Lab: May the (Brute) Force Be with You

A lab to reflect on forcing solutions

Daniel Bosk[1,2]

[1] School of Computer Science and Communication
KTH Royal Institute of Technology, SE-100 44 Stockholm
[2] Department of Information and Communication Systems
Mid Sweden University, SE-851 70 Sundsvall

Abstract. This lab focuses on brute forcing approaches, specifically the probability of finding *the* correct solution. It begins by finding the unknown plaintext of a given ciphertext and follows by some time to reflect on the probability of the plaintext indeed being the correct plaintext. The next step is to generate two keys, which given a ciphertext, will yield two equaly probable plaintexts.
We use some classical ciphers. Those are best covered in Chap. 1 of *Cryptography : theory and practice* [3] or "En introduktion till kryptografi" [2]. Finally, you should read about spurious keys and unicity distance. The recommended literature is Chap. 2 in [3].

1 Introduction

The idea of this assignment is to introduce the concept of brute forcing security mechanisms. The mechanism in question here is a simple monoalphabetic cryptographic algorithm. It also serves to help you reflect on deniability: how can you be sure that your solution is the correct one?

1.1 Aims

The aim of this assignment is to examine that you are:

- Able to reason about the security of basic security mechanisms.
- Have an understanding for plausible deniability.
- Able to make a proof of concept of how to break a simple and insecure mechanism.

2 Theory

If you do not have probability theory and statistics fresh in memory you are recommended to revise that. The text "Sannolikhetsteori" by Arnlind and Enblom [1] (in Swedish) treats this subject, you are recommended to read Sect. 1–4.

If you have previously taken (or are currently taking) a course on cryptography, the material from that course covering classical cryptography is enough.

Otherwise you are recommended to read “En introduktion till kryptografi” [2] (in Swedish) or Chap. 1 in *Cryptography : theory and practice* by Stinson [3].

Finally, you should read about spurious keys and unicity distance. The recommended literature is Chap. 2 in [3].

3 Assignment

The first part of the assignment is to break a monoalphabetic cipher. The intercepted text is the following:

```
TSVCFMSFQOÅCFMMBLVQTTLBUBQBUÄKEÖQSQPHMBNFCOQPHQBNNFQJMHPDG
NBSFNBSJLPDGLQXOSPHQBEJ
```

Find the corresponding plaintext of this ciphertext. When you have found a plaintext and the key, think about how certain you can be that this is indeed the correct key (and thus correct plaintext).

The second part of the assignment is about spurious keys [3, Chap. 2] and deniability. By spurious keys we mean a set of n keys $k_1, k_2, \ldots, k_n$ which all decrypt a ciphertext c to meaningful plaintexts $m_1, m_2, \ldots, m_n$. Your job is to construct such a ciphertext with $n = 2$, i.e. with two spurious keys k_1 and k_2, for the cryptosystem used in the first part of the assignment. The texts should be as long as possible (the longest meaningful plaintexts will receive an award), and they do not have to be both in Swedish or English — one plaintext in English and one in Swedish is fine.

Finding a spurious key algorithmically is possible, in this case, by generating plaintext using n-grams. However, using pen and paper is probably the most straightforward way, and probably the fastest for this short assignment.

4 Examination

You must submit your solutions to the assignment in a report (PDF-format) in the course platform. The report must contain the following:

1. The plaintext corresponding to the cryptotext given above with an explanation of what makes you sure this is the correct plaintext.
2. One ciphertext c, two keys k_1 and k_2 and the corresponding plaintexts m_1 and m_2, such that $\mathsf{Enc}_{k_1}(m_1) = c = \mathsf{Enc}_{k_2}(m_2)$. (It is not acceptable that either string contains incomprehensible text.) Explain your method for creating $c, k_1, k_2, m_1 . m_2$. Also discuss why we want to have spurious keys and how the length of the message affects the spurious keys.

References

[1] Joakim Arnlind and Andreas Enblom. “Sannolikhetsteori”. KTH:s matematiska cirkel 2007–2008, Kungliga Tekniska högskolan. 2007. URL: http://www.math.kth.se/cirkel/2007/kompendium07.pdf.

[2] Daniel Bosk. "En introduktion till kryptografi". 2015. URL: https://github.com/dbosk/introkrypt/releases/download/v1.0/introkrypt.pdf.

[3] Douglas R. Stinson. *Cryptography : theory and practice*. 3rd ed. Boca Raton: Chapman & Hall/CRC, 2006. ISBN: 1-58488-508-4 (Hardcover).