**0) Plan w skrócie**

1. Na komputerze **z internetem** budujesz package-registry.exe i pobierasz paczki integracji (ZIP).
2. Przenosisz to na **Windows Server offline**.
3. Uruchamiasz **EPR** z lokalnego katalogu paczek (jako usługę Windows).
4. Uruchamiasz **Kibanę** z **flagami CLI**: --xpack.fleet.registryUrl i --xpack.fleet.isAirGapped.

**1) Host z internetem — budowa binarki i zebranie paczek**

**1.1 Zbuduj package-registry.exe**

W PowerShell (Windows 10/11 lub Windows Server z dostępem do sieci):

# Wymagane: Git + Go (zgodna wersja z repo) + narzędzie Mage

git clone https://github.com/elastic/package-registry.git

cd package-registry

# Zainstaluj Mage (jednorazowo):

go install github.com/magefile/mage@latest

$env:PATH += ";$([Environment]::GetFolderPath('UserProfile'))\go\bin"

# Build (single-binary):

mage build

# => powstanie .\package-registry.exe

**1.2 Pobierz paczki integracji (ZIP) zgodne z Twoją Kibana/Stack**

Przykład dla Kibany 8.13.2 (zmień wersję według potrzeb):

# Sprawdź dostępne wersje paczki "system" (pod 8.13.2):

Invoke-WebRequest "https://epr.elastic.co/search?package=system&all=true&kibana.version=8.13.2" -OutFile system.json

# Pobierz konkretną wersję ZIP:

Invoke-WebRequest "https://epr.elastic.co/epr/system/system-1.53.0.zip" -OutFile "system-1.53.0.zip"

Tip: przygotuj listę paczek, które faktycznie będziesz używać (system, endpoint, elastic\_agent, windows, iis, nginx, postgres, itp.) i pobierz ich wersje kompatybilne z Twoją Kibana.

**1.3 Przenieś offline**

Skopiuj na pendrive / dysk zewnętrzny:

* package-registry.exe
* wszystkie pobrane \*.zip paczek

**2) Windows Server (offline) — katalogi i konfiguracja EPR**

**2.1 Utwórz strukturę i wgraj pliki**

New-Item -Type Directory C:\EPR\bin, C:\EPR\packages, C:\EPR\config | Out-Null

Copy-Item X:\package-registry.exe C:\EPR\bin\

Copy-Item X:\\*.zip C:\EPR\packages\

*(Zastąp X:\ literą Twojego pendrive’a.)*

**2.2 config.yml (EPR czyta paczki z dysku)**

Utwórz C:\EPR\config\config.yml:

server:

address: "0.0.0.0:8080" # adres/port nasłuchu HTTP

storage:

filesystem:

paths:

- "C:/EPR/packages" # używaj / w ścieżkach (zalecane)

cache:

max\_age: "1h" # opcjonalnie: nagłówki cache

EPR potrafi indeksować zarówno ZIP-y, jak i rozpakowane foldery paczek umieszczone w ścieżkach z listy paths:.

**3) Uruchomienie EPR (Windows Service)**

**3.1 Test na szybko (ręcznie)**

& "C:\EPR\bin\package-registry.exe" -config "C:\EPR\config\config.yml" -log-level info

W innym oknie:

(Invoke-WebRequest "http://127.0.0.1:8080/health").StatusCode # oczekiwane: 200

Invoke-WebRequest "http://127.0.0.1:8080/search?all=true" | Select-Object -ExpandProperty Content

Przerwij Ctrl+C po teście.

**3.2 Rejestracja jako usługa**

Najprościej przez sc.exe:

sc.exe create EPR binPath= "\"C:\EPR\bin\package-registry.exe\" -config \"C:\EPR\config\config.yml\" -log-level info" start= auto DisplayName= "Elastic Package Registry"

sc.exe start EPR

Uwaga: w sc.exe **musi** być spacja po = (to nie literówka).  
Jeśli chcesz uruchamiać na innym porcie/adresie, dodaj np. -address "0.0.0.0:8080" do binPath.

**3.3 Zapora (jeśli włączona)**

netsh advfirewall firewall add rule name="EPR 8080" dir=in action=allow protocol=TCP localport=8080

**3.4 (Opcjonalnie) TLS dla EPR**

Jeżeli potrzebujesz HTTPS:

1. Umieść certyfikat/klucz w C:\EPR\tls\ (CN/SAN zgodny z nazwą hosta).
2. Zmień usługę tak, by uruchamiała EPR z parametrami TLS (flagi zależą od wersji — sprawdź package-registry.exe --help; najczęściej są to opcje w rodzaju -tls-cert i -tls-key).
3. Otwórz port (np. 8443 lub 443) w zaporze.

*(Jeśli używasz prywatnego CA, do Kibany dodasz zaufanie przez NODE\_EXTRA\_CA\_CERTS — patrz sekcja 4.2.)*

**4) Kibana bez edycji kibana.yml — podejście 4A (flagi CLI)**

Założenie: Kibana jest uruchamiana jako **usługa Windows**.

**4.1 Ustawienie EPR i trybu air-gapped flagami CLI**

Zmień ścieżkę uruchomienia usługi Kibana, dopisując dwie flagi:

# Przykład: instalacja ZIP Kibany i standardowa usługa "Kibana"

sc.exe stop Kibana

sc.exe config Kibana binPath= "\"C:\kibana\bin\kibana.bat\" --xpack.fleet.registryUrl=http://<HOST\_EPR>:8080 --xpack.fleet.isAirGapped=true"

sc.exe start Kibana

* --xpack.fleet.registryUrl=http://<HOST\_EPR>:8080 — Kibana/Fleet korzystają z Twojego EPR.
* --xpack.fleet.isAirGapped=true — ogranicza oczekiwane połączenia na zewnątrz (tryb offline).

Jeśli Kibana była zainstalowana jako usługa w innej lokalizacji/nazwie, podmień ścieżki/nazwę usługi. Gdy masz wrapper NSSM lub inny menedżer, dodaj flagi CLI w jego konfiguracji startowej.

**4.2 (Opcjonalnie) Zaufanie do prywatnego CA**

Jeżeli EPR działa na HTTPS z certyfikatem z **prywatnego CA**, dodaj zmienną środowiskową **systemową** dla procesu Kibany:

setx NODE\_EXTRA\_CA\_CERTS "C:\kibana\certs\ca-cert.pem" /M

sc.exe stop Kibana

sc.exe start Kibana

Dzięki temu warstwa Node.js, na której działa Kibana, będzie ufała Twojemu CA przy łączeniu do EPR.

**5) Fleet — źródła binariów Agenta (ważne w offline)**

EPR to **tylko paczki integracji**. Aktualizacje binarne **Elastic Agenta** (oraz niektórych komponentów) standardowo pobierane są z **artifactów Elastic** w internecie. W trybie offline:

1. Na hostach z internetem **zbierz instalatory/archiwa Agenta** (Windows -windows-x86\_64.zip/.msi + pliki sum .sha512), przenieś na **wewnętrzny serwer HTTP** (np. http://repo.lan/elastic-agent/...).
2. W Kibanie wejdź w **Management → Fleet → Settings → Agent binary download sources** i dodaj swój URL (oznacz jako domyślny).
3. Od tej pory Fleet będzie pobierał binaria z Twojego źródła, bez wyjścia do internetu.

*(Ten krok nie wymaga edycji kibana.yml — robisz go w UI.)*

**6) Test i diagnostyka**

* **EPR żyje?**
* (Invoke-WebRequest "http://<HOST\_EPR>:8080/health").StatusCode # 200 = OK
* Invoke-WebRequest "http://<HOST\_EPR>:8080/search?kibana.version=8.13.2" | % Content

Jeśli zwraca JSON z paczkami — indeksacja działa, Kibana powinna je widzieć.

* **Kibana → EPR?**  
  Po starcie z flagami CLI wejdź w **Management → Fleet → Integrations**.  
  Jeśli wcześniej widziałeś błędy „Error connecting to package registry…”, powinny zniknąć. Katalog paczek będzie pochodził z Twojego EPR.
* **Najczęstsze problemy i szybkie fixy**
  + *Pusta lista paczek*: sprawdź, czy ZIP-y są w C:\EPR\packages i czy wersje paczek zgadzają się z Twoją wersją Kibany (filtr kibana.version).
  + *404/500 z EPR*: literówka w ścieżkach w config.yml (na Windows używaj ukośników /) lub brak uprawnień usługi do folderów.
  + *Kibana nadal łączy się na zewnątrz*: upewnij się, że **flagi CLI** naprawdę są w binPath usługi i Kibana została **zrestartowana**.
  + *Błąd certyfikatu (HTTPS EPR)*: dodaj/zweryfikuj NODE\_EXTRA\_CA\_CERTS i certyfikat CA; sprawdź też nazwę hosta w certyfikacie.
  + *Usługa EPR nie startuje*: najczęściej problem z cudzysłowami w binPath. Użyj dokładnie wzorca z sekcji **3.2**.

**7) (Opcjonalnie) Skrypty, które ułatwią życie**

**Pobieranie wielu paczek z listy (na hoście online):**

# packages.txt: po jednej linii: nazwa;wersja (np. system;1.53.0)

Get-Content .\packages.txt | ForEach-Object {

$n,$v = $\_.Split(';')

$url = "https://epr.elastic.co/epr/$n/$n-$v.zip"

Write-Host "Pobieram $url"

Invoke-WebRequest $url -OutFile "$n-$v.zip"

}

**Szybki smoke-test EPR (offline):**

$EprHost = "localhost:8080"

(Invoke-WebRequest "http://$EprHost/health").StatusCode

(Invoke-WebRequest "http://$EprHost/search?kibana.version=8.13.2").StatusCode

**Podsumowanie (2 zdania)**

* **EPR offline** uruchamiasz jako usługę Windows, serwując paczki z lokalnego katalogu ZIP-ów.
* **Kibanie** wskazujesz EPR i tryb air-gapped **bez kibana.yml**, dodając **flagi CLI** do binPath usługi: --xpack.fleet.registryUrl=http://<HOST\_EPR>:8080 --xpack.fleet.isAirGapped=true.