Poczta: najstarsza usługa internetu, która wciąż potrafi zepsuć piątek

Poziom: podstawy **Dotyczy: poczta elektroniczna, MailCow Dockerized**

Czas: ~15 min

Ta strona otwiera książkę o poczcie: skąd wzięła się poczta elektroniczna, dlaczego jest starsza niż sam internet w dzisiejszym rozumieniu, i dlaczego Zielona zamiast trzymać ją w Zimbrze, przenosi się na MailCow.

Skróty użyte na tej stronie (15)

Skrót	Co znaczy
DKIM	DomainKeys Identified Mail: cyfrowy podpis wiadomości e-mail
DMARC	Domain-based Message Authentication, Reporting and Conformance: reguła domeny mówiąca, co robić z mailem, który nie przeszedł SPF i DKIM
DNS	Domain Name System: książka telefoniczna internetu, zamienia nazwy na adresy IP
IBM	International Business Machines: firma, w której laboratorium powstał model relacyjny i pierwszy działający silnik SQL
IMAP	Internet Message Access Protocol: protokół, którym klient pocztowy pobiera wiadomości z serwera, zostawiając je tam zamiast kasować
IP	Internet Protocol: adres komputera w sieci; w elektryce: stopień ochrony obudowy (Ingress Protection)
IT	Information Technology: dział albo praca związana z komputerami i infrastrukturą firmy
LDAP	Lightweight Directory Access Protocol: protokół do odpytywania katalogu użytkowników (AD, FreeIPA)
MTA	Mail Transfer Agent: serwer przekazujący pocztę dalej, np. Postfix
PDF	Portable Document Format: format dokumentu, który wygląda tak samo na każdym komputerze, np. zakres akredytacji jednostki na stronie PCA
RFC	Request for Comments: oficjalny dokument opisujący standard internetowy, na przykład sposób działania danego protokołu; w ITIL także Request for Change, czyli wniosek o zmianę

Skrót	Co znaczy
SMTP	Simple Mail Transfer Protocol: protokół wysyłania poczty
SPF	Sender Policy Framework: rekord DNS mówiący, które serwery mogą wysyłać pocztę domeny
SSH	Secure Shell: szyfrowane zdalne logowanie do serwera
WWW	World Wide Web: strony i aplikacje otwierane w przeglądarce po HTTP albo HTTPS

Skąd się wzięło

> **Pojęcie: poczta elektroniczna (e-mail).** Sposób przesyłania wiadomości tekstowych między komputerami, starszy niż strony WWW, starszy niż sam internet w dzisiejszym znaczeniu tego słowa. Pierwsze systemy poczty elektronicznej działały już w latach sześćdziesiątych, ale tylko między użytkownikami tego samego komputera: wiadomość zostawało się w cudzym katalogu domowym na tej samej maszynie, nikt nigdzie jej nie wysyłał.

To, co dziś rozumiemy jako pocztę, czyli wiadomość wędrującą między różnymi komputerami w sieci, powstało w 1971 roku. Ray Tomlinson, inżynier pracujący dla firmy BBN nad siecią ARPANET (poprzedniczką internetu), połączył istniejący już program do poczty lokalnej z programem do przesyłania plików między maszynami. Potrzebował sposobu zapisania, do kogo i na jakim komputerze wysyła wiadomość, więc sięgnął po znak `@`, który w tamtym czasie służył głównie do oznaczania cen na klawiaturach maszyn do pisania i praktycznie nie pojawiał się w nazwach użytkowników. Adres w formacie `użytkownik@komputer` rozdzielał od razu dwie rzeczy: kto jest odbiorcą i gdzie go szukać. Sam Tomlinson po latach żartował, że nie pamięta treści pierwszej testowej wiadomości, prawdopodobnie był to ciąg losowych znaków z górnego rzędu klawiatury.

Przez kolejną dekadę każdy ośrodek akademicki pisał swój własny format przesyłania poczty, co czyniło komunikację między różnymi sieciami koszmarem. Porządek zaprowadził dopiero **Pojęcie: SMTP (Simple Mail Transfer Protocol).** Protokół tekstowy do przekazywania wiadomości między serwerami, opisany w 1982 roku przez Jona Postela w dokumencie RFC 821, obowiązujący (po wielu poprawkach) do dziś. Więcej o samym SMTP i drodze wiadomości piszemy na stronie Jak działa poczta: MTA, MDA, MUA, SMTP, kolejka.

Pierwszym powszechnie używanym programem realizującym SMTP po stronie serwera był Sendmail, napisany przez Erica Allmana z Uniwersytetu Kalifornijskiego w Berkeley. Sendmail zdominował internet na dwie dekady, ale zapłacił za to legendarnie nieczytelnym plikiem konfiguracyjnym: reguły trasowania poczty zapisywało się w składni tak gęstej, że administratorzy przechowywali gotowe fragmenty jak zaklęcia, nie do końca rozumiejąc, dlaczego działają. W 1998 roku Wietse Venema, holenderski badacz bezpieczeństwa pracujący wtedy dla IBM, opublikował

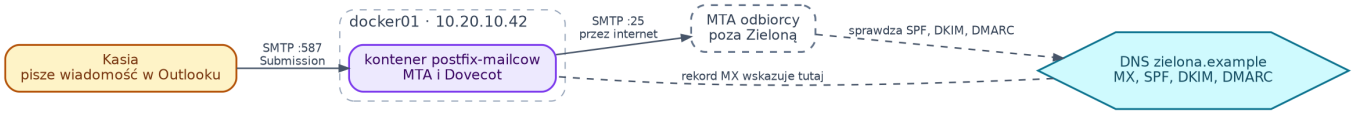
Postfixa: MTA zaprojektowanego od zera z myślą o bezpieczeństwie i czytelnej konfiguracji. Postfix, opisany szerzej na stronie Postfix: konfiguracja, do dziś jest jednym z najpopularniejszych silników pocztowych na świecie i stoi pod maską większości gotowych pakietów, łącznie z MailCow.

Wraz z rosnącym internetem rosła też skala nadużyć: pole `From:` można było przez dekady wpisać dowolnie, bez żadnej weryfikacji. Fala spamu i phishingu wymusiła powstanie trzech mechanizmów uwierzytelniających nadawcę: SPF (2000), DKIM (2004, połączenie dwóch konkurencyjnych propozycji Yahoo i Cisco) oraz DMARC (2012, spinający oba poprzednie). Opisujemy je szczegółowo na stronie Jak działa poczta: MTA, MDA, MUA, SMTP, kolejka: bez nich instalacja pocztowa trafia niemal automatycznie do spamu u odbiorców. Rekordy, które Zielona publikuje dla nich w strefie `zielona.example`, drogę DMARC do `p=reject` i szyfrowanie między serwerami opisuje osobna książka o domenie: strony SPF, DKIM i DMARC: trzy podejścia do poczty i systemy wysyłające w imieniu domeny, Droga DMARC od p=none do p=reject z raportami parsedmarc na docker01 i MTA-STS i TLS-RPT: szyfrowanie poczty między serwerami, a BIMI informacyjnie.

Prowadzenie własnego serwera pocztowego od zera, czyli ręczne złożenie Postfixa, Dovecota (serwera IMAP), filtra antyspamowego, webmaila i ochrony przed nadużyciami w jeden spójny system, jest projektem na tygodnie pracy i kolejne miesiące utrzymania. Stąd powstały gotowe pakiety pocztowe: komercyjna Zimbra Collaboration Suite (od 2005 roku, dziś rozwijana przez Synacor i Zextras) oraz otwartoźródłowy MailCow Dockerized, opisany krócej na stronie MailCow, łączący te same komponenty w zestaw kontenerów Docker uruchamianych jednym poleceniem.

Jak to działa w jednym obrazku

Wiadomość Kasi, wysłana z Outlooka do klienta spoza Zielonej, trafia najpierw do kontenera Postfixa wewnątrz MailCow na `docker01`, a stamtąd przez internet do serwera odbiorcy. Po drodze serwer odbiorcy sprawdza w DNS domeny `zielona.example` trzy rzeczy: czy adres IP, z którego przyszło połączenie, ma prawo wysyłać pocztę w imieniu tej domeny (SPF), czy podpis kryptograficzny w nagłówku się zgadza (DKIM) i co zrobić, jeśli któraś z tych dwóch kontroli zawiedzie (DMARC).



Droga wiadomości od Kasi przez MailCow na docker01 do serwera odbiorcy, z kontrolą DNS po drodze (kliknięcie otwiera pełny rozmiar)

Dlaczego Zielona

> **Z ŻYCIA WZIĘTE**
>
> Piątek, 15:50. Pani Halina wysyła do biura rachunkowego skan zamkniętego miesiąca z załączonym plikiem PDF. Mija dziesięć minut, telefon od księgowej: „nic nie doszło, sprawdziłam nawet spam". Tomek loguje się do starej Zimbry, przegląda kolejkę: wiadomość wisi z błędem `550 5.7.1 SPF check failed`, bo adres IP biurowego łącza od miesiąca nie jest już wpisany w rekordzie SPF domeny, odkąd dostawca internetu zmienił pulę adresów. Nikt tego nie zauważył, bo Zimbra nie ma żadnego czytelnego miejsca, w którym pokazałaby, że coś jest nie tak z autoryzacją domeny.

Zimbra działa w Zielonej od lat i działa poprawnie jako sam magazyn skrzynek, ale jej panel administracyjny rozdziela konfigurację MTA, LDAP i antyspamu na osobne narzędzia wiersza poleceń (`zmprov`, `zmcontrol`, edycję plików konfiguracyjnych ręcznie), opisane na stronie Zimbra. Diagnoza problemu takiego jak wyżej wymaga znajomości wewnętrznego, zmodyfikowanego Postfixa Zimbry i osobnych komend do każdego komponentu.

MailCow rozwiązuje ten sam zestaw problemów jednym webowym panelem administracyjnym: kolejka poczty, reguły antyspamowe silnika Rspamd, status DKIM i SPF danej domeny oraz dziennik zdarzeń każdej skrzynki widoczne są w jednym miejscu, bez logowania się po SSH. Do tego dochodzi mniejszy narzut utrzymania: aktualizacja to jeden skrypt (`update.sh`), a kopia zapasowa to jeden skrypt (`backup_and_restore.sh`), oba opisane na istniejącej stronie MailCow. To dlatego zespół IT Zielonej przenosi pocztę firmową z wygaszanej Zimbry na MailCow, zaczynając od instalacji opisanej na kolejnej stronie tej książki.

Mapa tej książki

Książka o poczcie łączy ogólną wiedzę o tym, jak działa poczta w ogóle, ze szczegółami dwóch konkretnych silników: Postfixa jako samodzielnego MTA i MailCow jako gotowego pakietu, którym Zielona zastępuje starą Zimbrę.

Strona	Na jakie pytanie odpowiada
Poczta: najstarsza usługa internetu, która wciąż potrafi zepsuć piątek	Skąd wzięła się poczta elektroniczna i dlaczego Zielona przechodzi z Zimbry na MailCow
Jak działa poczta: MTA, MDA, MUA, SMTP, kolejka	Kto jest kim na drodze wiadomości i co sprawdzają SPF, DKIM, DMARC
Postfix: konfiguracja	Jak skonfigurować Postfixa jako serwer wysyłający powiadomienia dalej

Strona	Na jakie pytanie odpowiada
Kolejka Postfiksa: podgląd, wymuszenie, czyszczenie	Jak sprawdzić, dlaczego wiadomość utknęła
Diagnostyka dostarczania	Jak czytać logi, gdy wiadomość nie dochodzi
MailCow	Jakie kontenery składają się na MailCow i jak wykonać kopię zapasową
Instalacja MailCow na docker01 od zera	Jak postawić MailCow od zera i podłączyć domenę `zielona.example`
MailCow: konta z Active Directory	Jak podłączyć logowanie do MailCow kontem domenowym
Zimbra	Jak obsłużyć starą pocztę, zanim zostanie wygaszona
MailCow: serwer pocztowy w kontenerach, który Zielona wybrała zamiast Zimbry	Osobna książka o MailCow: codzienna praca, kopia i aktualizacja oraz częste problemy

MailCow ma od niedawna własną książkę, którą otwiera strona z ostatniego wiersza tabeli: tam są codzienna praca, kopia zapasowa, aktualizacja i częste problemy.

Powiązane: Active Directory: konta, grupy, jednostki organizacyjne · Certyfikaty SSL: wystawienie, konwersja, weryfikacja · SPF, DKIM i DMARC: trzy podejścia do poczty i systemy wysyłające w imieniu domeny

Źródła: en.wikipedia.org/wiki/Ray_Tomlinson · en.wikipedia.org/wiki/Simple_Mail_Transfer_Protocol · en.wikipedia.org/wiki/Sendmail · en.wikipedia.org/wiki/Postfix_(software) · docs.mailcow.email

---

Revision #23
Created 2026-09-28 17:15:32 UTC by Tomasz Konieczny
Updated 2026-10-26 14:15:58 UTC by Tomasz Konieczny