Dependency & Registry Supply Chain Güvenlik Checklist Şablonu — Yazılım / Sunucu ve Güvenlik (v1.0)

Asset Amaç: Bu asset; yazılım kütüphane/dependency paketlerini ve container imaj kaynaklarını "güvenilir ve denetlenebilir" hale getirmek için hazırlanmıştır. Lockfile kullanımı, dahili paket deposu (*internal registry*) ve onaylı temel imaj (*approved base image*) standartlarını kurar; CI/CD pipeline üzerindeki güvenlik denetimleri (*policy gates*) ile riskli ya da zafiyetli güncellemelerin canlı ortama girmesini engellemeyi hedefler. Temel amaç; yazılım tedarik zinciri (*supply chain*) saldırı yüzeyini küçültmektir.

Kim Kullanır?: Backend / DevOps Mühendisleri, Platform Mühendisliği ve Otel / B2B Yazılım Ekipleri.

Nasıl Kullanılır?

1. İzin verilen paket kaynaklarını ve registry politikalarını belirleyerek hazırlık kontrol listesini doldurun.
2. Lockfile zorunluluğunu ve internal registry/scope adımlarını CI/CD pipeline kontrollerine (*CI gate*) bağlayın.
3. Container imaj tarama (*image scanning*) ve otomatik dağıtım engelleme politikalarıyla güvenlik döngüsünü kapatın.

CHECKLIST & SPRINT PLAN — Supply Chain Güvenliği

1) Ölçüm & Önceliklendirme Kontrol Listesi (Checklist)

- [ ] **Onaylı Registry Listesi Var:** NPM, PyPI, Maven ve OS paket yöneticileri için izin verilen resmi ve güvenilir kaynaklar (*allowlist*) tanımlandı.
- [ ] **Lockfile Zorunlu & CI'da Kontrol Ediliyor:** Projelerde package-lock.json, Pipfile.lock veya Cargo.lock kullanımı zorunlu; CI pipeline üzerinde sürümler doğrulanıyor.
- [ ] **Yeni Dependency Ekleme PR + Review Şart:** Projeye eklenecek her yeni harici kütüphane için kod gözden geçirme (*code review*) ve güvenlik onayı zorunluluğu getirildi.
- [ ] **Internal Registry (Paket) Kullanım Planı Var:** Kurumsal ve üçüncü parti paketlerin önbelleklenmesi ve sunulması için dahili registry (Nexus, Artifactory, GitHub Packages) mimarisi tasarlandı.
- [ ] **Internal Paket İsimlendirme Standardı Var:** Dependency confusion / typosquatting saldırılarını önlemek amacıyla kurumsal paketler için özel isim alanı (*scoped packages*, örn: @otel/package) tanımlandı.
- [ ] **Approved Base Image Listesi Var:** Dockerfile ve container yapılandırmalarında yalnızca güvenlik ekibi tarafından onaylanmış ve taranmış temel imajların (*distroless, alpine-slim vb.*) kullanımı kurala bağlandı.
- [ ] **Image Taraması CI/Registry'de Zorunlu:** Container imajları derleme aşamasında ve registry'e push edildiğinde otomatik güvenlik taramasından (*Trivy, Grype, Clair*) geçiriliyor.

- [ ] **Kritik CVE'de Build/Deploy Bloklanıyor:** Taramalarda tespit edilen Critical/High seviye zafiyetlerde CI build ve deployment süreçleri otomatik olarak durduruluyor.
- [ ] **Rebuild / Rollout Rutini Var:** Güncel base image ve yamaları içerecek şekilde konteynerlerin periyodik olarak yeniden derlenmesi ve dağıtılması otomatize edildi.
- [ ] **365 Gün Gözden Geçirme Takvimi Var:** Tedarik zinciri güvenlik politikalarının ve kütüphane bağımlılıklarının yılda en az bir kez kapsamlı denetimi planlandı.

2) Problem → Kök Neden → Çözüm Tablosu

Problem	**Kök Neden**	**Çözüm (Supply Chain Güvenliği)**
Sürpriz Paket Güncellemesi	Lockfile kullanılmaması veya CI üzerinde kontrol edilmemesi	**Lockfile Kullanımı** $\mathbf{+}$ **CI Policy Gate**
Dependency Confusion Riski	İç paket çözümlenirken hiyerarşisiz public registry kullanımı	**Internal Registry** $\mathbf{+}$ **Scoped Naming Standardı**
Typosquatting Saldırıları	İsim benzerliğine dayalı zararlı paketlerin fark edilmeden eklenmesi	**Registry Allowlist** $\mathbf{+}$ **PR Review / Security Check**
Eski & Zafiyetli Container Imajı	Temel imajların düzenli güncellenmemesi ve rebuild edilmemesi	**Image Scan** $\mathbf{\rightarrow}$ **Rebuild** $\mathbf{\rightarrow}$ **Kademeli Rollout**

3) 14 Günlük Uygulama Sprint Planı (Gün 1–14)

[Gün 1–5: Kaynak & Registry Politikaları] ⟶ [Gün 6–8: Base Image & Scan Pipeline] ⟶ [Gün 9–12: Pilot & Rebuild Rutini] ⟶ [Gün 13–14: Eğitim & Rapor]

- **Gün 1:** Onaylı registry listesinin (Allowlist) çıkartılması ve paket erişim politikalarının dökümante edilmesi.
- **Gün 2:** Tüm repository'lerde Lockfile zorunluluğunun getirilmesi ve CI kuralı olarak eklenmesi.
- **Gün 3:** Dependency değişiklikleri için Pull Request (PR) gözden geçirme ve güvenlik kural setinin tanımlanması.
- **Gün 4:** Internal Registry (Nexus/Artifactory/GitHub Packages) mimarisinin ve proxy/cache kurallarının tasarlanması.

- **Gün 5:** Kurumsal paket isimlendirme standardının (*Scoped Packages*) oluşturulması ve yapılandırılması.
- **Gün 6:** Onaylı temel imaj (*Approved Base Image*) listesinin ve Docker en iyi uygulama standartlarının kilitlenmesi.
- **Gün 7:** CI/CD pipeline'a otomatik container imaj tarama adımlarının (Image Scanning) entegre edilmesi.
- **Gün 8:** Kritik zafiyetlerde (Critical/High CVE) build/deploy süreçlerini kesen güvenlik kapısının (*Policy Gate*) aktif edilmesi.
- **Gün 9:** Belirlenen pilot uygulama ve repository üzerinde tüm supply chain kurallarının uygulanması ve test edilmesi.
- **Gün 10–12:** Eski imaj ve bağımlılıkların güncellenmesini sağlayan otomatik Rebuild & Rollout rutinlerinin kurulması.
- **Gün 13:** Tüm geliştirici ve DevOps ekipleri için supply chain güvenliği oryantasyon dokümantasyonu ve eğitim verilmesi.
- **Gün 14:** Yönetimsel sonuç raporunun hazırlanması ve sürdürülebilir 365 günlük gözden geçirme takviminin kilitlenmesi.

4) Öncesi / Sonrası KPI Tablosu

KPI Metriği	**Önce (Mevcut Durum)**	**Sonra (Sprint Sonrası)**	**Hedef Yönü**
Lockfile Uyum Oranı (%)	$\%35$ (Geliştirici bazlı değişken)	$\%100$ (CI kuralı ile zorunlu)	**Hedef ↑ (Yükselme)**
Kritik CVE Açık Süresi (MTTD/MTTR)	$\sim 45\text{ gün}$	$< 48\text{ saat}$	**Hedef ↓ (Düşüş)**
Onaylı Kaynak Kullanım Oranı (%)	$\%50$ (Doğrudan public registry)	$\%100$ (Internal/Allowlist üzerinden)	**Hedef ↑ (Yükselme)**
Build Blok Sayısı (Risk Engelleme)	0 (Denetim yok)	İzleme Altında (Riskli PR'lar bloklandı)	**Takip / İzleme**

Deliverables (Teslim Edilecek Çıktılar)

- **Registry Policy & Onaylı Kaynaklar (Allowlist) Dokümanı:** Kullanımına izin verilen resmi ve dahili depolara ait erişim kuralları.
- **Lockfile Kontrol & CI Policy Gate Scriptleri:** Pipeline aşamasında bağımlılık kilitlerini ve sürümlerini doğrulayan otomasyon araçları.
- **Internal Registry & Scoped Naming Mimarisi:** Dahili paket sunucusu kurulum kılavuzu ve kurumsal paket isimlendirme rehberi.
- **Approved Base Image Kütüphanesi & Scan Raporu:** Güvenliği doğrulanmış temel konteyner imajları ve zaafiyet engelleme raporu.

[Checklist Card]

Platform mühendisleri, DevOps uzmanları ve yazılım mimarları için tasarlanmış, 1200x1200px (1:1) kare formatında yüksek çözünürlüklü bir Supply Chain Security Checklist Kartı. Koyu karbon gri ve parlak mor/siyan tonlarındaki teknolojik zemin üzerinde; kütüphane kilit simgeleri (lockfile), container imaj tarama radarları, internal registry bağlantı şemaları ve 14 günlük supply chain sprint takvimi yer alıyor. Kartın üzerinde; "Registry Allowlist & Scoped Package Naming", "Mandatory Lockfile Enforcement & CI Policy Gates", "Approved Base Images & Automated Container Scanning" ve "CVE-Based Build Blocking & Rollout Routines" teslimat kalemi başlıkları, yanlarında parlak mor onay işaretleriyle scannable ve kurumsal bir checklist-card formatında sergileniyor.