talk about the Elephant in the Room with Tor...
Tor hidden services facilitate child abuse on an industrial scale
 - And the Tor project ***DOES NOT CARE!***
- I will be talking about actual cases and the scope of the problem
 - I studied these cases because they touched on significant policy issues surrounding searches and government hacking
- This will not be on the test beyond the following:
"Nick hates Tor's Hidden Services with the fires of a thousand suns"
and this is why...
 - And for the love of everything do not ever build something that has proved as loathsome as Tor

February 2, 2020, Sunrise, Florida

- A team of FBI agents in the Violent Crimes Against Children division, including special agents Daniel Alfin and Laura Schwartzenberger, attempted to serve a search warrant as part of a CSAM (Child Sexual Abuse Material) investigation
- Agents Alfin and Schwartzenberger were murdered by the suspect and three other agents injured
- I knew Dan professionally from his previous work involving CSAM and Tor...



The "Playpen" Investigation

- In 2015 the FBI managed to identify and capture the server hosting the "Playpen" child exploitation site:
Daniel Alfin was one of the lead investigators
- Playpen operated as a hidden service image board for posting CSAM
 - 250,000+ registered users, 20,000+ images
 - This represents thousands of abused children!
- But the site operator's are not the only problem...
The site users are a problem
 - A significant number are "hands-on" abusers:
Both because of their predilections and because creating new "content" is currency in these communities

To Deanonimize the Users...

- The FBI took over Playpen and ran the site for 2 weeks
- During those two weeks...
 - Disabled posting of new content, but continued to serve old content...
 - And added a post-login bonus: A zero-day attack on the Tor Browser Bundle
- Exploit payload: "phone home"
 - Not a general purpose shellcode, instead collect Ethernet Addresses, current user, and similar identifying information and contact an FBI server
- FBI calls this a NIT: "Network Investigation Technique"
- They had a warrant:
 - It described with particularity what it would search for, how it would work conceptually, etc...

Significant Impact

- 25 producers prosecuted, 350 arrests in the US alone
- Nearly 300 children identified or rescued from abusive situations worldwide, over 50 in the US
- But also two significant controversies:
- Was the warrant actually valid?
 - Answer ended up being "No, but 'good faith'....":
At the time there was no way to write a warrant that says "I want to search these computers, but we don't know where they are!"
- What should defendants be able to examine with regard to the exploit?
 - Answer largely ended up being "No, not actually relevant"
 - An in the weeds discussion by Susan Hennesey and myself is available here:
<https://www.lawfareblog.com/judicial-framework-evaluating-network-investigative-techniques>

The Problem:

These are communities of abusers

- There have been others both before and since
 - Before Playpen there was "Freedom Hosting": hosted close to 50 CSAM sites. If you want to be nauseated read the Freedom Hosting NIT warrant application
 - In 2017 an FBI style NIT was deployed on "GiftBox" (probably by the French): But it was captured by a site user and posted to Reddit...
 - In 2018 "Welcome to Video" was busted: Pay for CSAM with Bitcoin! Again, if you want to vomit read the indictments
- Communities create dangerous cycles of normalization
 - And larger communities are more dangerous:
See more mild versions that happened on Reddit with TheDonald, jailbait, creepshots, etc...
 - Self reinforcement behavior: "Its normal because others in the community do it" and the community becomes self justifying
 - See the "Jailbait" analysis in ***Twitter and Tear Gas***
 - Drives to extremes: Over the past decade, the age of CSAM victims has basically gotten younger... To the point where average age really can't get much lower

The Problem #2:

The Tor Project ***JUST DOES NOT CARE!***

- They treat this as "collateral damage" with a series of excuses. Here are actual justifications by Roger Dingledine (Founder):
- "But hidden services are in their infancy"
 - And in the same presentation talk about it being a 10 year old idea...
- "But hidden services are end-to-end authenticated"
 - Yeah, there is this thing call TLS...
- "But hidden services work through NATs"
 - Yeah, there is this thing called uPNP: You ask the NAT to allow inbound connections
 - Oh, or just use EC2...
- "But dissidents..."
 - Well, running Tor is very noticeable...
 - Plus you can "arbitrage host": Want to piss off China? Host in the US. Piss off the US? Host in Russia...
- "But Facebook/SecureDrop/Etc... has an onion service"
 - Uh, they don't actually need to be hidden! And work better when they aren't!

And A Different Problem: Grooming

- I never encountered Agent Schwarzenberger, but this was her specialty...
people who use electronic chat to groom child victims for exploitation
- In unencrypted chats, the chat-provider can ***theoretically*** try to detect this behavior
 - A case where classic Machine Learning tends to work pretty well if the results are human-reviewed for false-positives
- The problem grows even harder when dealing with encrypted chats
 - Since there is no longer a central server that can try to detect the behavior...
 - And the developers would probably resist adding an AI-snitch to the client

So Remember:

Child Abuse IS REAL

- Too often those in favor of security/privacy view claims about child abuse and CSAM as disingenuous...
 - It isn't helped that those wanting encryption backdoors will use claims about child abuse and CSAM in a disingenuous manner!
- But these problems are real
 - Grooming over messenger systems is a serious problem
 - Usually starting over some open system where children frequent...
 - Before moving onto encrypted messengers like iMessage and Facebook Messenger
 - Tor hidden services have created a CSAM "industry"