

Jak przekierować dodatkową domenę w dhosting

Instrukcja wdrożeniowa krok po kroku

Ta instrukcja pokazuje, jak skonfigurować dodatkową domenę tak, żeby bezpiecznie i na stałe prowadziła do głównego adresu strony.

Przykład: [twojadowena.com](#) → [twojadowena.pl](#)

W takim układzie **twojadowena.pl** jest głównym adresem strony. Pod **twojadowena.com** nie powstaje drugi WordPress ani druga kopia serwisu. Dodatkowa domena ma tylko skierować użytkownika do właściwego miejsca.

Cały proces wygląda tak: dodatkowa domena → pusta strona www → HTTPS → przekierowanie 301 → domena główna

Instrukcja powstała na podstawie konfiguracji wykonanej dla lukaspajak.pl i lukaspajak.com oraz problemów, które pojawiły się podczas wdrożenia.

Powiązany artykuł: techniczną część tego procesu rozwijam w artykule [Punkt wejścia. Jak ustawiłem główny adres strony](#). Opisuję tam, dlaczego mam dwie domeny, ale jeden główny adres, skąd pojawił się problem z aliasem i dlaczego przy przekierowaniu trzeba zadbać również o bezpieczne połączenie z domeną dodatkową.

1. Sprawdź i zaparkuj dodatkową domenę

Najpierw sprawdź, czy dodatkowa domena jest przypisana do Twojego konta hostingowego. W dPanelu przejdź do sekcji **Domeny** i odszukaj adres, który ma później przekierowywać użytkowników.

Jeżeli domeny nie ma na liście, wybierz **Dodaj** → **Zaparkuj domenę**, wpisz jej adres i zatwierdź.

Co oznacza parkowanie domeny? Domena jest adresem. Samo jej zarejestrowanie nie tworzy jeszcze miejsca na serwerze ani strony www. Zaparkowanie domeny oznacza powiązanie jej z konkretnym kontem hostingowym, żeby serwer wiedział, że ma ten adres obsługiwać.

Po wykonaniu tego kroku dodatkowa domena powinna być widoczna na Twoim koncie.

2. Zdecyduj, jak obsłużyć dodatkową domenę

Skoro dodatkowy adres ma tylko prowadzić do istniejącej strony, naturalnym rozwiązaniem może wydawać się **alias www**.

Co to jest alias www? Alias www to dodatkowy adres przypisany do istniejącej strony. Nie tworzy osobnej witryny ani własnej zawartości. Pozwala korzystać z innej domeny bez budowania pod nią drugiego serwisu.

W najprostszym ujęciu wygląda to tak: dodatkowa domena → istniejąca strona

Problem pojawia się wtedy, gdy dodatkowa domena ma działać również przez HTTPS.

Po co jest certyfikat bezpieczeństwa? Certyfikat bezpieczeństwa pozwala przeglądarce zweryfikować domenę i zestawień szyfrowane połączenie z serwerem. Dzięki temu strona może prawidłowo działać pod adresem zaczynającym się od **https://**.

HTTPS chroni dane przesyłane pomiędzy przeglądarką a serwerem. Ma znaczenie nie tylko przy formularzach, logowaniu czy płatnościach. Jest potrzebny również wtedy, gdy domena służy wyłącznie do przekierowania.

Jeżeli użytkownik wpisze **https://twojadomena.com**, przeglądarka najpierw musi połączyć się z domeną dodatkową. Dopiero później serwer może odpowiedzieć, że użytkownik powinien przejść pod **https://twojadomena.pl**.

Kolejność wygląda więc tak: twojadomena.com → bezpieczne połączenie HTTPS → przekierowanie → twojadomena.pl

Jeżeli pierwszy adres nie ma prawidłowego certyfikatu, użytkownik może zobaczyć ostrzeżenie dotyczące bezpieczeństwa jeszcze przed wykonaniem przekierowania.

Jakiego certyfikatu potrzebujesz? Do takiej konfiguracji wystarczy certyfikat typu **DV (Domain Validation)**, czyli certyfikat potwierdzający kontrolę nad domeną. Bezpłatny certyfikat Let's Encrypt jest właśnie takim rozwiązaniem i do zwykłej strony lub domeny przekierowującej jest wystarczający.

Istnieją również certyfikaty płatne, które mogą obejmować dodatkową weryfikację organizacji, rozszerzone warunki gwarancji lub inne usługi. Przy prostym przekierowaniu dodatkowej domeny nie są potrzebne.

Dlaczego nie używamy aliasu www? W tej konfiguracji potrzebujemy nie tylko dodatkowego adresu, ale również możliwości uruchomienia dla niego HTTPS, dostępu do własnego katalogu i zapisania reguły przekierowania. Dlatego zamiast aliasu tworzymy **pustą stronę www**.

3. Dodaj pustą stronę www i ustaw jej parametry

Co to jest pusta strona www? Pusta strona www tworzy dla domeny osobne miejsce na serwerze, ale nie instaluje żadnego systemu CMS.

CMS (Content Management System), czyli systemu zarządzania treścią. Pozwala tworzyć i edytować stronę bez ręcznego przygotowywania całego kodu. WordPress i Joomla są przykładami CMS-ów.

Tutaj CMS nie jest potrzebny, ponieważ pod dodatkową domeną nie będziemy publikować żadnych treści. Potrzebujemy natomiast własnego katalogu, certyfikatu bezpieczeństwa oraz dostępu do plików konfiguracyjnych.

W dPanelu przejdź do **Strony www** → **Dodaj** → **Pusta strona www** i wskaż dodatkową domenę.

Ustaw parametry pustej strony:

- **Wersja PHP** - wybierz wersję oznaczoną przez hosting jako zalecaną. W przypadku samego przekierowania PHP praktycznie nie będzie wykorzystywane, ale strona www wymaga kompletnej konfiguracji środowiska.
- **Wildcard** - wyłącz, jeżeli nie potrzebujesz obsługi nieistniejących subdomen. Wildcard powoduje, że ruch kierowany do nieistniejących subdomen może trafiać do tego samego konta www. Przy prostym przekierowaniu jednej domeny nie jest potrzebny.
- **Open_basedir** - pozostaw ochronę włączoną. Funkcja ogranicza dostęp skryptów do określonej przestrzeni na serwerze.
- **Konto FTP** - włącz utworzenie konta FTP i ustaw dla niego hasło. Konto będzie potrzebne, żeby wejść do katalogu dodatkowej domeny i utworzyć plik **.htaccess**.
- **Baza MySQL** - nie twórz bazy danych. Baza jest potrzebna np. WordPressowi, ale domena służąca wyłącznie do przekierowania nie zapisuje treści ani danych aplikacji.
- **Opis strony www** - możesz pozostawić pusty.

Po zapisaniu konfiguracji poczekaj, aż pusta strona pojawi się na liście **Strony www**.

4. Uruchom certyfikat bezpieczeństwa

Dodatkowa domena ma już własną konfigurację na serwerze. Teraz trzeba uruchomić dla niej HTTPS.

Otwórz utworzoną stronę www, przejdź do ustawień bezpieczeństwa i aktywuj bezpłatny certyfikat Let's Encrypt. Poczekaj, aż jego status zmieni się na aktywny.

Nie ustawiaj jeszcze przekierowania. Najpierw otwórz w przeglądarce **<https://twojadomena.com>**.

Na tym etapie możesz zobaczyć pustą stronę. To prawidłowe, ponieważ nie instalowaliśmy tam WordPressa ani żadnej treści. Najważniejsze jest to, żeby przeglądarka nie wyświetlała błędu certyfikatu ani ostrzeżenia o niezabezpieczonym połączeniu.

Dopiero po poprawnym uruchomieniu HTTPS przejdź do ustawienia przekierowania.

5. Ustaw stałe przekierowanie 301

Domena działa już przez HTTPS, ale nadal nie ma informacji, dokąd powinna skierować użytkownika. Teraz ustawimy właściwe przekierowanie.

Co to jest przekierowanie? Przekierowanie informuje przeglądarkę, że po wejściu pod jeden adres ma automatycznie przejść pod inny.

Przykład: `twojadena.com` → `twojadena.pl`

Użytkownik nie musi niczego klikać. Serwer przekazuje przeglądarce nowy adres, a przeglądarka przechodzi do niego automatycznie.

Dlaczego 301? Kod 301 oznacza przekierowanie stałe. Informuje przeglądarki i wyszukiwarki, że jeden adres został trwale przeniesiony pod inny. Ponieważ dodatkowa domena ma stale prowadzić do domeny głównej, używamy właśnie 301.

Gdzie zapisujemy przekierowanie? Regułę przekierowania zapiszemy w pliku **.htaccess**. Jest to plik konfiguracyjny serwera Apache, w którym można zapisywać m.in. reguły dotyczące adresów, dostępu i przekierowań.

Żeby dostać się do tego pliku, potrzebujemy dostępu FTP.

Co to jest FTP? FTP to skrót od File Transfer Protocol. Jest to sposób przesyłania i zarządzania plikami znajdującymi się na serwerze. Dzięki FTP możesz z własnego komputera wejść do katalogu utworzonej wcześniej pustej strony www i utworzyć w nim plik **.htaccess**.

Do połączenia możesz wykorzystać program FileZilla.

Jak połączyć się przez FTP? Dane do połączenia znajdziesz przy koncie FTP utworzonym wcześniej dla pustej strony. Potrzebujesz hosta, loginu, hasła oraz portu, jeżeli jest podany.

W FileZilli przejdź do **Plik** → **Menedżer stron** → **Nowa strona**, wpisz dane dostępowe, a w **Ustawieniach transferu** wybierz tryb **Pasywny**. Następnie kliknij **Połącz**.

Po prawej stronie programu zobaczysz katalogi znajdujące się na serwerze. Mogą pojawić się m.in. **awstats** oraz **public_html**. Interesuje Cię **public_html**.

Co to jest public_html? **public_html** to główny katalog publiczny danej strony www. Serwer korzysta z jego zawartości podczas obsługi domeny.

W przypadku dodatkowej domeny nie będziemy umieszczać tam strony. Potrzebujemy tego katalogu, żeby zapisać konfigurację przekierowania.

Jeżeli przy pierwszej próbie nie możesz wejść do **public_html**, rozłącz FileZillę, zamknij program, uruchom go ponownie i jeszcze raz nawiąż połączenie.

Jak utworzyć plik .htaccess? Jeżeli **.htaccess** już istnieje, przed zmianą zrób jego kopię. Jeżeli go nie ma, będąc w katalogu **public_html**, kliknij prawym przyciskiem myszy, wybierz **Utwórz nowy plik** i wpisz dokładnie **.htaccess**. Następnie kliknij utworzony plik prawym przyciskiem i wybierz **Podgląd/Edycja**.

Kropka na początku jest częścią nazwy. Plik nie powinien nazywać się **.htaccess.txt**. Utworzenie go bezpośrednio na serwerze ogranicza ryzyko, że edytor tekstowy automatycznie dopisze rozszerzenie.

Jaką regułę przekierowania dodać? Dla przekierowania **twojadomena.com** → **twojadomena.pl** użyj:

```
RewriteEngine On
RewriteCond %{HTTP_HOST} ^(www\.)?twojadomena\.com$ [NC]
RewriteRule ^(.*)$ https://twojadomena.pl/$1 [R=301,L]
```

W tym przykładzie:

- **twojadomena.com** to domena dodatkowa, z której ma następować przekierowanie,
- **twojadomena.pl** to domena główna, do której ma trafić użytkownik.

Jeżeli Twoje domeny mają inne końcówki, wpisz własne adresy, np. **mojadomena.eu** → **mojadomena.pl** albo **mojadomena.net** → **mojadomena.com**.

Co robi ta reguła? **RewriteEngine On** uruchamia mechanizm odpowiedzialny za przepisywanie adresów.

RewriteCond %{HTTP_HOST} ^(www\.)?twojadomena\.com\$ [NC] sprawdza, czy zapytanie dotyczy domeny dodatkowej, z www albo bez niego.

RewriteRule ^(.*)\$ https://twojadomena.pl/\$1 [R=301,L] przekierowuje użytkownika do domeny głównej.

R=301 oznacza stałe przekierowanie. **L** kończy dalsze przetwarzanie reguł po zastosowaniu tej. **\$1** zachowuje część adresu znajdującą się po nazwie domeny.

Dzięki temu **twojadomena.com/kontakt** → **twojadomena.pl/kontakt**, zamiast **twojadomena.com/kontakt** → **twojadomena.pl**.

Jak zapisać zmiany? Po wpisaniu reguły zapisz plik i zamknij edytor. Jeżeli FileZilla zapyta o przesłanie zmienionego pliku z powrotem na serwer, potwierdź. Na końcu sprawdź, czy **.htaccess** znajduje się w **public_html** dodatkowej domeny.

6. Sprawdź działanie przekierowania

Najpierw wykonaj prosty test w prywatnym oknie przeglądarki. Wpisz **https://twojadomena.com**. Powinieneś zostać automatycznie przeniesiony do **https://twojadomena.pl**.

Sprawdź również **<http://twojadomena.com>**. Ostatecznie użytkownik powinien trafić pod bezpieczny adres domeny głównej rozpoczynający się od **<https://>**.

Jeżeli chcesz obsługiwać również wariant z **www** i został on poprawnie skonfigurowany, przetestuj także **<http://www.twojadomena.com>** oraz **<https://www.twojadomena.com>**. Oba adresy powinny prowadzić do właściwej domeny głównej.

Jak sprawdzić przekierowanie podstrony? Wpisz np. **<https://twojadomena.com/kontakt>**. Prawidłowy wynik to **<https://twojadomena.pl/kontakt>**.

Ten test potwierdza, że reguła przekierowania zachowuje część adresu znajdującą się po nazwie domeny, zamiast kierować każdą podstronę wyłącznie na stronę główną.

7. Sprawdź kod 301 w narzędziach developerskich

Samo przejście pod właściwy adres potwierdza, że przekierowanie działa. Warto jednak sprawdzić, czy serwer rzeczywiście zwraca kod **301**.

W Google Chrome otwórz prywatne okno przeglądarki. Następnie kliknij prawym przyciskiem myszy na stronie i wybierz **Zbadaj**. Możesz też użyć skrótu **F12** albo **Ctrl + Shift + I**.

W narzędziach deweloperskich przejdź do zakładki **Sieć**.

Na górnym pasku tej zakładki zaznacz:

- **Zachowaj dziennik** - dzięki temu pierwszy adres pozostanie widoczny także po wykonaniu przekierowania,
- **Wyłącz pamięć podręczną** - dzięki temu Chrome nie będzie korzystał z zapisanej wcześniej wersji strony podczas testu.

Następnie wpisz w pasku adresu **<https://twojadomena.com>** i naciśnij Enter.

Na liście żądań w zakładce **Sieć** odśledź domenę dodatkową i kliknij ją. Po prawej stronie otwórz sekcję **Nagłówki**.

Gdzie sprawdzić kod 301? W części **Ogólne** znajdź pozycję **Kod stanu**. Prawidłowy wynik to **301 Moved Permanently**.

Gdzie sprawdzić adres docelowy? W nagłówkach odpowiedzi znajdź pozycję **Location**. Powinna wskazywać **<https://twojadomena.pl/>**.

Jeżeli testujesz podstronę, sprawdź również, czy zachowana została dalsza część adresu. **<https://twojadomena.com/kontakt>** powinno prowadzić do **<https://twojadomena.pl/kontakt>**.

Jak sprawdzić, czy nie ma zbędnego łańcucha? Najprostsza droga dla wejścia przez HTTPS wygląda tak: **<https://twojadomena.com>** → **301** → **<https://twojadomena.pl/>**.

Jeżeli w zakładce **Sieć** pojawia się kilka kolejnych przekierowań, np.

http://twojadena.com → **https://twojadena.com** →

https://www.twojadena.com → **https://twojadena.pl**, warto sprawdzić, czy konfiguracji nie da się uprościć.

8. Co zrobić, jeżeli coś nie działa

- **Domena pokazuje błąd HTTPS** - sprawdź, czy certyfikat dodatkowej domeny jest już aktywny. Przeglądarka musi najpierw prawidłowo połączyć się z domeną źródłową, a dopiero później może wykonać przekierowanie.
- **Nie możesz połączyć się przez FTP** - sprawdź host, login, hasło i dane konta przypisanego do właściwej strony www. Upewnij się również, że konto FTP zostało utworzone podczas konfiguracji pustej strony.
- **Nie możesz wejść do public_html** - rozłącz FileZillę, zamknij program, uruchom go ponownie i ponów połączenie.
- **.htaccess nie działa** - sprawdź, czy plik nazywa się dokładnie **.htaccess**, nie ma rozszerzenia **.txt**, znajduje się we właściwym **public_html**, zawiera prawidłową domenę źródłową i docelową oraz nie koliduje z innymi regułami.
- **Powstaje pętla przekierowań** - sprawdź, czy domena docelowa nie ma reguły kierującej z powrotem do domeny źródłowej.
- **Nie działa www** - fragment (**www.**)? pozwala regule uwzględnić wariant www, ale sam nie tworzy subdomeny ani nie zapewnia jej prawidłowej konfiguracji. Jeżeli chcesz korzystać z www, ten wariant również musi być poprawnie skonfigurowany i obsługiwany przez certyfikat.

9. Kontrola końcowa

Po zakończeniu sprawdź:

- ☐ dodatkowa domena jest zaparkowana,
- ☐ została utworzona jako pusta strona www,
- ☐ nie ma na niej niepotrzebnego CMS-a,
- ☐ nie utworzono zbędnej bazy MySQL,
- ☐ konto FTP zostało utworzone i działa,
- ☐ możesz wejść do **public_html**,
- ☐ certyfikat bezpieczeństwa jest aktywny,
- ☐ HTTPS działa bez ostrzeżeń,
- ☐ **.htaccess** znajduje się we właściwym katalogu,
- ☐ reguła zawiera prawidłową domenę źródłową,
- ☐ reguła wskazuje właściwą domenę główną,
- ☐ przekierowanie działa w prywatnym oknie,
- ☐ serwer zwraca kod **301**,
- ☐ nagłówek **Location** wskazuje właściwy adres,
- ☐ podstrony zachowują swoją ścieżkę,
- ☐ HTTP prowadzi ostatecznie do HTTPS,

- ☐ www działa, jeżeli ma być obsługiwane,
- ☐ nie występuje pętla ani zbędny łańcuch przekierowań.

Efekt końcowy

Po prawidłowej konfiguracji: **https://twojadena.com → 301 → https://twojadena.pl**

Dodatkowa domena ma własne bezpieczne połączenie, ale nie tworzy drugiej wersji serwisu. Jej zadaniem jest skierować użytkownika pod jeden główny adres.

Chcesz poznać kontekst tej konfiguracji? O tym, dlaczego zdecydowałem się zachować dwa naturalne warianty domeny, ale tylko jeden główny adres, piszę szerzej w artykule **Punkt wejścia. Jak ustawiłem główny adres strony.**

[\[Przejdź do artykułu\]](#)