

# Bezpieczeństwo sieci komputerowych

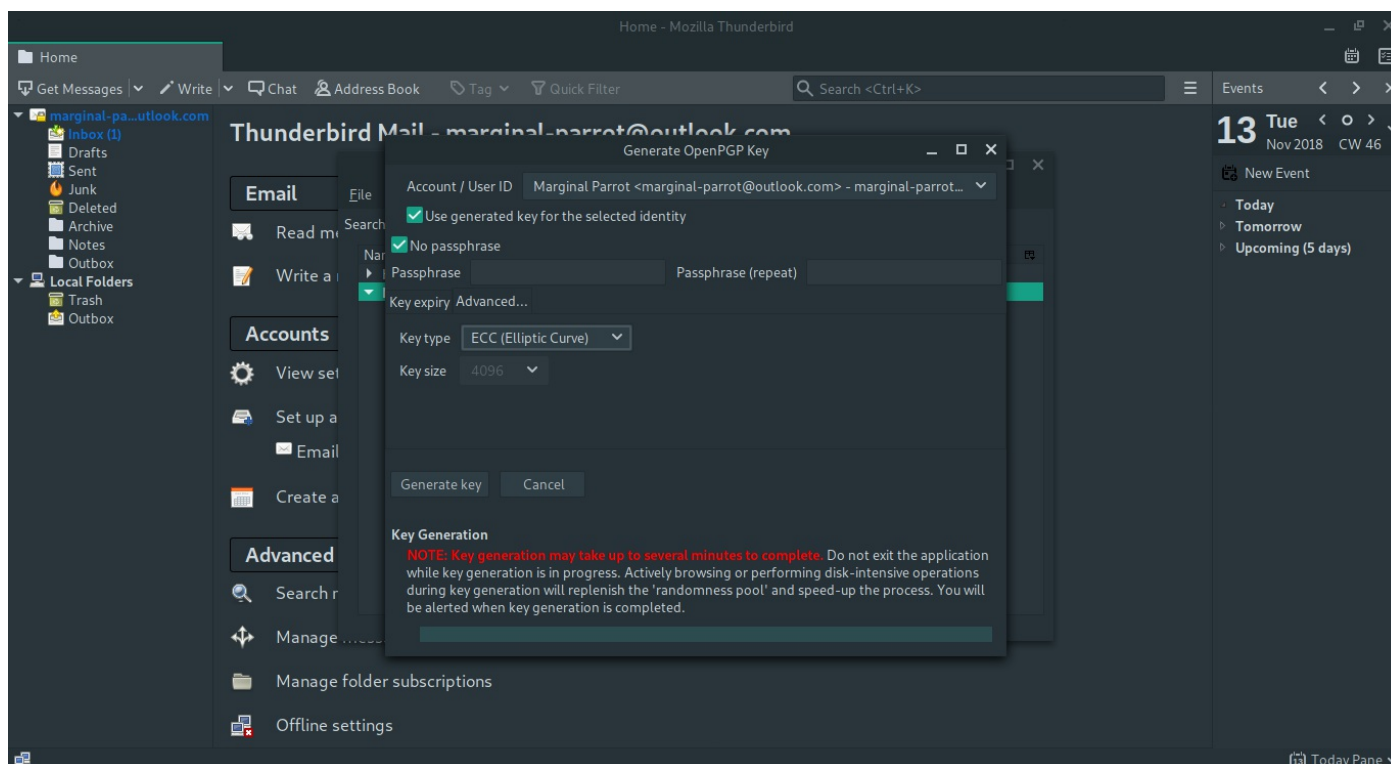
## Sprawozdanie z laboratorium

Data	Tytuł zajęć	Uczestnicy
30.10.2018 16:10	Kryptografia	Iwo Bujkiewicz (226203)

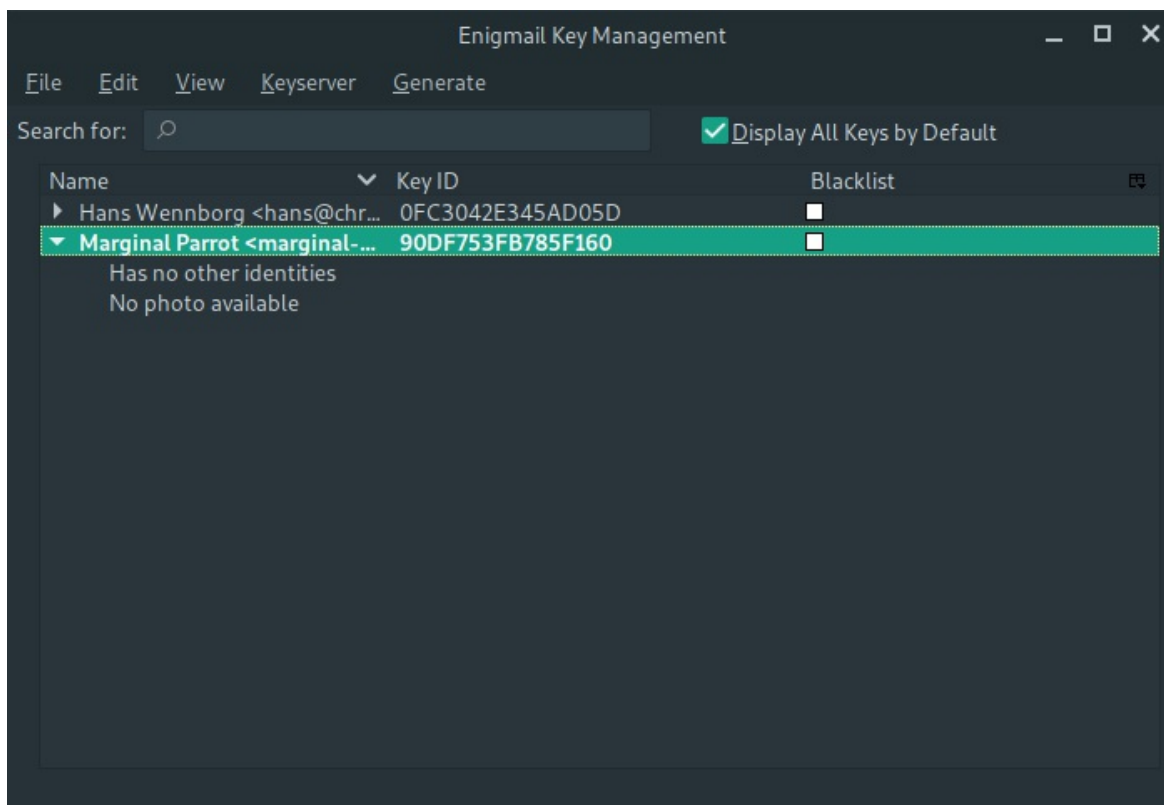
## Wyniki realizacji zadań

### Zadanie 2.

Podczas generowania pary kluczy przy użyciu wtyczki Enigmail w programie Thunderbird wybrano rodzaj klucza **Elliptic Curve**, co zaowocowało wygenerowaniem klucza Ed25519, oraz brak terminu ważności. Kryptografia oparta o klucze Ed25519 uważana jest za nieco skuteczniejszą od opartej o klucze RSA (często nawet w postaci 4096-bitowej), jednak nie jest tak powszechnie wykorzystywana ze względu na kompatybilność wsteczną.

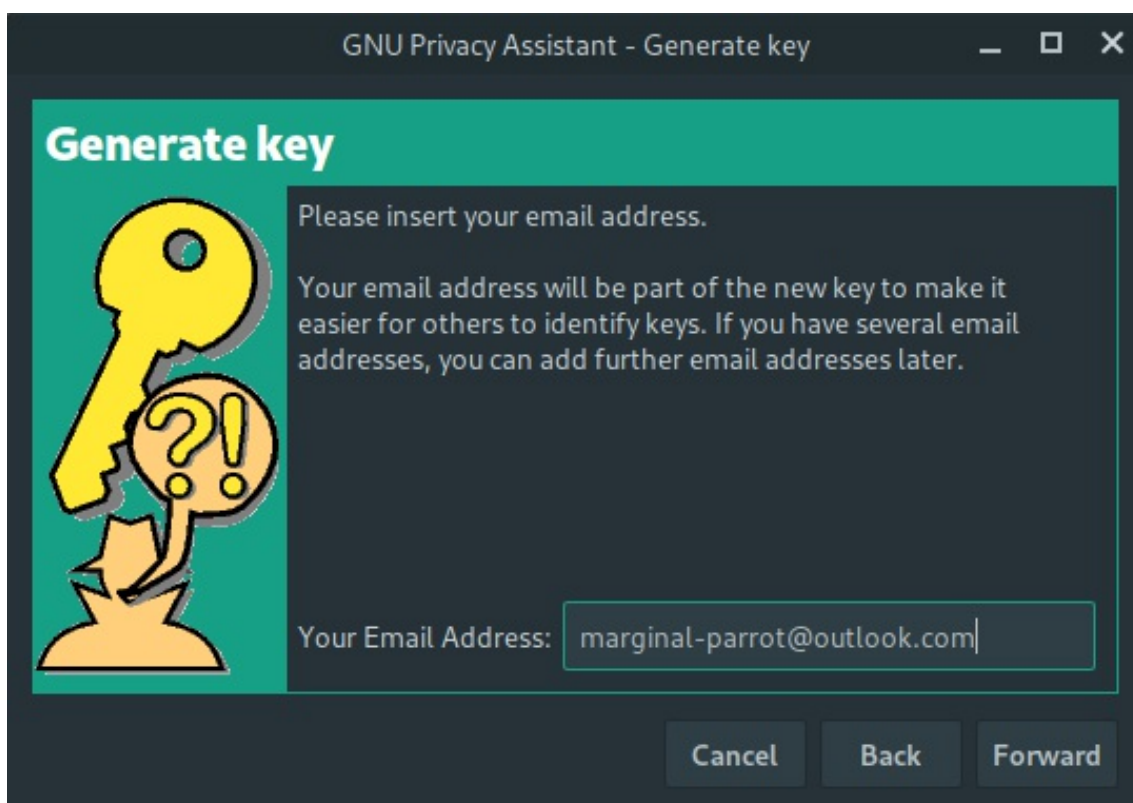


Generowanie pary kluczy przy użyciu Enigmail

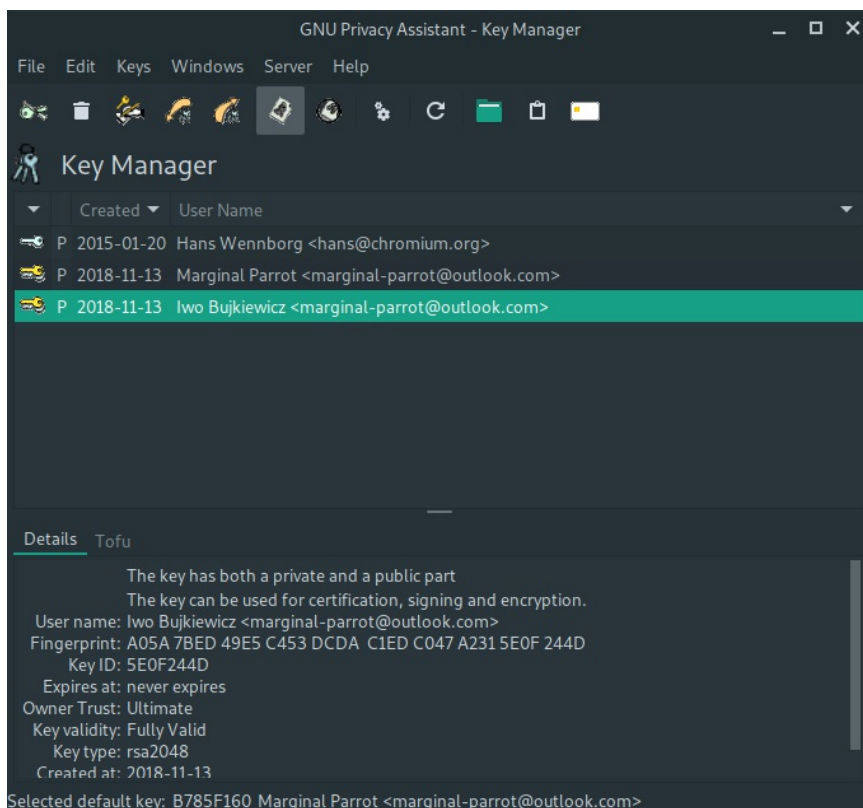


*Klucz publiczny wygenerowany przy użyciu Enigmail*

Podczas generowania pary kluczy przy użyciu GNU Privacy Assistant wybrano domyślny rodzaj klucza RSA w postaci 2048-bitowej oraz brak terminu ważności.



*Generowanie pary kluczy przy użyciu GPA*



*Para kluczy wygenerowana przy użyciu GPA*

Podczas generowania pary kluczy przy użyciu CLI GPG wybrano generowanie 4096-bitowego klucza RSA oraz termin ważności określony na 2 lata.

```
outfrost@Vaelastrasz:~  
File Edit View Search Terminal Help  
[outfrost@Vaelastrasz ~]$ man gpg  
[outfrost@Vaelastrasz ~]$ gpg --full-gen-key  
gpg (GnuPG) 2.2.10; Copyright (C) 2018 Free Software Foundation, Inc.  
This is free software: you are free to change and redistribute it.  
There is NO WARRANTY, to the extent permitted by law.  
  
Please select what kind of key you want:  
  (1) RSA and RSA (default)  
  (2) DSA and Elgamal  
  (3) DSA (sign only)  
  (4) RSA (sign only)  
Your selection? 1  
RSA keys may be between 1024 and 4096 bits long.  
What keysize do you want? (2048) 4096  
Requested keysize is 4096 bits  
Please specify how long the key should be valid.  
  0 = key does not expire  
  <n> = key expires in n days  
  <n>w = key expires in n weeks  
  <n>m = key expires in n months  
  <n>y = key expires in n years  
Key is valid for? (0) 2y  
Key expires at Thu 12 Nov 2020 14:47:52 CET  
Is this correct? (y/N) █
```

*Generowanie pary kluczy przy użyciu CLI GPG*

```
outfrost@Vaelastrasz:~  
File Edit View Search Terminal Help  
Comment:  
You selected this USER-ID:  
  "Iwo Bujkiewicz <marginal-parrot@outlook.com>"  
  
Change (N)ame, (C)omment, (E)mail or (O)kay/(Q)uit? o  
We need to generate a lot of random bytes. It is a good idea to perform  
some other action (type on the keyboard, move the mouse, utilize the  
disks) during the prime generation; this gives the random number  
generator a better chance to gain enough entropy.  
We need to generate a lot of random bytes. It is a good idea to perform  
some other action (type on the keyboard, move the mouse, utilize the  
disks) during the prime generation; this gives the random number  
generator a better chance to gain enough entropy.  
gpg: key F7BA1AE17D0A1E9B marked as ultimately trusted  
gpg: revocation certificate stored as '/home/outfrost/.gnupg/openpgp-revocs.d/74  
D86D6F9B9F372FD28C8D07F7BA1AE17D0A1E9B.rev'  
public and secret key created and signed.  
  
pub   rsa4096 2018-11-13 [SC] [expires: 2020-11-12]  
       74D86D6F9B9F372FD28C8D07F7BA1AE17D0A1E9B  
uid           Iwo Bujkiewicz <marginal-parrot@outlook.com>  
sub   rsa4096 2018-11-13 [E] [expires: 2020-11-12]  
  
[outfrost@Vaelastrasz ~]$
```

*Generowanie pary kluczy przy użyciu CLI GPG*

```
outfrost@Vaelastrasz:~  
File Edit View Search Terminal Help  
gpg: next trustdb check due at 2020-11-12  
/home/outfrost/.gnupg/pubring.kbx  
-----  
pub   rsa4096 2015-01-20 [SC] [expires: 2023-01-15]  
       B6C8F98282B944E3B0D5C2530FC3042E345AD05D  
uid           [ unknown] Hans Wennborg <hans@chromium.org>  
sub   rsa4096 2015-01-20 [E] [expires: 2023-01-15]  
  
pub   ed25519 2018-11-13 [SC]  
       EA925BDD64F857675E8ADF3E90DF753FB785F160  
uid           [ultimate] Marginal Parrot <marginal-parrot@outlook.com>  
sub   cv25519 2018-11-13 [E]  
  
pub   rsa2048 2018-11-13 [SC]  
       A05A7BED49E5C453DCDAC1EDC047A2315E0F244D  
uid           [ultimate] Iwo Bujkiewicz <marginal-parrot@outlook.com>  
sub   rsa2048 2018-11-13 [E]  
  
pub   rsa4096 2018-11-13 [SC] [expires: 2020-11-12]  
       74D86D6F9B9F372FD28C8D07F7BA1AE17D0A1E9B  
uid           [ultimate] Iwo Bujkiewicz <marginal-parrot@outlook.com>  
sub   rsa4096 2018-11-13 [E] [expires: 2020-11-12]  
  
[outfrost@Vaelastrasz ~]$
```

*Klucz publiczny wygenerowany przy użyciu CLI GPG*

Dla żadnego z kluczy prywatnych nie ustalono hasła szyfrującego klucz podczas przechowywania. Do wykonania kolejnych zadań wybrano parę 4096-bitowych kluczy RSA.



### Zadanie 3.

```
outfrost@Vaelastrasz: ~  
File Edit View Search Terminal Help  
[outfrost@Vaelastrasz ~]$ gpg --armor --output Bujkiewicz.pub.asc --export 7D0A1E9B  
[outfrost@Vaelastrasz ~]$ less Bujkiewicz.pub.asc  
[outfrost@Vaelastrasz ~]$ head Bujkiewicz.pub.asc  
-----BEGIN PGP PUBLIC KEY BLOCK-----  
  
mQINBFvq1igBEACu3TT8gY/8Vmh7gnsr0GmIBctFaGguBtM1C9p5N4kEhoyGqdW7  
yph0APCE1gToD/fZzQc/D4UoYIN39B04rIt2SoicU7Rq/6zvF8+Y6SZg5qGuyRdN  
VIYMLykrPCuTr1pa/69lwHuDedqWAhZN+SLAyHIHZbJPTqayFmy545CyG7Nm+cpJ  
jM9p1tfJp6AAs27b08cZdwtddMFqG3WEQJojsAL5p4zaJA5udccu00uei2vkLAKn  
kCQJiYIsosRVZCi3VMvUSE0PJDJbsod8pUSg8y7FbESvvVBukj85Bi0oQCYRtC3w  
0jkoh3DdVpzTd96o8jfYyIrHEiAeXWqGM/nc73eWtjLNR6QwwgkFEcNtujHqFf3k  
YuQVRgFw5XL79BkRW9UJn1/hx23stPVvbNpdGvGXWw3RBDt5uc2XLnL9cXlFwVF4  
wRNaAtNz91ifp0Zm/0z8K96vpxyPW8SmoasK5LXnRimAlqFtvMMhdtQq8kCEKpD  
[outfrost@Vaelastrasz ~]$ gpg --armor --output Bujkiewicz.asc --export-secret-ke  
ys 7D0A1E9B  
[outfrost@Vaelastrasz ~]$ head Bujkiewicz.asc
```

Klucze publiczny i prywatny wyeksportowane do plików

```
outfrost@Vaelastrasz: ~  
File Edit View Search Terminal Help  
-----BEGIN PGP PUBLIC KEY BLOCK-----  
  
mQINBFvq1igBEACu3TT8gY/8Vmh7gnsr0GmIBctFaGguBtM1C9p5N4kEhoyGqdW7  
yph0APCE1gToD/fZzQc/D4UoYIN39B04rIt2SoicU7Rq/6zvF8+Y6SZg5qGuyRdN  
VIYMLykrPCuTr1pa/69lwHuDedqWAhZN+SLAyHIHZbJPTqayFmy545CyG7Nm+cpJ  
jM9p1tfJp6AAs27b08cZdwtddMFqG3WEQJojsAL5p4zaJA5udccu00uei2vkLAKn  
kCQJiYIsosRVZCi3VMvUSE0PJDJbsod8pUSg8y7FbESvvVBukj85Bi0oQCYRtC3w  
0jkoh3DdVpzTd96o8jfYyIrHEiAeXWqGM/nc73eWtjLNR6QwwgkFEcNtujHqFf3k  
YuQVRgFw5XL79BkRW9UJn1/hx23stPVvbNpdGvGXWw3RBDt5uc2XLnL9cXlFwVF4  
wRNaAtNz91ifp0Zm/0z8K96vpxyPW8SmoasK5LXnRimAlqFtvMMhdtQq8kCEKpD  
[outfrost@Vaelastrasz ~]$ gpg --armor --output Bujkiewicz.asc --export-secret-ke  
ys 7D0A1E9B  
[outfrost@Vaelastrasz ~]$ head Bujkiewicz.asc  
-----BEGIN PGP PRIVATE KEY BLOCK-----  
  
lQcYBFvq1igBEACu3TT8gY/8Vmh7gnsr0GmIBctFaGguBtM1C9p5N4kEhoyGqdW7  
yph0APCE1gToD/fZzQc/D4UoYIN39B04rIt2SoicU7Rq/6zvF8+Y6SZg5qGuyRdN  
VIYMLykrPCuTr1pa/69lwHuDedqWAhZN+SLAyHIHZbJPTqayFmy545CyG7Nm+cpJ  
jM9p1tfJp6AAs27b08cZdwtddMFqG3WEQJojsAL5p4zaJA5udccu00uei2vkLAKn  
kCQJiYIsosRVZCi3VMvUSE0PJDJbsod8pUSg8y7FbESvvVBukj85Bi0oQCYRtC3w  
0jkoh3DdVpzTd96o8jfYyIrHEiAeXWqGM/nc73eWtjLNR6QwwgkFEcNtujHqFf3k  
YuQVRgFw5XL79BkRW9UJn1/hx23stPVvbNpdGvGXWw3RBDt5uc2XLnL9cXlFwVF4  
wRNaAtNz91ifp0Zm/0z8K96vpxyPW8SmoasK5LXnRimAlqFtvMMhdtQq8kCEKpD  
[outfrost@Vaelastrasz ~]$
```

Klucze publiczny i prywatny wyeksportowane do plików

74D86D6F9B9F372FD28C8D07F7BA1AE17D0A1E9B

Odcisk klucza publicznego

#### Zadanie 4.

Klucz publiczny wyeksportowano na serwer `pgp.mit.edu`.

```
outfrost@Vaelastrasz:~  
File Edit View Search Terminal Help  
sub  rsa2048 2018-11-13 [E]  
pub  rsa4096 2018-11-13 [SC] [expires: 2020-11-12]  
     74D86D6F9B9F372FD28C8D07F7BA1AE17D0A1E9B  
uid      [ultimate] Iwo Bujkiewicz <marginal-parrot@outlook.com>  
sub  rsa4096 2018-11-13 [E] [expires: 2020-11-12]  
  
[outfrost@Vaelastrasz ~]$ gpg --keyserver pgp.mit.edu --send-keys 7D0A1E9B  
gpg: sending key F7BA1AE17D0A1E9B to hkp://pgp.mit.edu  
[outfrost@Vaelastrasz ~]$ gpg --keyserver pgp.mit.edu --recv-keys 7D0A1E9B  
gpg: keyserver receive failed: No data  
[outfrost@Vaelastrasz ~]$ gpg --keyserver pgp.mit.edu --search-key marginal-parrot@outlook.com  
gpg: data source: http://pgp.mit.edu:11371  
(1)      Iwo Bujkiewicz <marginal-parrot@outlook.com>  
        4096 bit RSA key F7BA1AE17D0A1E9B, created: 2018-11-13, expires: 2020-11-12  
Keys 1-1 of 1 for "marginal-parrot@outlook.com". Enter number(s), N)ext, or Q)uit > 1  
gpg: key F7BA1AE17D0A1E9B: "Iwo Bujkiewicz <marginal-parrot@outlook.com>" not changed  
gpg: Total number processed: 1  
gpg:           unchanged: 1  
[outfrost@Vaelastrasz ~]$
```

*Ekspert klucza publicznego na serwer kluczy i wyszukiwanie klucza na serwerze kluczy*

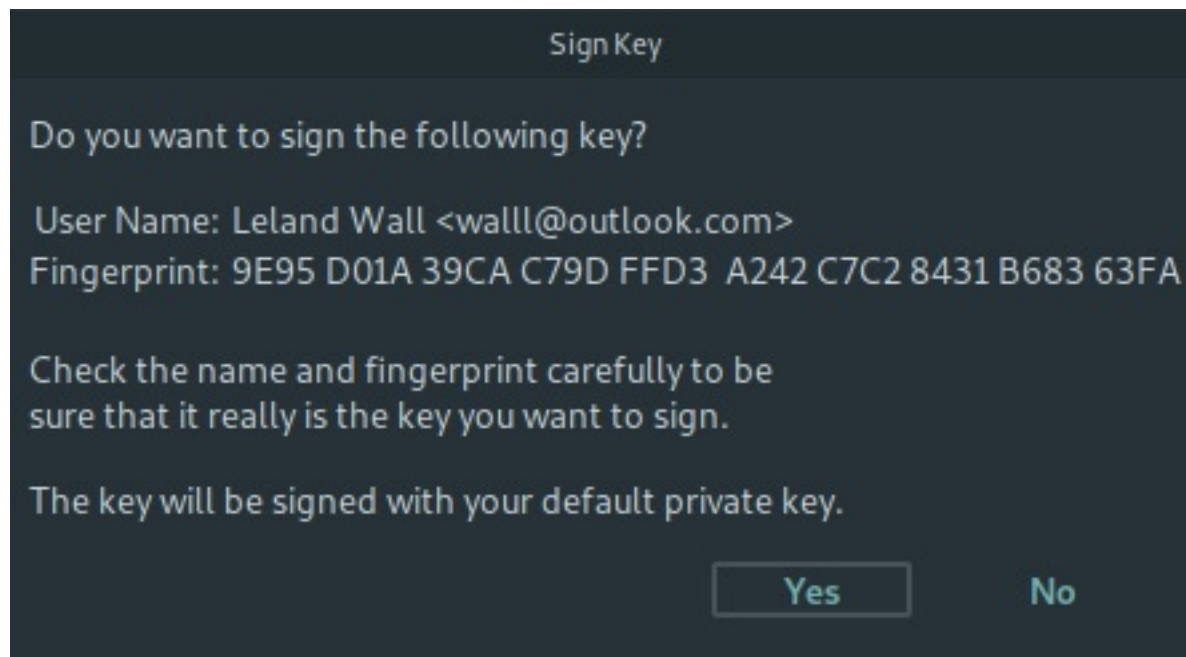
Klucz można unieważnić, importując w GPG certyfikat unieważnienia klucza, a następnie wysyłając zmodyfikowany w ten sposób klucz na serwery kluczy oraz do indywidualnych osób i organizacji.

```
outfrost@Vaelastrasz:~  
File Edit View Search Terminal Help  
how to use this.  
  
--generate-revocation name  
--gen-revoke name  
Generate a revocation certificate for the complete key. To only  
revoke a subkey or a key signature, use the --edit command.  
  
This command merely creates the revocation certificate so that  
it can be used to revoke the key if that is ever needed. To  
actually revoke a key the created revocation certificate needs  
to be merged with the key to revoke. This is done by importing  
the revocation certificate using the --import command. Then the  
revoked key needs to be published, which is best done by sending  
the key to a keyserver (command --send-key) and by exporting  
(--export) it to a file which is then send to frequent communi-  
cation partners.  
  
--generate-designated-revocation name  
--desig-revoke name  
Generate a designated revocation certificate for a key. This  
allows a user (with the permission of the keyholder) to revoke  
someone else's key.  
  
Manual page gpg(1) line 603 (press h for help or q to quit)
```

*Wycinek manuala GPG - sposób unieważnienia klucza*

## Zadanie 5.

Z uwagi na wykonywanie zadania w grupie ówczas jednoosobowej, wygenerowany został dodatkowy klucz fikcyjnej osoby.



*Podpisywanie zaimportowanego klucza publicznego*

```
outfrost@Vaelastrasz: ~/University/BSK
File Edit View Search Terminal Help
uid      [ unknown] Hans Wennborg <hans@chromium.org>
sub      rsa4096 2015-01-20 [E] [expires: 2023-01-15]

pub      ed25519 2018-11-13 [SC]
          EA925BDD64F857675E8ADF3E90DF753FB785F160
uid      [ultimate] Marginal Parrot <marginal-parrot@outlook.com>
sub      cv25519 2018-11-13 [E]

pub      rsa2048 2018-11-13 [SC]
          A05A7BED49E5C453DCDAC1EDC047A2315E0F244D
uid      [ultimate] Iwo Bujkiewicz <marginal-parrot@outlook.com>
sub      rsa2048 2018-11-13 [E]

pub      rsa4096 2018-11-13 [SC] [expires: 2020-11-12]
          74D86D6F9B9F372FD28C8D07F7BA1AE17D0A1E9B
uid      [ultimate] Iwo Bujkiewicz <marginal-parrot@outlook.com>
sub      rsa4096 2018-11-13 [E] [expires: 2020-11-12]

pub      rsa4096 2018-11-15 [SC] [expires: 2019-05-14]
          9E95D01A39CAC79DFFD3A242C7C28431B68363FA
uid      [ultimate] Leland Wall <walll@outlook.com>
sub      rsa4096 2018-11-15 [E] [expires: 2019-05-14]

[outfrost@Vaelastrasz BSK]$
```

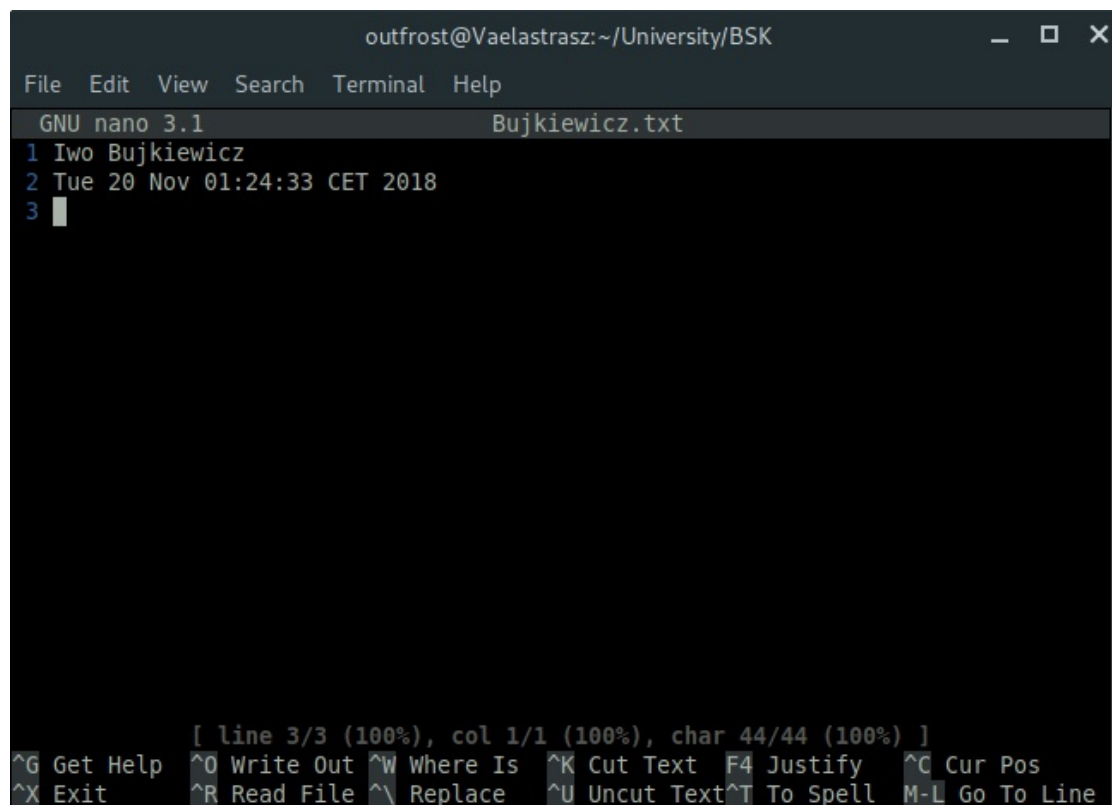
*Podpisany zaimportowany klucz publiczny*



## Zadanie 6.

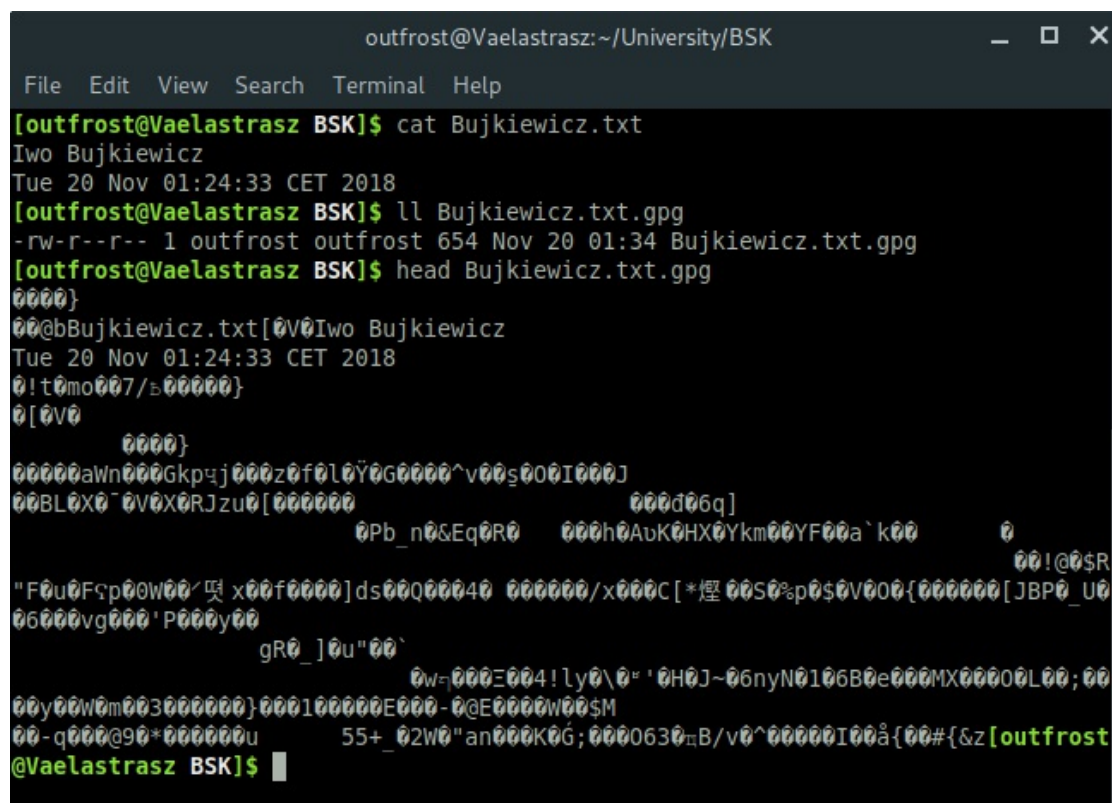
Plik tekstowy został podpisany przy użyciu następującej komendy.

```
$ gpg --default-key 7D0A1E9B --sign Bujkiewicz.txt
```



```
outfrost@Vaelastrasz:~/University/BSK
File Edit View Search Terminal Help
GNU nano 3.1 Bujkiewicz.txt
1 Iwo Bujkiewicz
2 Tue 20 Nov 01:24:33 CET 2018
3
[ line 3/3 (100%), col 1/1 (100%), char 44/44 (100%) ]
^G Get Help ^O Write Out ^W Where Is ^K Cut Text F4 Justify ^C Cur Pos
^X Exit ^R Read File ^\ Replace ^U Uncut Text ^T To Spell M-L Go To Line
```

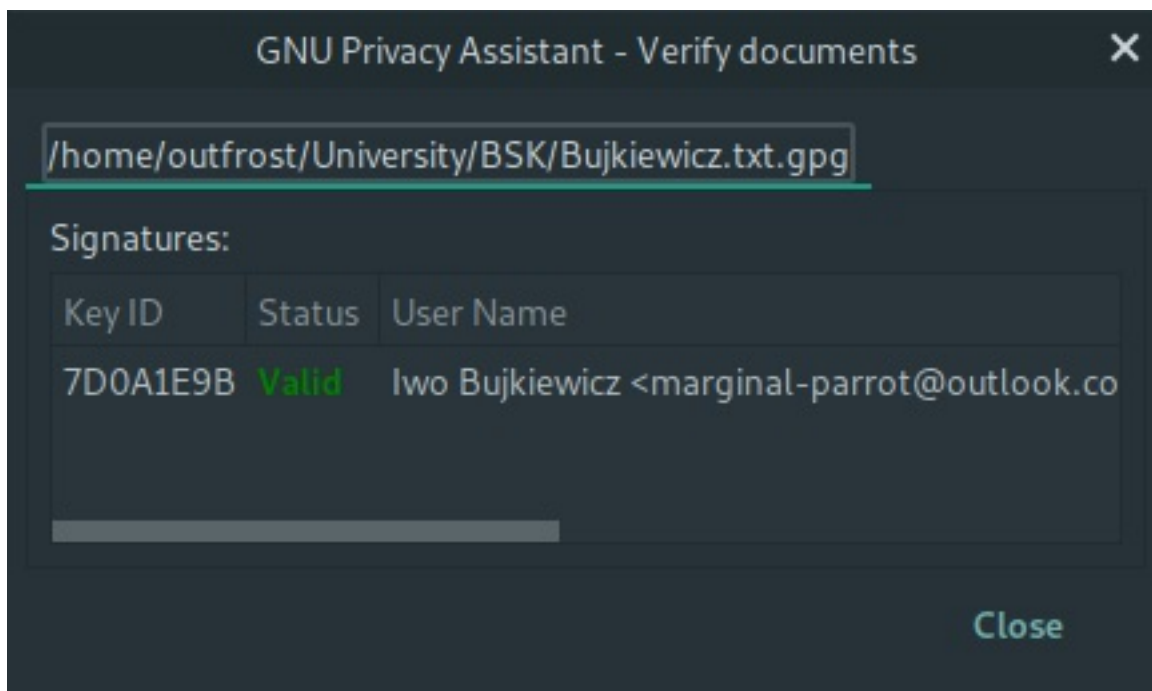
Zawartość pliku tekstowego



```
outfrost@Vaelastrasz:~/University/BSK
File Edit View Search Terminal Help
[outfrost@Vaelastrasz BSK]$ cat Bujkiewicz.txt
Iwo Bujkiewicz
Tue 20 Nov 01:24:33 CET 2018
[outfrost@Vaelastrasz BSK]$ ll Bujkiewicz.txt.gpg
-rw-r--r-- 1 outfrost outfrost 654 Nov 20 01:34 Bujkiewicz.txt.gpg
[outfrost@Vaelastrasz BSK]$ head Bujkiewicz.txt.gpg
0000}
00@bBujkiewicz.txt[0V0Iwo Bujkiewicz
Tue 20 Nov 01:24:33 CET 2018
0!t0mo007/500000}
0[0V0
0000}
00000aWn000Gkp4j000z0f0l0Y0G0000^v00s000I000J
00BL0X0~0V0X0RJzu0[000000 000d06q]
0Pb_n0&Eq0R0 000h0AbK0HX0Ykm00YF00a`k00 0
00!@0$R
"F0u0Fsp00W00/뵓 x00f0000]ds00Q00040 000000/x000C[*煙 00S0%p0$0V000{000000[JBP0_U0
06000vg000'P000y00
gR0_]0u"00`
0w=000E004!ly0\0"'0H0J~06nyN0106B0e000MX00000L00;00
00y00W0m003000000}000100000E000-0@E0000W00$M
00-q000@90*000000u 55+_02W0"an000K0G;0000630B/v0^00000I00â{00#{&z[outfrost
@Vaelastrasz BSK]$
```

Plik tekstowy podpisany przy użyciu CLI GPG





Weryfikacja podpisu przy użyciu GPA

#### Zadanie 7.

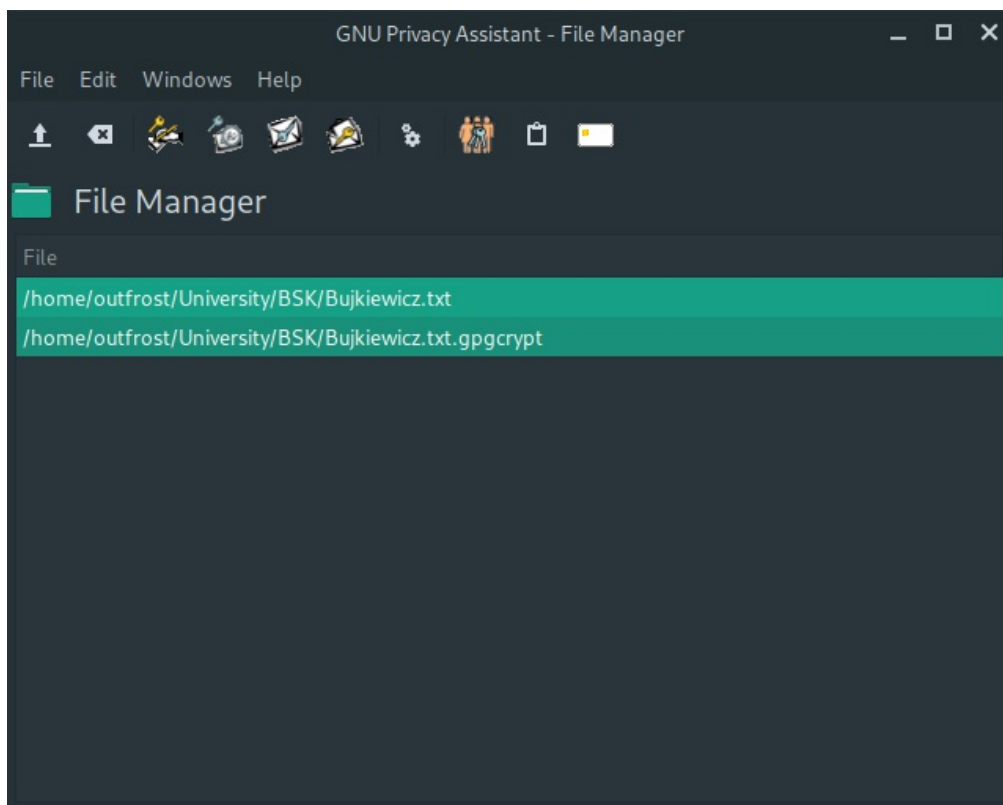
```
outfrost@Vaelastrasz: ~/University/BSK
File Edit View Search Terminal Help
[outfrost@Vaelastrasz BSK]$ gpg --armor --detach-sign BSK_2018_Lab2_krypt.pdf
File 'BSK_2018_Lab2_krypt.pdf.asc' exists. Overwrite? (y/N) y
[outfrost@Vaelastrasz BSK]$ cat BSK_2018_Lab2_krypt.pdf.asc
-----BEGIN PGP SIGNATURE-----

iHUEABYIAB0WIQTqklvdZPhXZ16K3z6Q33U/t4XxYAUCW/cJ+QAKCRCQ33U/t4Xx
YBZKAP9V9lZv3gLcI8e7yzZ25aZfZPUl0mWCXpE5z0hzkDLJggEAqTh6SRmmpCyR
HURT60G4o7XaU3/gBuK3VvW9EGZstw4=
=zJhU
-----END PGP SIGNATURE-----
[outfrost@Vaelastrasz BSK]$
```

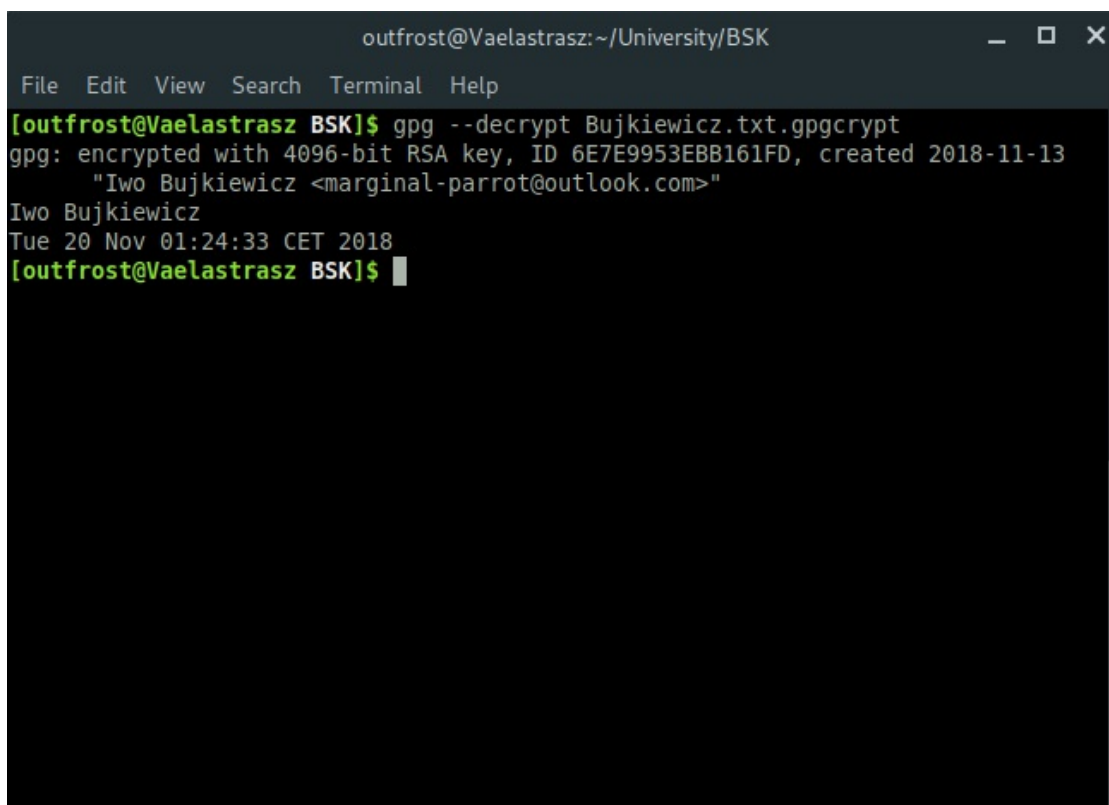
Podpis pliku PDF z instrukcją

#### Zadanie 8.

W GPA operacje na plikach wykonywane są za pośrednictwem okna **File Manager**, które umożliwia ich szyfrowanie, deszyfrowanie, podpisywanie oraz weryfikację podpisów. Zasyfrowany został ten sam plik, który był użyty do zadania 6. Jako odbiorca zasyfrowanego kluczem publicznym pliku wybrany został posiadacz głównego klucza prywatnego używanego w ćwiczeniu.



*Plik zaszyfrowany przy użyciu GPA*



*Plik odszyfrowany przy użyciu CLI GPG*

## Zadanie 9.

Zaszyfrowany został ten sam plik, który był użyty do zadania 6. Jako odbiorca zaszyfrowanego kluczem publicznym pliku wybrany został posiadacz głównego klucza prywatnego używanego w ćwiczeniu.

```
outfrost@Vaelastrasz: ~/University/BSK
File Edit View Search Terminal Help
74D86D6F9B9F372FD28C8D07F7BA1AE17D0A1E9B
uid      [ultimate] Iwo Bujkiewicz <marginal-parrot@outlook.com>
sub      rsa4096 2018-11-13 [E] [expires: 2020-11-12]

pub      rsa4096 2018-11-15 [SC] [expires: 2019-05-14]
9E95D01A39CAC79DFFD3A242C7C28431B68363FA
uid      [ultimate] Leland Wall <walll@outlook.com>
sub      rsa4096 2018-11-15 [E] [expires: 2019-05-14]

[outfrost@Vaelastrasz BSK]$ gpg --encrypt Bujkiewicz.txt
You did not specify a user ID. (you may use "-r")

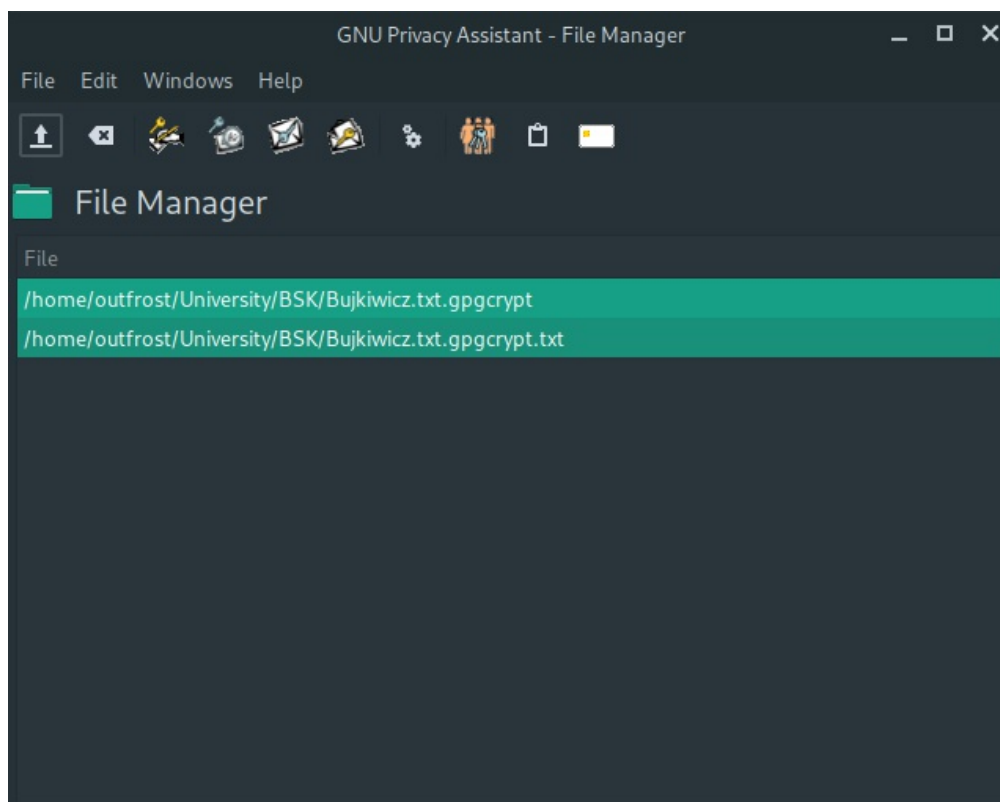
Current recipients:

Enter the user ID. End with an empty line: 7D0A1E9B

Current recipients:
rsa4096/6E7E9953EBB161FD 2018-11-13 "Iwo Bujkiewicz <marginal-parrot@outlook.com>"
>

Enter the user ID. End with an empty line:
File 'Bujkiewicz.txt.gpg' exists. Overwrite? (y/N) n
Enter new filename: Bujkiwicz.txt.gpgcrypt
[outfrost@Vaelastrasz BSK]$
```

*Plik zaszyfrowany przy użyciu CLI GPG*



*Plik odszyfrowany przy użyciu GPA*



## Zadanie 10.

```
outfrost@Vaelastrasz: ~/University/BSK
File Edit View Search Terminal Help
[outfrost@Vaelastrasz BSK]$ gpg --recv-keys 89DBEEDD6092A4F1576D83DDE02FABA5A9C05432
gpg: key E02FABA5A9C05432: 2 signatures not checked due to missing keys
gpg: key E02FABA5A9C05432: public key "Marcin Markowski <bsk2030@w4.pwr.pl>" imported
gpg: marginals needed: 3 completes needed: 1 trust model: pgp
gpg: depth: 0 valid: 4 signed: 0 trust: 0-, 0q, 0n, 0m, 0f, 4u
gpg: next trustdb check due at 2019-05-14
gpg: Total number processed: 1
gpg: imported: 1
[outfrost@Vaelastrasz BSK]$
```

*Klucz prowadzącego pobrany z serwera kluczy*

## Zadanie 11.

Właściwy plik z treścią zadania 11. został zidentyfikowany poprzez weryfikację podpisu każdego z plików. Zrzut ekranu demonstruje użyte komendy.

```
[outfrost@Vaelastrasz BSK]$ for i in {1..6}; do
> echo
> f="ZAD11_v$i.txt.asc"
> echo "$f"
> gpg --verify "$f"
> done

ZAD11_v1.txt.asc
gpg: Signature made Mon 08 Oct 2018 10:27:19 CEST
gpg: using RSA key 89DBEEDD6092A4F1576D83DDE02FABA5A9C05432
gpg: BAD signature from "Marcin Markowski <bsk2030@w4.pwr.pl>" [unknown]

ZAD11_v2.txt.asc
gpg: Signature made Mon 08 Oct 2018 10:27:19 CEST
gpg: using RSA key 89DBEEDD6092A4F1576D83DDE02FABA5A9C05432
gpg: BAD signature from "Marcin Markowski <bsk2030@w4.pwr.pl>" [unknown]

ZAD11_v3.txt.asc
gpg: Signature made Mon 08 Oct 2018 10:27:19 CEST
gpg: using RSA key 89DBEEDD6092A4F1576D83DDE02FABA5A9C05432
gpg: BAD signature from "Marcin Markowski <bsk2030@w4.pwr.pl>" [unknown]

ZAD11_v4.txt.asc
gpg: Signature made Mon 08 Oct 2018 10:27:19 CEST
gpg: using RSA key 89DBEEDD6092A4F1576D83DDE02FABA5A9C05432
gpg: BAD signature from "Marcin Markowski <bsk2030@w4.pwr.pl>" [unknown]

ZAD11_v5.txt.asc
gpg: Signature made Mon 08 Oct 2018 10:27:19 CEST
gpg: using RSA key 89DBEEDD6092A4F1576D83DDE02FABA5A9C05432
gpg: Good signature from "Marcin Markowski <bsk2030@w4.pwr.pl>" [unknown]
gpg: WARNING: This key is not certified with a trusted signature!
gpg: There is no indication that the signature belongs to the owner.
Primary key fingerprint: 89DB EEDD 6092 A4F1 576D 83DD E02F ABA5 A9C0 5432

ZAD11_v6.txt.asc
gpg: Signature made Mon 08 Oct 2018 10:27:19 CEST
gpg: using RSA key 89DBEEDD6092A4F1576D83DDE02FABA5A9C05432
gpg: BAD signature from "Marcin Markowski <bsk2030@w4.pwr.pl>" [unknown]
[outfrost@Vaelastrasz BSK]$
```

*Weryfikacja sygnatur plików z treścią zadania*

Następnie plik tekstowy został wyodrębniony z pliku z podpisem i odczytany w terminalu. Ze względu na kodowanie tekstu użyte podczas zapisu pliku ( `windows-1250` , zamiast `UTF-8` ) przy wyświetlaniu zawartości pliku terminal napotkał nieznane znaki.

```
outfrost@Vaelastrasz: ~/University/BSK
File Edit View Search Terminal Help
[outfrost@Vaelastrasz BSK]$ man gpg
[outfrost@Vaelastrasz BSK]$ gpg ZAD11_v5.txt.asc
gpg: WARNING: no command supplied. Trying to guess what you mean ...
gpg: Signature made Mon 08 Oct 2018 10:27:19 CEST
gpg: using RSA key 89DBEEDD6092A4F1576D83DDE02FABA5A9C05432
gpg: Good signature from "Marcin Markowski <bsk2030@w4.pwr.pl>" [unknown]
gpg: WARNING: This key is not certified with a trusted signature!
gpg: There is no indication that the signature belongs to the owner.
Primary key fingerprint: 89DB EEDD 6092 A4F1 576D 83DD E02F ABA5 A9C0 5432
[outfrost@Vaelastrasz BSK]$ nano ZAD11_v5.txt
ZAD11_v5.txt ZAD11_v5.txt.asc
[outfrost@Vaelastrasz BSK]$ cat ZAD11_v5.txt

*** ZADANIE 11 ***

Zapisać do pliku tekstowego imiona członków grupy.
Plik zaszyfrować za pomocą gpg algorytmem AES192 (tylko symetrycznym) z kluczem 'LABORKA'.
Obliczyć sumę kontrolną SHA-1 pliku (Kleopatra).
Komendy gpg, treść pliku przed i po zaszyfrowaniu oraz sumę kontrolną umieścić w sprawozdaniu.

*****
[outfrost@Vaelastrasz BSK]$
```

Zweryfikowana treść zadania 11.

Zadanie polegało na stworzeniu pliku tekstowego, zaszyfrowaniu go algorytmem AES-192 z kluczem **LABORKA**, i sprawdzeniu jego zawartości po zaszyfrowaniu oraz sumy kontrolnej SHA1.

Iwo

Treść pliku **Bujkiewicz-11.txt** przed zaszyfrowaniem

```
$ gpg -c --cipher-algo aes192 Bujkiewicz-11.txt
```

Komenda szyfrująca

```
gpg
c000EN000J0'0%60S}! 00Jr000s0<o00000#0N0F^70
m00Eu-0,d070000'80W00v0?
```

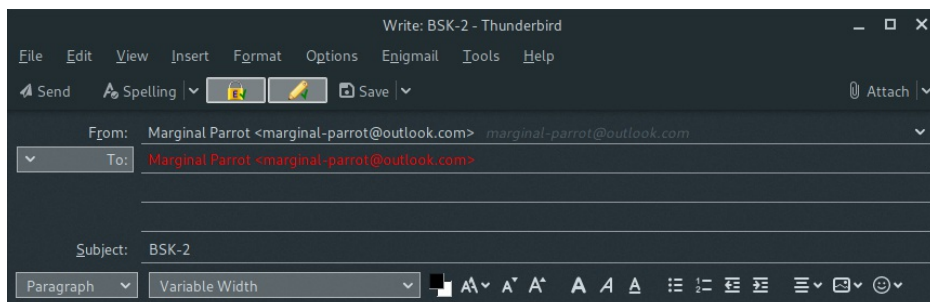
Treść pliku po zaszyfrowaniu (tekstowa reprezentacja pliku binarnego)

```
c46ce2f4d4a1101c19acf0097b872a0a2f65d926
```

Suma kontrolna SHA1

## Zadanie 12.

Z uwagi na wykonywanie ćwiczenia w grupie ówczas jednoosobowej, wiadomość email została zaszyfrowana, podpisana i wysłana na własny adres email.



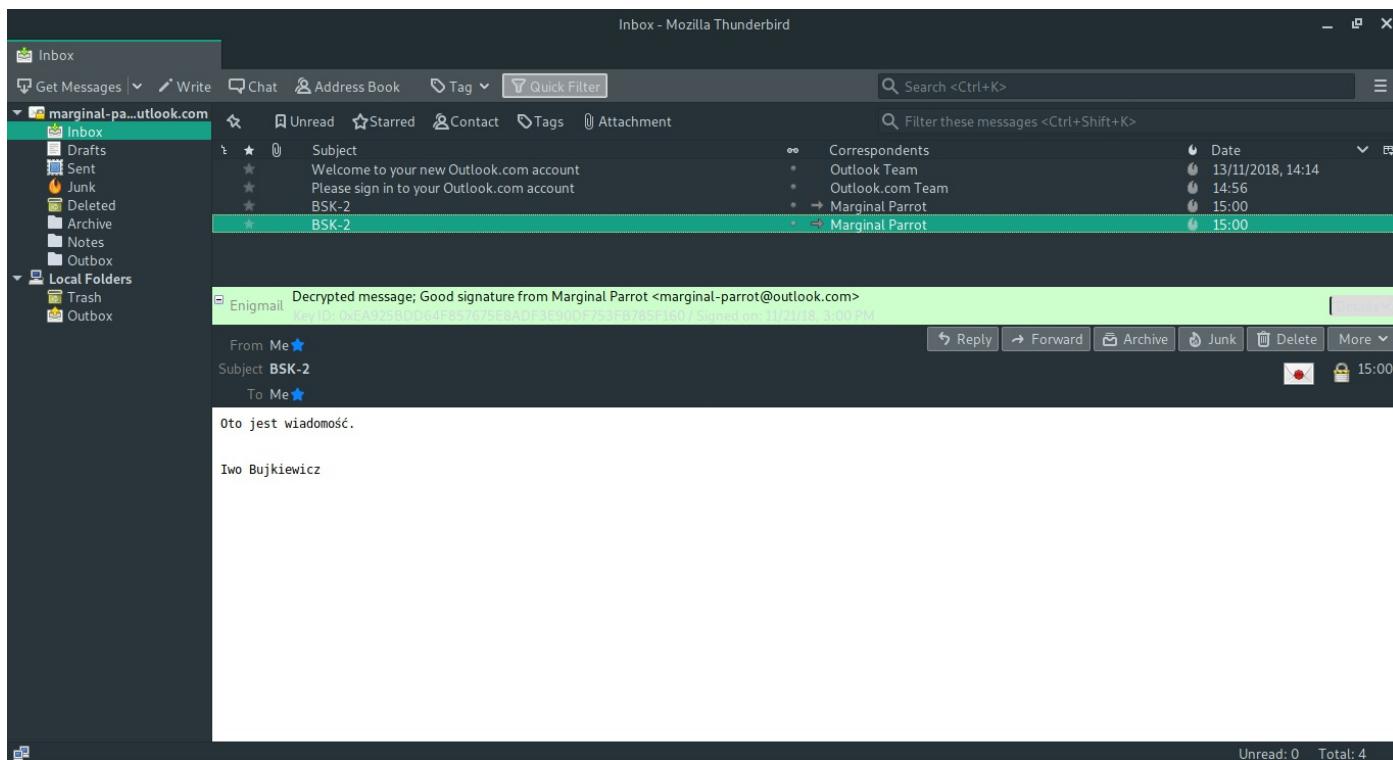
Oto jest wiadomość.

Iwo Bujkiewicz

English (United States)

*Wysyłana wiadomość email*

Enigmail automatycznie weryfikuje podpis i odszyfrowuje wiadomość przy odebraniu jej w poprawnej postaci.



*Odebrana wiadomość email, odszyfrowana po pomyślnej weryfikacji podpisu*



### Zadanie 13.

Do porównania wybrano algorytmy: domyślny (AES-128), AES-256 oraz Twofish. Szyfrowany był plik o identycznej zawartości, jak w zadaniu 6. Wielkość zaszyfrowanych plików okazała się być identyczna dla tych trzech algorytmów, różniły się one natomiast oczywiście zawartością.

```
[outfrost@Vaelastrasz BSK]$ gpg -c Bujkiewicz-13.txt
```

*Szyfrowanie symetryczne przy użyciu domyślnego algorytmu - AES-128*

```
outfrost@Vaelastrasz: ~/University/BSK
File Edit View Search Terminal Help
Copyright (C) 2018 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <https://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

Home: /home/outfrost/.gnupg
Supported algorithms:
Pubkey: RSA, ELG, DSA, ECDH, ECDSA, EDDSA
Cipher: IDEA, 3DES, CAST5, BLOWFISH, AES, AES192, AES256, TWOFISH,
        CAMELLIA128, CAMELLIA192, CAMELLIA256
Hash: SHA1, RIPEMD160, SHA256, SHA384, SHA512, SHA224
Compression: Uncompressed, ZIP, ZLIB, BZIP2
[outfrost@Vaelastrasz BSK]$ gpg -c --cipher-algo aes256 Bujkiewicz-13.txt
File 'Bujkiewicz-13.txt.gpg' exists. Overwrite? (y/N) n
Enter new filename: Bujkiewicz-13.txt.gpg-aes256
[outfrost@Vaelastrasz BSK]$ gpg -c --cipher-algo twofish Bujkiewicz-13.txt
File 'Bujkiewicz-13.txt.gpg' exists. Overwrite? (y/N) n
Enter new filename: Bujkiewicz-13.txt.gpg-twofish
[outfrost@Vaelastrasz BSK]$ ll Bujkiewicz-13*
-rw-r--r-- 1 outfrost outfrost 44 Nov 21 15:06 Bujkiewicz-13.txt
-rw-r--r-- 1 outfrost outfrost 122 Nov 21 15:07 Bujkiewicz-13.txt.gpg
-rw-r--r-- 1 outfrost outfrost 122 Nov 21 15:12 Bujkiewicz-13.txt.gpg-aes256
-rw-r--r-- 1 outfrost outfrost 122 Nov 21 15:13 Bujkiewicz-13.txt.gpg-twofish
[outfrost@Vaelastrasz BSK]$
```

*Szyfrowanie symetryczne przy użyciu algorytmów AES-256 oraz Twofish; porównanie rozmiarów plików*

```
outfrost@Vaelastrasz: ~/University/BSK
File Edit View Search Terminal Help
[outfrost@Vaelastrasz BSK]$ for f in Bujkiewicz-13*; do
> echo
> echo "$f:"
> cat "$f"
> done

Bujkiewicz-13.txt:
Iwo Bujkiewicz
Tue 20 Nov 01:24:33 CET 2018

Bujkiewicz-13.txt.gpg:
 00]50DE00`0v00w(0C0060007{3000m0H0|80<{0+t<0x0Je
00>000~T000b0j0
00<0
 0w00+0{/N0s0
Bujkiewicz-13.txt.gpg-aes256:
0 V0Bg0tq6000icb0000L00/S:U00\00x0[000IsV0p0 0y00eg0500qW0;@. V0g000f0
0)^0'00v8
050M00n00000j3`Rz
Bujkiewicz-13.txt.gpg-twofish:
0
0I00K00I00iQt00E0!F0W0cS00JL000jL00*h900dpj00sCQm0qd400000x0G0v00
00}0`+/0E0)0e[outfrost@Vaelastrasz BSK]$
```

*Porównanie zawartości zaszyfrowanych plików*