

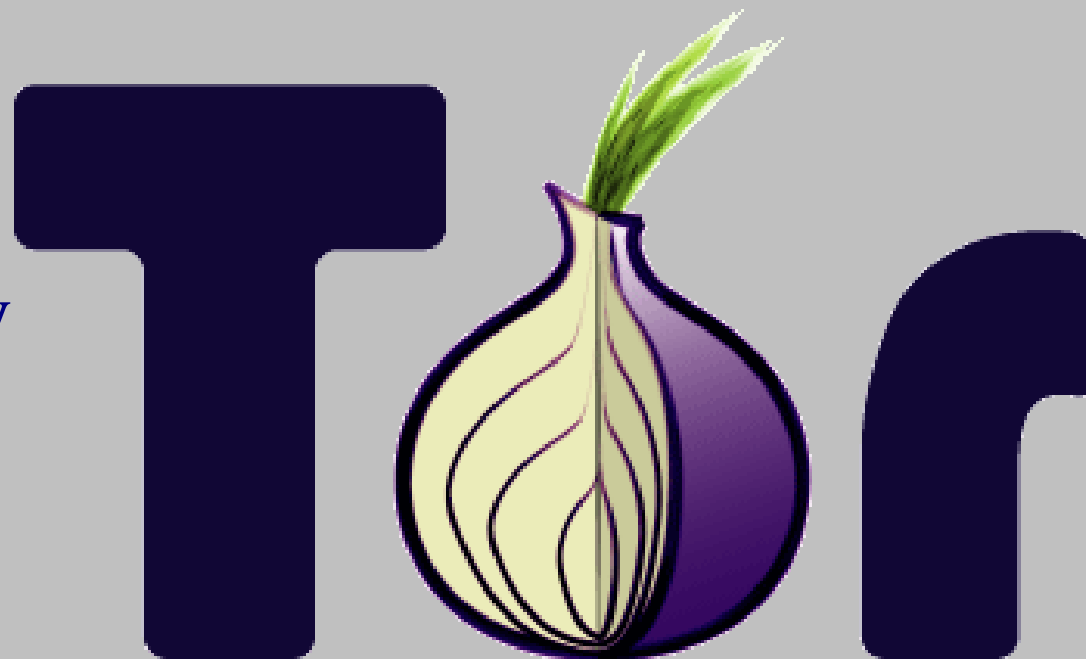
Securing the Tor Network

Mike Perry

Riverbed Technology

Black Hat USA 2007

Defcon 2007



Who am I?

- Volunteer Tor developer
- Forward+Reverse engineer
- Employed by Riverbed (shameless plug)
 - Leading manufacturer of WAN accelerators
 - 20-200X (not percent. X) improvement of CIFS
 - 5-50X improvement of MAPI/Exchange
 - Protocol independent data reduction
 - > 90% head to head win rate
 - Outselling Cisco accelerators 2:1

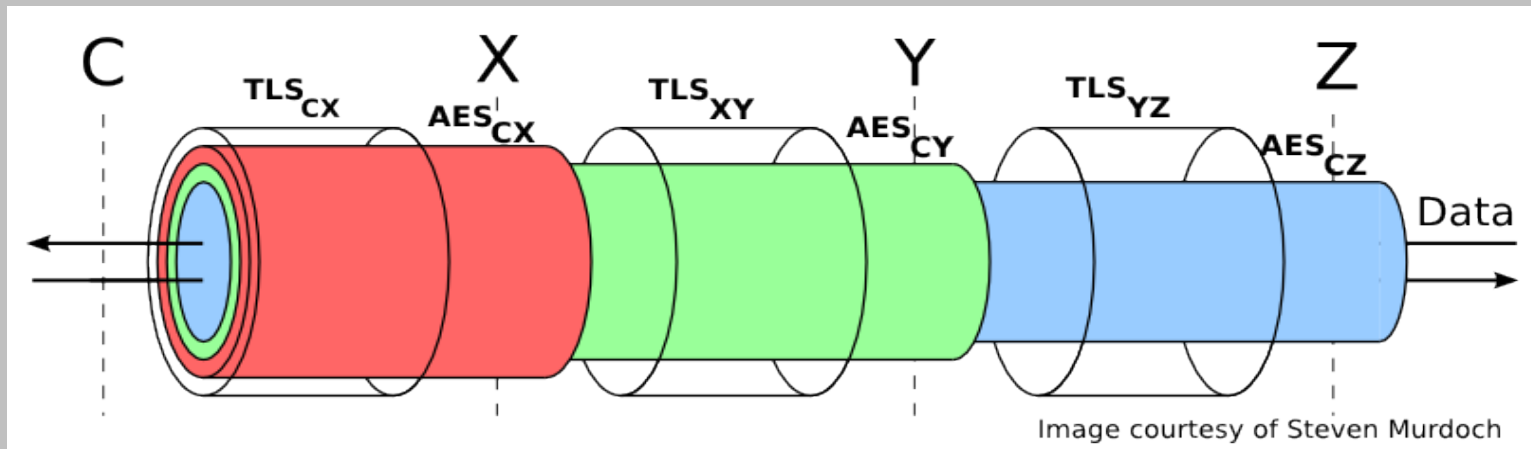
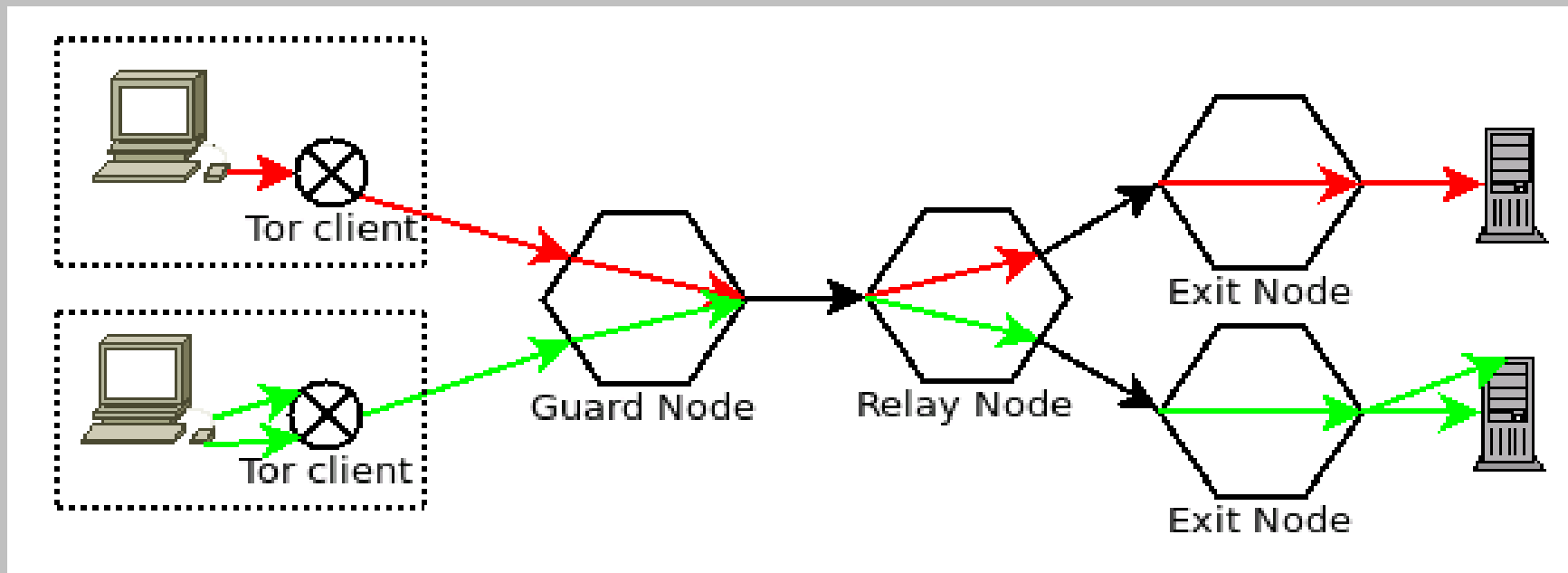
Preaching to the Choir

- Don't yet understand consequences of having lives+thoughts archived by IP, bought and sold
- Google may not be (that) evil, but what about ISPs, other search engines?
- Information can come back to bite in unexpected ways
 - Divorce cases
 - Lawsuits
 - Catalogs/Spam

What is Tor?

- Volunteer run relay network designed for privacy, anonymity, and censorship resistance.
- Client acts as SOCKS proxy
- Relays TCP connections (“streams”)
 - Multiplexed on encrypted paths (“circuits”)
- Circuits multiplexed over node-to-node TLS/SSL
- Circuits route through 3 nodes
 - “Guard”, “relay”, “exit”

Tor Routing



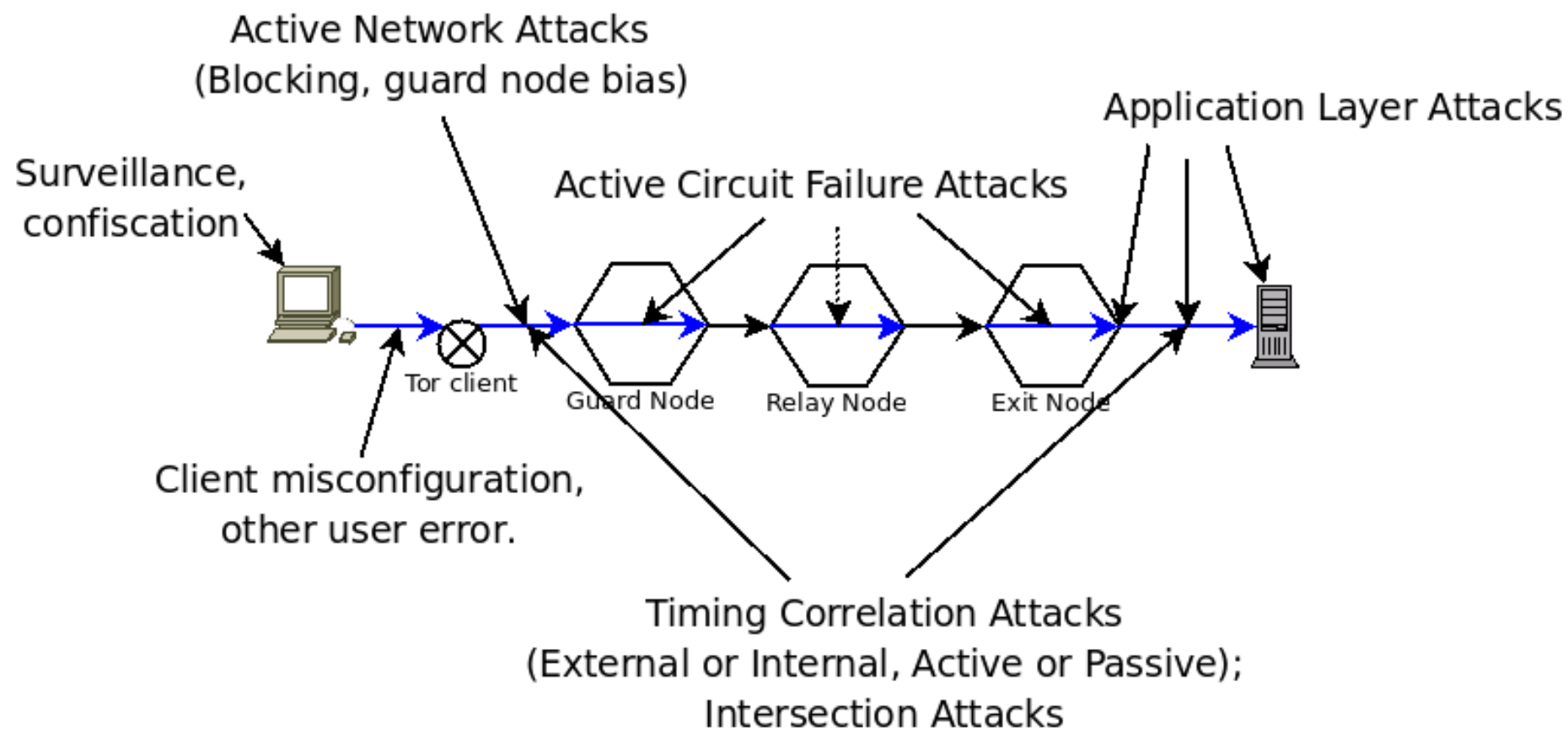
Classes of Attack

- Passive attacks
 - Packet and connection timing correlation
 - Fingerprinting of traffic/usage patterns
 - “Intersection Attacks” of multiple attributes of users
- Active attacks
 - Lying about bandwidth to get more traffic
 - Failing circuits to bias node selection
 - Modifying application layer traffic at exit

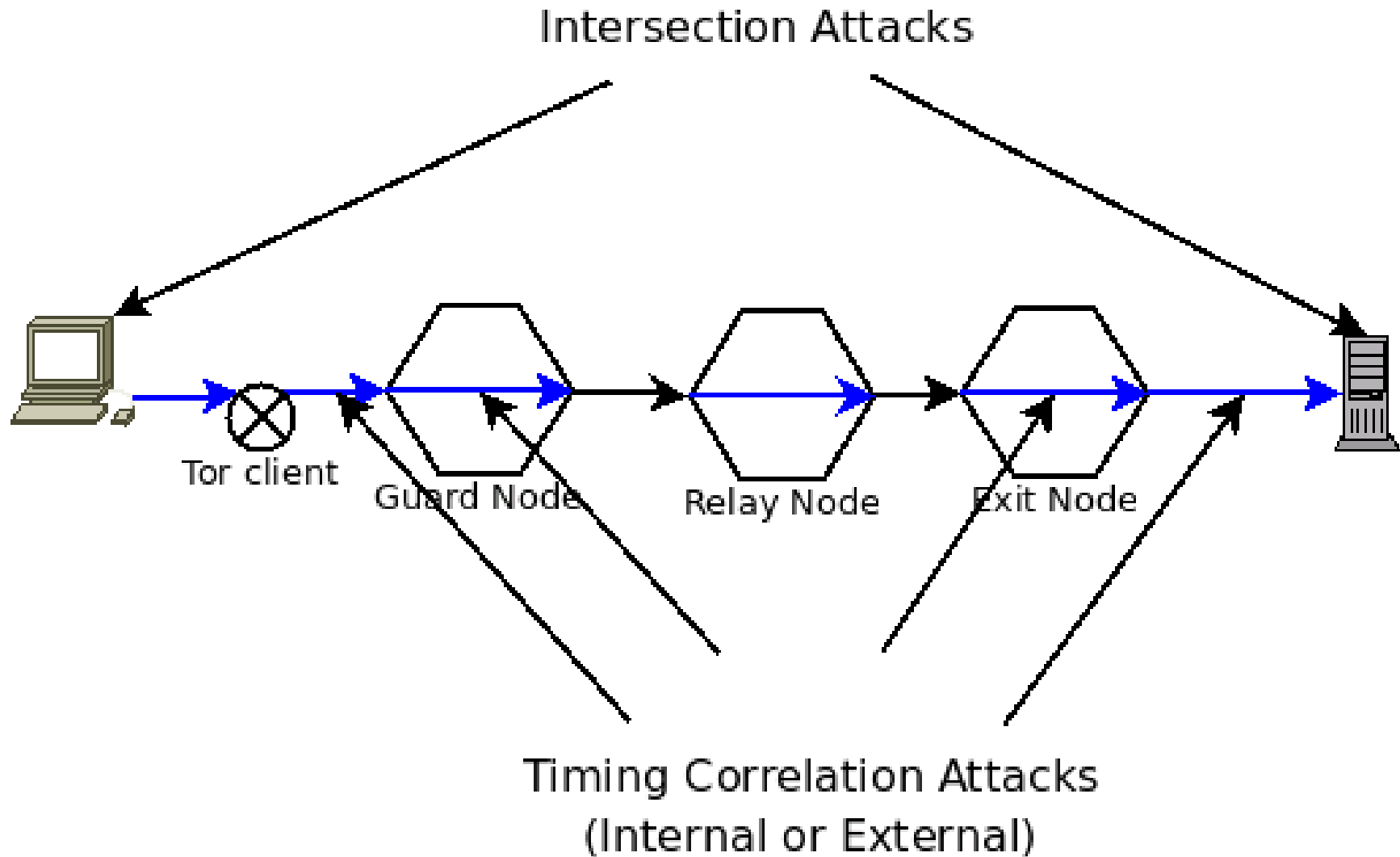
Position of Attack

- Internal
 - Node operator
 - Can differentiate circuits at guard and relay.
 - Able to differentiate streams per circuit at exit
- External
 - ISP or Echelon-style adversary
 - Assumed to be unable to see inside TLS streams
 - Likely frustrated to a large degree by running Tor as both node and client

Attack Points

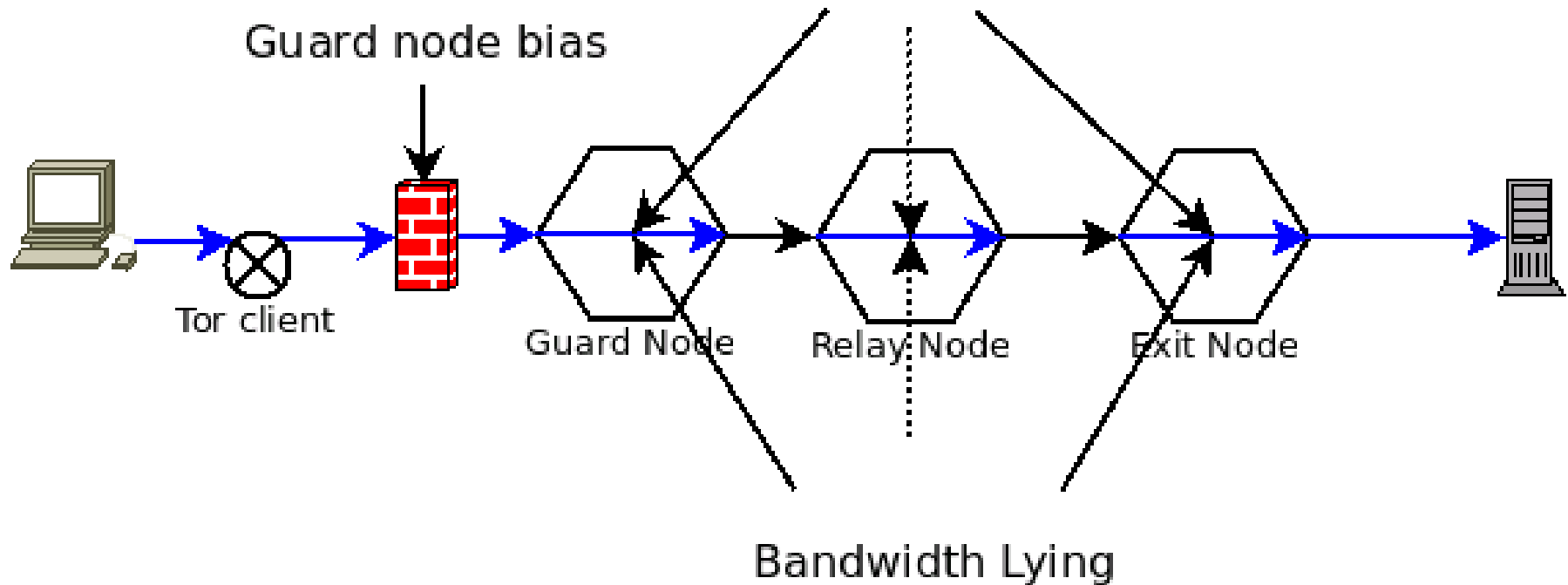


Passive Attacks

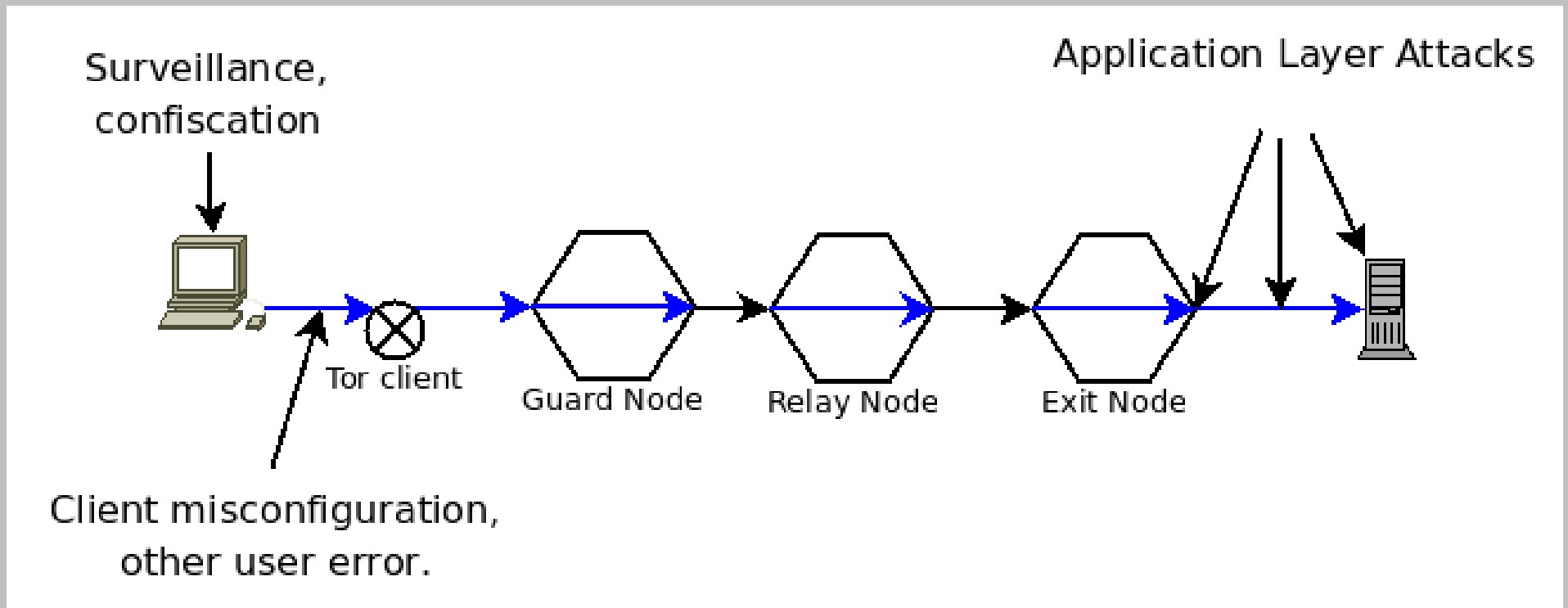


Active Attacks

Active Circuit Failure Attacks



Application Attacks



Questions/Intermission 1

Questions so far?

Approaches to Security

- Verify node operators (Ha!)
- Path selection hacks
- “Tor up from the floor up”
- Improve network speed and usability
- Scan nodes for modification/reliability
- Secure the applications (different threat model)

Path Selection Hacks

- /16 hack: No two nodes from same /16 netmask
 - Many ISPs have disjoint IP ranges..
- Guard nodes
 - Chosen from top 50% uptime, top 50% bandwidth
 - Foil “repetitive fetch” application layer attacks
 - Reduces long-term fingerprinting potential
 - Without rotation, can deter intimidation attacks
 - Difficult to do right. Typically still rotate
 - Essentially a time-tradeoff of risk

Tor Routers and LiveCDs

- JanusVM, Anonym.OS, xBVM
 - “Tor up from the floor up”
 - Address application-level attacks to bypass Tor
 - Block UDP
- Major flaw: Circuit reuse -> app correlation
 - AV software update, other ID-based software updates
 - AIM, ssh, email usage of different “nyms”
 - Media players checking recommended music, etc etc

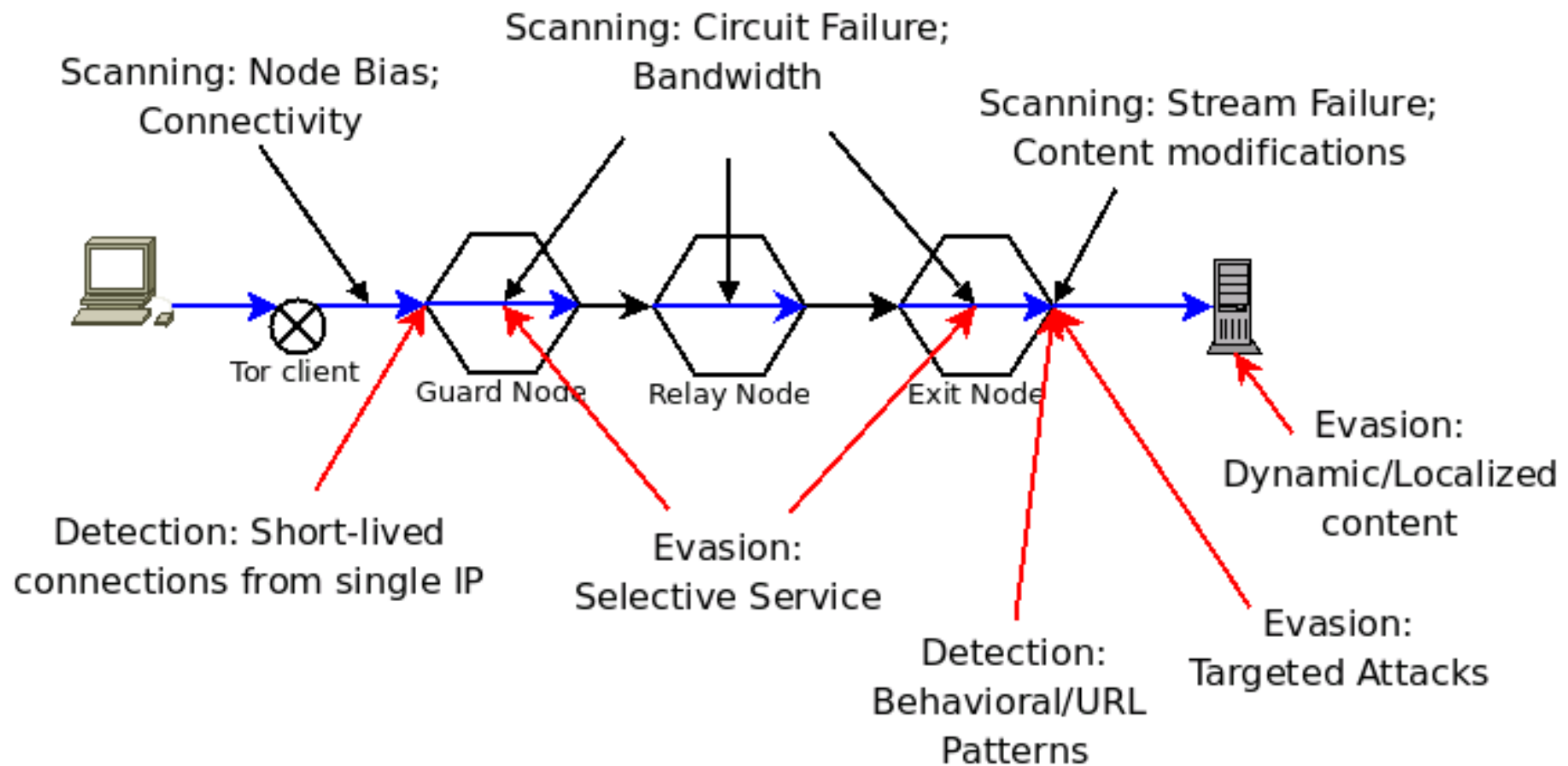
Improving Speed and Usability

- Key component of Tor security: Large userbase
 - Users have been harassed because of small anonymity sets! Whistleblower/Blogger scenario can be unsafe!
- Users want speed and ease of use
 - Many do not need as much anonymity
 - Two hop proposal (semi-controversial)
 - Intelligent path selection
 - Ensure network is evenly balanced and reliable

Centralized Network Scanning

- Tor control port is fun stuff
- Snakes on a Tor and TorFlow
 - Verifies md5 sums of googled URLs
 - Also verifies node reliability+bandwidth
- Works against incompetent+blanket adversaries
 - Actually found some broken+malicious nodes
- Does not work against selective adversaries
- Vulnerable to detection

Scanning Methods and Weaknesses



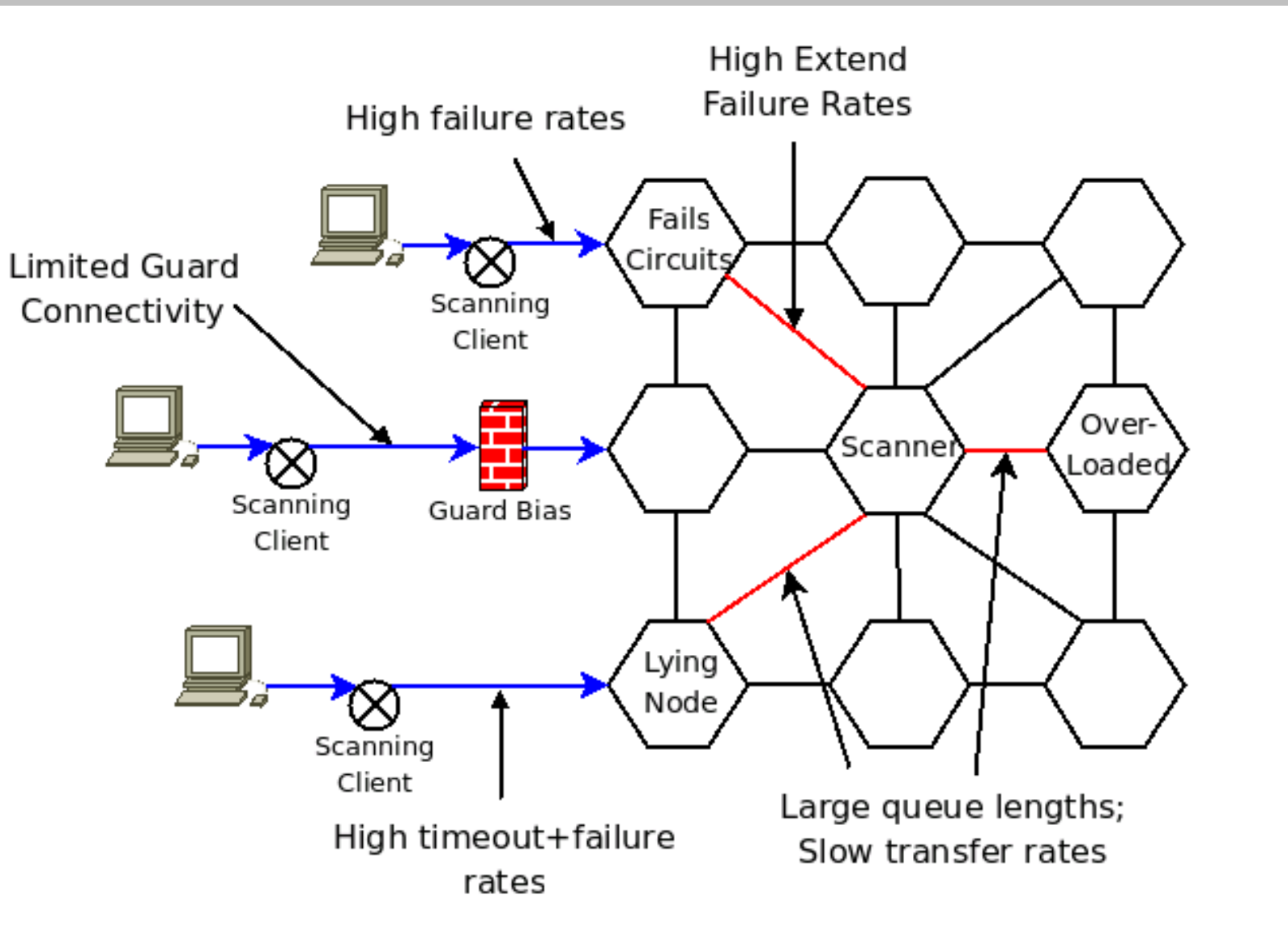
Stuff We Found Anyway

1. Chinese ISP doing SSL MITM
2. Popup blocking! :)
3. Google Analytics Blocking! <3
4. DNS Spoofing
5. SSH+SSL MITM
6. Overloaded nodes
7. Balancing issues :(

Decentralized Network Scanning

- Client-based:
 - Use reliability averages from TorFlow
 - Alert user if guard node fails more than X% circuits
 - Measure observed bandwidth/latency of nodes
- Node-based:
 - Gather statistics on average capacity and queue lengths to peers, compare to node rankings
 - Report major deviations or use as balancing feedback loop.

Passive Client+Node Based Scanning



Balancing Issues

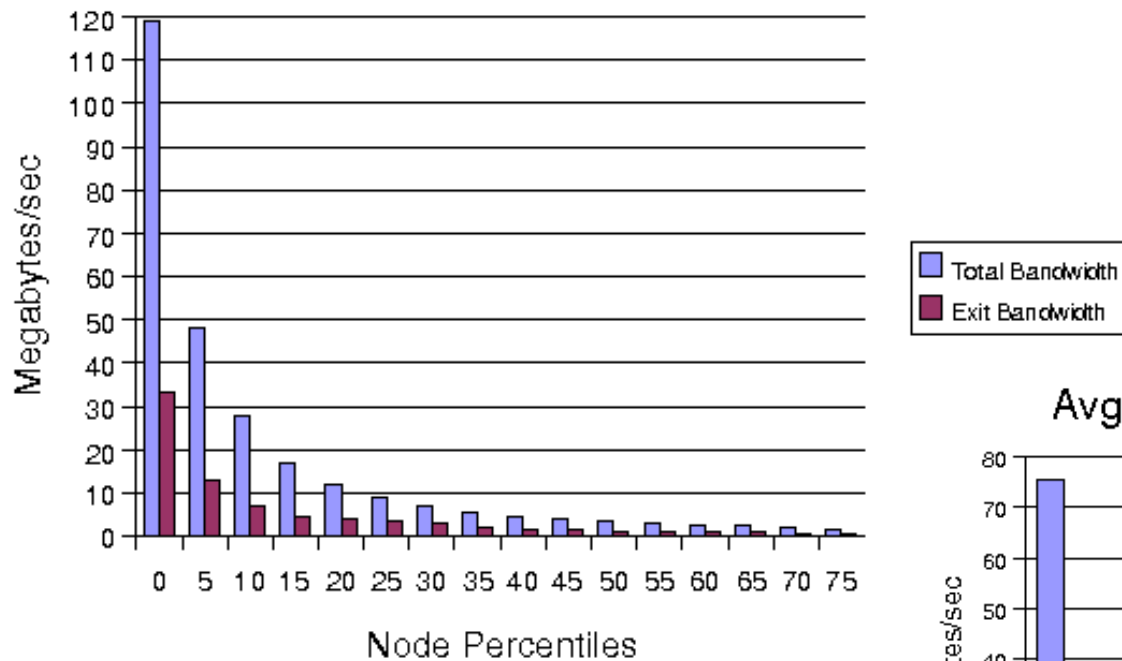
- Tor network is unbalanced
 - Guard node issues (bug #440)
 - Bandwidth clipping
- Detectable during scans
 - Top 5% of nodes have room for 7X more capacity
 - Next 10% of nodes have room for 3X more capacity
 - High circuit failure rates that drop off at 50% mark
 - High extend times that drop off at 50% mark

Scanning Methodology

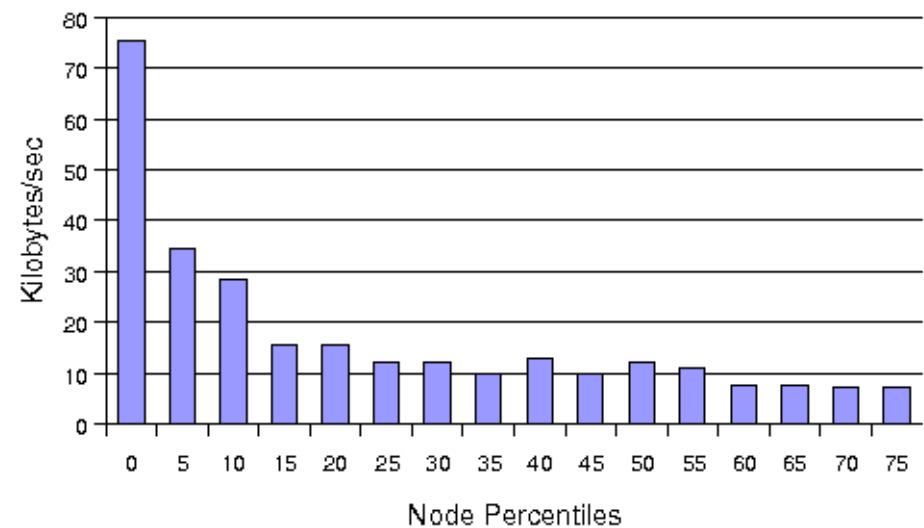
- Divide Tor network into 5-percentile segments
 - About 80 nodes each
- Circuit Scanning
 - Build 500 three hop paths for each range
 - Fetch ~20k file on each path
 - Count failures, track extend times
- Bandwidth Scanning
 - Fetch 512k file 200 times over two hop paths
 - Average the observed bandwidth for each range

Bandwidth (Mis)Balancing

Network Bandwidth by Percentile

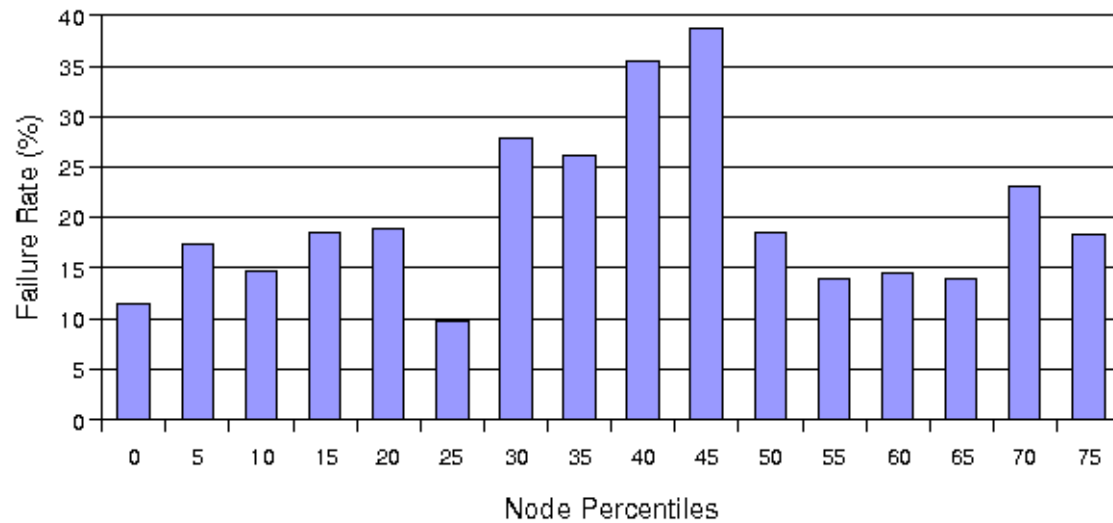


Avg Stream Bandwidth by Percentile

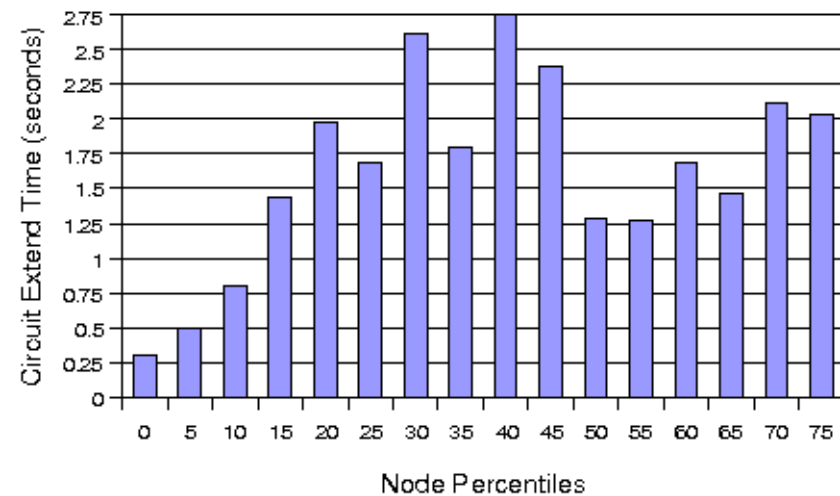


Side Effects of Unbalancing

Circuit Failure by Percentile



Avg Extend Times



Probability of Existing Tor Usability

- 70% Chance of choosing one unbalanced guard
 - Tor goal: 3 guards
- $.7 \times .7 \times .7 = 34\%$ chance of 3 unbalanced guards
 - Tor is likely unbearable for 34% of users
- $C(3,2) \times .7 \times .7 \times .3 = 44\%$ chance of 2/3 bad guards
- $C(3,1) \times .7 \times .3 \times .3 = 19\%$ chance of 1/3 bad guards
- $.3 \times .3 \times .3 = 3\%$ chance of 3/3 usable guards

Other Load Balancing Factors

- Insane exit policies
 - Allowing bittorrent, p2p, smtp..
- High uptime vs low uptime
- Scarce guard bandwidth
 - Avoid guards for relay choice
- Directory vs Node traffic
- Time of day
- Location

Questions/Intermission 2

Questions so far?

Securing the Application Layer

- Tor has a superset of the threat model most applications are written for.
 - No UDP!
 - Unique identifiers are bad
 - Proxy settings must be sacrosanct
 - Location information must not be transmitted
 - Updates are dangerous. Hostile network.

Tor's Web Attack Profile

1. Bypassing proxy settings
2. Correlation of Tor vs Non-Tor
3. History disclosure
4. Location information
5. Misc Anonymity set reduction
6. History records

Plugin Wall of Shame

- Flash v9
- Quicktime v7.2
 - RTSP proxy (does not apply to web streams)
- Windows Media Player v10.000000.4040
 - Has proxy settings. Even has a “No Bypass” option.
 - Still Ignores them
- Adobe Acrobat Reader Plugin v8.1
 - Leaks DNS
- mplayerplug-in

Solution: Improved TorButton

- Disable plugins while Tor is enabled
- Isolate dynamic content per Tor load state
- Cookie jars/cookie clearing
- Cache management
- History management
- User agent spoofing during Tor
- Timezone+Locale spoofing

TorButton Demo

- <http://gemal.dk/browserspy/basic.html>
- <http://gemal.dk/browserspy/css.html>
- <http://gemal.dk/browserspy/date.html>
- <http://gemal.dk/browserspy/plugins.html>
- <http://metasploit.com/research/misc/decloak/index>
- <http://ha.ckers.org/weird/CSS-history.cgi>
- <http://www.tjkdesign.com/articles/css%20pop%20>

Interesting Technical Details

- Context issues
- Tab tagging
- XPCOM hooking and XPCOM policies
- Javascript hooking

Final Thoughts

- Tor security \neq Internet security
 - Superset, actually
 - Adversary has different goals
 - Many apps do not consider privacy vulnerabilities as real vulnerabilities

Credits+Contributions

Scott Squires (Original TorButton Author)

Collin Jackson (History blocking+Cookie jars)

Johannes Renner (TorFlow contributions+research)

Nick & Roger (Advice, Tor in general)

Nitin, Dave, Thom (Advice, Moral Support)

“What can I do to help Tor?”

- Extra bandwidth? Run a node!
 - See conference CD for Linux 'tc' prioritization script
 - No need to impact your own traffic flows
- Post patches/plugins to your favorite apps to protect against info disclosure.
 - Work to raise awareness that privacy issues should be considered as part of security measures