Czytaj lub słuchaj artykuł online

Zeskanuj kod QR, aby otworzyć artykuł na XO.PL

Źródło artykułu:

https://xo.pl/it-na-wesolo-net-https

SIECI KOMPUTEROWE

HTTP i HTTPS, czyli od pocztówki do zamkniętej koperty

Dawno, dawno temu — a właściwie wcale nie tak dawno — ogromna część stron WWW działała bez **szyfrowania**. Nikt nie potrzebował hakera wyskakującego z szafy ani dramatycznej muzyki. Tak po prostu wyglądała codzienność Internetu. Żeby zrozumieć, co się zmieniło, odwiedźmy pewną pocztę, której kierownik kupił cztery miliony pocztówek.

Witamy na poczcie, w której wszystko jest na wierzchu

W miasteczku działała poczta. Mieszkańcy codziennie wysyłali wiadomości do znajomych, urzędów, sklepów i banków. Wszystko szło sprawnie. Był tylko jeden szczegół: **wszystkie wiadomości wysyłano na pocztówkach**. Bez kopert.

Klient podszedł do okienka.

— Dzień dobry, chciałbym wysłać wiadomość do banku.
— Oczywiście. Poproszę pocztówkę.
— Proszę.
— „Szanowni Państwo, moje hasło to...” Ooooo, ciekawe!
— PRZEPRASZAM, DLACZEGO PAN TO CZYTA?!
— Bo jest napisane.
— ALE TO PRYWATNA WIADOMOŚĆ!
— To dlaczego napisał ją pan na pocztówce?!
— BO INNYCH NIE MACIE!
— No właśnie. Następny proszę!

🤣🤣

Tak można sobie wyobrazić zwykły **HTTP — Hypertext Transfer Protocol**. Ten protokół pozwala przeglądarce poprosić serwer o stronę, obrazek czy inny zasób i odebrać odpowiedź. Tradycyjne połączenie HTTP nie szyfruje jednak treści transmisji. Ktoś mający dostęp do danych przesyłanych po drodze może je odczytać, a w pewnych sytuacjach także podmienić.

Oczywiście to nie znaczy, że *każdy człowiek na świecie* widzi każdą wiadomość. Tak samo nie każdy przypadkowy przechodzień dostaje do ręki wszystkie pocztówki. Problem polega na tym, że **pocztówkę można przeczytać bez otwierania czegokolwiek**.

Jak działa HTTP? Poproszę jedną stronę!

Przeglądarka podchodzi do okienka.

— Dzień dobry. Poproszę stronę główną.
— Jaką?
— `example.com`.
— Zaraz przyniosę.
— Dziękuję.
— Proszę. Oto strona.
— To wszystko?
— A czego pan jeszcze oczekiwał? Orkiestry?

W wielkim uproszczeniu HTTP działa właśnie tak: **żądanie i odpowiedź**. Przeglądarka mówi, czego potrzebuje, serwer odsyła odpowiedź. Nie trzeba rozumieć wszystkich technicznych szczegółów, żeby zrozumieć najważniejszą zasadę.

Na przykład przeglądarka może wysłać żądanie `GET /`, czyli „poproszę stronę główną", a serwer odpowiedzieć kodem `200 OK` i treścią strony. Same komendy HTTP nie są problemem — problemem w zwykłym HTTP jest brak **szyfrowania** połączenia.

A może zaczniemy używać kopert?

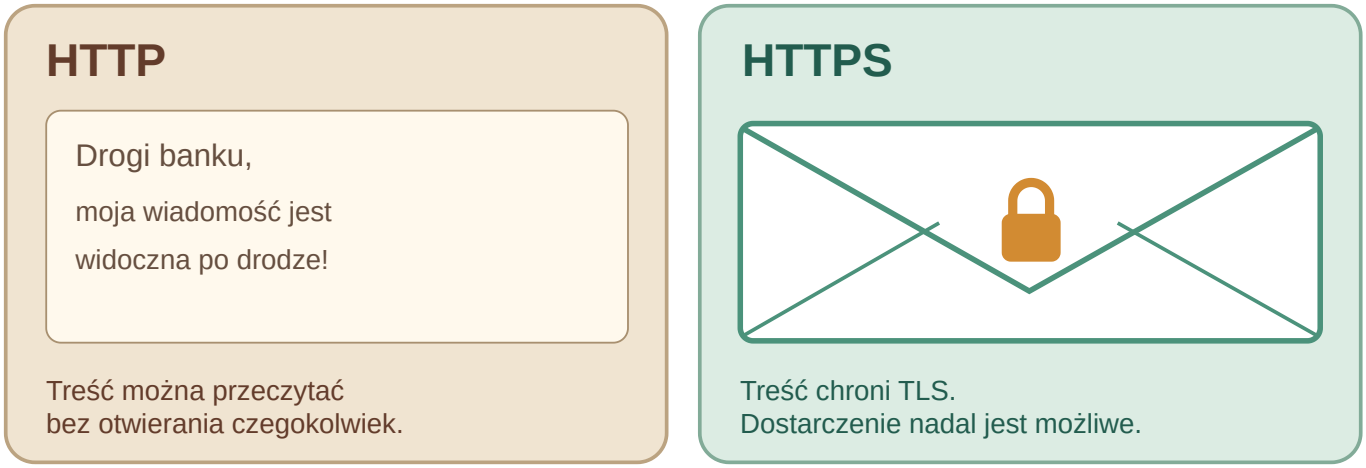
Pewnego dnia do urzędu przyjęto nowego pracownika.

— Panie kierowniku, mam propozycję.
— Słucham.
— Może zaczniemy wysyłać wiadomości w zamkniętych kopertach?
— W CZYM?!
— W kopertach. Żeby obcy nie mogli ich czytać.
— Ale od lat wysyłamy pocztówki!
— Wiem. Każdy, kto ma do nich dostęp, widzi treść.
— TO NIECH NIE CZYTA!
— Bezpieczeństwo nie może zależeć wyłącznie od cudzej uprzejmości.
— Nie wystarczy napisać „PROSZĘ NIE CZYTAĆ"?
— Nie.
— Nawet wielkimi literami?!
— Nawet.
— CHOLERA. TO TRZEBA KUPIĆ KOPERTY.

🤣🤣

Witamy w świecie **HTTPS — Hypertext Transfer Protocol Secure**. Najważniejsza wiadomość: **HTTPS nadal używa HTTP**! Przeglądarka nadal prosi, serwer nadal odpowiada. Zmienia się to, że komunikacja HTTP zostaje zabezpieczona przez **TLS**.

Dzięki TLS osoby po drodze nie powinny móc odczytać treści prawidłowo zabezpieczonej transmisji ani jej niezauważenie zmienić. To wielki krok naprzód, a nie dodatkowe utrudnienie wymyślone dla zabawy administratorów.



Dlaczego nie używaliśmy kopert od samego początku?

HTTPS wcale nie pojawił się wczoraj. Zabezpieczanie połączeń WWW zaczęło rozwijać się już w latach dziewięćdziesiątych. Jednak przez długi czas wiele zwykłych witryn korzystało z HTTP. HTTPS stosowano szczególnie chętnie przy logowaniu, płatnościach i bankowości.

Powody były praktyczne: **szyfrowanie** kosztowało więcej mocy obliczeniowej, zdobywanie i odnawianie **certyfikatów** wymagało pracy, a konfiguracja bywała trudniejsza niż dziś. Z czasem komputery stały się szybsze, oprogramowanie dojrzalsze, a **certyfikaty** i ich automatyczne odnawianie — łatwiej dostępne. Przeglądarki zaczęły również mocniej ostrzegać przed niezabezpieczonymi stronami.

Na poczcie tymczasem zwołano zebranie.

— Panie kierowniku, kolejne urzędy przechodzą na koperty!
— A co z naszym magazynem pocztówek?!
— Możemy je sprzedawać turystom.
— MAMY ICH CZTERY MILIONY!
— Może promocja?
— JAKA?!
— „Kup pocztówkę, dostaniesz kopertę gratis”.
— ZWALNIAM PANA!

🤣

Dziś HTTPS jest standardowym wyborem dla publicznych stron WWW. Nie oznacza to, że dawniej wszyscy byli nierozsądni. Technologia i oczekiwania dotyczące **bezpieczeństwa** po prostu się rozwinęły.

Koperta nie jest peleryną niewidką

Kierownik postanowił wypróbować nowy system. Włożył wiadomość do koperty i wręczył ją listonoszowi.

— Co jest w środku?
— Nie wiem.
— Doskonale! A dokąd pan ją niesie?
— Do banku.
— SKĄD PAN WIE?!
— Bo na kopercie jest adres.

— TO PROSZĘ GO ZAMAZAĆ!
— Wtedy nie będę wiedział, gdzie dostarczyć!
— NIE MOŻE PAN ZGADNĄĆ?!

🤣🤣

HTTPS chroni treść komunikacji HTTP, ale **nie ukrywa automatycznie wszystkich informacji o połączeniu**. Osoba obserwująca sieć może nadal znać **adresy IP**, porty, czas i przybliżoną ilość przesyłanych danych. W niektórych konfiguracjach widoczna może być także nazwa witryny, np. przez zapytania **DNS** lub informacje podczas zestawiania TLS. Istnieją techniki ograniczające część tych ujawnień, lecz sam napis HTTPS nie oznacza pełnej anonimowości.

Analogia jest prosta: koperta chroni list, ale poczta nadal musi wiedzieć, którędy przesłać przesyłkę.

Ktoś podmienił pocztówkę!

Klient zamówił z urzędu godziny otwarcia. Urząd napisał: „Pracujemy od 8:00 do 16:00”. Ktoś, kto miał możliwość ingerowania w drogę przesyłki, podmienił treść na: „Pracujemy od 3:00 do 4:00. Prosimy przynieść własne krzesło”.

O trzeciej nad ranem klient stanął pod budynkiem.

— DZIEŃ DOBRY! JESTEM!
— CZEGO PAN CHCE O TEJ PORZE?! — krzyknął ochroniarz.
— PRZYSZEDŁEM DO URZĘDU!
— O TRZECIEJ W NOCY?!
— TAK BYŁO NAPISANE!
— GDZIE?!
— NA POCZTÓWCE!
— PANIE KIEROWNIKU! KTOŚ ZNOWU PODMIENIA NASZĄ KORESPONDENCJĘ!

🤣🤣

HTTPS zapewnia nie tylko **poufność**, ale również **integralność** chronionej transmisji. To znaczy, że ktoś po drodze nie powinien móc zmienić wiadomości tak, aby odbiorca nie zauważył ingerencji. W HTTP bez TLS brak takiej ochrony na poziomie połączenia.

Jak rozpoznać właściwego odbiorcę?

Koperty działały znakomicie, aż pewnego dnia klient zadał niewygodne pytanie.

— Skąd wiem, że wysyłam wiadomość do prawdziwego banku?
— Bo na budynku napisano „BANK”.
— A jeśli oszust powiesił taki sam szyld?
— ...
— No właśnie.
— PANIE KIEROWNIKU! MUSIMY JESZCZE SPRAWDZAĆ, KTO ODBIERA KOPERTY!
— CZY W TYM URZĘDZIE NIGDY NIE BĘDZIE SPOKOJU?!

HTTPS obejmuje również sprawdzanie tożsamości serwera za pomocą mechanizmów TLS, w tym **certyfikatu**. Przeglądarka sprawdza między innymi, czy **certyfikat** pasuje do odwiedzanej nazwy domenowej i czy jest zaufany. Dzięki temu chronimy nie tylko zawartość rozmowy, lecz także ograniczamy ryzyko rozmowy z niewłaściwym serwerem.

Ale szczegóły **certyfikatów**, kluczy i urzędów certyfikacji to już osobna historia: *„**Certyfikaty** SSL/TLS, czyli skąd wiesz, z kim rozmawiasz?”*. Tam dopiero pojawią się dokumenty, pieczątki i urzędnicy z bardzo poważnymi minami.

HTTP, HTTPS, porty i przekierowania

— Panie kierowniku, gdzie kierować zwykłe pocztówki?
— Do okienka 80!
— A zabezpieczone koperty?
— Do 443!
— Dlaczego te numery?
— Bo tak się przyjęło.
— A jeśli ktoś ustawi inny numer?
— To będzie inne okienko. Ważne, żeby nadawca wiedział które.

Standardowo HTTP korzysta z portu **80**, a HTTPS z **443**, choć usługi mogą działać także na innych portach. Często po wpisaniu adresu `http://` serwer przekierowuje przeglądarkę na `https://`. To przydatne, ale dopiero po zestawieniu chronionego połączenia dane HTTP są zabezpieczone przez TLS; samo przekierowanie nie zamienia wcześniejszego, niezabezpieczonego żądania w tajną wiadomość.

Mechanizm **HSTS** może pomagać przeglądarce unikać kolejnych niezabezpieczonych połączeń z daną witryną. W przypadku wcześniej nieodwiedzanej witryny szczególną rolę może pełnić lista HSTS preload.

HTTP się nie skończył. Zmienił sposób podróżowania

— Czyli HTTP już nie istnieje? — pyta kierownik.
— Oczywiście, że istnieje.
— TO PO CO KUPILIŚMY KOPERTY?!
— Bo HTTPS to nadal HTTP, tylko przesyłany z ochroną TLS.
— A co z HTTP/2 i HTTP/3?
— To różne wersje sposobu wymiany żądań i odpowiedzi HTTP.
— CZY ONE TEŻ POTRZEBUJĄ POCZTÓWEK?!
— Nie, panie kierowniku. Naprawdę proszę już odłożyć ten katalog papierniczy.

HTTP nadal opisuje komunikację między przeglądarką a serwerem. Współczesne strony używają różnych wersji tego protokołu, m.in. HTTP/1.1, HTTP/2 i HTTP/3. W praktyce przeglądarki korzystają z nich przede wszystkim w zabezpieczonych połączeniach; HTTP/3 wykorzystuje QUIC z wbudowanym zabezpieczeniem opartym na TLS 1.3. To techniczny rozwój transportu, a nie powrót do niezabezpieczonych pocztówek.

Niezabezpieczony HTTP nadal może występować w testach, niektórych sieciach lokalnych lub między usługami, gdzie stosuje się inne zabezpieczenia. Przy publicznej stronie WWW HTTPS pozostaje właściwym punktem wyjścia.

Finał: ostatnia pocztówka

Po wielu miesiącach reorganizacji poczta działała jak należy. Wiadomości docierały, korespondencja była chroniona, a klientom nie trzeba już było przypominać, żeby nie wypisywali haseł na otwartych pocztówkach.

Kierownik postanowił uczcić sukces.

— Panowie! Wysyłamy wszystkim klientom zawiadomienie o uruchomieniu bezpiecznej korespondencji!
— Świetnie, panie kierowniku.
— Napisałem piękny tekst!
— Doskonale. Wysyłamy w kopertach?
— NIE! NA POCZTÓWKACH! MAMY ICH JESZCZE TRZY MILIONY DZIEWIĘĆSET TYSIĘCY!

🤣🤣🤣

Dyrekcja Poczty Internetowej informuje, że prywatnych wiadomości nie należy wysyłać na otwartych pocztówkach. Kierownik prosi jednak o wykorzystanie zapasów magazynowych przed końcem roku.