Kibana – konfiguracja APM RUM JS (Angular) z dostępem anonimowym

# Cel dokumentu

Poniższa checklista opisuje dokładnie, co należy włączyć w Kibanie (Fleet → APM), aby przyjmować anonimowe dane z APM RUM JS dla aplikacji Angular – bez użycia API Key lub secretu w przeglądarce.

# Checklista krok po kroku (Kibana → Fleet → APM)

1. Wejdź: Fleet → Agent policies → (Twoja policy) → APM → Edit integration.
2. W sekcji RUM (Real User Monitoring) zaznacz:  
    • Enable RUM  
    • Enable anonymous access for RUM
3. CORS – w polu RUM allowed origins dodaj dokładne originy aplikacji (schemat + host + port), np.:  
    • http://localhost:4200  
    • https://twoja-domena.pl  
   Nie używaj symbolu \* (każdy origin podaj osobno).
4. (Opcjonalnie, zalecane) Allowed service names – wpisz nazwy usług używane w inicjalizacji RUM (np. frontend-angular), aby ograniczyć kto może wysyłać dane przy włączonym anonymous.
5. (Opcjonalnie) Anonymous rate limiting – ustaw limity zdarzeń/IP, aby ograniczyć nadużycia.
6. Zapisz zmiany: Save & deploy changes.
7. Skopiuj APM Server URL z karty integracji i ustaw go w konfiguracji RUM po stronie Angulara jako serverUrl. To jedyny wymagany element po stronie JS – bez API Key/secretu.
8. Weryfikacja:  
    • Jeśli widzisz 401 dla …/intake/v2/rum/events → sprawdź, czy Enable anonymous access jest włączone.  
    • Jeśli widzisz błąd CORS → sprawdź, czy origin Twojej aplikacji jest dodany w RUM allowed origins (z prawidłowym portem).  
    • Po kilku minutach wejdź w APM → Services – powinna pojawić się usługa o nazwie podanej w konfiguracji RUM.

# Notatki

Brak API Key w JS jest zamierzony – przy włączonym „Enable anonymous access for RUM” APM Server przyjmuje zdarzenia z dozwolonych originów. Kluczy i sekretów nie zapisujemy w kodzie frontendu.