CKAN Mimari Açıklaması: Veri Ambarı ve DMZ Kullanımı

Bu doküman, CKAN'ın kurumsal bir veri ambarı (Data Warehouse) ile nasıl entegre olduğu ve güvenli bir ağ mimarisinde (DMZ) nasıl konumlandırıldığına dair teknik bir açıklama sunar.

1. Veri Akış Mimarisi (Data Warehouse Entegrasyonu)

CKAN, bir veri ambarındaki verileri iki temel yöntemle dış dünyaya açar:

A. Veri Alımı (Ingestion - Warehouse to CKAN)

- **Push Modeli (API):** Veri ambarı, ETL (Extract, Transform, Load) süreçlerinin sonunda CKAN'ın Action API uç noktalarını (`package_create` , `resource_create`) kullanarak veriyi CKAN'a iter.
- **Pull Modeli (Harvesting):** `ckanext-harvest` eklentisi kullanılarak, CKAN belirli aralıklarla veri ambarının sunduğu (CSW, WFS veya uzak CKAN) metadata kaynaklarını tarar ve günceller.
- **DataStore Yükleme:** Dosyalar (CSV, Excel vb.) yüklendiğinde `ckanext-xloader` veya `datapusher` bileşenleri bu dosyaları satır satır okuyarak queryable (sorgulanabilir) bir yapı olan **DataStore**'a (ikinci bir PostgreSQL veritabanı) aktarır.

B. Veri Dağıtımı (Distribution - CKAN to Public)

- **DataStore API:** Kullanıcıların tüm dosyayı indirmeden sadece ihtiyaç duydukları satırları SQL benzeri sorgularla (`datastore_search`) çekmesini sağlar.
- **Data Dumps:** DataStore içindeki yapılandırılmış tabloların otomatik olarak CSV, JSON veya XML formatlarında "dump" edilerek indirilmesine olanak tanır.

---

2. DMZ (Demilitarized Zone) Mimarisi

Güvenlik gereksinimleri nedeniyle CKAN genellikle çok katmanlı bir ağ yapısında kurulur.

```
graph TD
    subgraph "Public Internet (Dış Dünya)"
        Client["Kullanıcılar / API İstemcileri"]
    end

    subgraph "DMZ (Arındırılmış Bölge)"
        Nginx["Reverse Proxy (Nginx)"]
        StaticAssets["Statik Dosyalar (CSS/JS)"]
    end

    subgraph "Application Tier (Uygulama Katmanı)"
        CKAN["CKAN (Flask/uWSGI)"]
        Redis["Redis (Kuyruk ve Oturum)"]
    end

    subgraph "Secure Data Tier (Güvenli Veri Katmanı)"
        Postgres["PostgreSQL (Core DB + DataStore)"]
        Solr["Solr (Arama Motoru)"]
    end
```

```
Client -- HTTPS (Port 443) --> Nginx
Nginx -- Proxy --> CKAN
CKAN -- Internal API --> Redis
CKAN -- SQL --> Postgres
CKAN -- Search API --> Solr
```

Katmanların Görevleri:

1. **DMZ (Nginx):** Dış dünyaya açık tek noktadır. SSL terminasyonu yapar, statik dosyaları sunar ve uygulamayı doğrudan saldırılardan korur. Sadece uygulama katmanına (CKAN) erişim izni vardır.
2. **Application Tier (CKAN):** İş mantığının (Business Logic) çalıştığı yerdir. Dış dünyaya kapalıdır, sadece DMZ'den gelen istekleri kabul eder.
3. **Secure Data Tier (Database):** En korunaklı bölgedir. Sadece uygulama katmanından gelen veritabanı bağlantılarını kabul eder. Dış dünyadan bu katmana doğrudan hiçbir erişim yoktur.

3. Güvenlik ve Yetkilendirme

- **API Token:** Dış sistemler (Veri ambarı gibi) CKAN'a veri yazmak veya özel verilere erişmek için JWT tabanlı API Token'ları kullanır.
- **Organizasyonel Yetkilendirme:** Veriler organizasyon bazlı gruplandırılır. "Private" işaretlenen veriler, DMZ üzerinden erişilse dahi sadece yetkili kullanıcılar tarafından görülebilir.
- **IP Whitelisting:** Veri ambarından CKAN'a yapılan push işlemleri için güvenlik duvarı seviyesinde sadece ambarın IP adresine yazma yetkisi verilebilir.

> *[!IMPORTANT] CKAN'ın DataStore veritabanı, ana veritabanından (Core DB) ayrı tutulmalıdır. Bu, bir veri sızıntısı durumunda ana kullanıcı ve sistem meta verilerinin korunmasını sağlar.*