# Üebung 01

Module 1 - Introduction to Cryptography and Data Security

Knut Lucas Andersen
Master Applied Information Security
isits AG International School of IT Security

May 1, 2017

# Contents

# 1 Attack against a substitution cipher

## a) Amount of letters (A-Z) found in the ciphertext

| Character | Absolute amount | Relative amount |
|---|---|---|
| A | 22 | 1.51%. |
| B | 104 | 7.14%. |
| C | 122 | 8.37%. |
| D | 119 | 8.17%. |
| E | 43 | 2.95%. |
| F | 2 | 0.14%. |
| G | 70 | 4.80%. |
| H | 37 | 2.54%. |
| I | 0 | 0.00%. |
| J | 3 | 0.21%. |
| K | 58 | 3.98%. |
| L | 113 | 7.76%. |
| M | 31 | 2.13%. |
| N | 6 | 0.41%. |
| O | 19 | 1.30%. |
| P | 51 | 3.50%. |
| Q | 11 | 0.75%. |
| R | 94 | 6.45%. |
| S | 34 | 2.33%. |
| T | 32 | 2.20%. |
| U | 131 | 8.99%. |
| V | 45 | 3.09%. |
| W | 166 | 11.39%. |
| X | 81 | 5.56%. |
| Y | 61 | 4.19%. |
| Z | 2 | 0.14%. |

## b) Decrypting the ciphertext and c) Alphabetic substitution table

For both getting the amount, frequency and decrypted ciphertext, I wrote a program in C[1]. When mapping the frequency from the ciphertext to the frequency of English letters, the following table emerged.

[1] The program can be found on my GitHub (substitution_attack):
https://github.com/klAndersen/Msc-Appl-Info-Sec/tree/master/Module%201%20-%20Intro%20to%20Crypto/Uebung_01

| $A_{plaintext}$ | $\rightarrow$ | $U_{ciphertext}$ |
|---|---|---|
| E | $\rightarrow$ | W |
| T | $\rightarrow$ | U |
| A | $\rightarrow$ | C |
| O | $\rightarrow$ | D |
| I | $\rightarrow$ | L |
| N | $\rightarrow$ | B |
| S | $\rightarrow$ | R |
| H | $\rightarrow$ | X |
| R | $\rightarrow$ | G |
| D | $\rightarrow$ | Y |
| L | $\rightarrow$ | K |
| C | $\rightarrow$ | P |
| U | $\rightarrow$ | V |
| M | $\rightarrow$ | E |
| W | $\rightarrow$ | H |
| F | $\rightarrow$ | S |
| G | $\rightarrow$ | T |
| Y | $\rightarrow$ | M |
| P | $\rightarrow$ | A |
| B | $\rightarrow$ | O |
| V | $\rightarrow$ | Q |
| K | $\rightarrow$ | N |
| X | $\rightarrow$ | J |
| J | $\rightarrow$ | F |
| Q | $\rightarrow$ | Z |
| Z | $\rightarrow$ | I |

However, when attempting to decrypt the text, the text was not readable (see Listing 1).

Listing 1: Attempt at decrypting using developed C-program

```
STRRI?
LSW AF O STDT? LSAE'H FW PUDPIHT ON ROGT?

LSAE CI O FTAN BW LSI AF O?

YARF CILN, MTE A MDOP NIL ... IS! ESOH OH AN ONETDTHEONM
HTNHAEOIN, LSAE OH OE? OE'H A HIDE IG ... WALNONM, EONMRONM
HTNHAEOIN ON FW ... FW ... LTRR O HUPPIHT O'C BTEETD HEADE
GONCONM NAFTH GID ESONMH OG O LANE EI FAKT ANW STACLAW ON LSAE
GID EST HAKT IG LSAE O HSARR YARR AN ADMUFTNE O HSARR YARR EST
LIDRC, HI RTE'H YARR OE FW HEIFAYS.

MIIC. IIIIS, OE'H MTEEONM QUOET HEDINM. ANC STW, LSAE'H ABIUE
ESOH LSOHERONM DIADONM HIUNC MIONM PAHE LSAE O'F HUCCTNRW MIONM
EI YARR FW STAC? PTDSAPH O YAN YARR ESAE ... LONC! OH ESAE A MIIC
NAFT? OE'RR CI ... PTDSAPH O YAN GONC A BTEETD NAFT GID OE RAETD
LSTN O'VT GIUNC IUE LSAE OE'H GID. OE FUHE BT HIFTESONM VTDW
OFPIDEANE BTYAUHT ESTDT YTDEAONRW HTTFH EI BT A STRR IG A RIE IG
OE. STW! LSAE'H ESOH ESONM? ESOH ... RTE'H YARR OE A EAOR - WTAS,
EAOR. STW! O YAN YAN DTARRW ESDAHS OE ABIUE PDTEEW MIIC YAN'E O?
LIL! LIL! ESAE GTTRH MDTAE! CITHN'E HTTF EI AYSOTVT VTDW FUYS BUE
O'RR PDIBABRW GONC IUE LSAE OE'H GID RAETD IN. NIL - SAVT O BUORE
```

```
UP ANW YISTDTNE POYEUDT IG ESONMH WTE?

NI. NTVTD FONC, STW, ESOH OH DTARRW TXYOEONM, HI FUYS EI GONC IUE
ABIUE, HI FUYS EI RIIK GIDLADC EI, O'F QUOET COJJW LOES
ANEOYOPAEOIN ... ID OH OE EST LONC?

ESTDT DTARRW OH A RIE IG ESAE NIL OHN'E OE?

ANC LIL! STW! LSAE'H ESOH ESONM HUCCTNRW YIFONM EILADCH FT VTDW
GAHE? VTDW VTDW GAHE. HI BOM ANC GRAE ANC DIUNC, OE NTTCH A BOM
LOCT HIUNCONM NAFT ROKT ... IL ... IUNC ... DIUNC ... MDIUNC!
ESAE'H OE! ESAE'H A MIIC NAFT - MDIUNC!

O LINCTD OG OE LORR BT GDOTNCH LOES FT?

ANC EST DTHE, AGETD A HUCCTN LTE ESUC, LAH HORTNYT.

YUDOIUHRW TNIUMS, EST INRW ESONM ESAE LTNE ESDIUMS EST FONC IG
EST BILR IG PTEUNOAH AH OE GTRR LAH IS NI, NIE AMAON. FANW PTIPRT
SAVT HPTYURAETC ESAE OG LT KNTL TXAYERW LSW EST BILR IG PTEUNOAH
SAC ESIUMSE ESAE LT LIURC KNIL A RIE FIDT ABIUE EST NAEUDT IG EST
UNOVTDHT ESAN LT CI NIL.

HIUDYT: SEEP://LLL.YRTADLSOETROMSE.IDM/SOEYS/SSMEEM.EXE
```

When looking at the ciphertext, one thing that really stood out was the last line. This clearly was a link, so I therefore started to map the letters by hand for the link. The question now was, what could the text before the link be? By using a text editor, and replacing all the letters I had found via the link, I looked for singular occurences. Specifically, the apostrophes at the end of the words. In most English sentences, this is usually an 's' (e.g. "it's", "what's", etc.).

I also took another look at the listing that I got from the mapping. When looking at the listing, the mapping was done based on 'E' being the most used letter. However, compared to the manual decryption, 'T' was the value which the letter 'E' was mapped to. I therefore went on the assumption that for this text, plaintext letter 'E' was secondary, which meant that 'E' was represented by 'T's cipher letter. I did the same for the plaintext letter 'A', which gave me the partially readable ciphertext shown in Listing 2.

Listing 2: Partially decrypted text

```
heGGL?
whH aS D heYe? what's SH pVYpLse DB GDTe?

what PL D SeaB OH whL aS D?

MaGS PLwB, Eet a EYDp BLw ... Lh! thDs Ds aB DBteYestDBE
seBsatDLB, what Ds Dt? Dt's a sLYt LT ... HawBDBE, tDBEGDBE
seBsatDLB DB SH ... SH ... weGG D sVppLse D'P OetteY staYt
TDBPDBE BaSes TLY thDBEs DT D waBt tL SaNe aBH heaPwaH DB what
TLY the saNe LT what D shaGG MaGG aB aYEVSeBt D shaGG MaGG the
wLYGP, sL Get's MaGG Dt SH stLSaMh.

ELLP. LLLLh, Dt's EettDBE ZVDte stYLBE. aBP heH, what's aOLVt
thDs whDstGDBE YLaYDBE sLVBP ELDBE past what D'S sVPPeBGH ELDBE
tL MaGG SH heaP? peYhaps D MaB MaGG that ... wDBP! Ds that a ELLP
BaSe? Dt'GG PL ... peYhaps D MaB TDBP a OetteY BaSe TLY Dt GateY
wheB D'Qe TLVBP LVt what Dt's TLY. Dt SVst Oe sLSethDBE QeYH
```

```
DSpLYtaBt OeMaVse theYe MeYtaDBGH seeSs tL Oe a heGG LT a GLt LT
Dt. heH! what's thDs thDBE? thDs ... Get's MaGG Dt a taDG - Heah,
taDG. heH! D MaB MaB YeaGGH thYash Dt aOLVt pYettH ELLP MaB't D?
wLw! wLw! that TeeGs EYeat! PLesB't seeS tL aMhDeQe QeYH SVMh OVt
D'GG pYLOaOGH TDBP LVt what Dt's TLY GateY LB. BLw - haQe D OVDGt
Vp aBH MLheYeBt pDMtVYe LT thDBEs Het?

BL. BeQeY SDBP, heH, thDs Ds YeaGGH eJMDtDBE, sL SVMh tL TDBP LVt
aOLVt, sL SVMh tL GLLN TLYwaYP tL, D'S ZVDte PDFFH wDth
aBtDMDpatDLB ... LY Ds Dt the wDBP?

theYe YeaGGH Ds a GLt LT that BLw DsB't Dt?

aBP wLw! heH! what's thDs thDBE sVPPeBGH MLSDBE tLwaYPs Se QeYH
Tast? QeYH QeYH Tast. sL ODE aBP TGat aBP YLVBP, Dt BeePs a ODE
wDPe sLVBPDBE BaSe GDNe ... Lw ... LVBP ... YLVBP ... EYLVBP!
that's Dt! that's a ELLP BaSe - EYLVBP!

D wLBPeY DT Dt wDGG Oe TYDeBPs wDth Se?

aBP the Yest, aTteY a sVPPeB wet thVP, was sDGeBMe.

MVYDLVsGH eBLVEh, the LBGH thDBE that weBt thYLVEh the SDBP LT
the OLwG LT petVBDas as Dt TeGG was Lh BL, BLt aEaDB. SaBH peLpGe
haQe speMVGateP that DT we NBew eJaMtGH whH the OLwG LT petVBDas
haP thLVEht that we wLVGP NBLw a GLt SLYe aOLVt the BatVYe LT the
VBDQeYse thaB we PL BLw.

sLVYMe: http://www.MGeaYwhDteGDEht.LYE/hDtMh/hhEttE.tJt
```

Looking through the text, cipher letter 'D' is the only letter that is alone, which implied that this was the plaintext letter 'I'. The first word in the text also stands out. In the beginning it was uncertain what it could be, but given that it had to repetitive letters (cipher letter 'G'), this could mean that this would in plain text be "hello". It now also become more obvious that the word before the ciphered link had to be the word "source". Further investigation when looking at the link, and this partially decrypted sentence ("...Eet a Erip Bow ..."), the cipher text 'E' was equal to plaintext 'G'.

Listing 3: Partially decrypted text - part 2

```
hello?
whH aS i here? what's SH purpose iB liTe?

what Po i SeaB OH who aS i?

calS PowB, get a grip Bow ... oh! this is aB iBterestiBg
seBsatioB, what is it? it's a sort oT ... HawBiBg, tiBgliBg
seBsatioB iB SH ... SH ... well i suppose i'P Oetter start
TiBPiBg BaSes Tor thiBgs iT i waBt to SaNe aBH heaPwaH iB what
Tor the saNe oT what i shall call aB arguSeBt i shall call the
worlP, so let's call it SH stoSach.

gooP. ooooh, it's gettiBg Zuite stroBg. aBP heH, what's aOout
this whistliBg roariBg souBP goiBg past what i'S suPPeBlH goiBg
to call SH heaP? perhaps i caB call that ... wiBP! is that a gooP
BaSe? it'll Po ... perhaps i caB TiBP a Oetter BaSe Tor it later
wheB i'Qe TouBP out what it's Tor. it Sust Oe soSethiBg QerH
iSportaBt Oecause there certaiBlH seeSs to Oe a hell oT a lot oT
it. heH! what's this thiBg? this ... let's call it a tail - Heah,
tail. heH! i caB caB reallH thrash it aOout prettH gooP caB't i?
```

```
wow! wow! that Teels great! PoesB't seeS to achieQe QerH Such Out
i'll proOaOlH TiBP out what it's Tor later oB. Bow - haQe i Ouilt
up aBH cohereBt picture oT thiBgs Het?

Bo. BeQer SiBP, heH, this is reallH eJcitiBg, so Such to TiBP out
aOout, so Such to looN TorwarP to, i'S Zuite PiFFH with
aBticipatioB ... or is it the wiBP?

there reallH is a lot oT that Bow isB't it?

aBP wow! heH! what's this thiBg suPPeBlH coSiBg towarPs Se QerH
Tast? QerH QerH Tast. so Oig aBP Tlat aBP rouBP, it BeePs a Oig
wiPe souBPiBg BaSe liNe ... ow ... ouBP ... rouBP ... grouBP!
that's it! that's a gooP BaSe - grouBP!

i woBPer iT it will Oe TrieBPs with Se?

aBP the rest, aTter a suPPeB wet thuP, was sileBce.

curiouslH eBough, the oBlH thiBg that weBt through the SiBP oT
the Oowl oT petuBias as it Tell was oh Bo, Bot agaiB. SaBH people
haQe speculateP that iT we NBew eJactlH whH the Oowl oT petuBias
haP thought that we woulP NBow a lot Sore aOout the Bature oT the
uBiQerse thaB we Po Bow.

source: http://www.clearwhitelight.org/hitch/hhgttg.tJt
```

After getting this much of the text decrypted, the rest of the mapping was done based on what letters was most probable for the given word. The resulting mapping table is shown below.

| $A_{plaintext}$ | → | $U_{ciphertext}$ |
|---|---|---|
| A | → | C |
| B | → | O |
| C | → | M |
| D | → | P |
| E | → | U |
| F | → | T |
| G | → | E |
| H | → | R |
| I | → | D |
| J | → | I |
| K | → | N |
| L | → | G |
| M | → | S |
| N | → | B |
| O | → | L |
| P | → | A |
| Q | → | Z |
| R | → | Y |
| S | → | X |
| T | → | W |
| U | → | V |
| V | → | Q |
| W | → | K |
| X | → | J |
| Y | → | H |
| Z | → | F |

Quote:

> "hello? why am i here? what's my purpose in life?
>
> what do i mean by who am i?
>
> calm down, get a grip now ... oh! this is an interesting sensation, what is it? it's a sort of ... yawning, tingling sensation in my ... my ... well i suppose i'd better start finding names for things if i want to make any headway in what for the sake of what i shall call an argument i shall call the world, so let's call it my stomach."

## c) 1d) Key space

The key space for 26 letters is $26! \sim= 2^{88}$.

## d) 1e) Book and Author

Based on the link in the cipher-text (http://www.clearwhitelight.org/hitch/hhgttg.txt), the book is "The Hitchhiker's Guide to the Galaxy". The author is Douglas N. Adams.

# 2 Modulare Arithmetik I

## a) 5 * 9 mod 19

$$\begin{aligned} & 5 \times 9 \mod 19 \\ \Rightarrow\ & 45 \mod 19 \\ \Rightarrow\ & \frac{45}{19} \\ \Rightarrow\ & 19 \times 2 = 38; \\ & 45 - 38 = 7 \\ \Rightarrow\ & 45 \mod 19 = 7 \end{aligned} \tag{1}$$

# 3 Cäsar-Chiffre

## a) Alphabet mapping

When shifting letters, you just move N letters forward into the alphabet. If the offset is 6, then 'A'⇒ 'G', 'B' ⇒ 'H', etc. Which means that for the letters 'V' to 'Z' you get the following table:

| Plain letter | A | B | ... | V | W | X | Y | Z |
|---|---|---|---|---|---|---|---|---|
| Shifted letter | G | H | ... | B | C | D | E | F |

## b) Decrypt "pelcgbtencul"

The encrypted ciphertext "pelcgbtencul" with an offset with k=13 becomes the text "CRYPTOGRAPHY"[2]. This shift is known as ROT13.

## c) Using offset k=26

The use of the offset k=26 is not practical, because the offset is based on the amount of letters you shift. Since the alphabet contains 26 letters, you would end up going from 'A' + 26 letters ⇒ 'A'. You basically end up with a cipher text that is equal to the plaintext.

## d) Statistical attacks

Substitution cipher is more secure, because it uses a key of varying length. When it comes to Caesar cipher, it only uses an offset. Therefore, if you use a statistical attack and find that a given cipher is repeated often, one can assume that this letter is either an 'E' or 'T' (presuming English text). Then one could simply map the 'E' based on its belonging cipher,map the remaining letter based on the offset, and see if the plaintext "makes sense".

[2] Decrypted by using "shift_letters":
https://github.com/klAndersen/Msc-Appl-Info-Sec/tree/master/Module%201%20-%20Intro%20to%20Crypto/Uebung_01