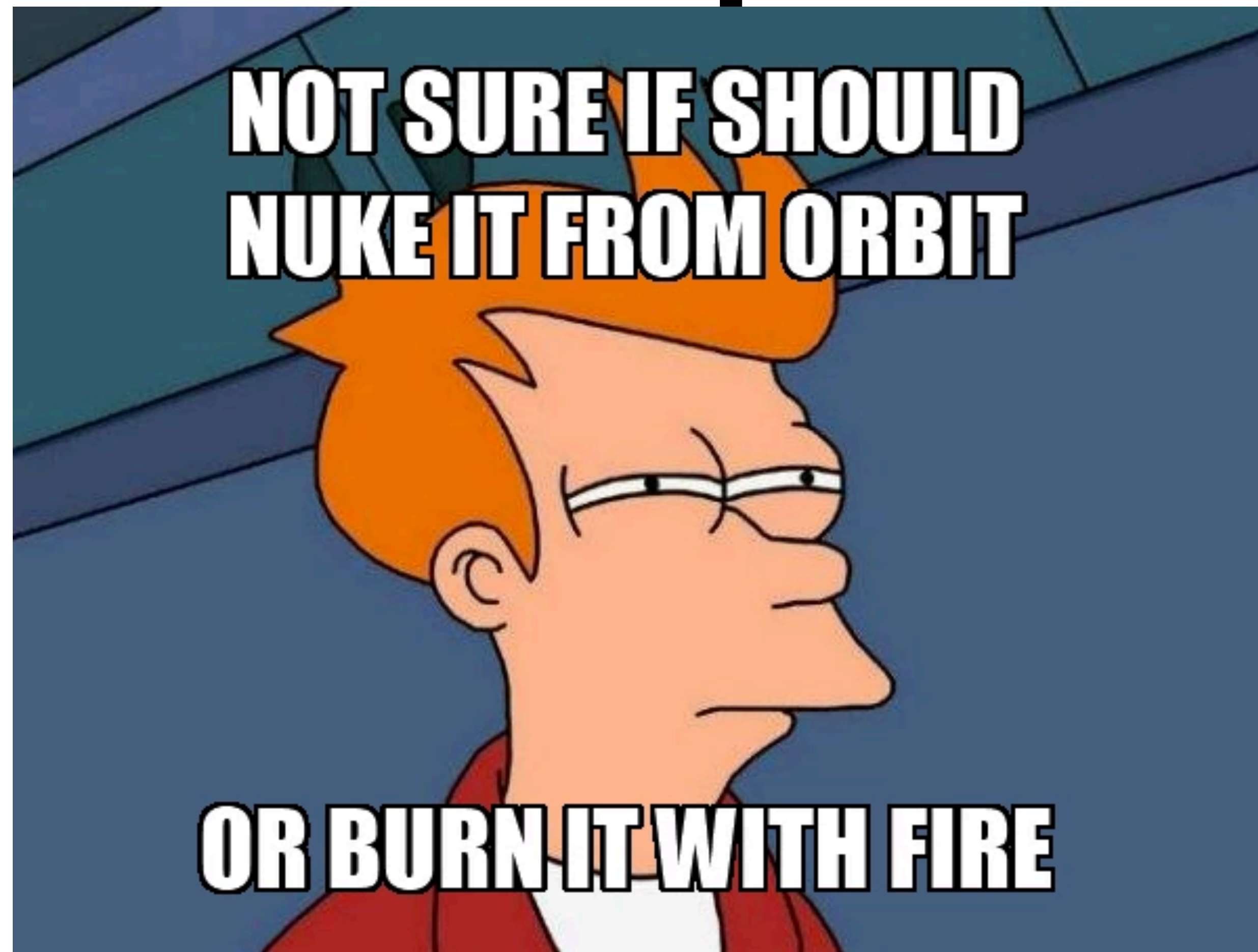


Misc Topics 2



Pre Lecture Facepalm...

From just a year ago!

Computer Science 161 Fall 2020

Weaver

Home / Cisco



Cisco



MENU



Cisco

Products & Services / Security /

Cisco Firepower Management Center

Hi

Summ

A vulnera
authentic
the *root* u

The vulne

could exp

allow an attacker to execute arbitrary cor



Centralize, integrate, and simplify management

This is your administrative nerve center for managing critical Cisco network security solutions. It provides complete and unified management over firewalls, application control, intrusion prevention, URL filtering, and advanced malware protection. Easily go from managing a firewall to controlling applications to investigating and remediating malware outbreaks.

[Watch 3-minute overview](#)[Watch demo now](#)

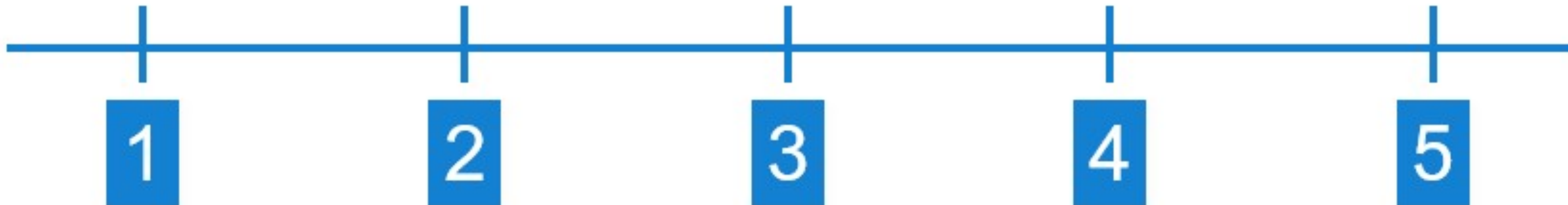
These vulnerabilities exist due to improper input validation. An attacker could exploit these vulnerabilities by sending crafted SQL queries to an affected device. A successful exploit could allow

Welcome to Hell Week....

- Mental pressure of "Curl up in a ball with a rifle" vs "Pretend everything is normal" continues...
- Highly likely Biden will be the winner:
But we won't know for sure for another few days at least....
- Really a massive screwup:
We ***should*** just have preliminary results announced on Friday at once for all states
- Today is More Off Topic Stuff:
 - Nukes
 - Tor Hidden Services
 - Sidechannels

And Checking In With Everyone Again...

How Are You on the Fauci Scale?



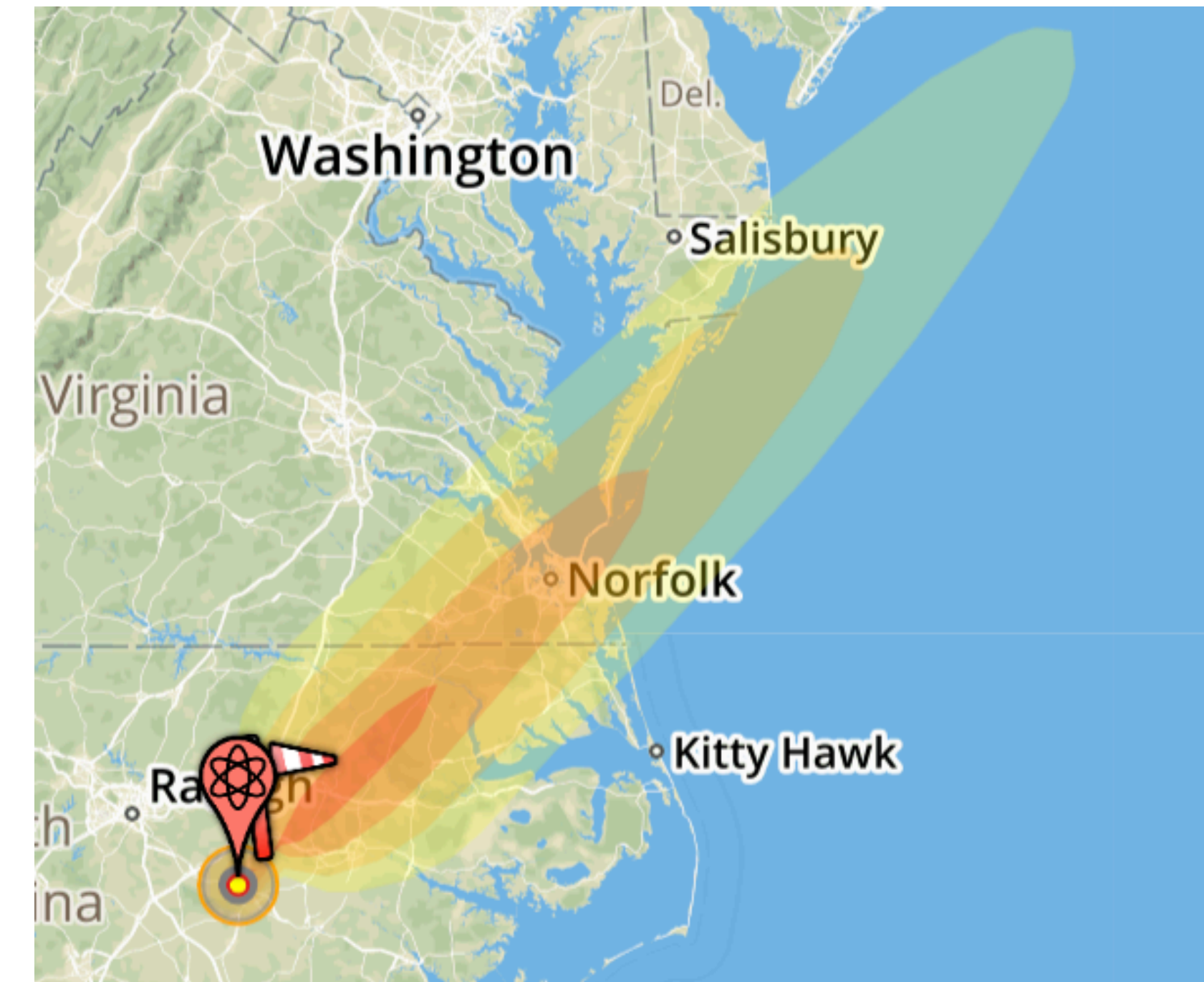
The Interesting Problem: Limiting Use

- Who might use a nuke without authorization?
 - Our "allies" where we station our nukes
 - Original motivation: Nukes stored in Turkey and Greece
 - Someone who can capture a nuke
 - This is what sold the military on the need for the problem:
We had nukes in Germany which **would** be overrun in case of a war with the USSR
 - Our own military
 - General Jack D Ripper scenario
- The ***mandated*** solution:
 - Permissive Access Link (PAL)



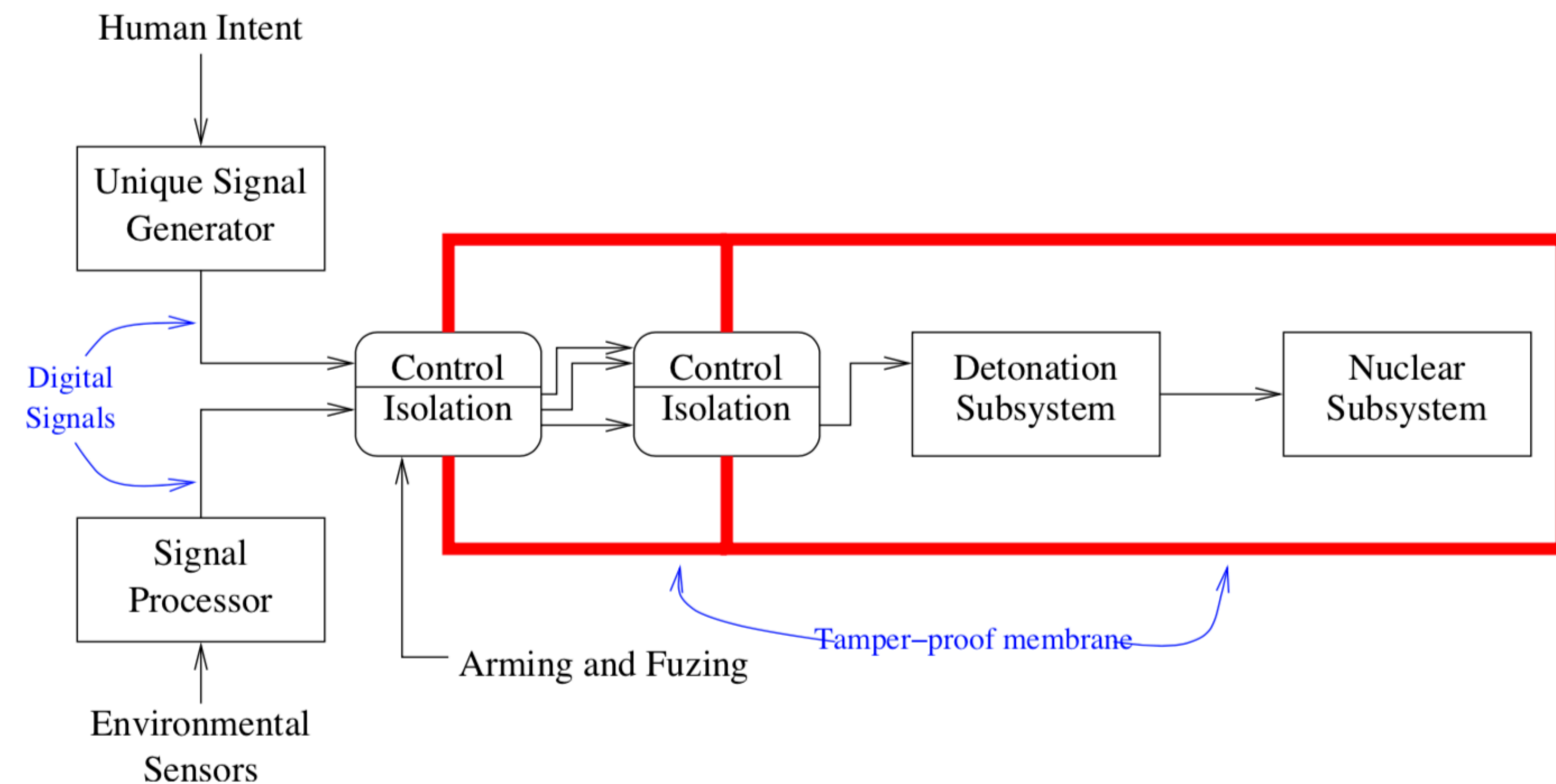
Nuke Safety Features

- One-point safety – no nuclear yield from detonation of one explosive charge.
- Strong link/weak link –
 - strong link provides electrical isolation;
 - weak link fails early under stress (heat, etc.)
- Environmental sensors – detect flight trajectory.
- Unique signal generator – digital signal used for coupling between stages.
- Insulation of the detonators from electrical energy.
- “Human intent” input.
- Tamper-resistant skin
- Use Control Systems
- Not always the case: In 1961 in South Carolina a B52 broke up
 - One of the two 4MT bombs **almost** detonated on impact, since it thought it was being dropped!



Bomb Safety Systems

- We have a "trusted base"
- Isolated inside a tamper-detecting membrane
 - Breach the membrane -> disable the bomb
- We have human input
 - Used to generate a signal saying "its OK to go boom"
 - The user interface to the PAL can follow the same path/concepts
- We have critical paths that we can block
 - Complete mediation of the signal to go boom!



Unique Signal Generator

- Part of the strong link
 - Prevent any detonation without clear, unambiguous showing of “human intent”
- A **safety** system, not a security system
- Looks for a 24-bit signal that is extremely unlikely to happen during any conceivable accident. (Format of input bits not safety-critical)
 - Accidents can generate random or non-random data streams
 - Desired signal pattern is unclassified!
- Unique signal discriminator locks up on a **single** erroneous bit
- At least partially mechanical

PALs

- Originally electromechanical. (Some weapons used combination locks!)
- Newest model is microprocessor-based. There may still be a mechanical component.
 - Recent PAL codes are 6 or 12 digits.
- The weapon will permanently disable itself if too many wrong codes are entered.
- PALs respond to a variety of codes – several different arming codes for different groups of weapons, disarm, test, rekey, etc.
- It was possible, though difficult, to bypass early PALs.
 - Some even used false markings to deceive folks who didn't have the manual.
- It does not appear to be possible to bypass the newest "CAT F" PAL.
 - Modern bombs don't work without the tritium boost-gas:
If you blow the gas you disable the nuke. Don't know if this is done or not

How are PALs built?

- We don't know, but some informed speculation from Steve...
- It is ***most likely*** based around the same basic mechanism as the unique signal generator
 - Gives a single point of control already in the system
 - Reports about it indicate that it was successfully evaluated in isolation
 - Take advantage of the existing trusted base of the tamper-resistant barrier around the warhead to protect the device

Deployment History

- Despite Kennedy's order, PALs were not deployed that quickly.
 - In 1974, there were still some unprotected nukes in Greece or Turkey
- PALs and use control systems were deployed on US-based strategic missiles by then
 - But the launch code was set to 00000000
 - Rational: the Air Force was more worried about failure to launch!
- A use control system was added to submarine-based missiles by 1997
- In 1981, half of the PALs were still mechanical combination locks

Steve Bellovin's Lessons Learned

- Understand what problem you're solving
- Understand ***exactly*** what problem you're solving
- If your abstraction is right:
you can solve the key piece of the overall puzzle
- For access control, find the One True Mandatory Path —
and block it.
 - And if there is more than one, you're doing it wrong!
- What is the real TCB of our systems?

Side Channels & Other Hardware Attacks: Worry

- A side channel attack requires measuring some other piece of information
 - EG, time, cache state, power consumption, etc...
- And using it to deduce a secret about the system
- Side channels are very, **very** powerful

Requirements

- Often the biggest limitation is attacker requirements
- Timing attack
 - Need to measure the timing of the operation with potentially very high precision
- Power attack
 - Need physical access to the device:
Generally only applicable to smart-cards and similar devices
- EMF ("Tempest")
 - Need close physical access
- Processor side-channel attacks
 - Need to co-locate the attacker code:
EG, cloud computing, web browsers, etc

Example Timing Attack: Keystrokes...

- User is inputting a password
 - And the user is using a Bluetooth keyboard...
 - Or the user is using a remote connection over ssh
- Someone nearby can observe when keys are pressed
 - They are sent immediately
 - But not ***what*** keys are pressed
- Can this leak sensitive information? Of course!

Timing Leakage

- Some keys are faster to press
- Can use this to model timing
 - Either generically or specific to the user
- Lots of ways to do this
 - Hidden markov models
 - Throw machine learning at it...
- Really really hard to hide
 - Can't delay interactive requests without adding latency
 - "Cover traffic" only adds additional data, can't remove the underlying signal
- From <https://people.eecs.berkeley.edu/~daw/papers/ssh-use01.pdf>

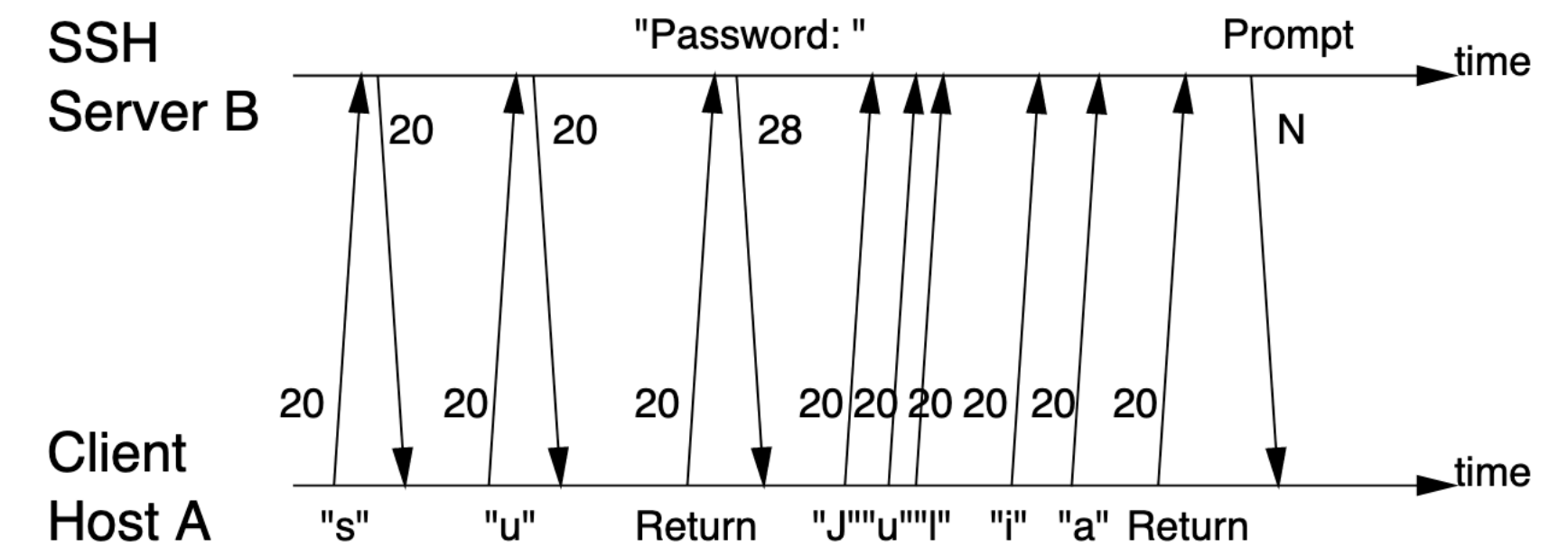


Figure 1: The traffic signature associated with running SU in a SSH session. The numbers in the figure are the size (in bytes) of the corresponding packet payloads.

Timing Attacks & Cryptography

- The classic timing attack:
 - Compute $y^x \bmod n$
- Easy solution ends up being

```
Let  $s_0 = 1$ .  
For  $k = 0$  upto  $w - 1$ :  
  If (bit  $k$  of  $x$ ) is 1 then  
    Let  $R_k = (s_k \cdot y) \bmod n$ .  
  Else  
    Let  $R_k = s_k$ .  
  Let  $s_{k+1} = R_k^2 \bmod n$ .  
EndFor.  
• Return  $(R_{w-1})$ .
```

- <https://www.paulkocher.com/TimingAttacks.pdf>

Implications: Public Key Operations Need "Constant Time"

- Optimizing cryptographic code can be dangerous...
 - Instead it needs to take the same amount of time no matter what the input is
 - Even compiler optimizations can be a problem
- First identified 20 years ago...
 - So you think we'd have solved it...
But you'd be wrong

Reminder DSA/ECDSA Brittleness...

- DSA algorithm
 - Global parameters: primes p and q , generator g
 - Message m , private key x , public key $y=g^x \bmod p$
 - Sign: select random k from 1 to $q-1$
 $r = (g^k \bmod p) \bmod q$ (retry if $r = 0$)
 $s = (k^{-1} (H(M) + xr)) \bmod q$ (retry if $s = 0$)
- k needs to be random and secret and unique
 - An attacker who learns or guesses k can find x
 - An attacker can even just try all possible k s if the entropy of k is low
 - Even just learning a few bits of k , and then having several signatures with different k for each one, and you break it!

Just ***A YEAR AGO***: The Minerva Attack



- A timing side-channel attack to get a few bits of k from the ECDSA signatures on Athena smart cards and lots of others
 - So have the smart card generate a lots of signatures
 - Then some math and brute force to get the actual x
- These devices were certified...
Including that they were supposed to resist timing attacks!
 - But, naturally, the certification doesn't actually test whether they are vulnerable to timing attacks...
- The root cause for many was a common code component:
The Atmel Toolbox 00.03.11.05 library

Guess the Problem Here...

- M10.6 the TSF shall provide digital signature confirming to EC-DSA standard.
 - Secure digital signature generate
 - Secure digital signature verify
 - Fast digital signature generate (**see note***)
 - Fast digital signature verify (**see note***)
 - M10.7 the TSF shall provide point multiplication on an elliptical curve, conforming to EC-DSA standard.
 - Secure multiply
 - Fast multiply (**see note***)
- * The **Fast** functions of M10.3, M10.4, M10.5, M10.7, M10.8, M10.9, do not offer any DPA/SPA protection and **must not** be used for secure data.

Guess the Problem Here...

- M10.6 the TSF shall provide digital signature confirming to EC-DSA standard.
 - Secure digital signature generate
 - Secure digital signature verify
 - Fast digital signature generate (**see note***)
 - Fast digital signature verify (**see note***)
 - M10.7 the TSF shall provide point multiplication on an elliptical curve, conforming to EC-DSA standard.
 - Secure multiply
 - Fast multiply (**see note***)
- * The **Fast** functions of M10.3, M10.4, M10.5, M10.7, M10.8, M10.9, do not offer any DPA/SPA protection and **must not** be used for secure data.

Once Again: Bad API

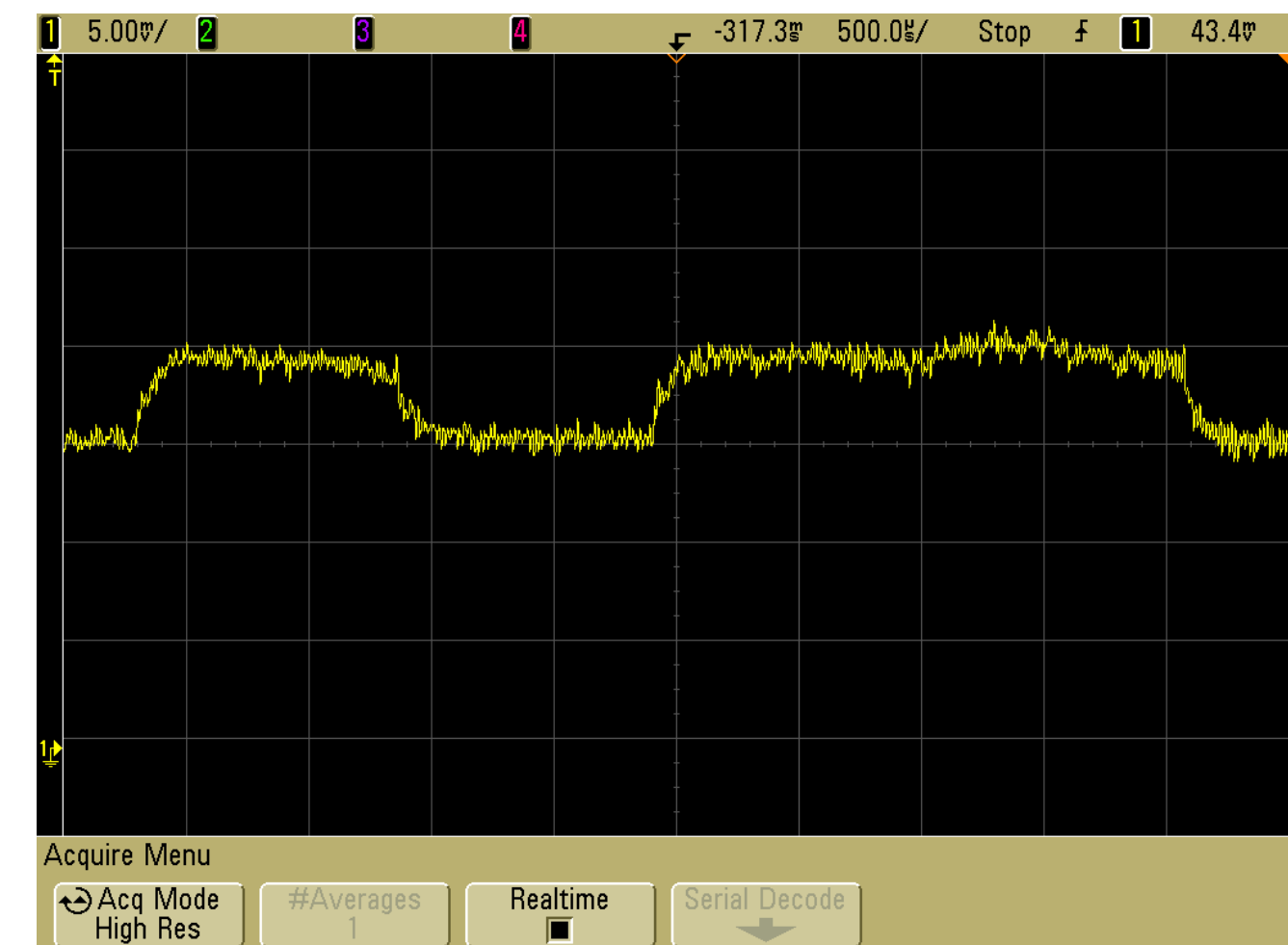
- Once again we have a case of “If you offer a programmer two ways, >50% of the time they will chose the wrong way”
 - In this case “why wouldn’t I chose the fast version?”
- You have a now growing list of “red flag/canary APIs”
 - `system()`, raw SQL, now this example
- Keep a growing list as a “cheat sheet”
- When you get to an existing software project...
 - Search the code for these APIs
- When you start a new project
 - **NEVER** use the dangerous version, even if you are using it safely... (EG, never use `system()`, only `execve()`)

Power Attacks: The Bane of Smart Cards...

- Smart Cards are effectively small computers
 - In a handy credit-card sized package...
- Some are used to hold secrets on behalf of the cardholder
 - So really, if the person holding the card can get the secrets, 🙋
- Some are used to hold secrets **from** the cardholder
 - So if the user can extract the secrets, 🙊
- The bane: Power Analysis
 - SPA == Simple Power Analysis
 - DPA == Differential Power Analysis

The Idea...

- Different operations use different amounts of power
 - EG, square vs multiply in RSA
- Hook up smart card to a reader that can measure the power
- Have it encrypt/sign something
- Look at the power trace to get information about hidden secrets
 - Including statistical techniques



Countermeasures...

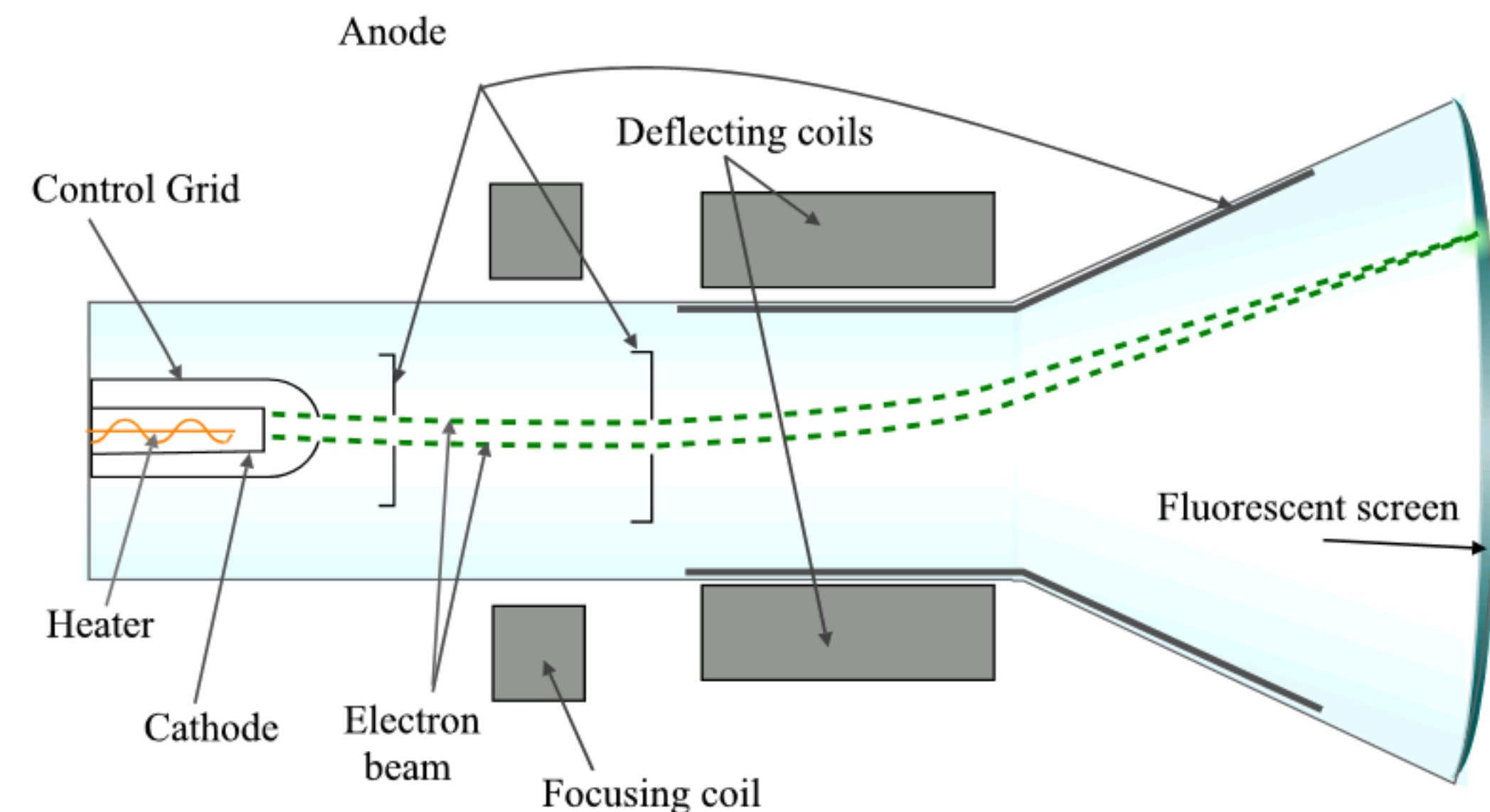
- Lots of work can make "simple" power analysis not work
 - But now you are using more power: Have to use the max all the time for the encryption
- Harder for more detailed differential analysis
 - Which can detect even small leaks
- If possible, punt!
 - Use your systems in a way where the person who holds the card is not your adversary!
- EG, you are building a “stored value” smart card
 - Option #1: The smart card has the value:
If you tamper with the smart card, you can change the value
 - Option #2: The smart card just has an ID:
You actually look up in the central database

Real Freaky: Electromagnetic Emissions...

Computer Science 161 Fall 2020

Weaver

- Every time a circuit switches...
 - It leaks out some radio frequency energy
- Some sources are even easier
 - A old-school monitor paints the image with an electron beam on the screen...
- Which means it is a radio!
 - Transmitting an image of the screen!
- Cheap, too
 - \$15 in 1984 for van Eck to read images off a monitor!



By Theresa Knott - en:Image:Cathode ray Tube.PNG,
CC BY-SA 3.0, <https://commons.wikimedia.org/w/index.php?curid=100143>

Solution:

The SCIF

- The US government's paranoia: The SCIF (Sensitive Compartmented Information Facility)
 - A room (or even a whole building) specifically designed for Top Secret "stuff"
 - Paranoia further enhanced by incidents like the "Project Gunman" Bug
- Multiple layers of security:
 - Physical access to the building
 - No outside electronics
 - With some caveats, fit bits can be OK depending...
 - No windows
 - Beam a laser at a window and can detect vibrations!
 - Electromagnetic shielding
 - So your cellphone wouldn't work in there anyway

And An Asside: The Second Coolest Bug ***EVER!***

Computer Science 161 Fall 2020

Weaver

- The "Project Gunman" bug
 - <https://www.cryptomuseum.com/covert/bugs/selectric/>
 - "Project Gunman" was the NSA effort to **remove** the bug...
- In the late 70s and early 80s, the USSR bugged the electric typewriters in the US embassy!
 - Modify the mechanism that selects which character the print head goes to with magnetically tagged pieces
 - Hide a pickup & transmitter in an aluminum support rail
 - Broadcast really close in spectrum to a major TV station
- We call this a "keylogger" when done in software



Project GUNMAN exhibit at the National Cryptologic Museum, Ft. Meade, MD, showing the metal bar that concealed the typewriter bug

And Funky Hardware SideChannels...

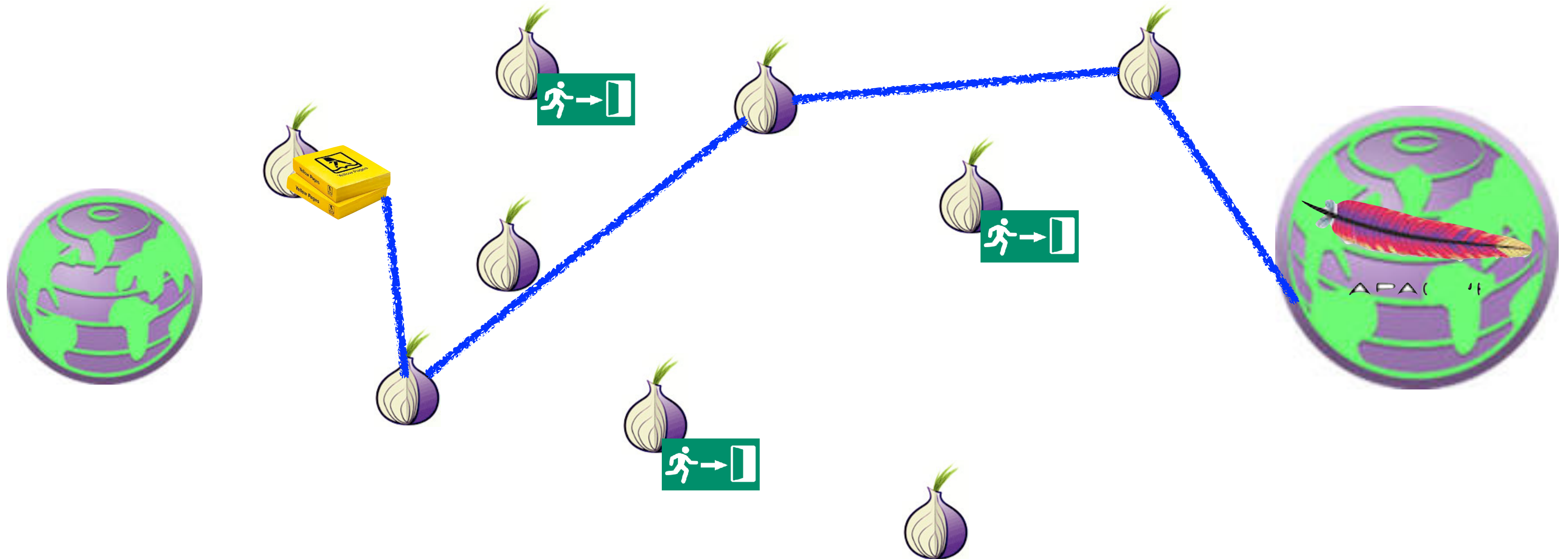
- The recent Meltdown and Spectre Intel bugs...
 - Both were effectively side-channels
- The key idea:
 - You could trick the speculative execution engine to compute on memory that you don't own
 - And that computation will take a different amount of time depending on the memory contents
- So between the two, you could read past isolation barriers
 - Meltdown: Read operating system (and other) memory from user level
 - Spectre: Read in JavaScript from other parts of the web browser

The Dark Web:

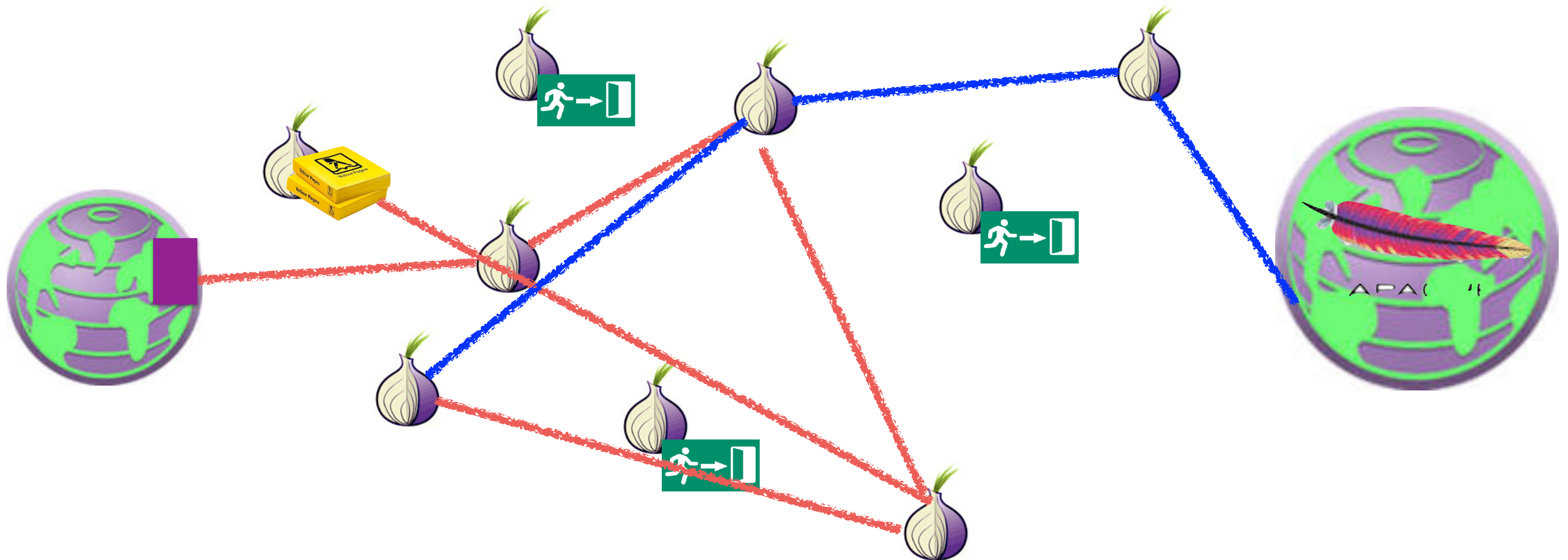
Tor Hidden Services aka .onion sites

- Services that **only** exist in the Tor network
 - So the service, not just the client, has possible anonymity protection
 - The “Dark Web”
- A **hash** of the hidden service's public key
 - <http://pwoah7foa6au2pul.onion>
 - AlphaBay, one of many dark markets
 - <https://facebookcorewwi.onion>
 - In this case, Facebook spent a lot of CPU time to create something distinctive
- 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

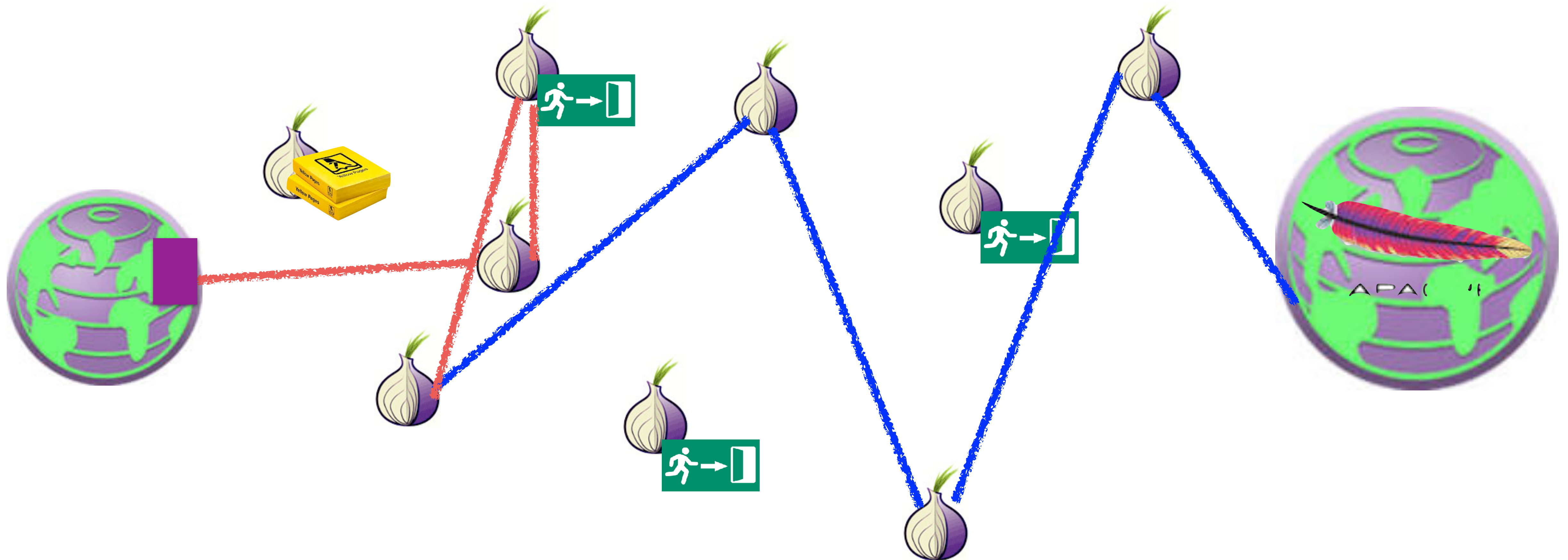
Tor Hidden Service: Setting Up Introduction Point

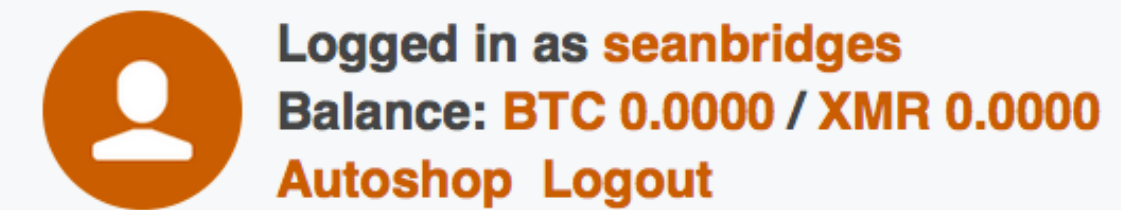


Tor Hidden Service: Query for Introduction, Arrange Rendezvous



Tor Hidden Service: Rendezvous and Data





[HOME](#)
[SALES](#)
[MESSAGES](#)
[ORDERS](#)
[LISTINGS](#)
[BALANCE](#)
[FEEDBACK](#)
[FORUMS](#)
[API](#)
[SUPPORT](#)



Joined:	Aug 30, 2016
Trust level:	Level 1
Total sales:	USD 0.00
Total orders:	USD 0.00

Search

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

Featured Listings



[FE 100%]

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

**[MS] EDITABLE HQ
TEMPLATES OF
DOCUMENTS
RE WORLDWIDE - GET
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!)

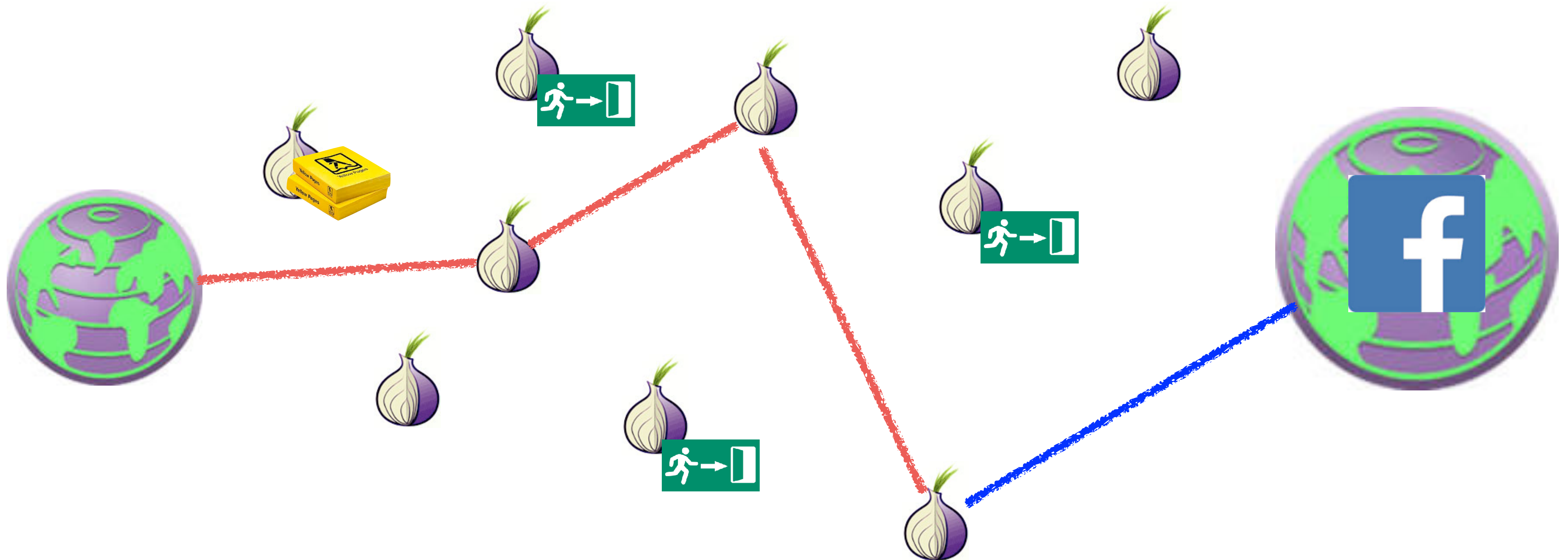
183848 - Other -
BitcoinThief

Buy: USD 600.00

Remarks...

- Want to keep your 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", CDNs like CloudFlare, or suitable "foreign" machine rooms:
And since CloudFlare will service the anti-Semitic shithheads like gab.ai and took forever to get rid of the actual nazis of Stormfront and the murderous shits of 8chan...
- Dark Markets
 - Marketplaces based on Bitcoin or other alternate currency
- Cybercrime Forums
 - Hoping to protect users/administrators from the fate of earlier markets
- Child Exploitation

The Dark Market Concept

- Four innovations:
- A censorship-resistant payment (Bitcoin)
 - Needed because illegal goods are not supported by Paypal etc
 - Bitcoin/cryptocurrency is the ***only game in town*** for US/Western Europe after the Feds smacked down Liberty Reserve and eGold
- 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
- 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” (e.g. 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 then the Child Exploitation types

- This is ***why*** I'm quite happy to see Tor Hidden Services ***burn!!!***
 - Because these do represent a serious problem:
The success against "PlayPen" shows just how major these are
- A far bigger systemic problem than the dark markets:
 - Dark markets are low volume, and not getting worse
 - Plus the libertarian attitude of "drug users are mostly harming themselves, its the drug-associated crime that is the problem"
 - No indication of any ***successful*** murder resulting from dark market activity
 - But these are harming others
- They are also harming Tor:
Tor itself is a very valuable tool for many legitimate uses, but the presence of the child exploitation sites on hidden services is a stain on Tor itself

Deanonymizing Hidden Services: Hacking...

- Most dark-net services are not very well run...
 - Either common off-the-shelf drek or custom drek
- And most have now learned ***don't ask questions on StackOverflow***
 - Here's looking at you, frosty...
- So they don't have a great deal of IT support services
 - A few hardening guides but nothing really robust

Onionscan...

- A tool written by Sarah Jamie Lewis
 - Available at <https://github.com/s-rah/onionscan>
- Idea is to look for very common weaknesses in Tor Hidden services
 - Default apache information screens
 - Web fingerprints
 - I believe a future version will check for common ssh keys elsewhere on the Internet
- Its really "dual use"
 - .onion site operators should use to make sure they aren't making rookie mistakes
 - Those investigation .onion sites should use to see if the target site made a rookie mistake!

Deanonymizing Visitors To Your Site

FBI Style

- Start with a Tor Browser Bundle vulnerability...
 - Requires paying for a decent vulnerability:
Firefox lacks sandboxing-type protections but you have to limit yourself to JavaScript
- Then take over the site you want to deanonymize visitors to...
- And simply hack the visitors to the site!
 - With a limited bit of malcode that just sends a “this is me” record back to an FBI-controlled computer



A History of NITs

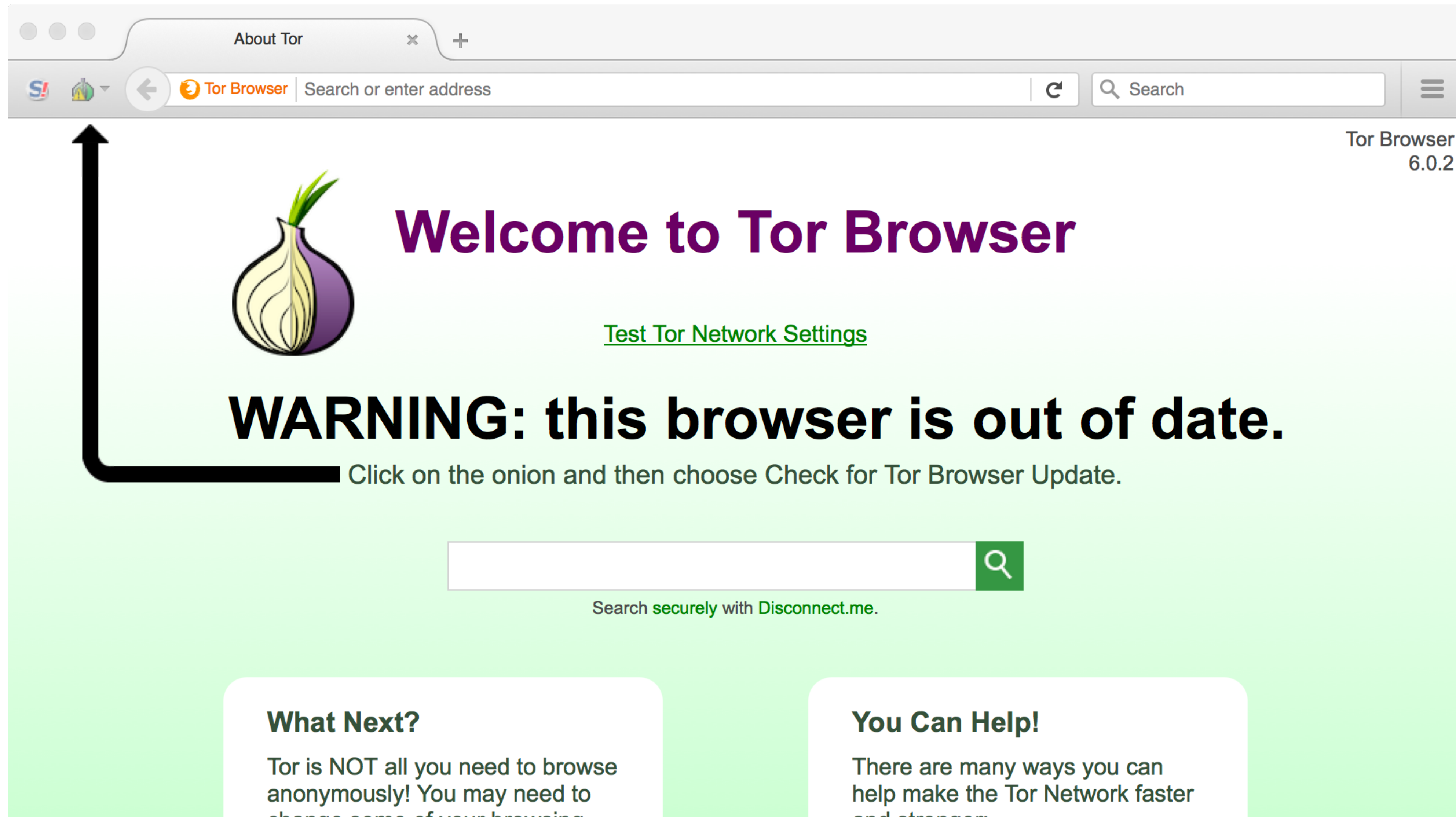
- The FBI calls their malicious code a NIT or Network Investigatory Technique
 - Because it sounds better to a magistrate judge than saying "we're gonna go hacking"
- The exploit attempts to take over the visitor's browser
- But the payload is small: just a "I'm this computer" sent over the Internet to an FBI controlled Internet address

A History of NITs: PedoBook

- The first known NIT targeting a hidden service was “PedoBook” back in 2012
 - Back then, many people used other web browsers to interact with Tor hidden services
 - NSA kept a database of these people, called ***EPICFAIL!***
- The NIT actually didn’t even qualify as malware
 - And a ***defense*** expert actually argued that it isn’t hacking and probably didn’t actually need a warrant
- Instead it was the “Metasploit Decloaking” flash applet:
 - A small bit of Flash which contacts the server directly, revealing the visitor’s IP address

A History of NITs: Freedom Hosting

- The second big NIT targeted FreedomHosting
 - A hosting provider for Tor Hidden services with an, umm, generous policy towards abuse
 - Hosted services included TorMail (a mail service through Tor) and child porn sites
- FBI replaced the entire service with a NIT-serving page
- Fallout:
 - Very quickly noticed because there are multiple legit users of TorMail
 - Targeted an older Firefox vulnerability in Tor Browser
- Tor browser switched to much more aggressive autoupdates:
Now you ***must*** have a zero-day for a NIT payload to work




The screenshot shows the Tor Browser interface. At the top, there is a tab labeled "About Tor" and a search bar with the text "Search or enter address". Below the search bar, the Tor Browser logo (an onion) is displayed. To the right of the logo, the text "Tor Browser 6.0.2" is visible. The main content area features a large purple heading "Welcome to Tor Browser" and a green link "Test Tor Network Settings". A prominent black warning message reads "WARNING: this browser is out of date." with a subtext "Click on the onion and then choose Check for Tor Browser Update." A black arrow points from the warning message to the onion logo. Below the warning, there is a search bar with a green search button and the text "Search securely with Disconnect.me." At the bottom, there are two white boxes with rounded corners. The left box is titled "What Next?" and contains the text "Tor is NOT all you need to browse anonymously! You may need to change some of your browsing...". The right box is titled "You Can Help!" and contains the text "There are many ways you can help make the Tor Network faster and stronger."

About Tor

Tor Browser | Search or enter address

Tor Browser 6.0.2



Welcome to Tor Browser

[Test Tor Network Settings](#)

WARNING: this browser is out of date.

Click on the onion and then choose Check for Tor Browser Update.

Search securely with Disconnect.me.

What Next?

Tor is NOT all you need to browse anonymously! You may need to change some of your browsing...

You Can Help!

There are many ways you can help make the Tor Network faster and stronger.

A History of NITs: Playpen

- The big one: PlayPen was a hidden service for child pornographers
 - In February 2015, the FBI captured the server and got a warrant to deploy a NIT to logged in visitors
 - The NIT warrant is public, but the malcode itself is still secret: >100,000 logins!
- What we do know:
 - This was big: hundreds of arrests, many abuse victims rescued
 - It almost certainly used a zero-day exploit for Tor Browser
- Courts are still hashing this out over two big questions
 - Is it valid under Rule 41?
 - **Most** have conclude "no, but a technical not constitutional flaw":
Good faith says that previous violations are OK, but not future violations
 - Does the defense have a right to examine the exploit?
 - I'll argue no, but some defense attorneys have successfully used a graymail technique initially
But followup hasn't replicated that success

A History of NITs: Three Years Ago

- Someone (probably the French police) captured a child porn site called the "GiftBox"
- They modified it to serve up a NIT
- The NIT payload was almost identical to the one in the Freedom Hosting case
 - Suggesting assistance from either the FBI or the FBI's contractor
- The exploit was a ***new*** zero-day exploit targeting Firefox
 - Patch released within ***hours***
 - And yes, it was a C-related memory corruption (naturally)

NITs won't work well in the future against Tor!

- The current Tor browser is a ***hard*** target
- Hardening will require that breaking Tor browser, even to just send a "I'm here" message, will require a chain of exploits
 - An information leakage to determine the address of a function and enough content in that function to enable an attack (break ASLR)
 - PLUS a conventional vulnerability
 - And now the Firefox rendering engine got sandboxed too...
 - And add in darknet users who are running without JavaScript
- Upshot: the current FBI exploit will need a massive upgrade if it will work at all!
 - And future exploits will be ***vastly*** more expensive and rarer
 - We should thank the FBI for their very valuable contributions to software hardening