Tor: The Second-Generation Onion Router

Roger Dingledine, Nick Mathewson, Paul Syverson

Daniel Medina
medina@columbia.edu
29 September 2004
Tor: Tor’s Onion Routing

Roger Dingledine, Nick Mathewson, Paul Syverson
Usenix Security 2004, Defcon 2004
Overview

Threats
Counter-measures
Demo
Discussion
Threat(s)

No Anonymity

Anonymous Tipsters
Personal Communication
Government Agencies
Threat(s)

http://www.onion-router.net/Sponsors.html

This research is supported in part at NRL's Center for High Assurance Computer Systems by:

- **Office of Naval Research (ONR), Information, Electronics, and Surveillance, Mathematical, Computer, and Information Sciences Division, Command and Control**
  
  Basic R&D work in addition to support for the coding of the first and second generation systems (6.2).

- **Defense Advanced Research Projects Agency (DARPA), Information Technology Office, Information Survivability, High Confidence Networking, Onion Routing: Anonymous Communications Infrastructure**
  
  Support for a contractor to assist in system performance assessments and system improvements (6.2).

- **NRL 6.2 Initiative**
  
  All other Onion Routing development and operation of the Testbed Onion Routing Network.

Just in case you didn’t believe me about the “Government Agencies” part
Threat(s)

Attackers

Network Service Provider
Content Providers
Government Agencies
Counter-measures

Some background…

A “mix”; which is the real message?

Layers (hence, “onion”), of encryption, per-hop

Anonymizing path (not data)

Use Privoxy for data protection

Images from slides by Dingledine
Counter-measures

Client initiates connection, and establishes session keys, builds path

Protect volunteer routers by building in bandwidth controls
Demo

http://freehaven.net/tor

# after "./configure && make && sudo make install", on Mac OS X
dyn-wireless-244-195:/ medina$ /usr/local/bin/tor
Sep 27 12:09:18.858 [notice] tor_init(): Tor v0.0.8. This is experimental software.
   Do not use it if you need anonymity.
Sep 27 12:09:18.860 [notice] Configuration file '/usr/local/etc/tor/torrc' not present,
   using reasonable defaults.
   Tor has successfully opened a circuit. Looks like it's working.
Sep 27 12:09:23.258 [warn] fetch_from_buf_socks(): Your application (using socks5 on port 80)
   is giving Tor only an IP address. Applications that do DNS resolves
   themselves may leak information.
   Consider using Socks4A (e.g. via privoxy or socat) instead.
Demo

Configure Proxies to Access the Internet

- Direct connection to the Internet
- Auto-detect proxy settings for this network
- Manual proxy configuration
  - Use the same proxy for all protocols
  - HTTP Proxy: Port: 0
  - SSL Proxy: Port: 0
  - FTP Proxy: Port: 0
  - Gopher Proxy: Port: 0
  - SOCKS Host: localhost Port: 9050
  - SOCKS v4
  - No Proxy for: localhost, 127.0.0.1
    Examples: mozilla.org, net.nz, 192.168.1.0/24
  - Automatic proxy configuration URL:

[Options: Cancel, OK]
Demo

Verify

http://junkbusters.com/cgi-bin/privacy

Remote host: one of the proxies
User-Agent, other client info leaked
Discussion