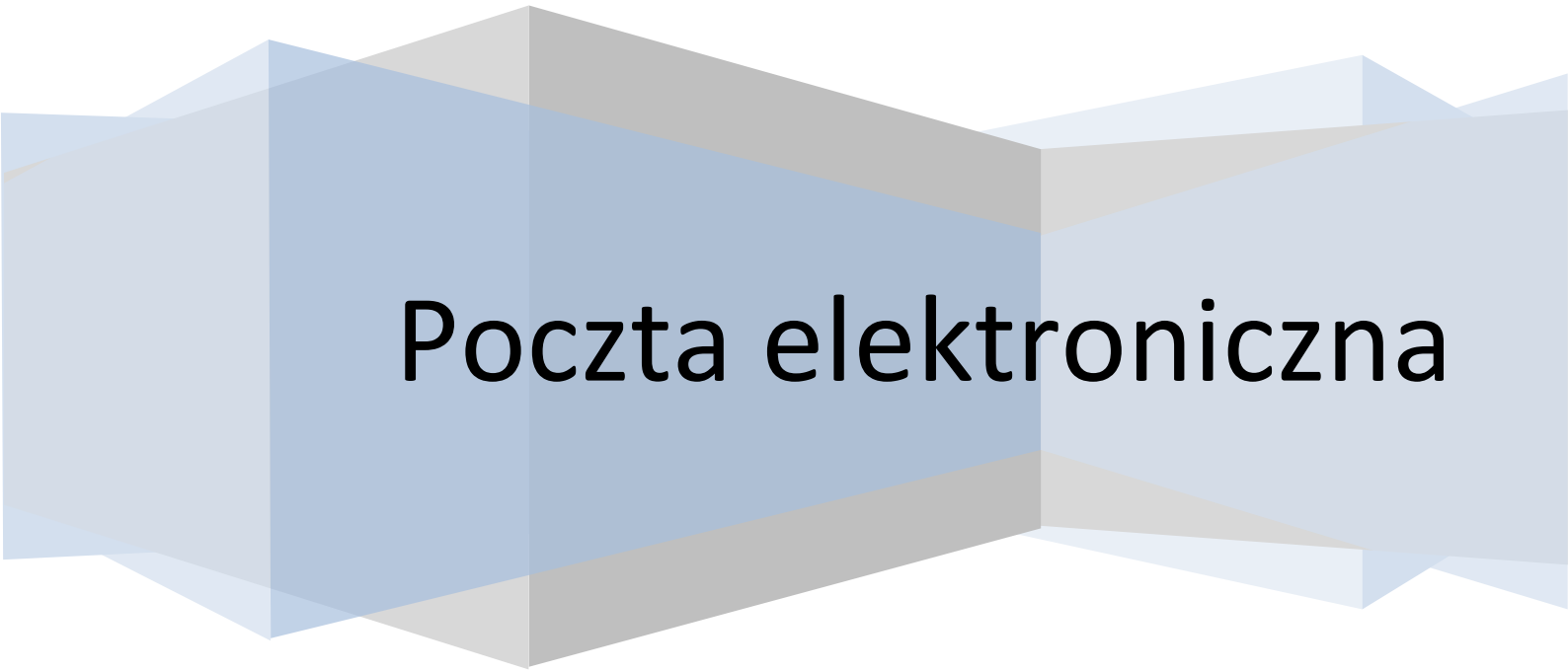
Politechnika Krakowska

Katedra Automatyki i Technik Informacyjnych

Laboratorium Sieci Komputerowych

2010/2011



Poczta elektroniczna

1. Wprowadzenie

Poczta elektroniczna (znana również jako *e-mail*) jest jedną z najpopularniejszych usług sieciowych. Pozwala ona na wymianę informacji pomiędzy pojedynczymi osobami, a także grupami osób (listy dyskusyjne). We wprowadzeniu przedstawiono charakterystyki trzech protokołów pocztowych: *SMTP*, *POP3* i *IMAP*. Wszystkie te protokoły wykorzystują TCP jako protokół warstwy transportowej.

1.1. SMTP

SMTP (Simple Mail Transfer Protocol) jest protokołem niezawodnego przesyłania wiadomości e-mail za pomocą prostych komend tekstowych – RFC 2821. Protokół zaczął być szeroko używany w początkach lat 80-tych dwudziestego wieku. W 1995 protokół został rozszerzony (*ESMTP*) – RFC 1869. Jednym z pierwszych programów do przesyłania poczty wykorzystującym SMTP był `sendmail`.

Standardowo serwer implementujący ten protokół nasłuchuje na porcie 25. Podstawowy SMTP składa się z kilkunastu komend (ESMTP wnosi kilka nowych). Tabela 1.1 przedstawia komendy i ich znaczenie.

SMTP		ESMTP	
Komenda	**Opis**	**Komenda**	**Opis**
HELO	Nawiązanie połączenia SMTP	AUTH	Autoryzacja
MAIL	Nadawca listu	DSN	Powiadamianie o statusie doręczenia
RCPT	Odbiorca listu	EHLO	Nawiązanie połączenia, spis dostępnych komend
DATA	Rozpoczęcie treści wiadomości	ETRN	Przesłanie kolejki listów przeznaczonych do podłączającego się serwera
RSET	Przerwanie sesji SMTP	PIPELINING	Potokowe przesyłanie wiadomości
VRFY	Sprawdzenie obecności skrzynki pocztowej o podanej nazwie	SIZE	Określenie wielkości przesyłanej wiadomości
NOOP	Podtrzymywanie połączenia	8BITMIME	Wsparcie dla 8-bitowego kodowania wiadomości
HELP	Wypis dostępnych komend	RESTART	Wznowienie przesyłania wiadomości przy rozłączeniu
QUIT	Zakończenie sesji SMTP		

Tabela 1.1: Podstawowe komendy protokołu SMTP i jego rozszerzenia ESMTP.

Kody zwrotne SMTP informują na bieżąco o statusie serwera. Każda wysłana przez klienta komenda wymaga w odpowiedzi kodu zwrotnego od serwera. Tabela 1.2 przedstawia odpowiedzi zwrotne.

Kody zwrotne			
Kod	**Opis**	**Kod**	**Opis**
211	Odpowiedź stanu systemu lub pomocy systemowej	500	Błąd składniowy, polecenie nierozpoznane
214	Komunikat pomocy	501	Błąd składniowy w parametrach lub argumentach
220	Usługa gotowa	502	Polecenie nie zostało implementowane
221	Usługa zamyka kanał transmisyjny	503	Zła kolejność poleceń
250	Żądane działanie poczty OK, zakończone	504	Parametr polecenia nie został implementowany
251	Użytkownik nielokalny; przekaż do ścieżki przekazywania	550	Żądane działanie poczty nie zostało podjete: skrzynka pocztowa niedostępna
354	Rozpocznij wprowadzanie poczty	551	Użytkownik nielokalny; spróbuj wykorzystać ścieżkę przekazywania
421	Usługa niedostępna, zamykam kanał transmisyjny	552	dane działanie poczty zostało przerwane: przekroczona alokacja pamięci
450	Żądane działanie poczty nie zostało podjęte: skrzynka pocztowa niedostępna	553	dane działanie nie zostało podjęte: niedozwolona nazwa skrzynki pocztowej
451	Żądane działanie zostało przerwane	554	Transakcja nie powiodła się
452	Żądane działanie nie zostało podjęte: niewystarczająca ilość pamięci systemowej		

Tabela 1.2: Odpowiedzi protokołu SMTP

Poniżej znajduje się przykład sesji SMTP. Pierwsza odpowiedź serwera występuje przy otwarciu gniazda połączenia (S – serwer, K – klient).

```
S: 220 beta.gov Simple Mail Transfer Service Ready
K: HELO alpha.edu
S: 250 beta.gov
K: MAIL FROM:<studentA@alpha.edu>
S: 250 OK
K: RCPT TO:<studentB@beta.gov>
S: 250 OK
K: RCPT TO:<studentC@beta.gov>
S: 550 No such user here
K: RCPT TO:<studentC@beta.gov>
S: 250 OK
K: DATA
S: 354 Start mail input; end with <CR><LF>.<CR><LF>
K: ... tu jest treść przesyłki ...
K: ... tyle linii, ile potrzeba ...
```

```
K: <CR><LF>.<CR><LF>
S: 250 OK
K: QUIT
S: 221 beta.gov Service closing transmission channel
```

Najpopularniejsze serwery SMTP to *Postfix*, *Exim*, *Sendmail, qmail*, *Novell Netmail*, *Microsoft Exchange Server*.

1.2. POP3

Post Office Protocol version 3 to protokół internetowy z warstwy aplikacji pozwalający na odbiór poczty elektronicznej ze zdalnego serwera do lokalnego komputera poprzez połączenie TCP/IP. W niektórych angielskojęzycznych programach pocztowych POP3 określany jest jako *Incoming Mail Server* (serwer poczty przychodzącej). Protokół POP3 jest standardowym protokołem pobierania poczty e-mail (alternatywą dla niego jest IMAP). Protokół POP3 kontroluje połączenie między klientem poczty e-mail POP3 a serwerem, na którym poczta e-mail jest przechowywana. Wcześniejsze wersje protokołu POP, czyli POP (czasami nazywany POP1), POP2 zostały całkowicie zastąpione przez POP3. Zwykle jeżeli ktoś mówi o protokole POP ma na myśli jego wersję 3.

Protokół POP3 ma trzy stany przetwarzania związane z obsługą połączenia między serwerem poczty a klientem poczty e-mail POP3: stan uwierzytelniania, stan transakcji i stan aktualizacji.

W stanie uwierzytelniania klient poczty e-mail POP3, który łączy się z serwerem, musi zostać uwierzytelniony, aby użytkownicy mogli pobrać swoją pocztę e-mail. Jeśli nazwa użytkownika i hasło podane przez klienta poczty e-mail są takie same jak te na serwerze, użytkownik zostaje uwierzytelniony i przechodzi do stanu transakcji. W przeciwnym wypadku użytkownik otrzymuje komunikat o błędzie i nie wolno mu nawiązać połączenia w celu pobrania poczty e-mail.

Komendy w stanie uwierzytelniania	
Komenda	**Opis**
USER username	służy do podania nazwy skrzynki klienta
PASS password	służy do podania hasła klienta. String to ciąg znaków mogących posiadać spację
QUIT	pozwala na wyjście z sesji w jej początkowej fazie

Tabela 1.3: Komendy protokołu POP3 używane w trakcie autoryzacji.

W danej chwili ze skrzynką pocztową może być połączony tylko jeden klient; dodatkowe żądania połączenia ze skrzynką pocztową są odrzucane.

W stanie transakcji klient wysyła polecenia POP3, które serwer odbiera i odpowiada na nie zgodnie z protokołem POP3.

Komendy w stanie transakcji	
Komenda	**Opis**
STAT	Pokazuje zawartość skrzynki. W odpowiedzi serwer przesyła linię: +OK. nn mm, gdzie nn oznacza liczbę wiadomości w skrzynce , a mm oznacza wielkość wszystkich wiadomości w

	skrzynce wyrażoną w oktetach
LIST msg	Wylistowuje zawartość skrzynki. Gdy nie podamy żadnych parametrów serwer odpowiada +OK. X messages (XXX octets). Gdy podamy parametr, który oznacza numer wiadomości serwer może odpowiedzieć np.: +OK. 2 200 (co oznacza, że wiadomość nr 2 ma wielkość 200 oktetów)
RETR msg	Przesłanie wiadomości o numerze msg do klienta w celu jej odczytania
DELE msg	Zaznaczenie wiadomości msg do usunięcia np.: DELE 1
NOOP	Serwer nic nie robi tylko przesyła pozytywną odpowiedź (+OK)

Tabela 1.4: Komendy protokołu POP3 używane podczas transakcji.

Stan aktualizacji zamyka połączenie między klientem a serwerem. Jest to ostatnie polecenie wysłane przez klienta. Kiedy klient wydaje komendę wyjścia ze stanu transakcji (QUIT) sesja wchodzi w stan aktualizacji (UPDATE).

Komunikacja POP3 może zostać zaszyfrowana z wykorzystaniem protokołu SSL. Jest to o tyle istotne, że w POP3 hasło przesyłane jest otwartym tekstem, o ile nie korzysta się z opcjonalnej komendy protokołu POP3, APOP.

Opcjonalne komendy protokołu POP3 pozwalają klientowi w wygodniejszy sposób zarządzać swoją skrzynką pocztową zachowując przy tym nadal niekwestionowaną prostotę zarządzania protokołem POP3.

Komendy opcjonalne	
Komenda	**Opis**
TOP msg n	msg oznacza numer wiadomości, n to liczba linii. Po wydaniu tej komendy serwer przesyła nagłówek wiadomości i pierwsze *n* linii wiadomości. Funkcja przydatna w przypadku dużych wiadomości, które lepiej wstępnie przejrzeć bez potrzeby transportowania ich na terminal klienta.
UIDL msg	Serwer przesyła linię nazywaną *unique-id listing* opisującą daną wiadomość. Kod identyfikacyjny składa się z liter z zakresu od 0x21 do 0x7E (od 33 do 126 dziesiętnie), które w całości identyfikują wiadomość w skrzynce.
APOP name digest	name to nazwa skrzynki a digest to MD5 digest string (oba argumenty są niezbędne). Ta komenda umożliwia nie przesyłanie hasła w jego czystej wersji poprzez sieć, zapobiegając wykryciu go przez osoby postronne.

Tabela 1.5: Opcjonalne komendy protokołu SMTP.

Przykładowa sesja POP3 (S – serwer, K – klient). Pierwsza odpowiedź serwera występuje przy otwarciu gniazda połączenia.

```
S: +OK POP3 server ready <1896.697170952@dbc.mtview.ca.us>
K: APOP mrose c4c9334bac560ecc979e58001b3e22fb
S: +OK mrose's maildrop has 2 messages (320 octets)
K: STAT
S: +OK 2 320
```

```
K: LIST
S: +OK 2 messages (320 octets)
S: 1 120
S: 2 200
S: .
K: RETR 1
S: +OK 120 octets
S: <serwer przesyła wiadomość 1>
S: .
K: DELE 1
S: +OK message 1 deleted
K: QUIT
S: +OK dewey POP3 server signing off
```

1.3. IMAP

IMAP – Internet Mail Access Protocol – został zaprojektowany jako zastępca POP3. Stworzony w 1986 roku oferuje o wiele większe możliwości niż protokół POP. Oryginalna wersja została kilkakrotnie zmieniona, obecnie (od 1996r.) używany jest protokół IMAP4rev1, zdefiniowany w RFC 3501. Ważną cechą tego protokołu jest wsparcie dla mechanizmów bezpiecznego uwierzytelniania i przesyłania wiadomości – używany jest tu TLS.

Serwer IMAP standardowo nasłuchuje na porcie 143. Protokół IMAP składa się z ciągu komend klienta i odpowiedzi serwera. Każda komenda wysłana przez program kliencki opatrzona jest *tagiem*, który jest unikatowy dla każdej kolejnej komendy w obrębie sesji. Tag jest zazwyczaj krótkim (kilko znakowym) tekstem. Używany jest on także w odpowiedzi do identyfikacji odpowiedzi na daną komendę.

Komendy		**Odpowiedzi**	
NOOP		OK.	text
LOGIN	user password	NO	text
LOGOUT		BAD	text
SELECT	Mailbox	FLAGS	flags_list
BBOARD	bulletin_board	SEARCH	sequence
FIND_MAILBOXES	Pat tern	BBOARD	string
CHECK		MAILBOX	string
EXPUNGE		BYE	text
COPY	sequence mailbox		
FETCH	sequence data		
STORE	sequence data values		
SEARCH	search_program		

Tabela 1.6: Komendy i odpowiedzi protokołu IMAP. Kolumna druga zawiera parametry komend, a kolumna czwarta dane zawarte w odpowiedzi.

Przykładowa sesja protokołu IMAP4 przedstawiona została poniżej (S – serwer, K – klient). Pierwsza odpowiedź serwera występuje (tak jak i wcześniej) przy otwarciu gniazda połączenia.

```
S: * OK [CAPABILITY IMAP4REV1 LOGIN-REFERRALS STARTTLS AUTH=LOGIN]     cap.city.org IMAP4rev1
2003.339
            at Wed, 13 Apr 2005 01:38:58 -0400 (EDT)
K: A1 LOGIN mailtest password
S: A1 OK [CAPABILITY IMAP4REV1 IDLE NAMESPACE MAILBOX-REFERRALS BINARY         UNSELECT SCAN SORT
THREAD=REFERENCES THREAD=ORDEREDSUBJECT         MULTIAPPEND] User mailtest authenticated
K: A2 SELECT Inbox
S: * 2 EXISTS
   * 2 RECENT
   * OK [UIDVALIDITY 1113370837] UID validity status
   * OK [UIDNEXT 3] Predicted next UID
   * FLAGS (¥Answered ¥Flagged ¥Deleted ¥Draft ¥Seen)
   * OK [PERMANENTFLAGS (¥* ¥Answered ¥Flagged ¥Deleted ¥Draft ¥Seen)]         Permanent flags
   * OK [UNSEEN 1] first unseen message in /var/mail/mailtest
   A2 OK [READ-WRITE] SELECT completed
K: A3 FETCH 2 BODY[HEADER]
S: * 2 FETCH (BODY[HEADER] {670}
Return-Path:
X-Original-To: mailtest@cap.city.org
Delivered-To: mailtest@cap.city.org
Received: from node18.city.org (node18 [192.168.5.38])
        by cap.city.org (Postfix) with ESMTP id A291B2B15C
        for ; Tue, 12 Apr 2005 22:23:53 -0400 (EDT)
Received: from me?here.com (unknown [192.168.5.250])
        by node18.city.org (Postfix) with SMTP id 4653B14112
        for ; Tue, 12 Apr 2005 22:24:03 -0400 (EDT)
To: guru@there.com
From: d@t.com
Subject: Forged e-mail
Message-Id: <20050413022403.4653B14112@node18.city.org>
Date: Tue, 12 Apr 2005 22:24:03 -0400 (EDT)


)
* 2 FETCH (FLAGS (¥Recent ¥Seen))
A3 OK FETCH completed
K: A4 FETCH 2 BODY[TEXT]
S: * 2 FETCH (BODY[TEXT] {88}
Hey,
The "To:" and "From:" are non-existent, but you still get the e-mail.
bye, bye
)
A4 OK FETCH completed
K: A5 LOGOUT
S: * BYE cap.city.org IMAP4rev1 server terminating connection
A5 OK LOGOUT completed
```

Najpopularniejszymi serwerami pocztowymi obsługującymi protokoły POP i IMAP są *Courier*, *Cyrus* oraz *Dovecot*.

1.4. Cel ćwiczenia

Celem ćwiczenia jest zapoznanie się z podstawowymi aspektami pracy, konfiguracji i administracji serwerem pocztowym oraz funkcjonowaniem protokołów związanych z obsługą poczty elektronicznej.

2. Konfiguracja serwerów poczty wychodzącej, przychodzącej, filtrów antyspamowych i Webmaila.

2.1. Konfiguracja programu Exim4

W systemie Debian Linux istnieje dogodne narzędzie umożliwiające podstawową konfigurację serwera poczty – Exim4 – wywoływane z linii komend (z uprawnieniami root):

```
dpkg-reconfigure exim4-config
```

W kolejnych ekranach tego narzędzia przeprowadzana jest konfiguracja agenta MTA. Możliwa jest także konfiguracja programu bezpośrednio w plikach konfiguracyjnych znajdujących się w /etc/exim4/. Przedstawiony zostanie sposób konfiguracji przy pomocy narzędzia systemowego.

W pierwszym ekranie wybieramy przeznaczenie konfigurowanego programu. Dostępne możliwości:

- **ośrodek internetowy**

Serwer działa jako serwer pocztowy ogólnie dostępny, e-mail'e przesyłane są bezpośrednio do MTA w domenach adresatów. W większości przypadków poczta wysyłana przez program bez zdefiniowania domeny internetowej zostaje odrzucona.

- **poczta wysyłana przez pośrednika**

Wiadomości dostarczone do MTA są przekazywane do innego MTA (pośrednika) za pomocą protokołu SMTP, loginu i hasła.

- **poczta wysyłana przez pośrednika bez dostarczania lokalnego**

Jak wyżej, z tym, że poczta nie jest dostarczana do lokalnych użytkowników bezpośrednio (bez pomijania serwera pośredniczącego).

- **tylko dostarczanie lokalne**

W tym przypadku poczta jest dostarczana wyłącznie do lokalnych (zdefiniowanych w aplikacji/systemie) kont pocztowych.

- **nie konfiguruj w tym momencie**

Ta opcja nas nie interesuje.

Konfiguracja – ośrodek internetowy

1. Po wybraniu tej opcji w następnym ekranie podajemy nazwę pocztową. Jest to nazwa dopisywana do adresów bez domeny. Powinna to być pełna nazwa domeny, np. dla: jacek@pk.edu.pl – *pk.edu.pl*.
2. Następnie konfigurujemy adresy na których ma nasłuchiwać Exim. Jeżeli pozostawimy puste miejsce program będzie nasłuchiwał na wszystkich dostępnych interfejsach. Podanie wartości 127.0.0.1 powoduje, że program będzie działał lokalnie – w obrębie użytkowników na tej maszynie.

3. W kolejnym kroku podajemy tzw. *domeny lokalne*. Wiadomości zaadresowane na którąś z tych domen będą obsługiwane w zakresie lokalnym. Domyślnie lokalnymi domenami są: *localhost* oraz *[host]* (nazwa hosta).
4. Podajemy domeny dla których system ma przekazywać wiadomości. Opcja ta jest przydatna w przypadku budowania systemu pocztowego w organizacji (np. firmie) gdzie poczta jest przekazywana z kilku systemów do systemu głównego (bramki), który jest widoczny 'na zewnątrz'. Jeżeli nie ma przekazywać żadnej poczty zostawić puste.
5. Dalej wprowadzamy zakresy adresów IP dla których program ma bezwarunkowo przyjmować pocztę działając na zasadzie pośrednika. Jeżeli podamy pusty ciąg to program nie będzie pośrednikiem. W tym przypadku jeżeli łączymy się do serwera z sieci powinniśmy podać zakres adresów z tej sieci, jeżeli chcemy aby poczta byłą przekazywana tylko z tego komputera należy wprowadzić np. 127.0.0.1/24.
6. Przy pytaniu o utrzymywanie ilości zapytań na minimalnym poziomie najlepiej zaznaczyć *Nie*.
7. Ustawiamy format dostarczania lokalnej poczty na *Maildir*, ze względu na to, że dalej używany będzie program Courier.
8. Ostatni krok to wybór przechowywania konfiguracji – w jednym pliku lub podzielonej na wiele osobnych dla każdej sekcji – zalecany jest podział na wiele plików.

Konfiguracja – poczta wysyłana przez pośrednika

System pocztowy można skonfigurować w ten sposób by odbierał wiadomości od użytkowników wewnątrz systemu, dostarczaj je lokalnie – jeśli wiadomość jest adresowana lokalnie – bądź wysyłał przez pośrednika – w przypadku adresacji zewnętrznej – przy użyciu konta w innym systemie pocztowym. Należy w pierwszym ekranie narzędzia konfiguracyjnego wybrać opcję drugą (lub trzecią). Konfiguracja przebiega podobnie jak we wcześniejszym opisie. Po kroku nr 5, ustawiamy adres pod którym znajduje się pośrednik przyjmujący pocztę wychodzącą. Następnie decydujemy czy domeny w nagłowkach *From*, *Reply-To*, *Return-Path* mają być podmieniane. Jeżeli zdecydujemy się na to, w następnym ekranie zostaniemy poproszeni o wpisanie domeny na jaką mają być podmieniane.

Aby dokończyć konfigurację z dostarczaniem przez pośrednika, należy jeszcze wpisać w pliku /etc/exim4/passwd.client login i hasło do pośrednika poczty, w formacie:

```
serwer:login:haslo
```

Inne ustawienia serwera exim4

Exim posiada możliwość ograniczania przestrzeni pocztowej dla użytkownik, dzięki czemu można odciążyć system plików od monitorowania skrzynek. Należy wyedytować (zakładamy konfigurację rozbitą) plik:
`/etc/exim4/conf.d/transport/30_exim4-config_maildir_home`
(dla formatu *Maildir*) lub
`/etc/exim4/conf.d/transport/30_exim4-config_mail_spool`
(dla formatu *mbox*). Możliwe ograniczenia:

- ograniczenie ilość miejsca w skrzynce: `quota = XX` (np. `quota = 20M`)
- limit rozmiaru wiadomości: `message_size_limit = XX` (np. `message_size_limit = 2M`)

W obu przypadkach istnieje także możliwość zdefiniowania makra, i określenia dla konkretnych użytkowników oddzielnych ograniczeń.

Inne pliki używane do konkretnych konfiguracji serwera (pełny opis: manual exim4_files):

- `/etc/aliases` – udostępnia mechanizm do przekierowywania wiadomości do lokalnych odbiorców
- `/etc/email-addresses` – używane do nadpisywania adresów e-mail użytkowników lokalnych (format: konto: adres@email)
- `/etc/exim4/local_host_blacklist` – lista adresów IP i nazw hostów od których wiadomości nie będą przyjmowane
- `/etc/exim4/local_host_whitelist` – lista adresów IP i nazw hostów od których wiadomości będą przyjmowane (nadpisuje plik wyżej)
- `/etc/exim4/local_sender_blacklist` – lista nadawców od których serwer nie będzie przyjmował wiadomości do wysłania
- `/etc/exim4/local_sender_whitelist` – podobnie jak z `local_host_whitelist`
- `/etc/exim4/local_sender_callout`
- `/etc/exim4/local_rcpt_callout`
- `/etc/exim4/local_domain_dnsbl_whitelist`
- `/etc/exim4/hubbed_hosts` – lista zawierająca dane służące do nadpisywania rekordów MX
- `/etc/exim4/passwd` – nazwy kont oraz ich hasła gdy exim pracuje jako lokalny serwer
- `/etc/exim4/passwd.client` – nazwy kont oraz hasła gdy exim przekazuje pocztę do innych serwerów SMTP
- `/etc/exim4/exim.crt` - certyfikaty służące do inicjalizacji połączeń TLS
- `/etc/exim4/exim.key` – klucze należące do certyfikatów w `exim.crt`

2.2. Konfiguracja programu *Courier*

Courier oferuje obsługę protokołów POP3 oraz IMAP, obydwa wraz z SSL. Konfiguracja programu w dużej części jest automatyczna. Po zainstalowaniu paczek z obsługą SSL automatycznie zostaną wygenerowane klucze oraz certyfikaty.

Pliki konfiguracyjne znajdują się w `/etc/courier`. Oto za co są odpowiedzialne poszczególne pliki:

- `authdaemonrc` – konfiguracja demona autoryzacji i identyfikacji użytkowników. W pliku tym ustawiany jest m. in. poziom logowanych informacji (DEBUG_LOGIN) oraz moduł do obsługi samej autoryzacji – dostępnych jest kilka, w tym domyślnie PAM.
- `imapd`, `imapd-ssl` – specyficzne dla protokołu IMAP ustawienia, a także numer portu na którym demon ma nasłuchiwać, nazwa folderu przechowującego wiadomości (dla każdego użytkownika taki sam). W drugim pliku ustawienia kluczy i certyfikatów, szczegóły szyfrowania, a także numer portu.
- `pop3d`, `pop3d-ssl` - podobnie jak wyżej.

Podczas instalacji z paczki .deb większość opcji jest ustawiana na standardowe wartości, jedyne co pozostaje do zrobienia to otwarcie odpowiednich portów na serwerze oraz konfiguracja użytkowników. Zazwyczaj używany jest tutaj format *Maildir*. Każdy użytkownik powinien mieć przygotowany folder na pocztę przy użyciu narzędzia `maildirmake` – program dostarczany wraz Courier'em, np.:

```
maildirmake -S /home/user/Maildir
```

Można stworzyć dodatkowe foldery, w których będą przechowywane wiadomości: wysłane, usunięte, spam, itp. Dla każdego z tych folderów należy wywołać powyższą komendę (w folderze głównym poczty – tutaj /home/user/Maildir), np. dla wiadomości usuniętych:

```
maildirmake -S .Trash
```

Następnie, aby użytkownik po zalogowaniu się do Webmaila (lub innego klienta pocztowego) widział te foldery należy jest 'subskrybować', definiując odpowiednie wpisy w pliku courierimapsubscribed, w głównym katalogu pocztowym, np. (dla powyższego przykładu):

```
INBOX
INBOX.Trash
```

Inną istotną rzeczą jest skonfigurowanie PAM'u (bo domyślnie jest używany i zalecany przy autoryzacji i identyfikacji) tak aby zawierał odpowiednie pliki wiążące moduł PAM z demonami protokołów. Pliki pop3 oraz imap powinny znajdować się w /etc/pam.d/. Zawartość takiego pliku jest następująca:

```
# PAM configuration file for Courier POP3 daemon
@include common-auth
@include common-account
@include common-password
@include common-session
```

2.3. Konfiguracja filtru antyspamowego *SpamAssassin* z programem *Exim4*

Podobnie jak podczas wcześniejszych programów, instalacja SpamAssasina z paczki uwalnia nas od większości pracy. Po instalacji należy włączyć demona – w pliku /etc/default/spamassasin, znajduje się odpowiednia do tego opcja (ENABLED = 1) oraz kilka systemowych ustawień (przykładowo priorytet procesu, np: NICE="--nicelevel 15", domyślnie zakomentowana). Właściwa konfiguracja demona znajduje się w /etc/spamassasin/local.cf. W przypadku naszej konfiguracji istotny jest parametr required_score, który oznacza próg 'punktów' po którym wiadomość jest uznawana za spam. Opcje use_bayes oraz bayes_auto_learn włączają 'uczenie się' filtru.

W przypadku konfiguracji z program Exim, istnieje kilka możliwości zintegrowania SpamAssasin'a. W tej instrukcji przedstawiony zostanie sposób wykorzystujący dodatek *exiscan*, jako że pozwala on na odrzucanie wiadomości oznaczonych jako spam w czasie odbierania wiadomości od innego MTA. Jest to najwydajniejsza konfiguracja. Wiadomość przechodzi przez filtr antyspamowy, który 'zwraca' nam informację o prawdopodobieństwie czy dana wiadomość jest spamem, oto dostępne zmienne:

- $spam_score – 'punkty' (np. 3.4), im więcej tym prawdopodobieństwo że wiadomość jest spamem jest większa
- $spam_score_int – j/w, z tym że jest to liczba całkowita (np. 3.4 → 34)
- $spam_bar – łańcuch znaków składający się z '+', których jest tyle ile wynosi część całkowita z $spam_score
- $spam_report – raport dostarczony ze SpamAssasina

Aby wykorzystać te informacje musimy wyedytować plik /etc/exim4/conf.d/acl/40_exim4-config_check_data. Należy dodać w nim dodać odpowiednie reguły:

```
# dodanie nagłówków do wszystkich wiadomości o rezultacie filtru
warn  spam = nobody:true
```

```
      add_header = X-Spam-Score: $spam_score ($spam_bar)
      add_header = X-Spam-Report: $spam_report

# dodanie nagłówka: '*SPAM* temat' w przypadku gdy wynik (score) jest
# powyżej ustalonego progu
warn  spam = nobody
      add_header = X-Spam-Subject: *SPAM* $h_Subject:
      add_header = X-Spam-Status: Yes

# odmowa odebrania wiadomości gdy wynik jest większy niz 12
deny  message = This message scored $spam_score spam points.
      spam = nobody:true
      condition = ${if >{$spam_score_int}{120}{1}{0}}
```

Aby wiadomości oznaczone jako spam (X-Spam-Status: Yes) trafiały bezpośrednio po odebraniu do folderu ze spałem należy posłużyć się *filtrem*. Każdy użytkownik Exim'a może tworzyć swoje filtry, umieszczając je w swoim katalogu domowym w pliku .forward. Aby założyć globalny filtr należy umieścić wpis o filtrze w którymś z głównych plików konfiguracyjnych, np. w /etc/exim4/conf.d/main/02_exim4-config_options:

```
system_filter = "/etc/exim4/system.filter" # plik z filtrem
system_filter_user = Debian-exim
system_filter_group = Debian-exim
system_filter_pipe_transport = address_pipe
system_filter_file_transport = address_file
system_filter_reply_transport = address_reply
```

Przykładowy filtr, zmieniający nagłówek z tematem wiadomości gdy zostanie ona oznaczona jako SPAM, plik /etc/exim4/system.filter:

```
# Exim
if $h_X-Spam-Status: CONTAINS "Yes"
then
  headers remove Subject # usunięcie tematu
  headers add "Subject: $header_X-Spam-Subject" # podmiana
  headers remove X-Spam-Subject # usunięcie nagłówka
  finish
endif
```

Przykładowy filtr użytkownika umieszczający wiadomości oznaczone jako SPAM w katalogu SPAM, plik .forward:

```
# Exim
if $h_X-Spam-Status: CONTAINS "Yes"
then
  save $home/Maildir/.Spam/
  finish
endif
```

Reguły w SpamAssassin

Istnieje możliwość definiowania własnych reguł. Reguły można przechowywać globalnie (np. w pliku /etc/spamassasin/local.cf) lub oddzielnie dla każdego użytkownika (~/.spamassasin/user_prefs).

Dodając nową regułę, tworzymy jej symboliczną nazwę, następnie za pomocą wyrażeń regularnych definiujemy wyszukiwane wzorce, w treści wiadomości i/lub w nagłówkach. Definiujemy także liczbę 'punktów kary', która zostaje dodana do wiadomości w przypadku gdy reguła pasuje do wiadomości (punkty te mają negatywne znaczenie - im jest ich więcej tym większe prawdopodobieństwo, że wiadomość jest spamem). Przykład reguły:

```
body    TEST1 /test/i
header TEST2 Subject =~ /test/i
meta    TEST  (TEST1 && TEST2)
score   TEST  2
describe      TEST  Przykladowa regula
```

W pierwszej linii deklarujemy regułę o nazwie TEST1, która sprawdza (bez względu na wielkość liter) czy treść wiadomości zawiera słowo: *test*. Druga reguła (TEST2) robi podobną rzecz, ale sprawdza pod tym kątem nagłówek wiadomości - tutaj temat. Trzecia linia definiuje kolejną regułę, która jest spełniona wtedy, gdy jedna z dwóch pierwszych jest spełniona. W linii czwartej nadana zostaje punktacja, w piątek opis reguły.

2.4. Konfiguracja Webmaila

W ćwiczeniu użyty zostanie *RoundCube* Webmail. Instalacja aplikacji jest bardzo prosta (jak większości pakietów w systemie Debian Linux). Jako że jest to klient poczty dostępny z poziomu przeglądarki WWW, dodatkowo zainstalowany zostanie serwer www: apache2.

Po zainstalowaniu paczki roundcube, program konfiguracyjny pyta o ustawienia bazy danych, w tym ćwiczeniu użyta zostanie baza danych *SqlLite*. Następnie należy przejść do edycji pliku /etc/apache/apache2.conf i dodać na końcu pliku linijkę ustawiającą alias do aplikacji webmaila:

```
Alias  /usr/share/roundcube /roundcube
```

oraz zrestartować serwer www (wymagane uprawnienia *root'a*):

```
# /etc/init.d/apache2 restart
```

Aplikacja będzie dostępna z naszej maszyny, np. pod adresem: http://localhost/roundcube.

Konfiguracja samej aplikacji znajduje się w pliku /etc/roundcube/main.inc.php. Aby był to klient obsługujący jedynie nasz system pocztowy, ustawiamy zmienna:

```
$rcmail_config['default_host'] = 'localhost';
```

domyślny język:

```
$rcmail_config['language'] = 'pl_PL';
```

Możemy także zmienić nazwę klienta, widoczną w nagłówku User-Agent

```
$rcmail_config['useragent'] = 'RoundCube Webmail/0.2.1';
```

oraz nazwę wyświetlaną w pasku tytułu przeglądarki

```
$rcmail_config['product_name'] = 'RoundCube Webmail';
```

W pliku konfiguracyjnym znajduje się wiele innych opcji, które są opisane w kodzie.

3. Realizacja ćwiczenia

3.1. Konfiguracja maszyn do przeprowadzenia ćwiczenia

Realizacja ćwiczenia przeprowadzana jest na systemie Debian GNU/Linux 5.0.4 Lenny. W systemie zainstalowane są i pozostawione bez konfiguracji następujące aplikacje będące tematem ćwiczenia:

- exim4,

- courier-imap, courier-pop (plus rozszerzenia o SSL),
- SpamAssassin,
- RoundCube Webmail.

Ponadto w systemie zainstalowany jest *Sun VirtualBox*, oraz przygotowana maszyna wirtualna z *Windows XP Volume Edition*. W systemie Windows zainstalowano klienta e-mail *Mozilla Thunderbird* oraz *PuTTY*.

Poszczególne nazwy hostów oraz ważniejsze ustawienia sieciowe przedstawia Tabela 3.1. W obydwu systemach utworzone zostały dwa konta użytkowników – Tabela 3.2. **W system Debian utworzony został także specjalny użytkownik: operator, hasło: sk2010. Należy zalogować przy użyciu tego konta, i wykonywać dalsze instrukcje.** Uprawnienia *root'a* uzyskiwane powinny być za pomocą programu sudo.

Nazwa hosta	sk-server	sk-client
Adres IP	172.22.7.131	172.22.7.132
Maska sieci	255.255.255.128	255.255.255.128
Otwarte porty	25 – smtp 80 – http 110 – pop3 143 – imap 993 – imap-ssl 995 – pop3-ssl	brak

Tabela 3.1: Konfiguracja ustawień sieciowych poszczególnych maszyn.

Nazwa użytkownika	Ludomir	helena
Pełna nazwa	Ludomir Lubomirski	Helena Helińska
Hasło	ludomir1	helena1

Tabela 3.2: Użytkownicy w systemie Debian. W systemie Windows użytkownicy są identyczni, ale bez haseł.

3.2. Instrukcje do wykonania

Po dokonaniu zmian w plikach konfiguracyjnych należy zrestartować odpowiedni demon tak by zmiany zaszły w systemie:

```
sudo /etc/init.d/(demon) restart
```

(demon) to odpowiednio:

- exim4
- courier-pop
- courier-pop-ssl
- courier-imap
- courier-imap-ssl
- spamassassin

1. Konfiguracja podstawowa systemu.

1.1. Należy skonfigurować program exim, by działał jako ośrodek internetowy, nazwę pocztową ustawić na `sk-server.pk.edu.pl`, program powinien przyjmować połączenia z dowolnego miejsca w sieci, format przechowywanych wiadomości ustawić na `Maildir` oraz zastosować konfigurację rozproszoną na wiele plików.

1.2. Utworzyć w katalogach domowych każdego z użytkowników foldery do przechowywania wiadomości e-mail – `Maildir` (zwrócić uwagę na uprawnienia). Jeżeli użyjemy innej nazwy folderu trzeba to uwzględnić w konfiguracji zarówno exim'a jak i courier'a.

1.3. Odpalić wirtualną maszynę z systemem Windows, zalogować się na użytkownika `ludomir`.

1.4. Używając klienta telnetu **(putty)**, połączyć się z systemu Windows do systemu pocztowego w celu wysłania wiadomości z konta `ludomir` na konto `helena`. W treści wiadomości należy wpisać:

```
Subject: Wiadomosc tetowa
Message-ID: 123456789.test@sk-server.pk.edu.pl
Wiadomosc do Heleny.
```

UWAGA: w systemie Windows należy użyć: sk-server.pk.edu.pl zamiast adresu IP podczas łączenia do serwera pocztowego, odpowiednie ustawienia zostały już wprowadzone w systemie by to umożliwić.

1.5. Używając tej samej aplikacji ponownie połączyć się z systemem pocztowym w celu odebrania wiadomości (połączyć się na konto `helena`). Połączyć się przez port 110 lub 143 (wybrać jeden z protokołów – POP3 lub IMAP).

1.6. Poczynić obserwacje dotyczące następujących zagadnień:

- Nagłówki wiadomości podczas odbierania poczty.
- Treści logów exim'a: `/var/log/exim4/mainlog`
- Co identyfikuje wiadomość w systemie pocztowym?

2. Konfiguracja protokołu IMAP i webmaila.

2.1. Utworzyć dodatkowe foldery w katalogu `Maildir` każdego użytkownika:

- `.Drafts` – szkice wiadomości
- `.Sent` – wiadomości wysłane
- `.Trash` – kosz
- `.Spam` – spam

Następnie należy stworzyć plik subskrypcjami do tych folderów dla courier'a.

2.2. Skonfigurować klienta e-mail (Windows) dla każdego z użytkowników, tak by do odbioru poczty używał protokołu IMAP i szyfrowania SSL.

2.3. Skonfigurować aplikację webmaila **Roundcube** w systemie Debian:

- ustawić domyślny host na: `sk-server.pk.edu.pl`
- zmienić nazwę wyświetlaną w pasku tytułu na: *Sieci Komputerowe 2010*
- wyłączyć moduł sprawdzania pisowni

- włączyć moduł edytora HTML

2.4. Zalogować się przez webmaila na konto `helena` i wysłać wiadomość na konto `ludomir`, następnie z konta `ludomir` wysłać wiadomość zwrotną za pomocą klienta pocztowego w systemie Windows.

2.5. Obserwacje.

- Porównać logi exim'a podczas wysyłania obydwu wiadomości.
- Co można powiedzieć o sposobie dostarczania wiadomości (klient webmail a klient na komputerze klienckim) na podstawie logów programu?

3. Filtrowanie wiadomości.

3.1. Ustawić w exim'ie każdemu z użytkowników limit wysyłanej wiadomości na 1 megabajt.

3.2. Spróbować wysłać wiadomość o rozmiarze większym niż 1 megabajt. Następnie sprawdzić logi exim'a.

3.3. Włączyć filtr antyspamowy. Włączyć 'uczenie się' *SpamAssassina*. Zintegrować exima ze SpamAssassinem. Wiadomość powinna być oznaczana jako spam w przypadku minimum 3.5 'punktów'. W przypadku gdy wiadomość 'uzbiera' więcej niż 12 tychże punktów wiadomość powinna być odrzucana podczas odbierania.

3.4. Wysłać 3 wiadomości z jednego konta na drugie, używając najpierw webmaila a następnie klienta pocztowe na systemie Windows, wg podanych informacji:

Nr.	Temat	Treść wiadomości	HTML
1.	Uważaj na dewelopera!	Chociaż realia się zmieniły i to deweloper szuka klienta, a nie klient dewelopera, to wciąż decydujemy się na zakup mieszkania na etapie dziury w ziemi – twierdzi UOKiK. Kupując mieszkanie na rynku pierwotnym przede wszystkim trzeba zwrócić uwagę – radzi UOKiK – na jej treść. Może być bowiem najeżona kruczkami prawnymi.	NIE
2.[1]			TAK
3.[2]	Test	XJS*C4JDBQADN1.NSBN3*2IDNEN*GTUBE-STANDARD-ANTI-UBE-TEST-EMAIL*C.34X	NIE

3.5. Podglądnąć logi exima i `/var/log/mail.log` po wysłaniu każdej z wiadomości.

3.6. Dokonać odpowiednich kroków tak by, dla obydwu użytkowników, wiadomości oznaczone jako spam trafiały do odpowiedniego folderu w skrzynce odbiorczej (folderu Spam).

Podpowiedź: należy użyć filtru lokalnego.

3.7. Dodać filtr, który będzie dodawał na końcu tematu napis: "(SPAM)", w przypadku gdy wiadomość oznaczona jest jako spam. Filtr powinien działać dla każdego użytkownika w systemie.

3.8. Obserwacje.

- Informacje w logach na temat przebiegu testu wiadomości przez filtr antyspamowy.

[1] Temat i treść mają być puste.
[2] Ciąg znaków w treści wiadomości bez żadnych przerw, w 1 linii.

- Nagłówki dodane do wiadomości.
- Jak działa odrzucanie wiadomości na poziomie SMTP?
- Ile punktów otrzymały wiadomości wysłane z webmaila? Dlaczego?
- Ile punktów dostają kolejne wiadomości w pkt. 3.4. i 'za co'?

4. Reguły filtru SpamAssassin.

4.1. Dodać odpowiednie reguły do SpamAssassin'a tak by wiadomości wysłane z webmaila zainstalowanego na systemie Debian, otrzymywały dodatkowo 1 punkt.

4.2. Dodać odpowiednie reguły do SpamAssassin'a, tak by wiadomości zawierające w temacie lub treści wiadomości adresy e-mail były oznaczane jako spam (trzeba dodać odpowiednią ilość punktów).[3]

4.3. Przetestować filtry wysyłając odpowiednie wiadomości.

4.4. Obserwacje.

A. Odnaleźć charakterystyczne dla działania filtru wpisy w logach (`/var/log/mail.log`).

B. Pytania

B.1. Na podstawie czego można śledzić wiadomość obiegającą różne systemy pocztowe?

B.2 Dlaczego plik `.forward` przenoszący wiadomości oznaczone jako spam do innego folderu został zastosowany lokalnie, a nie globalnie (filtr systemowy)?

5. Zadania dodatkowe.

5.1. Skonfigurować system pocztowy tak by wiadomości do odbiorców nie lokalnych były wysyłane za pomocą skrzynki pocztowej utworzonej w jakimś ogólnodostępnym, darmowym systemie pocztowym.

4. Podsumowanie

System pocztowy oparty MTA Exim i Courier cechuje kilka rzeczy. Po pierwsze Exim daje ogromne możliwości konfiguracyjne i administracyjne, wsparcie dla narzędzi antywirusowych (*ClamAV*) i antyspamowych (*SpamAssasin*). Courier cechuje się natomiast prostotą konfiguracji (m.in. dostarczana jest aplikacja *WebAdmin*, umożliwiająca konfigurację przez przeglądarkę) oraz obsługą wielu protokołów. Wybrana przez nas aplikacja Webmail – *RoundCube*, jest bodajże 'najprzyjaźniejszą' z dostępnych darmowych aplikacji tego typu, cechuję ją ładna szata graficzną, użycie *AJAX'a*, edytor HTML, sprawdzanie pisowni, itp.

W sprawozdaniu powinny znaleźć się: treść własnego zadania z zakresu protokołów i aplikacji poczty elektronicznej, podanie jego rozwiązania, a także wnioski płynące z wykonania ćwiczenia oraz ewentualne obserwacje. W sprawozdaniu należy też uwzględnić uwagi dotyczące tej instrukcji (tj. zauważone błędy i nieścisłości, propozycje zmian i poprawek) i systemu uruchomieniowego

5. Literatura dodatkowa

- Philip Hazel, *Specification of the Exim Mail Transfer Agent*, http://exim.org/exim-html-4.69/doc/html/spec_html/

[3] Należy zapoznać się z wyrażeniami regularnymi.

- Benjamin Mako Hill, David B. Harris, Jaldhar Vyas, *Debian GNU/Linux 3.1. Biblia*, Helion 2006 (*Rozdział 14: Serwery pocztowe*)
- *Courier mail server's Documentation*, http://www.courier-mta.org/documentation.html
- *SpamAssassin Documentation*, http://spamassassin.apache.org/full/3.3.x/doc/
- Jeffrey E. F. Friedl, *Wyrażenia regularne*, Helion 2001

Instrukcja opracowana przez:

Jacek Sokół
Paweł Przywara

Ćwiczenie: Poczta elektroniczna

Opieka merytoryczna:

dr inż. Piotr Andrzej Kowalski, dr inż. Szymon Łukasik, mgr inż. Sławomir Żak