Bezpieczeństwo poczty elektronicznej

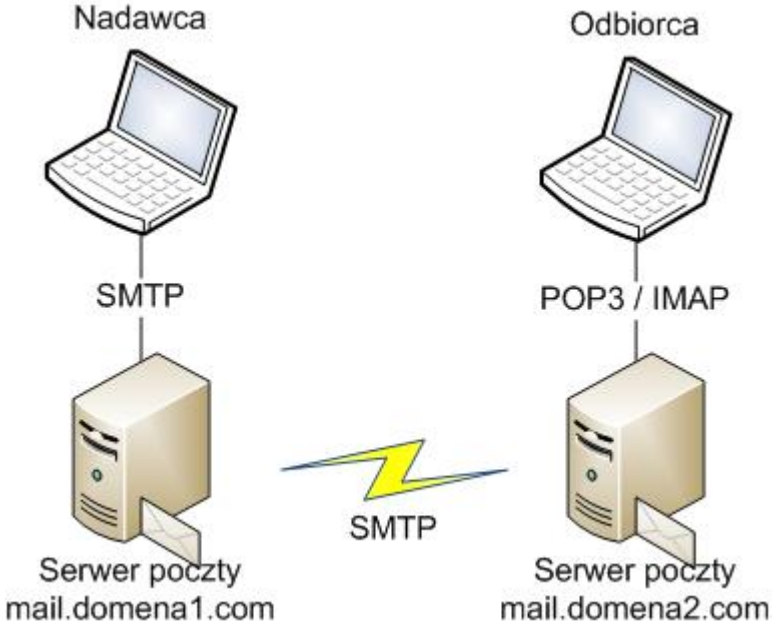
Mariusz Goch

Wydział Elektroniki i Technik Informacyjnych
Politechnika Warszawska

W aktualnych czasach bezpieczeństwo komunikacji stało się jednym z najistotniejszych czynników poprawnego funkcjonowania każdej instytucji. Najpowszechniejszą formą komunikacji jest poczta elektroniczna. W tym artykule postaram się przedstawić problemy bezpieczeństwa całej usługi oraz metody zapobiegawcze. W pierwszej części zaprezentuję jaką drogę pokonuje pojedyncza wiadomość e-mail, opiszę wykorzystywane do tego protokoły oraz zagrożenia z tym związane. Następnie opiszę różne metody zwiększania poziomu bezpieczeństwa serwera udostępniającego taką usługę, a na koniec poruszę temat walki z niechcianą pocztą.

1 Bezpieczeństwo transportu wiadomości

Wiadomość e-mail w zależności od konfiguracji serwerów pocztowych może podróżować pomiędzy nadawcą a odbiorcą w bardzo odmienny sposób. Podstawowy scenraiusz został zaprezentowany na poniższym rysunku.



Wiadomość jest tworzona za pośrednictwem dowolnego klienta pocztowego, który następnie za pomocą protokołu SMTP komunikuje się z serwerem pocztowym nadawcy i mu ją przekazuje. Ten z kolei wyszukuje w bazie DNS rekordy MX domeny odbiorcy. Rekordy te wskazują na serwery odpowiedzialne za obsługę poczty dla danej domeny. W ten sposób lokalizowany jest serwer pocztowy odbiorcy, z którym następnie jest realizowana komunikacja również za pomocą protokołu SMTP. Skutkiem tego wiadomość przekazana jest na serwer pocztowy odbiorcy. Aby móc ją odczytać odbiorca łączy się z nim przy użyciu jednego z dwóch protokołów POP3 lub IMAP i pobiera ją do swojego lokalnego programu pocztowego.

Przyjrzyjmy się teraz trochę bliżej wykorzystanym tutaj protokołom. SMTP czyli Simple Mail Transfer Protocol jest względnie prostym protokołem tekstowym. Jak pokazano wyżej służy on do wysyłania wiadomości oraz jej transportu pomiędzy serwerami pośredniczącymi. Ze względu na to że rekordy MX w DNSie przechowują tylko informacje o lokalizacji serwera, to usługa realizująca obsługę tego protokołu musi być uruchomiona na stałym porcie: 25. Poniżej zamieściłem przykładową komunikację.

```
220 test.com ESMTP Easy.Server
EHLO dom
250-test.com
250-PIPELINING
250-SIZE 20000000
250-ETRN
250-STARTTLS
250-AUTH LOGIN PLAIN
250-AUTH=LOGIN PLAIN
250-ENHANCEDSTATUSCODES
250-8BITMIME
250 DSN
AUTH LOGIN
334 VXNlcm5hbWU6                          [Username:]
cG9jenRhQHRlc3QuY29tCg==             [poczta@test.com]
334 UGFzc3dvcmQ6                          [Password:]
dGVzdAo=                                      [test]
235 2.0.0 Authentication successful
MAIL FROM:<poczta@test.com>
250 2.1.0 Ok
RCPT TO:<poczta@test2.com>
250 2.1.5 Ok
DATA
354 End data with <CR><LF>.<CR><LF>
Wiadomość
.
250 2.0.0 Ok: queued as 723F62090F1
```

```
QUIT
221 2.0.0 Bye
```

Na wstępie obie strony połączenia się wzajemnie przedstawiają. W tym wypadku serwer przedstawił się jako `test.com`, natomiast klient pocztowy jako `dom`. Następnie serwer wysyła do klienta informacje o swojej konfiguracji oraz rozszerzeniach protokołu jakie obsługuje. Najważniejsze informacje z powyższego przykładu to:j

- `SIZE 20000000` - maksymalna wielkość wiadomości w bajtach
- `STARTTLS` - możliwość włączenia szyfrowania komunikacji TLS
- `AUTH=LOGIN PLAIN DIGEST-MD5 CRAM-MD5` - dostępne metody uwierzytelniania

Po wymienieniu tych informacji zostało przeprowadzone uwierzytelnianie metodą `LOGIN`. Występuje ono tylko w sytuacji gdy skrzynka pocztowa nadawcy jest obsługiwana przez dany serwer, czyli przy komunikacji zainicjowanej przez klienta pocztowego. Dzięki temu tylko właściciel danej skrzynki jest uprawniony do wysyłania z niej wiadomości.

Po uwierzytelnieniu klient za pomocą dwóch komend `MAIL FROM` i `RCPT TO` przekazuje odpowiednio adres nadawcy i odbiorcy. Każdorazowo serwer wysyła odpowiedź czy akceptuje podany adres. Przykładowo gdyby etap uwierzytelniania został pominięty, natomiast podany adres nadawcy wskazywałby na ten serwer to zostałby zwrócony komunikat odmowny.

Następnie po komendzie `DATA` przesyłany jest właściwy kod wiadomości.

W powyższym przykładzie serwer poinformował klienta, że obsługuje cztery różne metody uwierzytelniania. Dwie pierwsze `PLAIN` i `LOGIN` są oparte na kodowaniu `base64`. Służy ono do zapisu ciągu bajtów za pomocą ciągu znaków i jest całkowicie odwracalne. W związku z tym obie metody nie dają żadnego zabezpieczenia przed podsłuchaniem transmisji. Różnią się one tylko realizacją przesyłania danych. W metodzie PLAIN wysyłany jest tylko jeden komunikat, który stanowi zakodowany ciąg: nazwa użytkownika, skrzynki oraz hasło, oddzielone znakiem o kodzie 0. W drugiej metodzie, jak było widać na przykładzie, dane te są przesyłane oddzielnie.

Pozostałe dwa rozwiązania `CRAM-MD5` i `DIGEST-MD5` są oparte na funkcji haszującej MD5, a dokładniej na kodowaniu HMAC (keyed-Hash Message Authentication Code). Kodowanie to nie jest odwracalne, dzięki czemu już zabezpiecza hasło przed podsłuchaniem. Pierwsza z tych metod uwierzytelnia jedynie klienta względem serwera. Druga natomiast również na odwrót, dzięki czemu zabezpiecza dodatkowo przed atakiem podszywania, czyli sytuacją gdyby komuś udało się zmodyfikować routing bądź wpisy w DNS i zmusić klienta pocztowego do połączenia z własnym serwerem.

Na podsłuch oprócz danych uwierzytelniających narażona jest również sama treść wiadomości. W tym celu zostało zaprojektowane specjalne rozszerzenie STARTTLS. W przypadku gdy obie strony je obsługują, inicjator połączenia

może wykonać komendę STARTTLS, która rozpoczyna komunikację szyfrowaną protokołem TLS (Transport Layer Security). Dodatek ten może być wykorzystywany zarówno przy połączeniach klienta pocztowego z serwerem, jak i pomiędzy serwerami.

Jak już wcześniej wspomniałem poza protokołem SMTP przy transporcie poczty wykorzystywane są również protokoły POP3 (Post Office Protocol version 3) lub IMAP (Internet Message Access Protocol). Do ich obsługi na serwerze uruchamiane są niezależne aplikacje. Standardowo nasłuchują one na portach:

POP3	110
IMAP	143

Poza tym zwykle uruchamia się je również w wersjach z obsługą szyfrowania połączeń poprzez SSL:

POP3S	995
IMAPS	993

Oba protokoły posiadają metody uwierzytelniania tak jak w przypadku SMTP. Oto podstawowe komendy protokołu POP3:

`AUTH`	wybór sposobu uwierzytelniania
`USER`	identyfikator użytkownika
`PASS`	hasło
`LIST`	pobierz listę wiadomości
`RETR`	pobierz wiadomość
`DELE`	usuń wiadomość
`QUIT`	koniec

2 Zabezpieczanie serwera pocztowego

Jednym z podstawowych problemów serwerów pocztowych jest open relay. Pod tym określeniem kryje się serwer niezabezpieczony przed nieautoryzowanym dostępem, a dokładniej jest możliwe wysłanie za jego pomocą nowej wiadomości bez uwierzytelnienia. Taki serwer w bardzo szybkim tempie może zostać wykorzystany do rozsyłania spamu, bądź próby podszycia pod inną osobą lub instytucję. Co więcej adres IP na którym on się znajduje w tym samym tempie może trafić na listy RBL, co może bardzo utrudnić, albo wręcz uniemożliwić wysyłanie wiadomości.

Problem ten jest oczywiście powiązany z protokołem SMTP, który w trakcie transportu wiadomości pomiędzy serwerami nie może wymagać uwierzytelnienia. W przypadku klasycznych konfiguracji z pojedynczym serwerem obsługującym domenę powinien zostać ustawiony wymóg uwierzytelnienia przy każdej próbie wysłania listu, którego nadawca znajduje się na liście lokalnych skrzynek pocztowych. Oczywiście cała sytuacja bardzo się komplikuje, jeśli do obsługi poczty wykorzystywana jest większa ilość serwerów pełniących różne funkcje.

Innym zagadnieniem jest kontrola poprawności podawanych adresów e-mail, czyli wartości pól `MAIL FROM` i `RCPT TO` protokołu SMTP. Oczywiście na wstępie należy sprawdzić czy adres ma poprawną składnię. Dodatkowo powinna nastąpić kontrola samej domeny, czyli sprawdzenie jej zgodności ze standardem FQDN - Fully Qualified Domain Name.

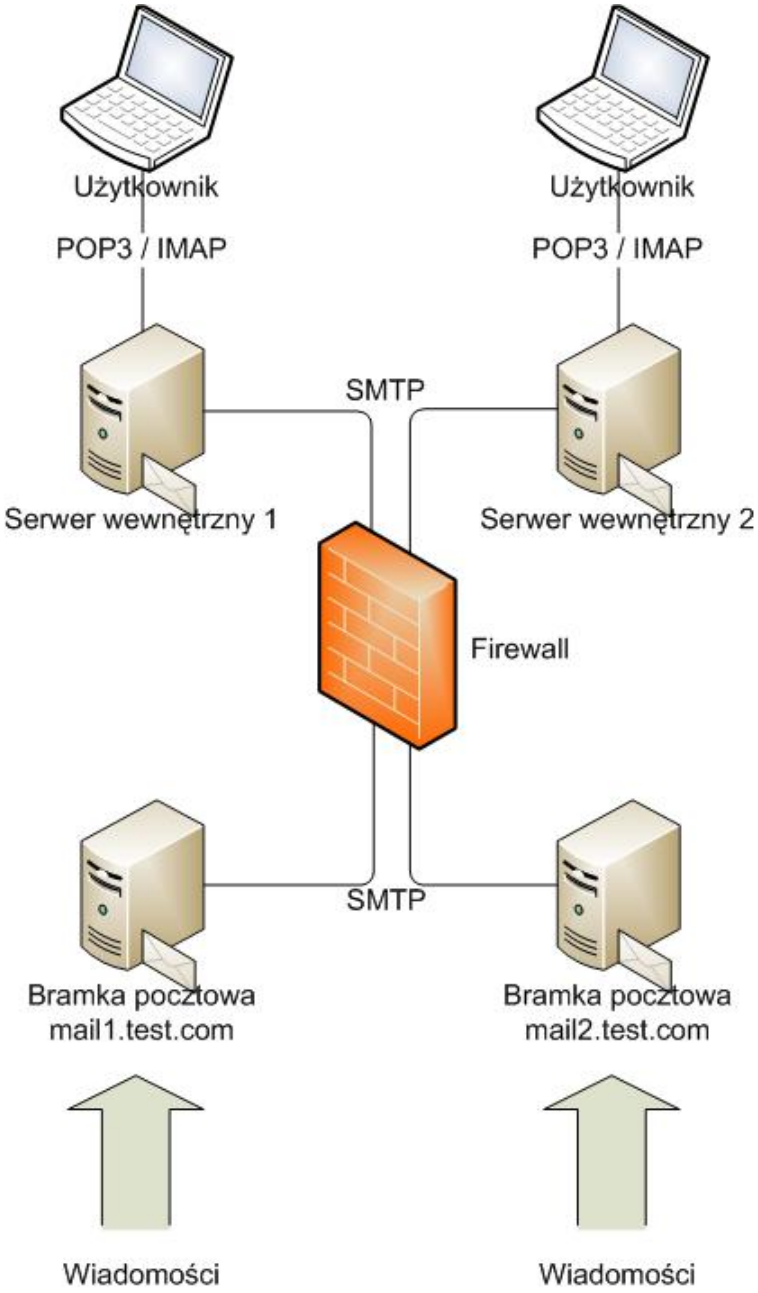
Kolejnym istotnym, choć rzadko implementowanym testem, jest kontrola zgodności adresu nadawcy i nazwy uwierzytelnionego użytkownika, oczywiście tylko w sytuacji gdy uwierzytelnienie miało miejsce. Jeżeli taka dodatkowa kontrola nie nastąpi, to każdy użytkownik będzie posiadał ukryte uprawnienie do wysłania wiadomości jako dowolny inny. Niestety tą lukę wciąż można znaleźć u wielu różnych dostawców usług hostingowych.

Czasami, w związku z zalewem niechcianej poczty, stosuje się również weryfikację adresu nadawcy w momencie odbioru wiadomości od innego serwera, a dokładniej w sytuacji gdy dana skrzynka nie jest obsługiwana lokalnie. W tym celu następuje próba wysłania wiadomości zwrotnej. Aby to zrealizować musi zostać zlokalizowany serwer nadawcy i nawiązane z nim połączenie. Następnie zgodnie z protokołem SMTP musi zostać podany adres nadawcy wiadomości testowej. Najczęściej jest tutaj stosowany adres administratora. Później jako adres odbiorcy podawany jest adres nadawcy oryginalnej wiadomości. Jeżeli serwer go zaakceptuje, to połączenie jest kończone, a oryginalna wiadomość przyjmowana. Niestety cała operacja jest dość kosztowna, w związku z tym stosuje się tutaj dodatkową pamięć podręczną w której przechowywane są zweryfikowane adresy.

Kolejnym problemem który chciałbym tutaj poruszyć jest odporność na awarie i przeciążenia serwerów. Podstawowym tutaj lekarstwem jest redundancja. Protokół SMTP dzięki swojej prostocie i elastyczności umożliwia jej zrealizowanie na wiele różnych sposobów.

Na początek możemy dodać do systemu zapasowe serwery SMTP. Możemy to zrealizować za pomocą wielu rekordów MX o różnych priorytetach. W ten sposób w przypadku awarii serwera głównego wiadomość zostanie przejęta przez zapasowy i umieszczona w kolejce w celu późniejszego dostarczenia. Niestety nie rozwiązuje to problemu dostępu do skrzynki w sytuacji przeciążenia, ponieważ wiadomości wciąż są składowane na serwerze podstawowym.

W związku z tym dla systemów o dużym obciążeniu można zbudować zwielokrotnione bramki pocztowe. Ich zadaniem byłoby tylko odbieranie wiadomości z zewnątrz i ich analiza pod kątem niechcianej poczty, co w gruncie rzeczy zabiera największą ilość zasobów w całym procesie. Następnie wiadomości byłyby przekazywane dalej na serwery odpowiedzialne już tylko za przechowywanie skrzynek i ich udostępnianie użytkownikom. Dodatkowo takie serwery mogą być umieszczone w lokalnej sieci firmowej bez dostępu z zewnątrz, co znacznie zwiększyłoby poziom bezpieczeństwa.



3 Niechciana poczta

Czym jest niechciana poczta? Otóż jest to tzw spam oraz wszelkiego rodzaju wirusy i exploity, czyli ogólnie wiadomości na których odbiór użytkownik nie wy-

raził zgody. Wszystkie wiadomości przechodzą przez serwer pocztowy odbiorcy i właśnie tutaj powinny zostać zweryfikowane. W tym celu instalowane są specjalne filtry antywirusowe oraz realizacje różnych koncepcji walki ze spamem.

Wiele system wykorzystuje system punktowy. Kolejno wykonywane są różnego rodzaju testy i na ich podstawie przydzielane są punkty. Jeśli ilość punktów przekroczy pewien próg, to z dużym prawdopodobieństwem dana wiadomość jest niepożądana. Najbardziej rozpowszechniona jest tutaj analiza sygnatur. Wiadomość jest sprawdzana pod kątem obecności pewnych słów kluczowych często wykorzystywanych w spamie. Odmiennym podejściem jest analiza heurystyczna. Użytkownik zaznacza wiadomości, które uznaje się za spam, a system na ich podstawie stara się wygenerować własną bazę sygnatur. Następnie nowe przesyłki są porównywane z tą bazą i określane jest prawdopodobieństwo że dana wiadomość jest niepożądana.

Oprócz wyżej wymienionych rozwiązań, coraz częściej wykorzystywane są bazy RBL - Realtime Blackhole Lists. Są to globalne bazy danych o hostach podejrzanych o rozsyłanie niechcianej poczty i administrowane przez różnego rodzaju instytucje walki ze spamem. Dane są tutaj udostępniane w przejrzystej formie rekordów DNS, natomiast zapytania są budowane jako reverse DNS. Przykładowo jeśli nadawcą wiadomości był host o adresie IP `a.b.c.d`, to wysyłane jest zapytanie `d.c.b.a.serwer.list.rbl`. Jeśli taki rekord będzie istniał to z dużym prawdopodobieństwem daną wiadomość można odrzucić.

Odmienne podejście prezentuje koncepcja greylisty. Poprawnie funkcjonujący system mailowy próbuje wysłać dostarczyć daną wiadomość przez określony okres czasu, natomiast serwery wykorzystywane do rozsyłania spamu robią to tylko raz. Dzieje się tak dlatego, że adresy mailowe spamerzy pozyskują w najróżniejszy sposób i nie ma pewności że faktycznie są one jeszcze aktualne, bądź w ogóle kiedykolwiek istniały. W związku z tym serwer pocztowy otrzymując nową wiadomość pobiera jej nagłówki i kończy połączenie. W przypadku gdy wysłanie tej wiadomości zostanie powtórzone, to zostanie ona zaakceptowana. Poważną wadą tej idei jest wprowadzenie opóźnienia w dostarczeniu wiadomości, które niestety jest często niedopuszczalne.

Istnieje jeszcze wiele różnych rozwiązań walki ze spamem, jednak nie ma jednego rozwiązania idealnego. Prawdziwe efekty można uzyskać dopiero łącząc wiele różnych metod w jeden spójny system.

Literatura

[1] Ralf Hildebrandt, Patrick Koetter "The Book of Postfix"