Zabezpieczanie komunikacji pocztowej, integracja mechanizmów kryptograficznych (OpenPGP, S/MIME oraz certyfikacji) z usługami pocztowymi, filtracja treści

1. Wprowadzenie

Coraz częściej w Internecie zdarzają się próby podszywania pod inne osoby (konta pocztowe innych osób), rozsyłane są wiadomości reklamowe oraz tak zwany SPAM. Istnieje wiele sposobów przeciwdziałania tego typu poczcie i co pewien czas jakaś firma postanawia zaproponować nowe rozwiązanie. Niestety jak dotychczas żadne rozwiązanie nie wyeliminowało całkowicie problemy niechcianej poczty (SPAMu). Natomiast jest na to bardzo proste rozwiązanie, które jest już zauważane przez ludzi, jednak oczywiście jest to w pewnym stopniu problematyczne. Mianowicie rozwiązanie to polega na zastosowaniu podpisu elektronicznego, który byłby stosowany do podpisu każdej wiadomości. Jednak jak wszystkim wiadomo podpis elektroniczny jest dosyć drogi, dlatego można wykorzystać inne metody podpisywania poczty nie korzystający z typowego podpisu elektronicznego, przykładami darmowych rozwiązań tego typu jest pakiet PGP (Pretty Good Privacy), jeszcze innym rozwiązaniem jest stosowanie certyfikatów poświadczających wiarygodność adresu poczty elektronicznej.

Wadą przedstawionego powyżej rozwiązania jest to, że aby w pełni wykorzystać z takiego sposobu obrony przed niechcianą pocztą jest posiadanie kluczy publicznych wszystkich osób, od których możemy otrzymać pocztę, a to może być trudne do wykonania. Bardzo często użytkownicy nie chcą przekonać się do stworzenia swoich kluczy i podpisywania wszystkich wiadomości.

Celem ćwiczenia jest zapoznanie się z mechanizmem PGP oraz z certyfikatami dla adresów e-mail.

2. Mechanizm OpenPGP

Standardy PGP (RFC 1991) i OpenPGP (RFC 2440) umożliwiają podpisywanie oraz szyfrowanie plików (w tym korespondencji pocztowej) metodami asymetrycznymi i symetrycznymi. W implementacji GnuPG stosowane są szyfry RSA, DSA lub ElGamal dla szyfrowania asymetrycznego oraz 3DES, CAST5, Blowfish, Twofish, i AES (128b, 192b i 256b) dla szyfrowania symetrycznego. Podpis elektroniczny obsługują algorytmy MD5, SHA-1 i SHA-256 oraz RIPEMD-160. Ponadto w implementacji tej możliwe jest dokonywanie kompresji szyfrowanej treści algorytmami ZIP lub ZLIB.

Zarządzanie kluczami kryptograficznymi

Wszystkie operacje w systemie GnuPG wykonuje uniwersalne narzędzie gpg. Program ten umożliwia użytkownikowi m.in. wygenerowanie pary kluczy asymetrycznych o wybranej długości oraz utrzymuje zbiory („pęki") kluczy publicznych innych użytkowników.

Wygenerowanie klucza

```
local> gpg --gen-key
```

Pęki kluczy przechowywane są domyślnie w katalogu ~/.gnupg w plikach: pubring.gpg oraz secring.gpg. Bieżącą zawartość pęku posiadanych kluczy publicznych można wyświetlić

poleceniem :

```
local> gpg --list-keys
```

lub dla kluczy prywatnych:

```
local> gpg --list-secret-keys
```

Przykładowy wynik działania pierwszego z tych dwóch poleceń może wyglądać następująco:

```
/home/jbond/.gnupg/pubring.gpg
------------------------------
pub  1024D/3BF84E43 2004-06-25 James Bond
<spsk007@cs.put.poznan.pl>
sub  1024g/8B423A22 2004-06-25
```

Wygenerowany klucz publiczny można udostępnić, np. eksportując do pliku tekstowego w formacie ASCII:

```
local> gpg --export -a -o ~/.gpgkey
```

Pozyskanie czyjegoś klucza publicznego

Pozyskanie klucza prywatnego może nastąpić:

- z pliku przesłanego lub pobranego (np. poprzez WWW) od właściciela klucza:

```
local> gpg --import plik_z_kluczem
```

- lub z serwera kluczy:

```
local> gpg --keyserver certserver.pgp.com --recv-key
0xBB7576AC
```

Podpisywanie i szyfrowanie wiadomości

Program gpg pozwala podpisać dowolny plik, w szczególności list, swoim kluczem prywatnym:

```
local> gpg --sign plik
```

W powyższym przykładzie wynik pojawi się w postaci skompresowanej w pliku o nazwie plik.pgp. Aby uzyskać podpis w postaci czytelnej (bez kompresji) należy wykonać polecenie:

```
local> gpg --clearsign plik
```

Wówczas powstanie plik tekstowy w formacie ACII o nazwie plik.asc. Nazwę pliku wynikowego możemy zmieniać opcją `-o nazwa_pliku`.

Program gpg pozwala też zaszyfrować wiadomość kluczem publicznym konkretnego odbiorcy:

```
local> gpg --recipient odbiorca -at -o list.txt plik
```

a także połączyć szyfrowanie z podpisem elektronicznym:

```
local> gpg -se -r odbiorca -at -o list.txt plik
```

Deszyfracja i weryfikacja wiadomości

Do deszyfrowania i weryfikowania podpisu służą opcje, odpowiednio, `-d` (`--decrypt`) oraz `--verify`:

```
local> gpg -d list

local> gpg --verify list
```

Szyfrowanie plików

Do symetrycznego szyfrowania plików służy opcja `-c` programu gpg:

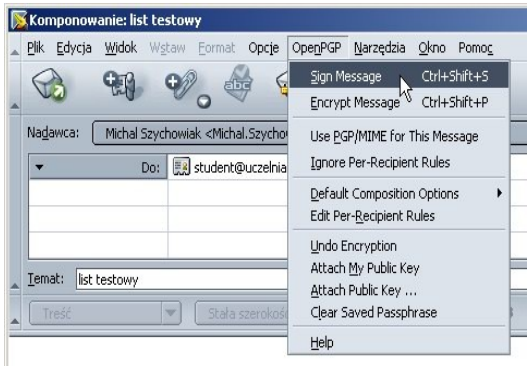
```
local> gpg -c -o szyfrogram plik.txt
```

lub:

```
local> gpg -symmetric --cipher-algo AES256 -o szyfrogram
plik.txt
```

3. Integracja mechanizmów kryptograficznych z popularnymi klientami poczty elektronicznej

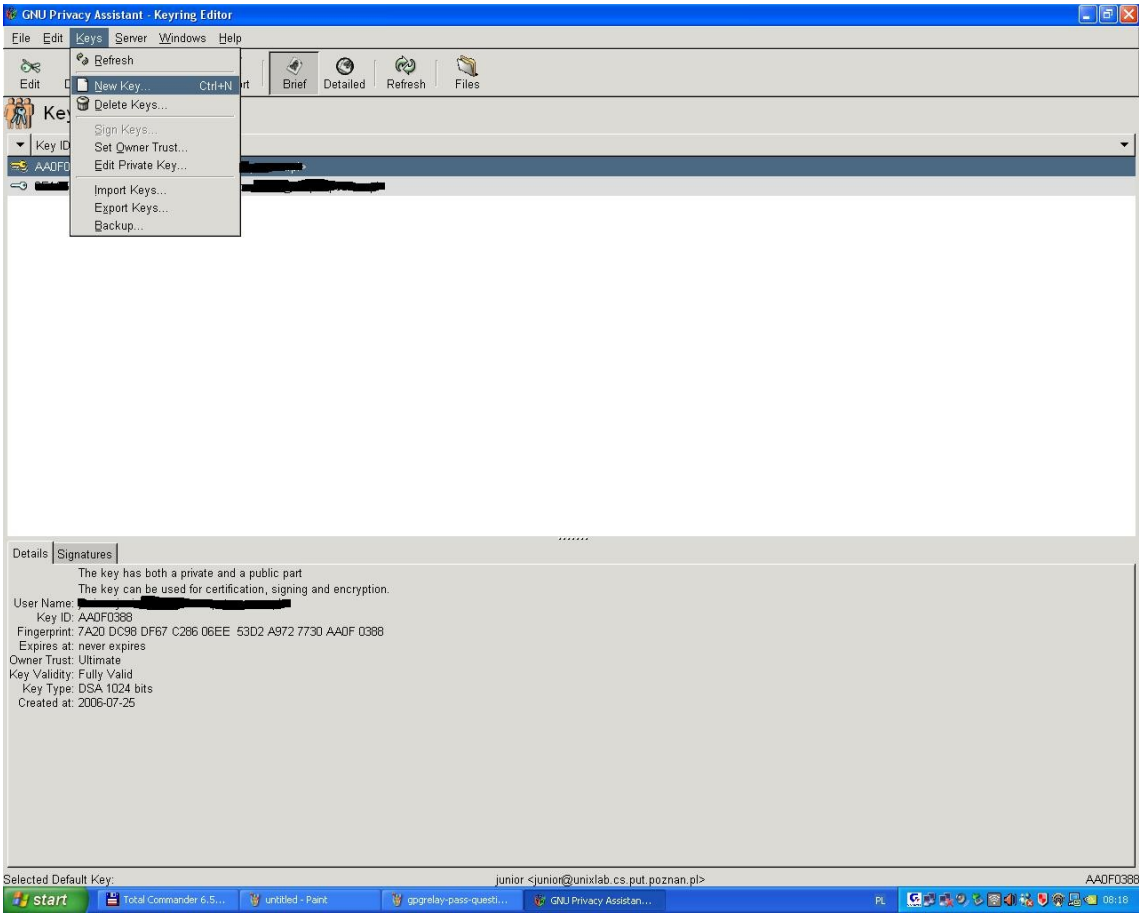
Niektóre programy klientów posiadają możliwość wykorzystania mechanizmów kryptograficznych: szyfrowania i / lub podpisywania korespondencji pocztowej. Taką możliwość posiadają np. popularne produkty z rodziny Mozilla (Mozilla Mail lub Thunderbird) z rozszerzeniem enigmail (na platformę Windows to rozszerzenie trzeba zainstalować oddzielnie). Enigmail integruje klienta pocztowego z popularnym i powszechnie stosowanym w Internecie systemem PGP (np. pakiet OpenPGP lub GnuPG, niezależnie instalowane w systemie). System PGP umożliwia m.in. certyfikację kluczy pocztowych metodą wzajemnego zaufania (*Web of Trust*).



Niestety nie wszystkie programy klienckie poczty elektronicznej posiadają rozszerzenia umożliwiające bezpośrednie korzystanie z podpisywania wiadomości przy użyciu mechanizmu PGP. Dosyć popularny klient poczty (MS Outlook Express) niestety nie posiada darmowych dobrze pracujących rozszerzeń zapewniających integrację z PGP. Zamiast tego wszystkie wiadomości podpisane z wykorzystaniem tego mechanizmu są pokazywane jako lista załączników do poczty. Aby jednak móc wykorzystywać mechanizm PGP można wykorzystać mechanizm pośrednika, który będzie pośrednikiem przy wysyłaniu i odbieraniu poczty elektronicznej (w ten sposób całkowicie uniezależniamy się od konkretnego klienta poczty elektronicznej, gdyż każdy klient może korzystać z takiego pośrednika).

Przykładowym programem zapewniającym stworzenie mechanizmu pośrednika jest program GnuPGRelay [GR] instalacje jego wymaga wcześniej zainstalowania aplikacji GnuPG [GPG] dostępnej w internecie, często również w innych pakietach programów, polecić można pakiet WinPT [WPT] zawierającą oprócz GnuPG szereg innych aplikacji umożliwiających szyfrowanie plików, katalogów itp.

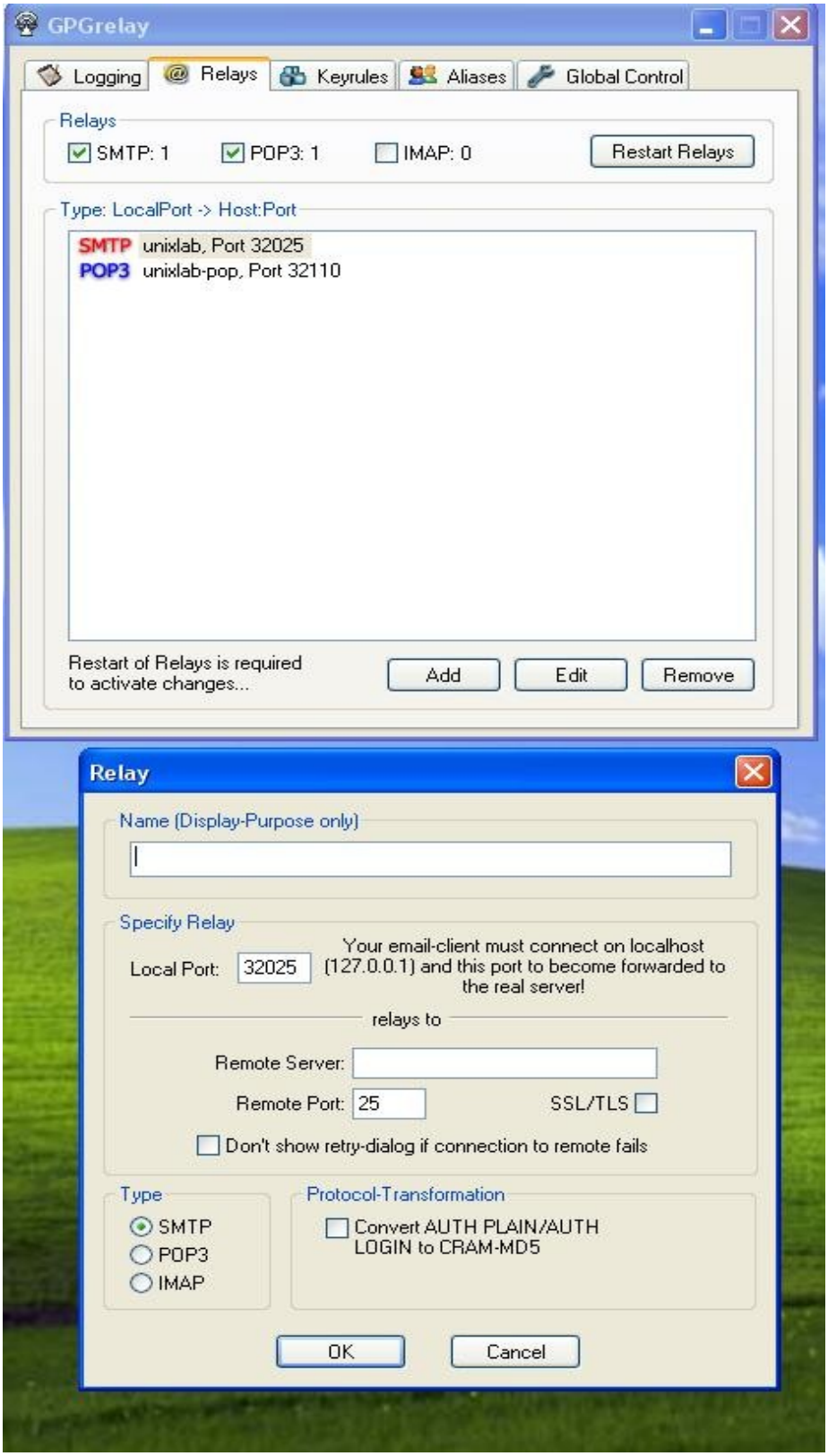
Po zainstalowaniu pakietu WinPT należy utworzyć własną parę kluczy PGP



W aplikacji tej również można dodawać klucze publiczne zaufanych osób, od których będziemy otrzymywać podpisaną pocztę celem weryfikacji podpisów.

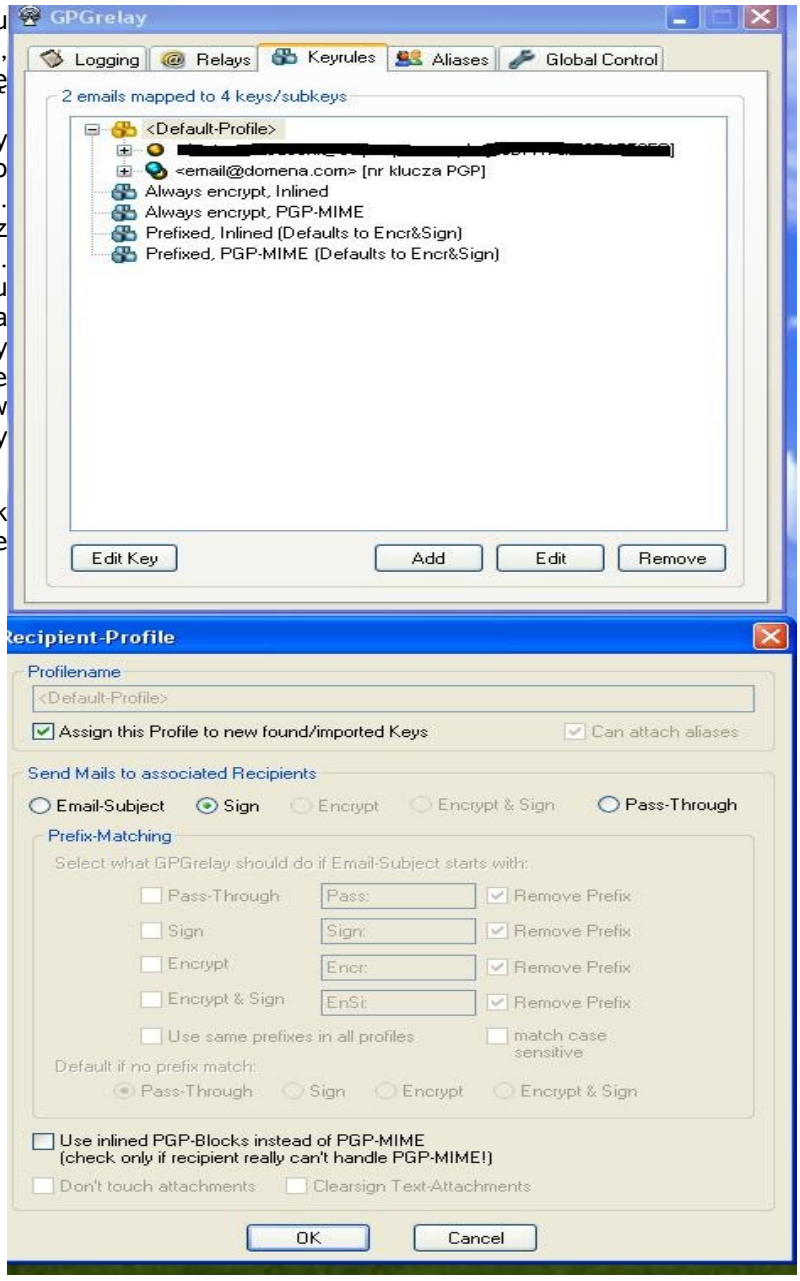
Następnie instalujemy aplikacje GnuPGRelay, którą musimy skonfigurować do wysyłania i odbierania poczty. Dokładniej należy ustawić serwery poczty przychodzącej (POP3) i wychodzącej (SMTP), aby móc korzystać z pośrednika, dodatkowo wymaga to przekonfigurowania aplikacji klienta poczty, aby kontaktował się z pośrednikiem lokalnym jako serwerem poczty przychodzącej i wychodzącej.

Poniższy rysunek przedstawia konfigurację serwerów poczty w pośredniku:


GPGrelay
Logging
Relays
Keyrules
Aliases
Global Control
Relays
SMTP: 1
POP3: 1
IMAP: 0
Restart Relays
Type: LocalPort -> Host:Port
SMTP unixlab, Port 32025
POP3 unixlab-pop, Port 32110
Restart of Relays is required to activate changes...
Add
Edit
Remove
Relay
Name (Display-Purpose only)
Specify Relay
Local Port:
32025
Your email-client must connect on localhost (127.0.0.1) and this port to become forwarded to the real server!
relays to
Remote Server:
Remote Port:
25
SSL/TLS
Don't show retry-dialog if connection to remote fails
Type
SMTP
POP3
IMAP
Protocol-Transformation
Convert AUTH PLAIN/AUTH LOGIN to CRAM-MD5
OK
Cancel

Po skonfigurowaniu pośrednika i klienta poczty, należy ustawić preferencję wykorzystywania mechanizmu PGP – należy ustawić profile i przypisać do nich odpowiednie klucze. Klucze pobierane są z głównej aplikacji GnuPG. Wybór i edycja profilu pozwala określić czy poczta ma być podpisywana, czy szyfrowana, czy obie akcje równocześnie lub w ostateczności nie musimy nic z pocztą robić.

Poniższy rysunek przedstawia konfiguracje profili:



Klucze PGP są zaszyfrowane dlatego pośrednik musi znać hasło do klucza prywatnego właściciela klucza, lub każdorazowo może się pytać o to hasło. Ekran pytania się o hasło poniżej:

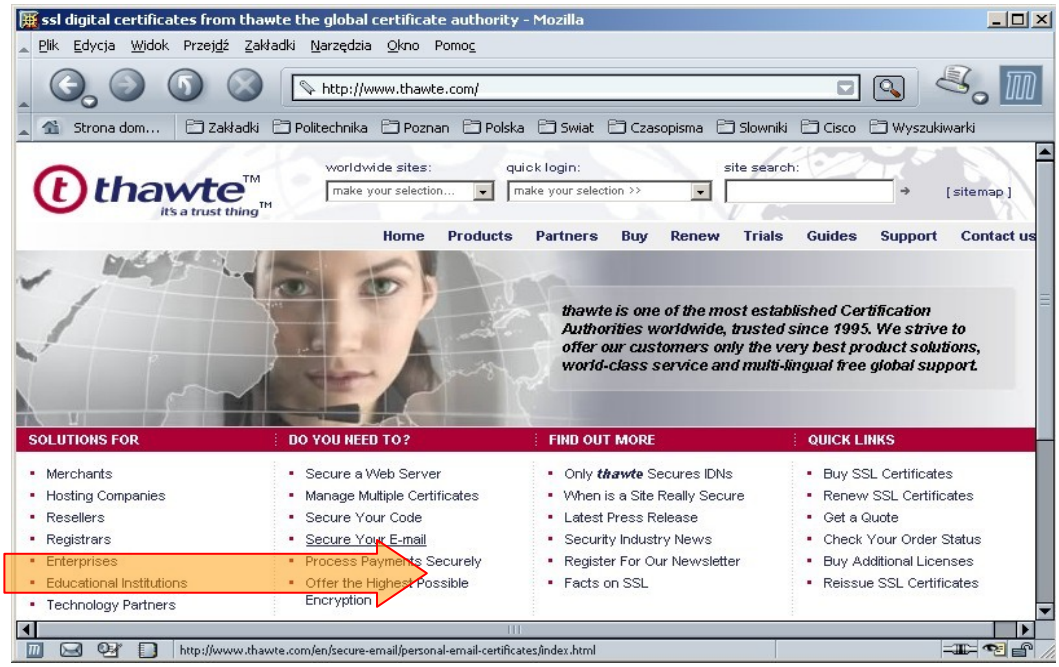
Możliwości wykorzystywania pośrednika nie ograniczają się do konkretnego klienta poczty elektronicznej. Oczywiście pośrednik taki ma również ograniczenia, gdyż na etapie

tworzenia wiadomości nie możemy określić czy wiadomość ma być podpisywania i/lub szyfrowana, tylko musimy określić to poprzez profile w pośredniku.

Certyfikaty adresów pocztowych

Jednym z bardziej popularnych ośrodków certyfikacji w Internecie jest firma Thawte. Wiele przeglądarek WWW i aplikacji pocztowych zawiera certyfikat głównego urzędu certyfikacji tej firmy, co pozwala ufać podpisom tego urzędu:

Korzystając ze strony http://www.thawte.com można pozyskać darmowy certyfikat poświadczający własny adres pocztowy:



4. Zadania

1. Za pomocą narzędzi pakietu GnuPG wygeneruj pęki kluczy kryptograficznych szyfrowania asymetrycznego dla swojego konta pocztowego, a następnie udostępnij klucz publiczny poprzez usługę finger.
2. Wykorzystaj pakiet GnuPG do ochrony korespondencji z wybranym użytkownikiem.
3. Dokonaj zaszyfrowania wybranego pliku metodą szyfrowania symetrycznego narzędziami pakietu GnuPG.
4. Zainstaluj system GnuPG (jeśli korzystasz z systemu Windows) oraz rozszerzenie Enigmail klienta pocztowego Mozilla Thunderbird.
5. Korzystając w programie Thunderbird z funkcji OpenPGP→Key Management wygeneruj swoją parę kluczy kryptograficznych.
6. Wyślij podpisany list do siebie samego. Sprawdź reakcję systemu.
7. Wyślij podpisany list do innego użytkownika ze swojej grupy i odbierz jego list.
8. Zweryfikuj poprawność podpisu otrzymanego listu.
9. Zrealizuj komunikację z szyfrowaniem całej przesyłki pocztowej
10. Wykonać instalację pośrednika GnuPGRelay wraz z WinPT oraz przekonfigurowanie dowolnego klienta poczty elektronicznej do korzystania z pośrednika i przetestowanie jego możliwości.
11. Wykorzystując stronę firmy thawte.com utworzyć darmowy certyfikat do poczty elektronicznej i zapisać go w klienci poczty elektronicznej (Thunderbird), a następnie wysłać list podpisany przy użyciu certyfikatu.
12. Zweryfikować otrzymaną pocztę podpisaną przy użyciu certyfikatu, sprawdzić poprawność podpisu.

5. Problemy do dyskusji

- Dlaczego podpisy PGP nie są zbyt często wykorzystywane przez zwykłego użytkownika?
- Dlaczego klient poczty elektronicznej wymaga podania certyfikatu lub klucza publicznego nadawcy poczty?
- Dlaczego popularny klient poczty Ms Outlook bardzo dziwnie odbiera pocztę podpisaną z wykorzystaniem mechanizmu PGP, pod warunkiem, że nie posiada mechanizmu do obsługi tego mechanizmu?
- Który z wymienionych wyżej systemów jest lepszy i dlaczego (PGP czy certyfikaty)?
- Czy istnieje prosta metoda wykorzystania mechanizmów kryptograficznych do zabezpieczenia poczty elektronicznej?
- Przykłady innych mechanizmów zapewniających bezpieczeństwo przesyłanych danych pocztowych (o ile takowe istnieją)

6. Bibliografia

[PGP] obszerny poradnik PGP: http://www.rossde.com/PGP/

[GPG] strona domowa i dokumentacja projektu GnuPG: http://www.gnupg.org

[WPT] WinPT - http://www.stud.uni-hannover.de/~twoaday/

[GOE] GnuPG for Outlook Express http://winpt.cityofcambridge.net/gpgoe.html

[G4W] GnuPG for Windows http://www.gpg4win.org/

[GR] GnuPG Relay http://sites.inka.de/tesla/gpgrelay.html