# Wykrywanie ataków sieciowych - lab 5

## Zadanie 1

Przykład nagłówków emaila oznaczonego jako SPAM:

```
X-Spam-Flag:YES
X-Spam-Score:7.584
X-Spam-Level:*******
X-Spam-Status:Yes, score=7.584 tagged_above=-999 required=6.2
tests=[BAYES 00=-1.9. DKIM INVALID=0.1, DKIM_ SIGNED=0.1,
HTML IMAGE ONLY 12=2.059. HTML MESSAGE=0.001, MIME HTML ONLY=0.723.
MART_ALT_ DIFF=0.79, MSGID FROM MTA HEADER=0.001,
SB_GIF_AND_NO_URIS=2.199, SPF_HELO_NONE=0.001, SPF_SOFTFAIL=3.5,
T OBFU PDF ATTACH=0.011 autolearn=no autolearn_force=no
```

Przykład nagłówków emaila nieoznaczonego jako SPAM:

```
X-Spam-Flag: NO
X-Spam-Score: -5.377
X-Spam-Level:
X-Spam-Status: No, score=-5.377 required=10 tests=[BAYES_00=-1.9,
  DKIMWL_WL_MED=-0.001, DKIM_SIGNED=0.1, DKIM_VALID=-0.1,
  DKIM_VALID_AU=-0.1, DKIM_VALID_EF=-0.1,
  HEADER_FROM_DIFFERENT_DOMAINS=0.249, HTML_FONT_LOW_CONTRAST=0.001,
  HTML_IMAGE_RATIO_04=0.001, HTML_MESSAGE=0.001, RCVD_IN_DNSWL_HI=-5,
  RCVD_IN_MSPIKE_H3=0.001, RCVD_IN_MSPIKE_WL=0.001, RDNS_NONE=0.793,
  SPF_HELO_FAIL=0.001, SPF_SOFTFAIL=0.665, T_KAM_HTML_FONT_INVALID=0.01,
  URIBL_BLOCKED=0.001] autolearn=ham autolearn_force=no
```

## Zadanie 2

- **BAYES_00** - prawdopodobieństwo spamu Bayesa wynosi od 0% do 1%. Prawdopodobieństwo spamu Bayesa jest metodą identyfikacji SPAMu z wykorzystaniem technik przetwarzania języka naturalnego i odzyskiwania informacji.
- **DKIM_SIGNED** - wiadomość posiada podpis DKIM lub DB, niekoniecznie zweryfikowany
- **DKIM_VALID** - wiadomość posiada przynajmniej jeden zweryfikowany podpis DKIM lub

DK

- **HTML_FONT_LOW_CONTRAST** - kolor czcionki dokumentu HTML jest podobna do koloru tła
- **HTML_IMAGE_RATIO_04** - dokument HTML ma niski stosunek powierzchni tekstu do powierzchni obrazu
- **HTML_MESSAGE** - wiadomość maila zawiera kod HTML
- **RCVD_IN_DNSWL_HI** - nadawca wylistowany na http://www.dnswl.org/, wysoka ufność
- **SPF_HELO_FAIL** - niepowodzenie weryfikacji sprawdzającej, czy SPF: HELO pasuje do rekordu SPF

## Zadanie 3

Przykładem algorytmu opartym na *Proof-of-work* jest algorytm *Hashcash*. Jest on łatwo dostępny zwykle w postaci plugina do klienta pocztowego. Polega na wegenerowaniu tekstowej pieczęci i dodaniu jej do nagłówka maila w celu odowodnienia, że nadawca poświęcił rozsądny czas pracu procesora do obliczenia funkcji skrótu przed wysłaniem wiadomości. Zadaniem odbiorcy jest jedynie zweryfikowanie pieczęci i dodanie jej do bazy danych, jeśli weryfikacja przebiegnie pomyślnie. Założeniem skuteczności algorytmu jest fakt, że spammerom zwykle zależy, aby wysłać jak najwięcej wiadomości znikomym kosztem, co jest niemożliwe, jeśli przed wysłaniem każdej z wiadomości trzeba wygenerować pieczęć.

## Zadanie 4

Szanka znajduje się w pliku `ham`.
Mielonka znajduje się w pliku `spam`.

Po przenalizowaniu mielonki spamassassinem wynik wyniósł 11.7 pkt. Najbardziej krytyczne elementy, które wpłynęły na ocenę i które należałoby zmienić, aby wiadomość została zaakceptowana:

- sprepareowane id wiadomości - należałoby zamieścić prawdziwe id wiadomości, zamiast spreparowanego
- brak X-MimeOLE - jeśli wiadomość zawiera nagłówek X-MSMail-Priority, należy również zamieścić X-MimeOLE
- znaleziony odcisk palca ze zbiorczych maili
- testy MSOE_MID_WRONG_CASE oraz SPOOFED_FREEMAIL - pierwszy jest warty aż 3.7 pkt, jednak spamassassin nie dostarczył opisu tych testów
- brak nawiasów w sekcji odbiorcy