# Kryptoanaliza stosowana 2023

Lista zadań nr 11

Na zajęcia 19 grudnia 2023

## Zarządzanie kluczami i szyfrowanie plików za pomocą GnuPG

**Zadanie 1 (1 pkt).** Przygotuj krótkie omówienie sposobu generowania kluczy w OpenPGP. Jakiego rodzaju algorytmy (poza RSA) są dostępne w Twojej instalacji GPG? Zajrzyj do pracy: Birger Schacht, Peter Kieseberg, An Analysis of 5 Million OpenPGP Keys, *J. Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications* **11**(3):107–140, Sept. 2020 i przygotuj jej krótkie omówienie.

**Zadanie 2 (1 pkt).** Wygeneruj klucz OpenPGP RSA-4096 tylko do certyfikacji z atrybutami:

| | |
|---|---|
| Real name: | *Twoje imię i nazwisko w takim brzmieniu, w jakim występuje w serwisie SKOS.* |
| Email address: | *Twój-numer-albumu*`@uwr.edu.pl` (tj. adres Twojej studenckiej skrzynki pocztowej). |
| Validity period: | 5 lat |

Dodaj do niego podklucze RSA-4096 ważne jeden rok:

- do szyfrowania,
- do podpisywania.

Uwaga: to zadanie najwygodniej rozwiązać używając polecenia `gpg -full-gen-key` (wybieramy opcję `RSA (sign only)`), a następnie `gpg -edit-key` (polecenia `change-usage` i `add-key`).

**Zadanie 3 (1 pkt).** Wymień się kluczami publicznymi z kolegami. Jeśli używasz niezaufanego kanału, zaproponuj jakiś sposób weryfikacji *fingerprintów*. Podpiszcie sobie wzajemnie swoje klucze publiczne.

**Zadanie 4 (1 pkt).** Wygeneruj plik tekstowy zawierający publiczne części swoich kluczy (najlepiej wraz z podpisami kolegów) i zgłoś go w zadaniu w serwisie SKOS na stronie zajęć jako pojedynczy plik o nazwie *`Twój-numer-albumu`*`.asc`.

Zwróć uwagę, że system SKOS zweryfikuje tożsamość osoby przesyłającej klucz, więc prowadzący będzie mógł im zaufać w takim samym stopniu, w jakim ufa systemom SKOS i UWr CAS.

**Zadanie 5 (1 pkt).** Dla bezpieczeństwa zrób *backup* swoich kluczy prywatnych, np. na osobnym pendrivie. Usuń następnie główny klucz prywatny z komputera (jest on potrzebny jedynie do podpisywania kluczy publicznych — własnych, na nowo wygenerowanych oraz kluczy znajomych). Do wykonania *backupu* użyj też programu `paperkey`.

**Zadanie 6 (1 pkt).** Zadanie wykonaj wspólnie z inną osobą, z którą wymieniłeś się kluczami. Za pomocą polecenia `gpg` zaszyfrujcie kluczem publicznym kolegi i podpiszcie własnym kluczem prywatnym jakieś pliki tekstowe. Przekażcie je sobie (dowolnym kanałem, np. wysyłając jako załącznik poczty). Sprawdźcie, że potraficie je odszyfrować i sprawdzić podpisy.

**Zadanie 7 (1 pkt).** Utwórz jakiś plik w formacie PDF, a następnie zaszyfruj go na trzy sposoby:

1. poleceniem `gpg` używając klucza publicznego kolegi;
2. poleceniem `gpg` za pomocą szyfrowania symetrycznego (wybierz AES128) używając hasła, które ustaliłeś wspólnie z kolegą;
3. poleceniem `pdftk` używając hasła, które ustaliłeś wspólnie z kolegą (wybierz AES128). Zorientuj się, jakie przeglądarki PDF są najczęściej używane (nie tylko w Linuksie/BSD, ale także w środowiskach macOS i MS Windows). Które *nie* wspierają szyfrowania na hasło? Co jeszcze, poza szyfrowaniem, potrafi program `pdftk`?

Porównaj wygodę i bezpieczeństwo przesyłania pliku PDF zaszyfrowanego każdą z tych trzech metod.

## Szyfrowanie poczty za pomocą GnuPG

Zadanie 7 pokazuje, że w przypadku szyfrowania plików wartość dodana kryptografii asymetrycznej i OpenPGP jest niewielka. Protokół OpenPGP jest jednak *bardzo* użyteczny, jeśli używamy go do automatycznego szyfrowania poczty elektronicznej. W tym celu należy korzystać z protokołów ESMTP i IMAP (lub POP3), a zatem zamiast przeglądarki WWW i protokołu HTTPS używać osobnego programu MUA, wspierającego te protokoły oraz protokół OpenPGP. Wiele agentów MUA posiada *pluginy* do programu GPG (Mutt, KMail). Inne (np. Mozilla Thunderbird) korzystają z biblioteki RNP (fork biblioteki NetPGP napisanej w 2016 roku przez developera NetBSD). W ostatnim roku zarówno GMail, jak i Office365 zablokowały możliwość stosowania protokołów `PLAIN`, `LOGIN` lub `Cleartext` podczas uwierzytelniania w protokołach IMAP i POP3, a jedynym dostępnym protokołem uwierzytelniania pozostał OAUTH2. Agent MUA musi więc także wspierać ten protokół. Spośród trzech wymienionych wyżej programów KMail wspiera OAUTH2 tylko dla IMAP i tylko w serwisie GMail. Zasadniczo pozostają więc do wyboru tylko dwa agenty: Thunderbird (dla początkujących) i Mutt (dla zaawansowanych).

Osoby bardzo wrażliwe na punkcie bezpieczeństwa mogą nie chcieć eksperymentować (a więc także potencjalnie skorumpować bezpieczeństwo) swojego konta studenckiego. Poniższe zadania można wykonać korzystając z jakiegokolwiek innego konta pocztowego udostępniającego protokoły ESMTP, IMAP i uwierzytelnianie za pomocą OAUTH2. W szczególności może być to serwer uruchomiony i skonfigurowany na własnym komputerze (choć wówczas pewnie zadania 11 nie da się wykonać).

**Zadanie 8 (1 pkt).** Zainstaluj na swoim komputerze program Thunderbird i skonfiguruj go do odbierania (IMAP) i wysyłania (ESMTP) poczty z Twojego studenckiego konta w domenie `uwr.edu.pl`. Wczytaj do Thunderbirda swoje wygenerowane klucze OpenPGP. Uwaga: Thunderbird nie obsługuje breloków z odłączonym kluczem głównym, nie usuwaj go więc z breloka. Przygotuj krótką prezentację pokazującą krok po kroku, co i jak należy skonfigurować.

**Zadanie 9 (1 pkt).** Skonfiguruj `gpg` jako plugin w Thunderbirdzie. W ten sposób możesz używać różnych własności `gpg`, które nie są dostępne w bibliotece RNP (np. odłączonego klucza głównego lub karty chipowej). Przygotuj krótką prezentację pokazującą krok po kroku, co i jak należy skonfigurować.

**Zadanie 10 (3 pkt).** Zainstaluj na swoim komputerze program Mutt i skonfiguruj go do odbierania (IMAP) i wysyłania (ESMTP) poczty z Twojego studenckiego konta w domenie `uwr.edu.pl`. Skonfiguruj na nim OpenPGP z wygenerowanymi wcześniej kluczami. Przygotuj prezentację pokazującą krok po kroku, co i jak należy skonfigurować.

**Zadanie 11 (1 pkt).** Z profilu prowadzącego w serwisie SKOS:

`https://skos.ii.uni.wroc.pl/user/profile.php?id=10`

pobierz klucz publiczny prowadzącego, znajdujący się w pliku `tomasz.wierzbicki.asc`:

```
pub   rsa4096 2022-01-31 [C]
      A3233FC467180159DDED189588AB5489AE7F30BA
uid           [ unknown] Tomasz Wierzbicki <tomasz.wierzbicki@uwr.edu.pl>
sig 3        88AB5489AE7F30BA 2022-01-31  Tomasz Wierzbicki <tomasz.wierzbicki@uwr.edu.pl>
sub   rsa4096 2022-01-31 [S]
sig          88AB5489AE7F30BA 2022-01-31  Tomasz Wierzbicki <tomasz.wierzbicki@uwr.edu.pl>
sub   rsa4096 2022-01-31 [E]
sig          88AB5489AE7F30BA 2022-01-31  Tomasz Wierzbicki <tomasz.wierzbicki@uwr.edu.pl>
```

(Prowadzący rozwiązał zadanie 2 w podobny sposób, w jaki oczekuje od Ciebie). Ze swojego studenckiego adresu *Twój-numer-albumu*`@uwr.edu.pl` wyślij pod adresem `tomasz.wierzbicki@uwr.edu.pl` zaszyfrowane i podpisane pozdrowienia Bożonarodzeniowe (albo złorzeczenia, że lista zadań jest nudna).