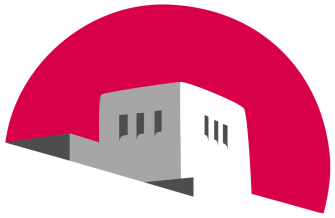
Three Researchers, Five Conjectures: An Empirical Analysis of TOM-Skype Censorship and Surveillance

Jeffrey Knockel · Jedidiah R. Crandall · Jared Saia
Computer Science Department
University of New Mexico



UNM SCHOOL *of* ENGINEERING

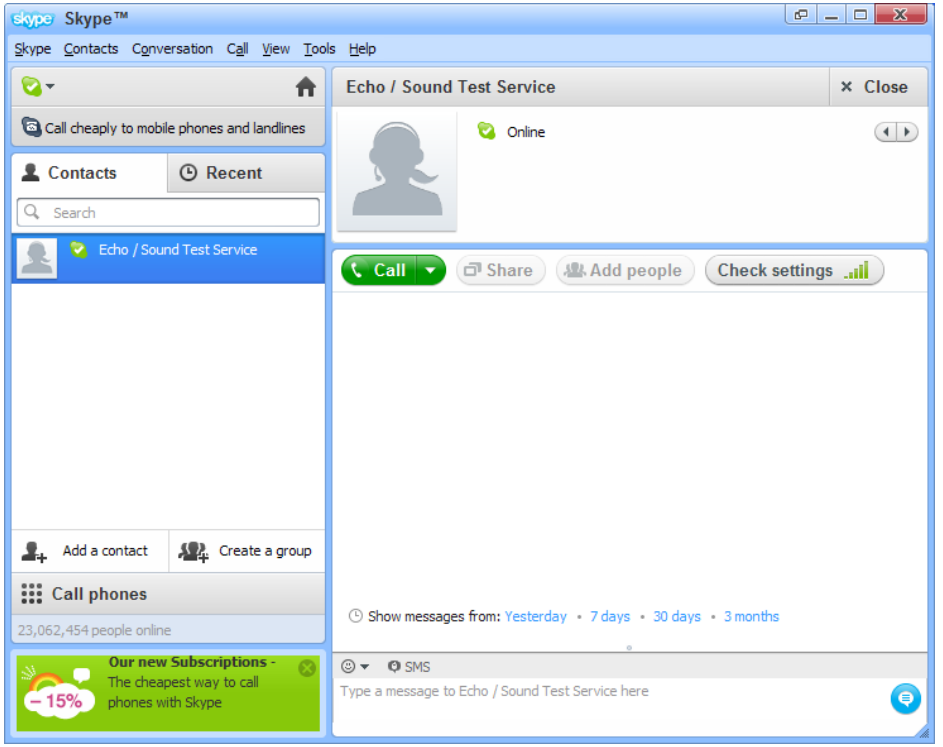
Secrets

- If you find a locked box at the bottom of the ocean, something cool must be inside...
- When something is encrypted, it must be important...

```
9B1A82546F
8D2C76449F
339604A813883E
B4019C319603C5
5F40564C70
E8A81580228D3B
920BD3E1FD
E3CFAF03B6EC8D
```

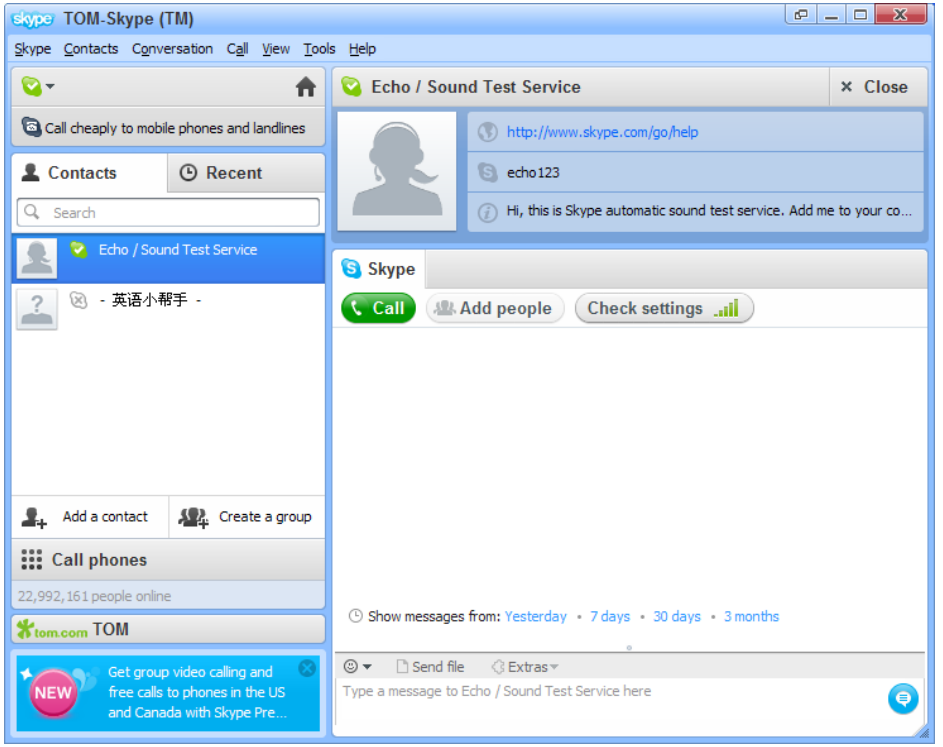
Skype

- Skype
  - Developed by Skype Limited
  - Text chat
  - Internet voice and video calls
  - Over 650 million users



TOM-Skype

- TOM-Skype
  - Modified version of Skype by TOM Group Limited, a China-based media company
  - Uses Skype's network
  - Over 80 million users
  - In China, http://www.skype.com HTTP redirects to http://skype.tom.com



Previous Work

- Nart Villeneuve's work (2008)
  - TOM-Skype censors and surveils text chat
  - Found some censored words (not exhaustive)
- Our work
  - Exact keyword lists and surveillance
  - Discernment between censored *vs*. surveilled keywords
  - Five conjectures on Internet censorship

Empirical Analysis

- TOM-Skype uses “keyfiles”
  - List of keywords triggering censorship and/or surveillance of text chat
  - One built-in
  - At least one other downloaded

3.6-4.2 Keyfiles

- TOM-Skype 3.6-3.8 downloads from

  http://skypetools.tom.com/agent/newkeyfile/keyfile
- TOM-Skype 4.0-4.2 downloads from

  http://a[1-8].skype.tom.com/installer/agent/keyfile
- Both use same encryption:

```
3690269933
AE187A2E9A
0BB4348F29
BC078825614C5D
14A3248E0D
B948DC2C
. . .
```

3.6-4.2 Keyfiles

- To crack: point skypetools.tom.com DNS queries to our server
- TOM-Skype downloads our keyfile
- Binary search to find “fuck”

```
. . .
1EB412B019
77B543CE52 # fuck
980684268425 99
. . .
```

3.6-4.2 Keyfiles

- To crack: point skypetools.tom.com DNS queries to our server
- TOM-Skype downloads our keyfile
- Binary search to find “fuck”
- Perform chosen ciphertext attack
- See what gets censored

```
77B543CE52 # fuck
77B543CE53 # fucl
77B543CE54 # fucm
. . .
77B341CC50 # duck
. . .
```

3.6-4.2 Keyfiles

- To crack: point skypetools.tom.com DNS queries to our server
- TOM-Skype downloads our keyfile
- Binary search to find “fuck”
- Perform chosen ciphertext attack
- See what gets censored
- Pattern emerges

```
77B543CE52 # fuck
77B543CE53 # fucl
77B543CE54 # fucm
. . .
77B341CC50 # duck
. . .
```

procedure DECRYPT ($C_{0..n}$, $P_{1..n}$)

  for i ← 1,n do

    $P_i = (C_i \oplus 0x68) - C_{i-1} \pmod{0xff}$

  end for

end procedure

5.0-5.1 Keyfiles

- TOM-Skype 5.0-5.1 downloads keyfiles from

  http://skypetools.tom.com/agent/keyfile
- TOM-Skype 5.1 downloads surveillance-only keyfile from

  http://skypetools.tom.com/agent/keyfile_u
- AES encrypted in ECB mode
- Key reused from TOM-Skype 2.x
- When encoded in UTF16-LE, 32 bytes:

  `0sr TM#RWFD,a43`
- Half of bytes printable ASCII, other half null (weak)

TOM-Skype 4.0-5.1 Surveillance

- TOM-Skype 5.0: no surveillance
- Reverse engineered TOM-Skype 5.1's surveillance
- Discovered key for TOM-Skype 4.0-4.2, 5.1
- Encrypts surveillance traffic with DES key in ECB mode:

  `X7sRUjL\0`

- First seven bytes printable ASCII, 8th byte is null-terminator of the Delphi string:

```
0045BDBC  FF FF FF FF 07 00 00 00
0045BDC4  58 37 73 52 55 6A 4C 00
```

TOM-Skype 4.0-5.1 Surveillance

- Example surveillance message from 4.0-4.2:

```
jdoe falungong 4/24/2011 2:25:53 AM 0
```

- Message author followed by triggering message followed by the date and time
- 0 or 1 indicates message is outgoing or incoming, respectively
- Example surveillance message from 5.1:

```
falungong 4/24/2011 2:29:57 AM 1
```

- 5.1 does not report username
- 5.1 does not report outgoing messages

TOM-Skype 3.6-3.8 Surveillance

- TOM-Skype 3.6-3.8 encrypts surveillance traffic with a different DES key
- Reverse engineering it required circumventing Skype's built-in anti-debugging measures
- Why not before? TOM-Skype 5.1 sends surveillance messages from an outside process called `ContentFilter.exe`
- Our strategy: DLL injection, a way to execute our own code inside of TOM-Skype's process...

TOM-Skype 3.6-3.8 Surveillance

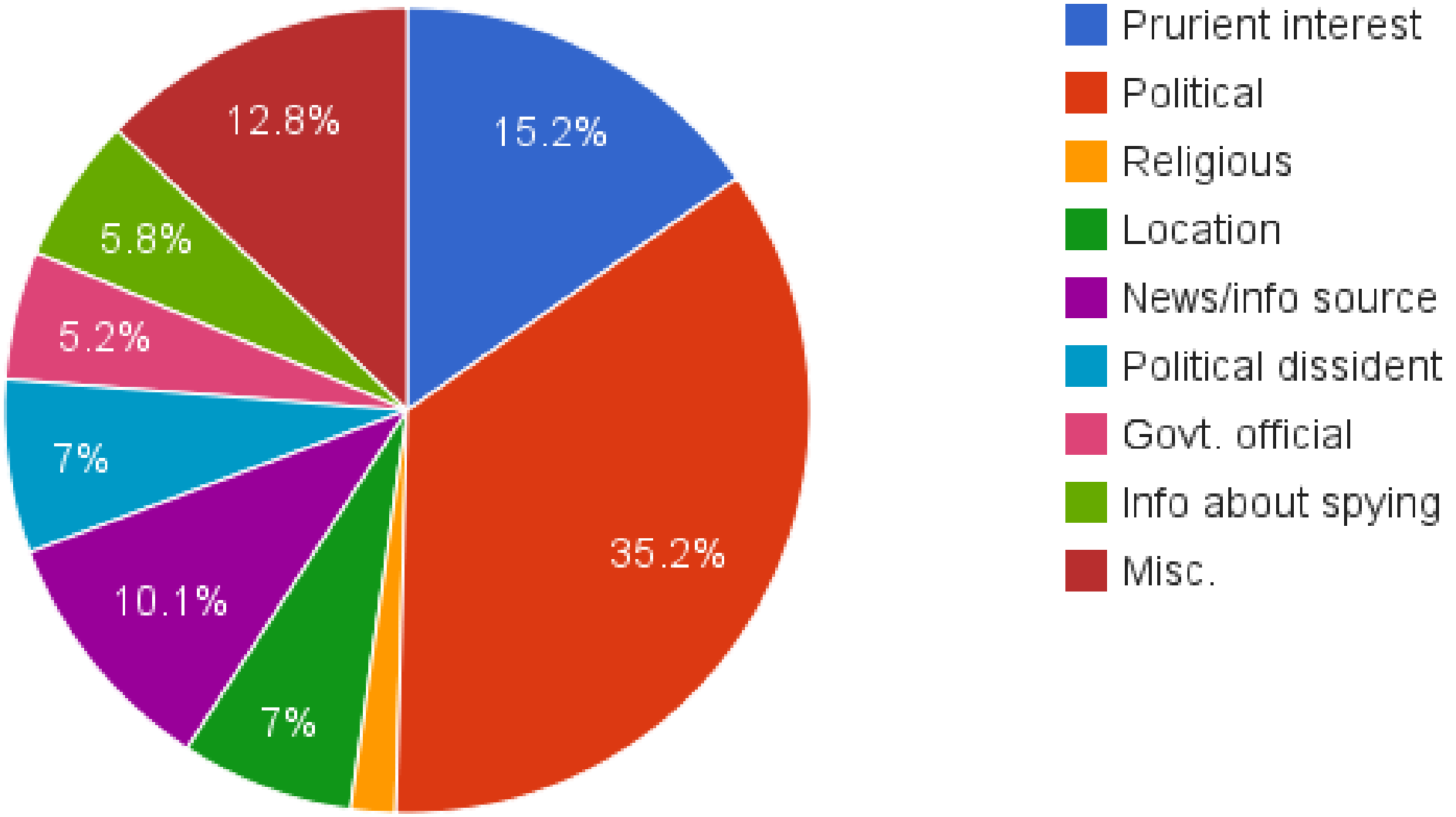
- Hook our code into timer function called before encryption
- Our code sleeps for 20 seconds
- Attach with debugger
- Suspend all other threads
- Resume sleeping thread
- In switch statement, we observed the following DES key used:

  32bnx23l

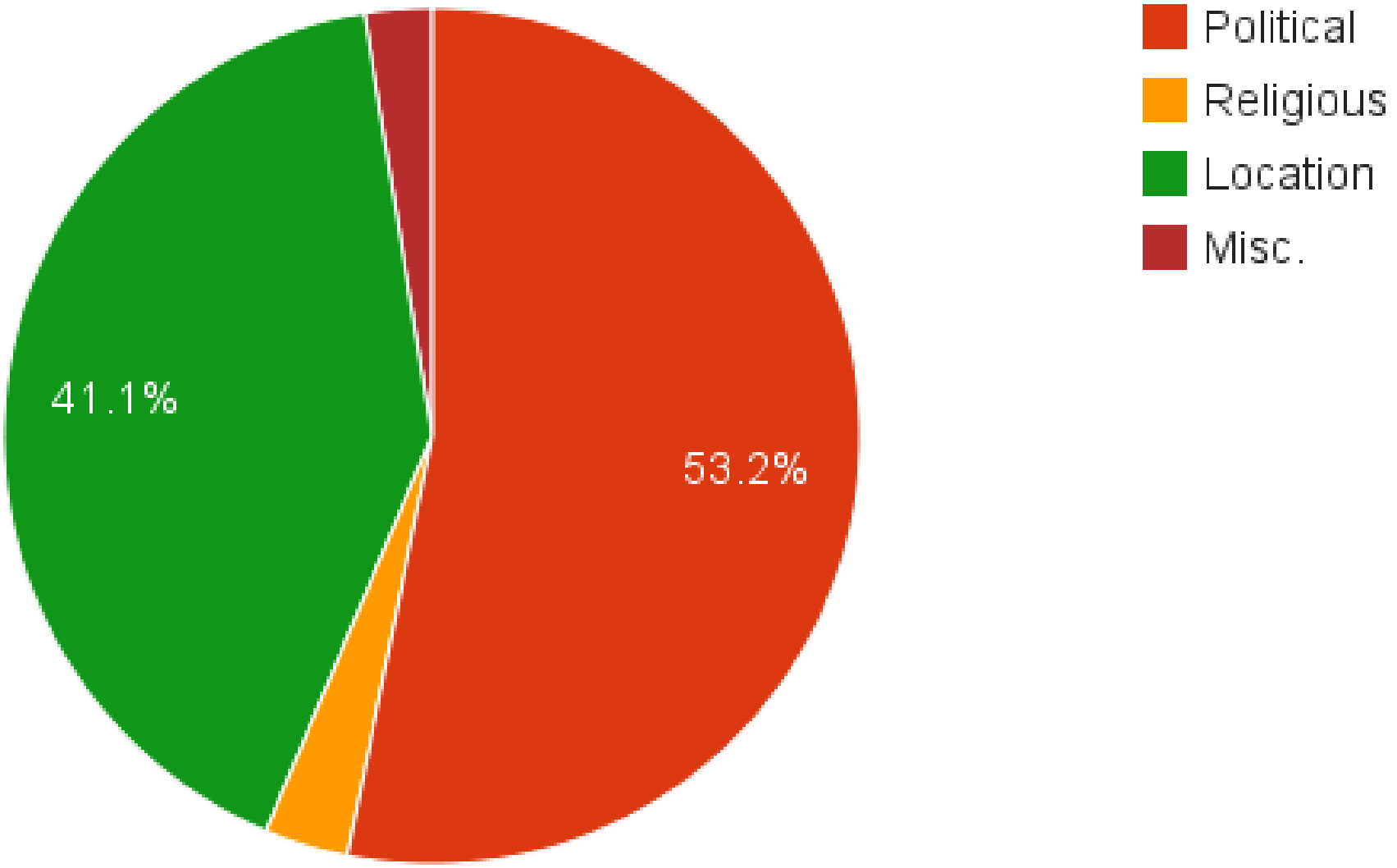
- Surveillance messages are same as 4.0-4.2

```
ADD DH,AH
CMP EAX,33B200ED
JMP SHORT Skype.00ED3DE8
MOV DL,32
JMP SHORT Skype.00ED3DE8
MOV DL,62
JMP SHORT Skype.00ED3DE8
MOV DL,6E
JMP SHORT Skype.00ED3DE8
MOV DL,78
JMP SHORT Skype.00ED3DE8
MOV DL,32
JMP SHORT Skype.00ED3DE8
MOV DL,33
JMP SHORT Skype.00ED3DE8
MOV DL,6C
JMP SHORT Skype.00ED3DE8
MOV DL,24
JE SHORT Skype.00ED3DF0
JNZ SHORT Skype.00ED3DF0
```

5.0-5.1 Downloaded Keyfile



5.1 Surveillance-only Keyfile



Censored Keywords

- Keyfile contained political words (35.2%)
  - 六四 (“64,” in reference to the June 4th Incident)
  - 拿着麦克风表示自由 (Hold a microphone to indicate liberty)
- Prurient interests (15.2%)
  - 操烂 (Fuck rotten)
  - 两女一杯 (Two girls one cup)

Censored Keywords

- News/info sources (10.1%)
  - 中文维基百科 (Chinese language Wikipedia)
  - BBC 中文网 (BBC Chinese language)
- Political dissidents (7%)
  - 刘晓波 (Liu Xiaobo)
  - 江天勇 (Jiang Tianyong)
- Locations (7%)
  - 成都 春熙路麦当劳门前 (McDonald's in front of Chunxi Road in Chengdu)

Surveillance-only

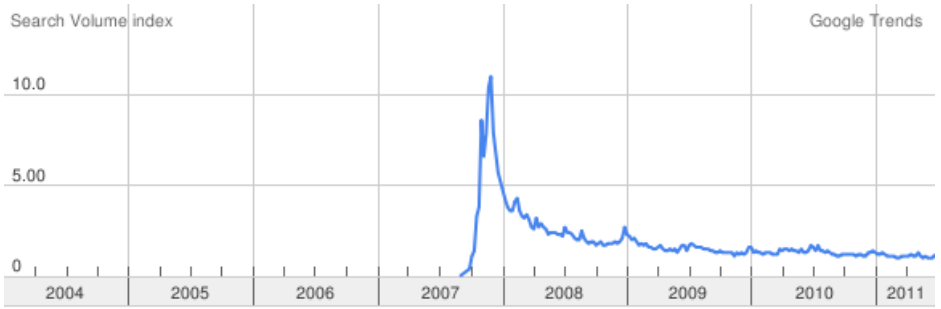
- Mostly political and locations
  - Almost all related to demolitions of homes in Beijing for future construction
  - A few related to illegal churches
  - A couple company names

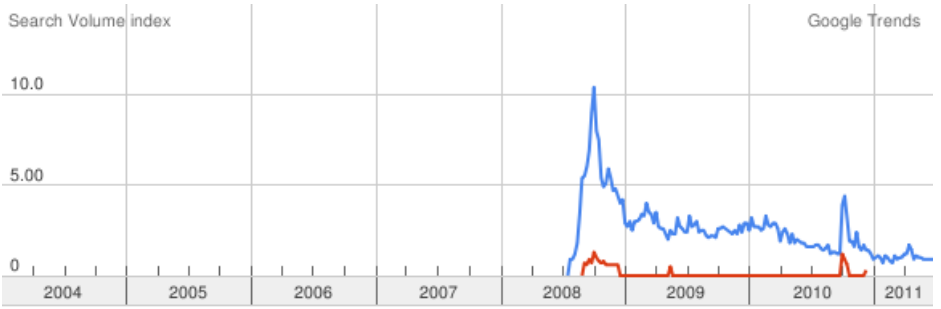
Conjectures

1. **Effectiveness Conjecture**: *Censorship is effective, despite attempts to evade it.*

- Inspired by phrases in keyfiles taken from documents that did not get as widely distributed as the authors had probably intended

Conjectures





2. **Spread Skew Conjecture**: *Censored memes spread differently than uncensored memes.*

- Inspired by Google trends data for “two girls one cup” in English (left) vs. Chinese (right)

Conjectures

3. **Secrecy Conjecture**: *Keyword based censorship is more effective when the censored keywords are unknown and online activity is, or is believed to be, under constant surveillance.*

- Inspired by TOM-Skype's efforts to keep list of censored words and surveillance traffic secret

Conjectures

4. **Peer-to-peer vs. Client-server Conjecture**: *The types of keywords censored in peer-to-peer communications are fundamentally different than the types of keywords censored in client-server communications.*

- Inspired by the high number of proper nouns in TOM-Skype's keyfiles compared to other lists (such as for GET request filtering)

Conjectures

5. **Neologism Conjecture**: *Neologisms are an effective technique in evading keyword based censorship, but censors frequently learn of their existence.*

- Example: 六四 (64), 陆肆 (sixty four), but not “32 + 32” or “8 squared,” which we have seen in Web forums

Conclusion and Future Work

- Found exact keyword lists and surveillance traffic
- We got lucky with TOM-Skype
  - P2P network encrypted, not owned by China
- Future data sources?
  - QQ Chat seem to censor over the network

For keyword lists, machine and human translations, and source code, see
http://cs.unm.edu/~jeffk/tom-skype/

(This URL is also in our paper.)

Acknowledgments

This material is based upon work supported by the National Science Foundation under Grant Nos. CCR #0313160, CAREER #0644058, CAREER #0844880, and TC-M #090517.

Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.