Tor update 2012

Roger Dingledine
The Tor Project
https://torproject.org/
Today's plan

• 0) Crash course on Tor
• 1) History of Tor censorship attempts
• 2) Attacks on low-latency anonymity
• 3) Tor performance issues
• 4) Next research questions
Alice makes a session key with R1
...And then tunnels to R2...and to R3
Today's plan

- 0) Crash course on Tor
- 1) Recent censorship
- 2) Pluggable transport work
- 3) Simulations / Performance
- 4) Anonymity questions
China starting in 2011

- DPI for SSL handshakes that offer the Firefox 3 ciphersuite. Follow-up with Tor handshake, create a one-hop circuit, blacklist.
- Half of probes coming from a single IP
- The rest coming from dialup China IPs
Tricks that work now in China

- Change your cipher suite (but 0xff!)
- Lower MSS during the SSL handshake
- Split SSL strings across TCP packets
- Drop first two SYNs
- They appear to be scanning with real Tor 0.2.2.x clients
- Philipp Winter's paper at FOCI '12
Iran filters SSL (Feb 2012)

- They cut all SSL – so no gmail, no facebook, etc
- After a few days, they cut OpenVPN by DPI, blocked SOCKS handshakes, etc
- We deployed a trial Obfsproxy bundle, which got 5000+ users.
Directly connecting users from Iran

The Tor Project - https://metrics.torproject.org/
Directly connecting users from Kazakhstan

The Tor Project - https://metrics.torproject.org/
Directly connecting users from Ethiopia

The Tor Project - https://metrics.torproject.org/
Directly connecting users from the Syrian Arab Republic

The Tor Project - https://metrics.torproject.org/
Total relay bandwidth

Advertised bandwidth
Bandwidth history

The Tor Project - https://metrics.torproject.org/
Time in seconds to complete 50 KiB request

Measured times on all sources per day

- Median
- 1st to 3rd quartile

The Tor Project - https://metrics.torproject.org/
Time in seconds to complete 1 MiB request

Measured times on all sources per day

- Median
- 1st to 3rd quartile

The Tor Project - https://metrics.torproject.org/
Time in seconds to complete 5 MiB request

*Measured times on all sources per day*

- Orange: Median
- Light orange: 1st to 3rd quartile

The Tor Project - https://metrics.torproject.org/
What else we've been up to (1)

- Tor network up to almost 3000 relays
- Pluggable transport interface now tested, bugfixed, and deployed
- New “DisableNetwork” config option
- New “v3” connection handshake that doesn't use SSL renegotiation
- Abandoned the Torbutton toggle model (continue to maintain our Firefox fork)
What else we've been up to (2)

- We have a Farsi blog now (Arabic soon)
- IPv6 bridges working (used in China)
- Isolate streams by domain/destination/application rather than time interval
- Hired a new core developer! (Want to hire browser hacker, QA automation person, etc)
- Security analysis on Ultrasurf
Today's plan

• 0) Crash course on Tor
• 1) Recent censorship
• 2) Pluggable transport work
• 3) Simulations / Performance
• 4) Anonymity questions
Obfsproxy

- Doesn't deal with packet volume/timing (so look for frequent 586 byte packets)
- Each side sends random key, then $E(MAC(key), magic, padlen, padding)$
- So you can test a flow to see if it has the right format when you decrypt it (or DPI inside it) – just like you can recognize and unzip flows to DPI inside them
- Need some ECDHE approach
Flashproxy

- The next transport I want to try deploying
- Paper published at PETS 2012
- Reimplemented with Websockets, but still a few issues
- Currently the end user needs a public IP address
- Facilitator still centralized/blockable
Skypemorph

- Tries to match Skype traffic in packet timing/volume (but assumes they're drawn from independent distributions)
- Sends at a fixed rate that's a function of network conditions
- Sends whether the user is clicking on something or not?
Dust

- Designed for UDP, where the first packets exchange a key
- Brandon Wiley is also working on a Google Summer of Code 2012 project to develop a Python pluggable transport library
Stegotorus

- Space efficiency overhead?
- Where do you get the covertexts from?
- Should you draw the covertexts from a distribution, or from a library of templates?
Recent Tor design proposals

- Proposal 190: Authentication to bridge before it will talk Tor protocol
- Proposal 191: Authentication of bridge before client will authenticate to it
- Proposal 198: be more flexible about what ciphersuites we can advertise
- Proposal 199: Bridgefinder
Putting the pieces together

- Pluggable transport of some kind
- Some scanning-resistance property for the bridges, e.g. “address knocking” design
- Address (or credential) distribution strategies
- Reachability testing to know whether to cycle the address
- Defend against protocol-level bridge enumeration
Measuring bridge reachability

- Passive: 1) bridges track incoming connections by country
- 2) Clients self-report blockage (e.g. via some other bridge)
- Active: 1) direct scans
- 2) Measure remotely via FTP reflectors
- 3) Bridges test for duplex blocking
Ways to find bridges (1)

- 1) Overwhelm the public address distribution strategies
- 2) Run a non-guard non-exit relay and look for connections from non-relays.
- 3) Run a guard relay and look for protocol differences
- 4) Run a guard relay and do timing analysis
- 5) Run a relay and probe clients as they connect to you
Ways to find bridges (2)

6) Scan the Internet for things that talk Tor
7) Break into Tor Project infrastructure (or break the developers)
8) Watch the bridge authority do its reachability tests
9) Watch your border firewall and DPI for Tor flows
10) Zig-zag between bridges and users
BridgeDB needs a feedback cycle

- Measure how much use each bridge sees
- Measure bridge blocking
- Then adapt bridge distribution to favor efficient distribution channels
- Need to invent new distribution channels
- Need more and changing bridge addresses
Tradeoffs to consider

- Blank address space, or Apache “unconfigured” page?
- Logging to detect enumeration attacks, vs not logging to protect user privacy
- BridgeDB strategies (recaptcha, etc)
- Strategies for using address pools intelligently
Today's plan

- 0) Crash course on Tor
- 1) Recent censorship
- 2) Pluggable transport work
- 3) *Simulations / Performance*
- 4) Anonymity questions
Performance issues

- Get more capacity / scale better
- Load balancing (bw measurement)
- Flow control / round-trip latency
- Throttling / scheduling
Capacity / scaling

- Incentives papers
  - Gold star design
  - Braids
  - Paul/Rob/Aaron's paper
- Everybody-a-relay
- The AES implementations in OpenSSL 1.0.1 are 10x faster than before
Load balancing / measurement

- “Bandwidth authority” scripts (Should measure latency, socket exhaustion)
- EigenTor
- Congestion-aware path selection
- Recognize poor guard performance, switch?
- Give out Guard flag more freely (Tariq's paper. Entropy? Tradeoffs?)
- Raise the min threshold for the Fast flag?
Flow control / round-trip latency

- N23 still worth exploring
  - Especially for slow client connections
- “Double door” effect from independent read and write rate limits
- Comparison of Tor datagram designs
- Optimistic data in begin cells
Simulation work

- Upcoming CSET paper on comparing Shadow to ExperimenTor, and on realistic simulation parameters
- George Danezis and Ryan Henry both working on privacy-preserving measurement
- Still need to down-sample both the network and the client load
- Some bugs and mysteries remain
Throttling / scheduling

- Ian and Can's EWMA paper for circuits
- Rob's “throttle at entry nodes” work
- Micah's too
- Nadia's student's “two conns” approach
- Refill token buckets 10/s, not 1/s
- Round-robin between each circuit, or between each TLS conn?
Today's plan

- 0) Crash course on Tor
- 1) Recent censorship
- 2) Pluggable transport work
- 3) Simulations / Performance
- 4) Anonymity questions
Operational attacks

- You need to use https – correctly.
- Don't use Flash.
- Who runs the relays?
- What local traces does Tor leave on the system?
- ...Different talk.
Traffic confirmation

• If you can see the flow into Tor and the flow out of Tor, simple math lets you correlate them.
• Feamster's AS-level attack (2004), Edman's followup (2009), Murdoch's sampled traffic analysis attack (2007).
Countermeasures?

- Defensive dropping (2004)? Adaptive padding (2006)?
- Traffic morphing (2009), Johnson (2010)
- Tagging attack, traffic watermarking
Tor gives three anonymity properties

• #1: A local network attacker can't learn, or influence, your destination.
  – Clearly useful for blocking resistance.

• #2: No single router can link you to your destination.
  – The attacker can't sign up relays to trace users.

• #3: The destination, or somebody watching it, can't learn your location.
  – So they can't reveal you; or treat you differently.
Tor's safety comes from diversity

• #1: Diversity of relays. The more relays we have and the more diverse they, the fewer attackers are in a position to do traffic confirmation.

• #2: Diversity of users and reasons to use it. 60000 users in Iran means almost all of them are normal citizens.
Website fingerprinting

• If you can see an SSL-encrypted link, you can guess what web page is inside it based on size.

• Does this attack work on Tor? Open-world vs closed-world analysis.

• Considering multiple pages (e.g. via hidden Markov models) would probably make the attack even more effective.
Low-resource routing attacks

- Bauer et al (WPES 2009)
- Clients use the bandwidth as reported by the relay
- So you can sign up tiny relays, claim huge bandwidth, and get lots of traffic
- Fix is active measurement.
Long-term passive attacks

- Matt Wright's predecessor attack
- Overlier and Syverson, Oakland 2006
- The more circuits you make, the more likely one of them is bad
- The fix: guard relays
Denial of service as denial of anonymity

- Borisov et al, CCS 2007
- If you can't win against a circuit, kill it and see if you win the next one
- Guard relays also a good answer here.
Epistemic attacks on route selection

- Danezis/Syverson (PET 2008)
- If the list of relays gets big enough, we'd be tempted to give people random subsets of the relay list
- But, partitioning attacks
Congestion attacks (1)

- Murdoch-Danezis attack (2005) sent constant traffic through every relay, and when Alice made her connection, looked for a traffic bump in three relays.
- Couldn't identify Alice – just the relays she picked.
Congestion attacks (2)

- Hopper et al (2007) extended this to (maybe) locate Alice based on latency.
- Chakravarty et al (2008) extended this to (maybe) locate Alice via bandwidth tests.
- Evans et al (2009) showed the original attack doesn't work anymore (too many relays, too much noise) – but “infinite length circuit” makes it work again?
Throughput fingerprinting

- Build a test path through the network. See if you picked the same bottleneck node as Alice picked.
Profiling at exit relays

- Tor reuses the same circuit for 10 minutes before rotating to a new one.
- (It used to be 30 seconds, but that put too much CPU load on the relays.)
- If one of your connections identifies you, then the rest lose too.
- What's the right algorithm for allocating connections to circuits safely?
Declining to extend

- Tor's directory system prevents an attacker from spoofing the whole Tor network.
- But your first hop can still say “sorry, that relay isn't up. Try again.”
- Or your local network can restrict connections so you only reach relays they like.
Attacks on Tor

- Pretty much any Tor bug seems to turn into an anonymity attack.
- Many of the hard research problems are attacks against all low-latency anonymity systems. Tor is still the best that we know of – other than not communicating.
- People find things because of the openness and thoroughness of our design, spec, and code. We'd love to hear from you.
Upcoming Tor news

- Making Torbutton the central controller?
- More modular schedulers
- Build/QA automation
- Apparmor/seatbelt/selinux for TBB. Then VM?
- TBB for Android
- NAT-piercing libraries?
- OONI