Steganographic Communication Using TCP Inter Burst Delays

Florian Kemmer

TÜBIX

13. Juni 2015

Who am I?

About me

2014	MSc *Network Systems Engineering* Plymouth, UK
2012	BSc *Computer Networking* Furtwangen, Germany
misc	one term in Austria in IT-Security worked one year in Academia

Contact

mail@fkemmer.de
http://www.fkemmer.de

TOC

Introduction

The Problem

Cryptography doesn't work

"Conventional cryptography is like shipping a safe in an armored car with a regiment of soldiers around it. Everyone knows that theres something secret inside"[14]

Metadata gets you killed. . . literally

"We kill people based on metadata"

—former NSA/CIA director General Michael Hayden [13]

Hidden in plain sight

- From greek: “Hidden text”
- Information embedded into carrier data
- Hidden within texts, images, videos, . . .

Hidden in plain sight

(a) sha512: 20e37f[...]16aeae

(b) sha512: 27f6fb[...]65e956

Figure 1: `outguess -d message.txt -p100 angel_noSteg.jpg angel_steg.jpg`

Use cases

- Watermarking
- Hidden Communication
- Information Leakage

Network Steganography

General advantages

- Pictures/other media stored “forever” on the Internet
- thus available for forensic investigation
- Network traffic is rather volatile

Classic Steganography

[Eth][IP][UDP/TCP][**PAYLOAD**]

- Payload gets modified as described before
- Protocols producing much traffic preferred
- Widely used with VoIP [11]
- Capacity depending on generated traffic

Header Modification

[Eth][IP][**UDP/TCP**][PAYLOAD]

- Using unused or unspecified data fields
- Capacity depending on number of packets
- Easily defeated by *traffic normalisation* [16]
- Hardly used

Covert Timing Channels

- Retransmissions [8, 9]
- Reordering [1]
- Delays

Inter-arrival times/Inter Packet Delays

[Packet] Δt [Packet]

```
/usr/sbin/tcpdump -n -ttt -r ...

00:00:01.125810 IP 192.168.0.12.32822 > 141.28.100.151.21: ...
00:00:00.040138 IP 141.28.100.151.21 > 192.168.0.12.32822: ...
00:00:00.000100 IP 192.168.0.12.32822 > 141.28.100.151.21: ...
00:00:00.038057 IP 141.28.100.151.21 > 192.168.0.12.32822: ...
00:00:00.000195 IP 192.168.0.12.32822 > 141.28.100.151.21: ...
00:00:01.494845 IP 192.168.0.12.32822 > 141.28.100.151.21: ...
00:00:00.038338 IP 141.28.100.151.21 > 192.168.0.12.32822: ...
00:00:00.000304 IP 141.28.100.151.21 > 192.168.0.12.32822: ...
00:00:00.000104 IP 192.168.0.12.32822 > 141.28.100.151.21: ...
```

On/Off timing channels

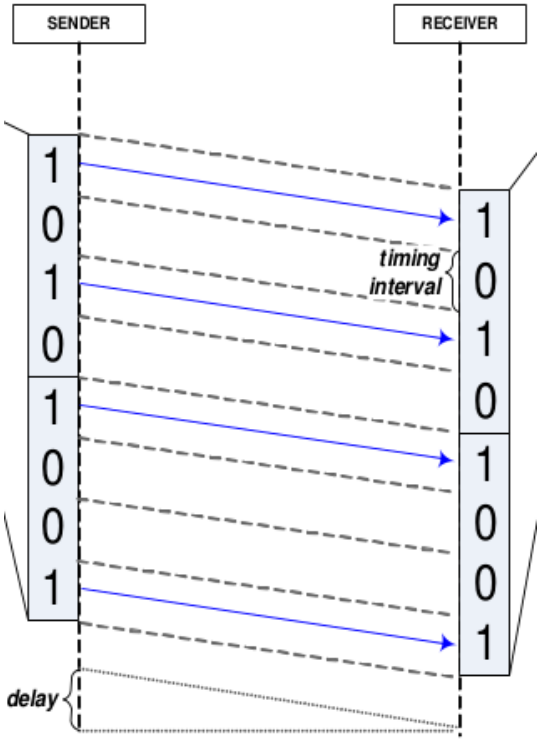
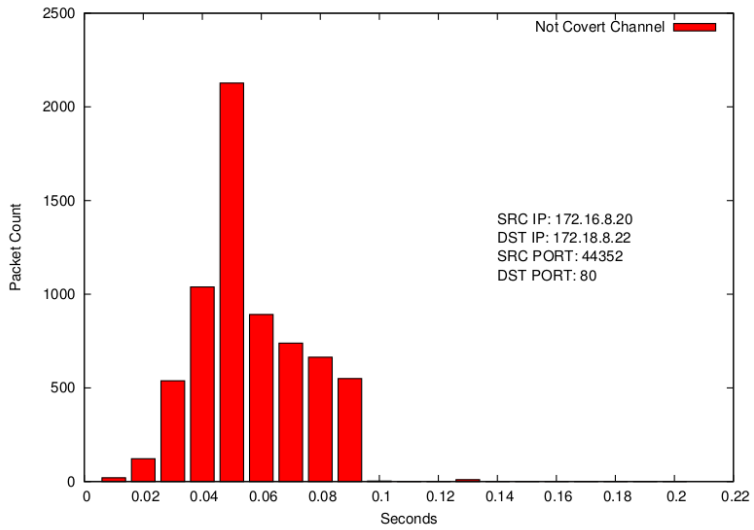
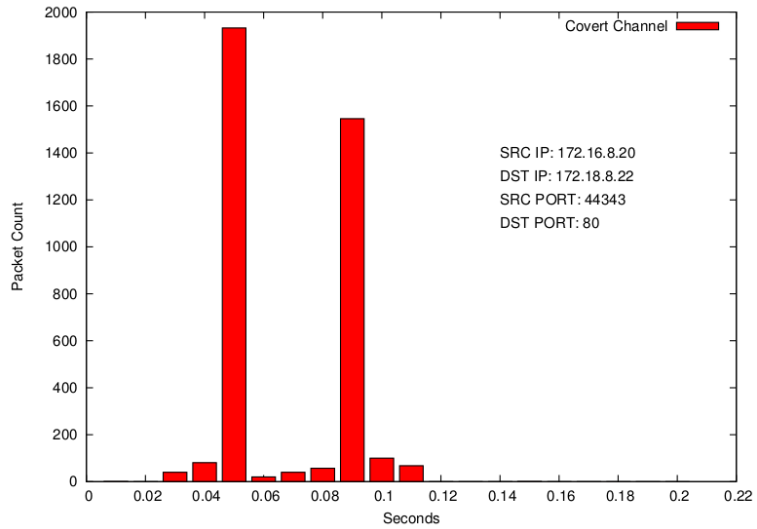


Figure 2: On/Off timing channel [4].

"Morse codes"



(a) Ordinary traffic pattern.



(b) Modified traffic pattern.

Figure 3: "Morse codes" [2]

Timing is everything

- Very sensitive constructs
- Many things can happen on the way through the Internet

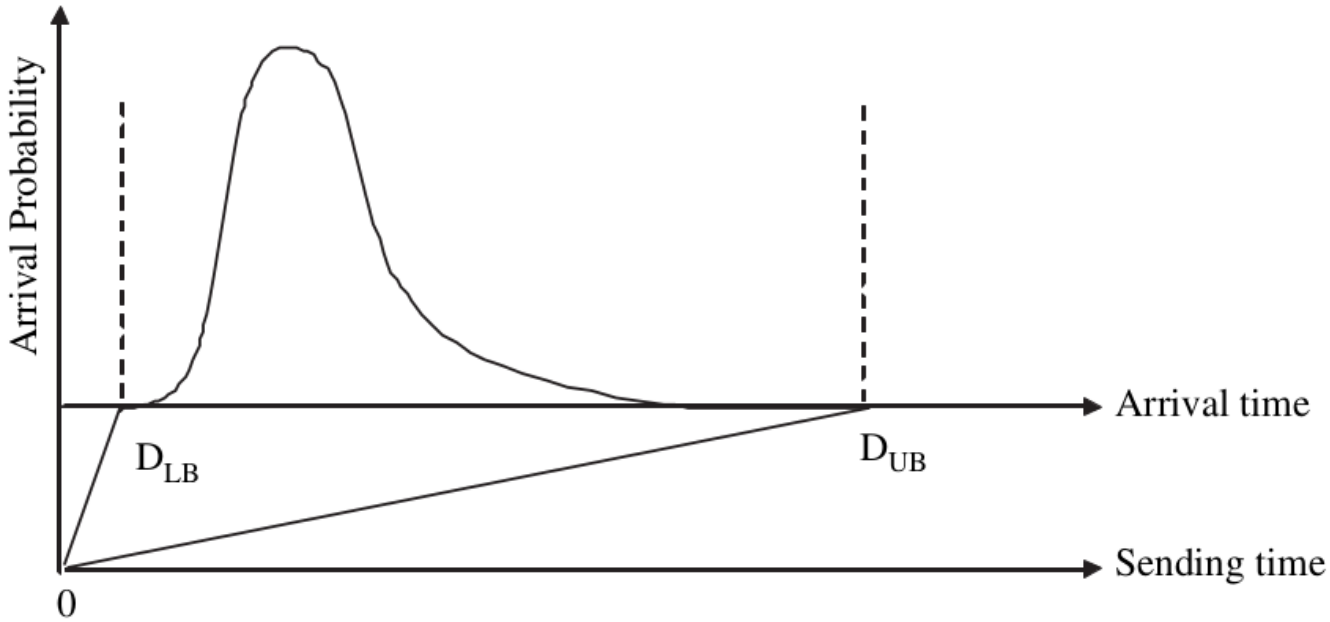


Figure 4: Arrival Distribution of packet sent at $t = 0$ [16].

Summary

- Networks offer plenty of options to hide messages
- Varying channel capacity
- Typically hard to detect
- Mainly researched in “Information Leakage”

MSc Project

Scenario description

- We're the good guys now!
- Fight censorship/surveillance
- Cryptography still doesn't work here
- Store hidden information in inter-arrival times

Differences to previous scenarios

- Two-way communication desired
- Control over both sending and receiving host
- Ability to generate traffic
- Use *TCP* based protocol (for fun and profit)

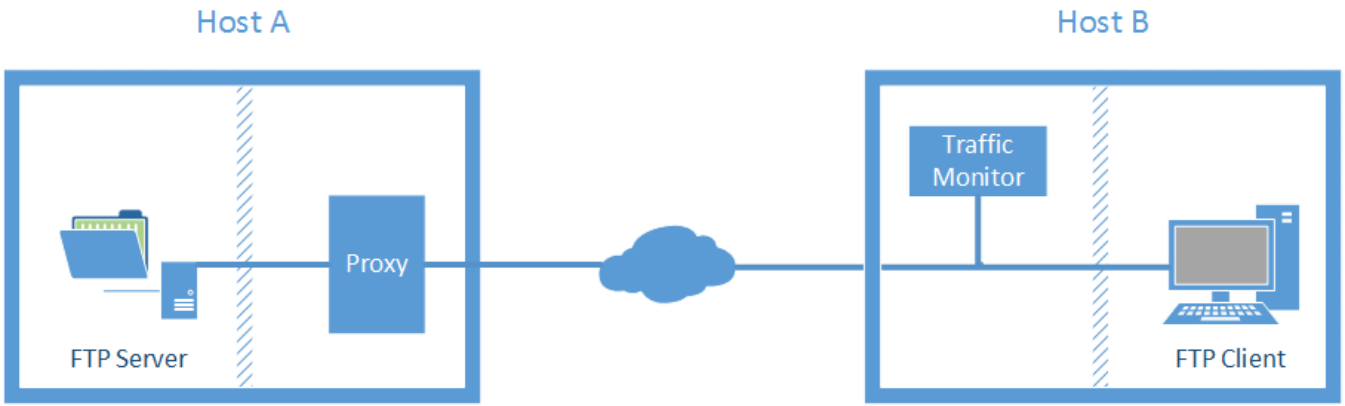
Selection Criteria for cover protocol

1. TCP based
2. High volume (packet count)
3. Commonly used
4. User independent
5. Bi-directional data flow
6. Timely asymmetric

Comparison of protocols (simplified)

	TCP based	*High volume*	*Commonly used*	*User independent*	*two-way*	*timely asymmetric*
HTTP	✓	✗	✓	✗	✗	✓
VoIP	✗	✓	✓	✗	✓	✗
VoD	✗	✓	✓	✓	✗	✓
SSH	✓	✗	✓	✗	✓	✗
FTP	✓	✓	✓	✓	✓	✓

Architecture



Sender

- Converts text into binary
- Applies Error-correcting Codes
- Delays outgoing packets accordingly

Receiver

- Observes IPDs of incoming packets
- Converts them back
- Doesn't have to be real-time recording with `tcpdump` in first place is sufficient

TCP's bursty nature

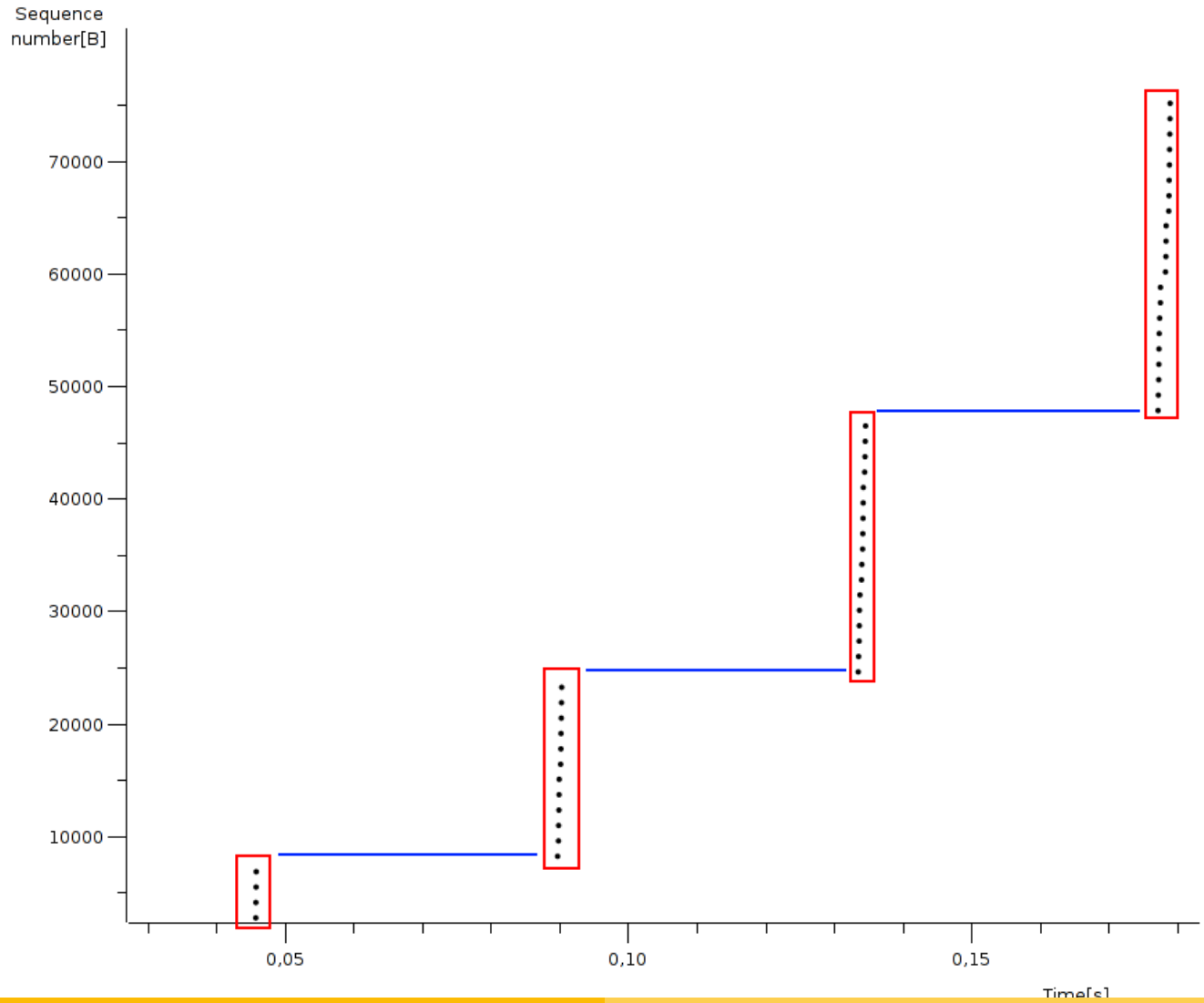
- Multiple packets combined into *burts*
- Within bursts: IPDs defined by bottleneck bandwidth; not by sender
- Muss less packets usable for hiding information

TCP's bursty nature

Figure 5: Usable IPDs intervals with UDP.

Figure 6: Usable IPDs intervals with TCP.

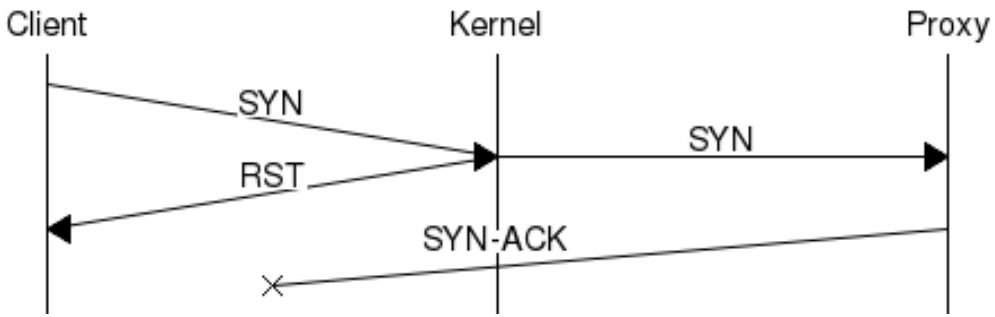
Inter Burst Delays



RAW Sockets

- First attempt to create sending proxy
- Incoming `TCP SYN` never reached proxy
- Kernel killed handshake with `TCP RST` before

RAW Sockets

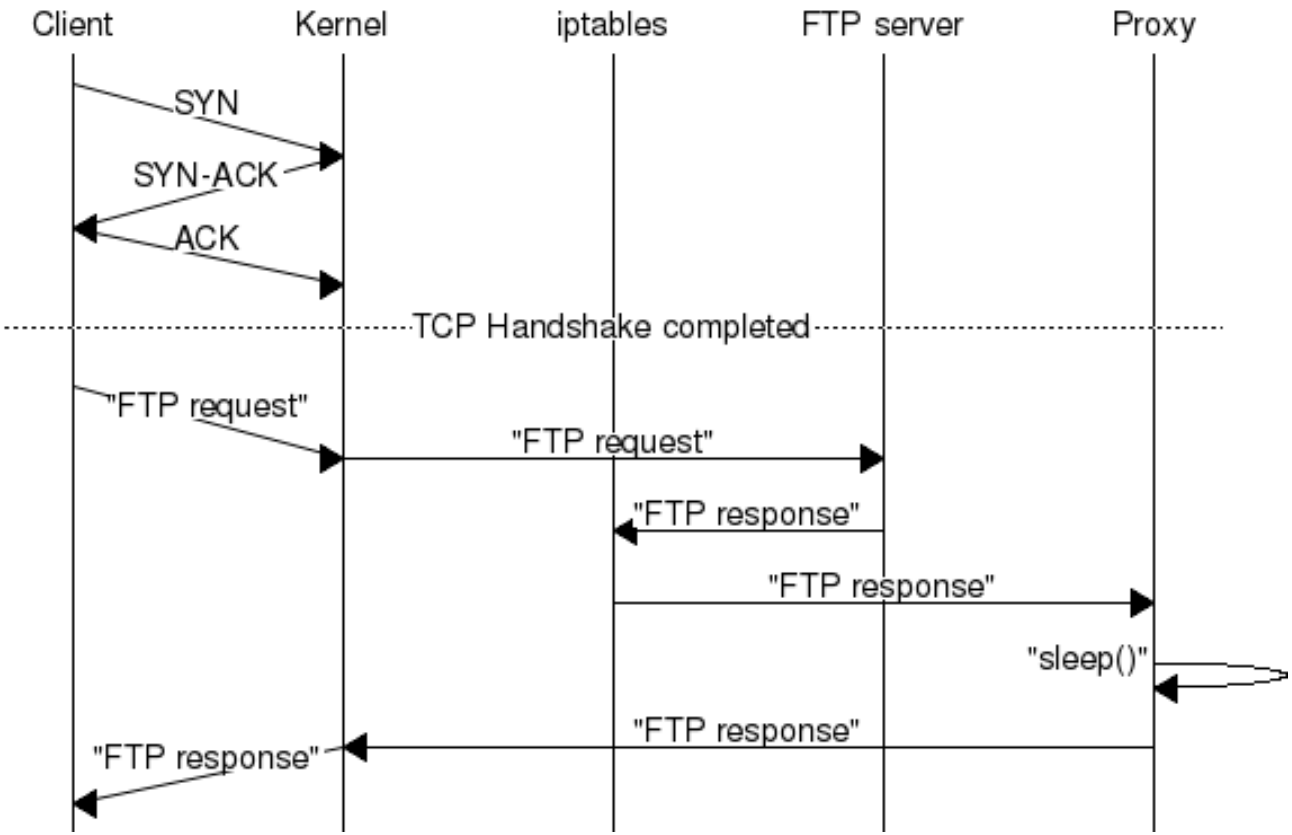


nfqueues

- In combination with iptables
- At the cost of platform independence
- Used to redirect packets internally

```
/sbin/iptables -A OUTPUT -p tcp --sport 21 -j NFQUEUE --queue-num
21
```

nfqueues


Client
Kernel
iptables
FTP server
Proxy
SYN
SYN-ACK
ACK
TCP Handshake completed
"FTP request"
"FTP request"
"FTP response"
"FTP response"
"sleep()"
"FTP response"
"FTP response"

The Internet: It's dangerous to go alone

- Many things can happen to packets on the way through the Internet
- Dealing with corrupted information
- *Automatic Repeat Request* (ARQ) (e.g. TCP)
- Embed Parity Information (e.g. Hamming Codes)

The Internet: It's dangerous to go alone

- Two basic things can happen:
- Bits get substituted:
  $0010\ \ 0110 \Rightarrow 0011\ \ 0100$
- Bits get lost:
  $0010\ \ 0110 \Rightarrow 0010\ \ 011$

Substitution Errors

- Long known problem; intensively researched
- Hamming [6], LDPC [5], Turbo Codes [3]
- Parity information to counter bit flips
- Hamming Codes for Prototype

Insertion/Deletion Errors

- "Potentially catastrophic" [17]
- One bit lost in the beginning and the rest is scrambled
- "Lack of good codes" and "not adequately understood" [10]

Sellers Markers

- Defined by Sellers Jr [12]
- Appending known Sequence to each block, e.g. 001
- On receiver side: Compare actual value to expected value

Evaluation

Let's see how we've done

Metrics

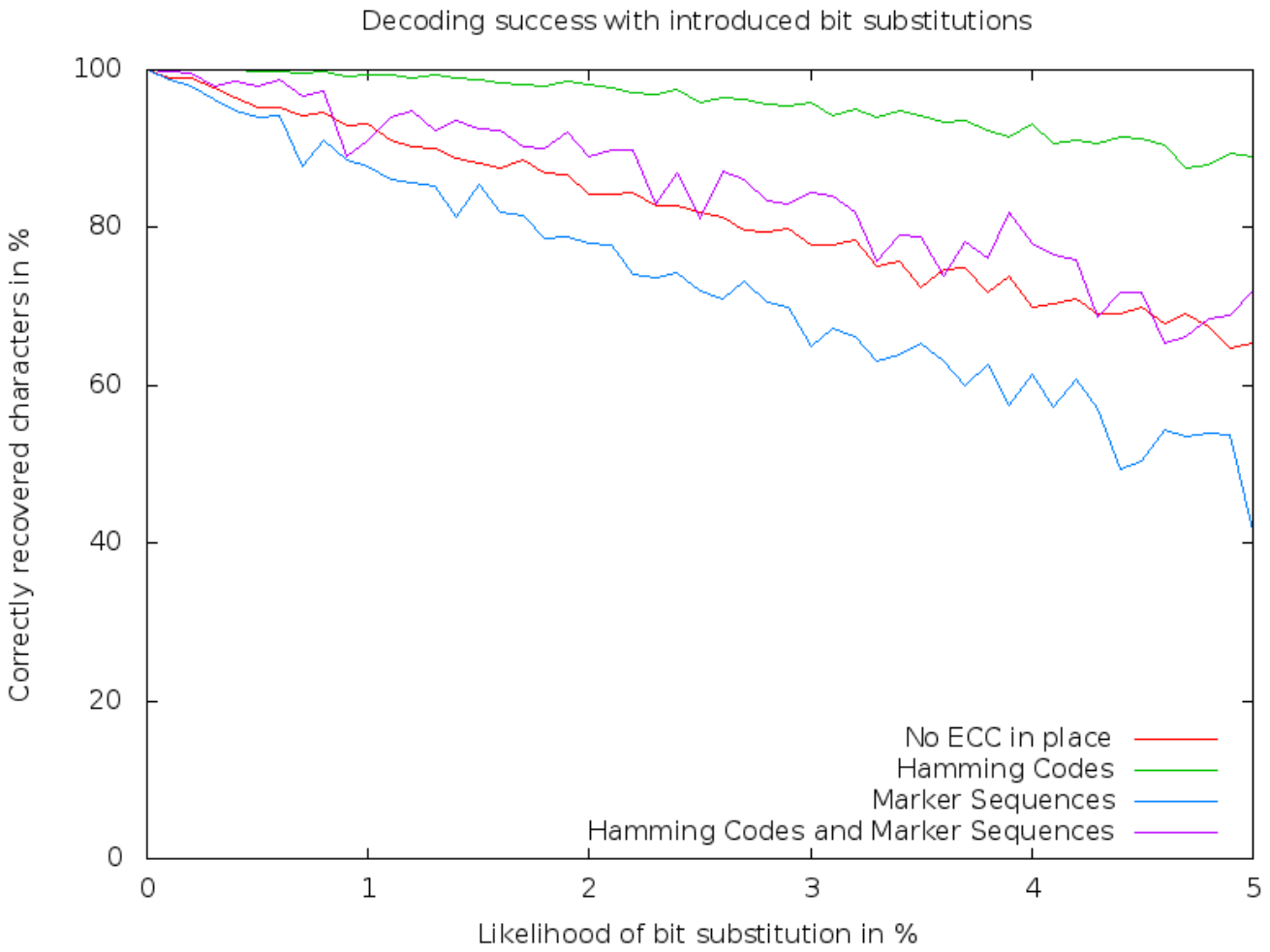
- Robustness
- Speed & Efficiency
- Stealthiness

Connection Robustness

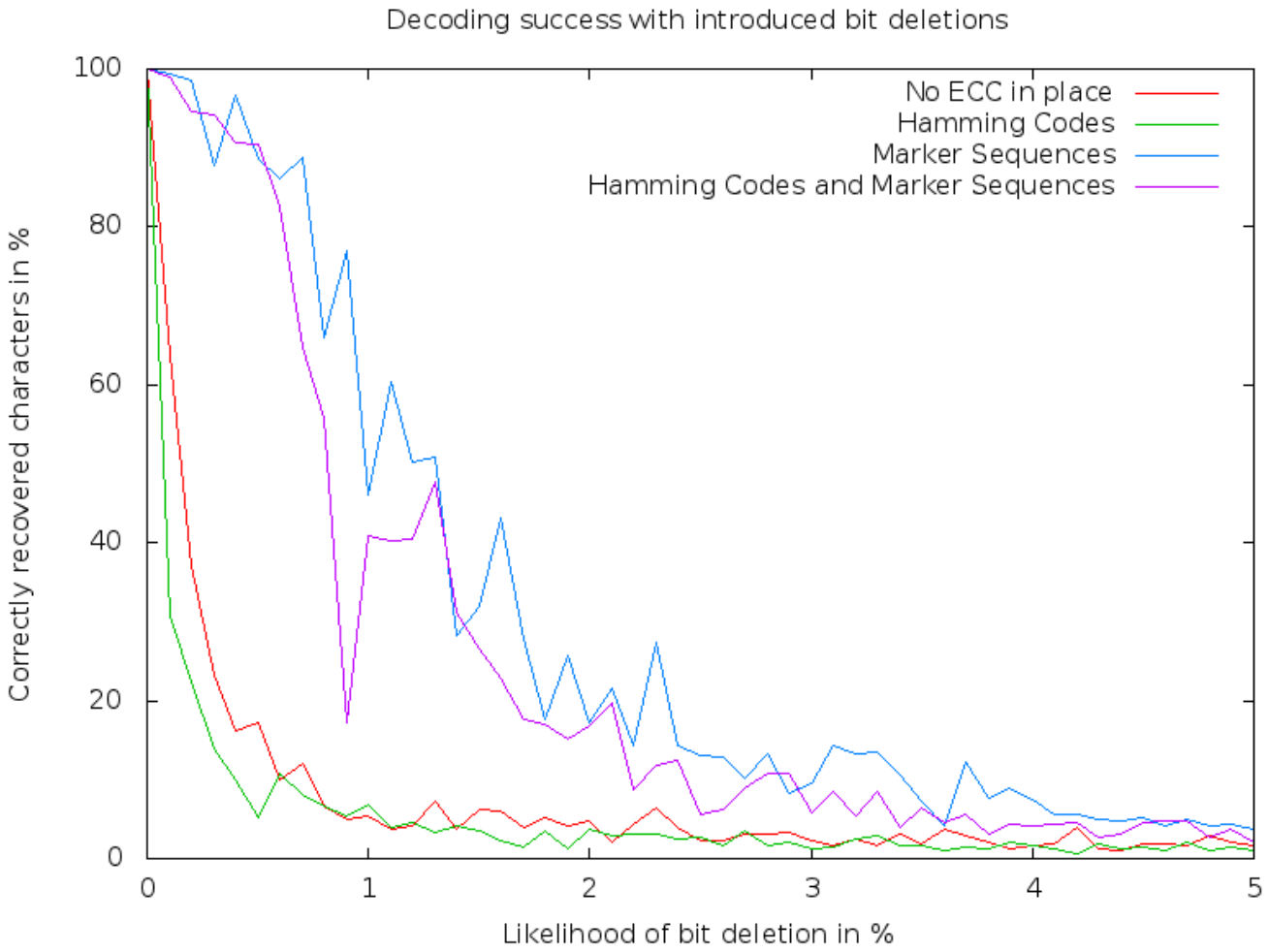
id	concealed bits	bit deletions	bit insertions	$0 \rightarrow 1$	$1 \rightarrow 0$
35217	(698b):	0.00%	0.00%	0.14%	0.00%
37228	(714b):	0.00%	0.14%	0.14%	0.00%
37914	(687b):	0.00%	0.00%	0.87%	0.00%
39671	(176b):	0.00%	0.57%	0.57%	0.00%
42046	(691b):	0.00%	2.60%	0.14%	0.14%
43247	(226b):	0.00%	0.00%	0.44%	0.00%
43712	(675b):	0.89%	0.00%	0.15%	0.44%
44906	(1383b):	0.00%	0.00%	0.07%	0.00%
48102	(1380b):	2.10%	0.07%	0.07%	1.96%
48693	(690b):	0.00%	0.14%	0.14%	0.00%
		0.30%	0.35%	0.27%	0.25%

Table 1: Sample robustness of a wired connection (experimentally obtained)

Channel Robustness



Channel Robustness



Speed & Efficiency

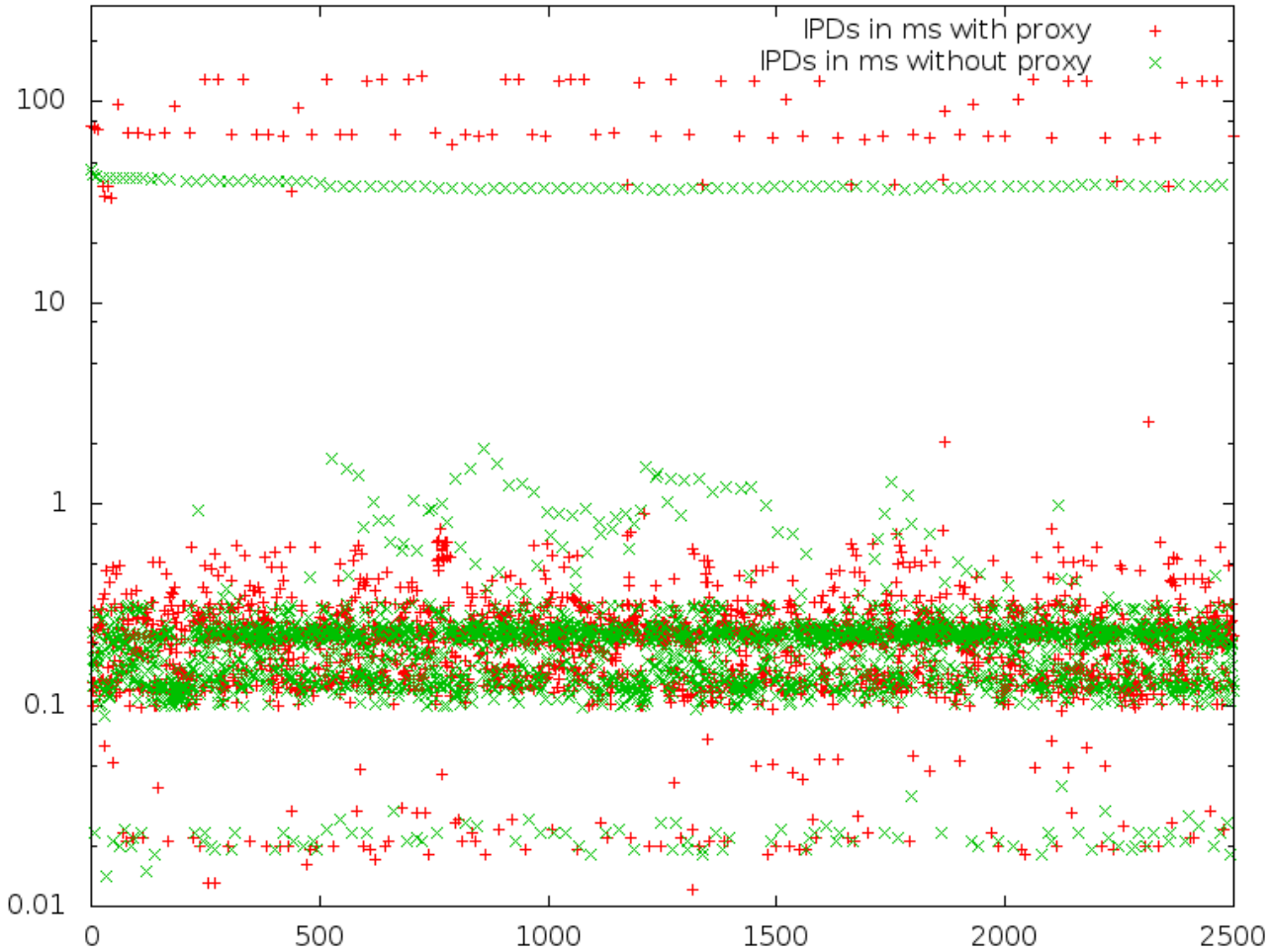
- Hidden bits per second [15]
- Hidden bits per transferred byte
- Transferred bytes per second

Speed & Efficiency

	FTP				Steganographic Proxy		
	No Proxy		With Proxy				
bytes	duration	bytes/s	duration	bytes/s	hidden bits	bytes/bit	bits/s
10485760	6.58 s	1556.1 kB	15.88s	644.7 kB	165	63,550.06	10.39
10485760	6.71 s	1525.2 kB	14.78s	693.1 kB	149	70,374.23	10.08
10485760	6.80 s	1506.8 kB	14.54s	704.3 kB	148	70,849.73	10.18
10485760	6.75 s	1517.6 kB	15.91s	643.7 kB	165	63,550.06	10.37
10485760	6.70 s	1529.4 kB	16.57s	618.1 kB	167	62,788.98	10.08
10485760	10.12 s	1011.9 kB	14.82s	691.0 kB	151	69,442.12	10.19
10485760	6.94 s	1475.3 kB	16.65s	615.1 kB	161	65,128.94	9.67
10485760	7.10 s	1442.4 kB	14.78s	692.9 kB	148	70,849.73	10.01
10485760	6.87 s	1489.7 kB	14.98s	683.6 kB	155	67,650.06	10.35
10485760	6.70 s	1528.3 kB	15.00s	682.6 kB	149	70,374.23	9.93

Table 2: Evaluation of FTP transmission speeds and steganographic performance.

Stealthiness



Conclusion

Conclusion

- Hooray, it works!

Conclusion

- Hooray, it works!
- . . . sort of
  - Better/more efficient ECC
  - Better code
  - Support for active FTP
  - . . .

Thank you!

Questions?

- mail@fkemmer.de
- https://tuebix2015.titanpad.com/kemmer-network-steganography-pad

References I

Adel El-Atawy and Ehab Al-Shaer. "Building covert channels over the packet reordering phenomenon". In: *INFOCOM 2009, IEEE*. IEEE, 2009, 2186–2194. URL: http://ieeexplore.ieee.org/xpls/abs%5C_all.jsp?arnumber=5062143 (visited on 22/08/2014).

Vincent Berk, Annarita Giani and George Cybenko. *Detection of Covert Channel Encoding in Network Packet Delays*. Tech. rep. Department of Computer Science, Dartmouth College, Nov. 2005. URL: http://www.ists.dartmouth.edu/library/149.pdf.

C. Berrou, A. Glavieux and P. Thitimajshima. "Near Shannon limit error-correcting coding and decoding: Turbo-codes. 1". In: vol. 2. IEEE, 1993, pp. 1064–1070. ISBN: 0-7803-0950-2. DOI: 10.1109/ICC.1993.397441. URL: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=397441 (visited on 22/08/2014).

References II


Serdar Cabuk, Carla E Brodley and Clay Shields. "IP covert timing channels: design and detection". In: *Proceedings of the 11th ACM conference on Computer and communications security*. 2004, pp. 178–187.


Robert G. Gallager. "Low-density parity-check codes". In: *Information Theory, IRE Transactions on* 8.1 (1962), 21–28. URL: http://ieeexplore.ieee.org/xpls/abs%5C_all.jsp?arnumber=1057683 (visited on 22/08/2014).


Richard W Hamming. "Error detecting and error correcting codes". In: *Bell System technical journal* 29.2 (1950), pp. 147–160.


Wojciech Mazurczyk, Krzysztof Cabaj and Krzysztof Szczypiorski. "What are suspicious VoIP delays?" In: *Multimedia Tools and Applications* 57.1 (2012), pp. 109–126.


Wojciech Mazurczyk, Milosz Smolarczyk and Krzysztof Szczypiorski. "Hiding Information in Retransmissions". In: *CoRR* abs/0905.0363 (2009).

References III

Wojciech Mazurczyk, Miłosz Smolarczyk and Krzysztof Szczypiorski. "On information hiding in retransmissions". In: *Telecommunication Systems* (Sept. 2011). ISSN: 1018-4864, 1572-9451. DOI: 10.1007/s11235-011-9617-y. URL: http://link.springer.com/10.1007/s11235-011-9617-y (visited on 26/11/2013).

Michael Mitzenmacher. "A Survey of results for deletion channels and related synchronization channels". en. In: *Probability Surveys* 6 (2009), pp. 1–33. ISSN: 1549-5787. DOI: 10.1214/08-PS141. URL: http://www.i-journals.org/ps/viewarticle.php?id=141%5C&layout=abstract (visited on 24/06/2014).

Rainer Poisel. *Mobile VoIP Steganography*. Vienna, Oct. 2010. URL: https://deepsec.net/docs/Slides/2010/DeepSec%5C_2010%5C_Mobile%5C_VoIP%5C_Steganography.pdf (visited on 05/06/2014).

F. Sellers Jr. "Bit loss and gain correction code". In: *Information Theory, IRE Transactions on* 8.1 (1962), 35–38. URL: http://ieeexplore.ieee.org/xpls/abs%5C_all.jsp?arnumber=1057684 (visited on 24/06/2014).

References IV

The Johns Hopkins Foreign Affairs Symposium Presents: The Price of Privacy: Re-Evaluating the NSA. Apr. 2014. URL: https://www.youtube.com/watch?v=kV2HDM86XgI (visited on 19/06/2014).

Peter Wayner. *Disappearing cryptography: information hiding: steganography & watermarking*. 3rd ed. Amsterdam ; Boston: Morgan Kaufmann Publishers, 2009. ISBN: 9780123744791.

Jingzheng Wu et al. "Improving performance of network covert timing channel through Huffman coding". In: *Mathematical and Computer Modelling* 55.1-2 (Jan. 2012), pp. 69–79. ISSN: 08957177. DOI: 10.1016/j.mcm.2011.01.051. URL: http://linkinghub.elsevier.com/retrieve/pii/S0895717711000690 (visited on 23/02/2014).

Lihong Yao et al. "A study of on/off timing channel based on packet delay distribution". In: *Computers & Security* 28.8 (Nov. 2009), pp. 785–794. ISSN: 01674048. DOI: 10.1016/j.cose.2009.05.006. URL: http://linkinghub.elsevier.com/retrieve/pii/S0167404809000510 (visited on 12/12/2013).

References V

Raman Yazdani and Masoud Ardakani. "Reliable Communication over Non-Binary Insertion/Deletion Channels". In: *IEEE Transactions on Communications* 60.12 (Dec. 2012), pp. 3597–3608. ISSN: 0090-6778. DOI: 10.1109/TCOMM.2012.100812.110547. URL: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6334504 (visited on 24/06/2014).

Xiaochao Zi et al. "Implementing a passive network covert timing channel". In: *Computers & Security* 29.6 (Sept. 2010), pp. 686–696. ISSN: 01674048. DOI: 10.1016/j.cose.2009.12.010. URL: http://linkinghub.elsevier.com/retrieve/pii/S0167404809001485 (visited on 12/12/2013).