Celari Wallet — iOS PXE Engine Durum Raporu

22 Şubat 2026 | Versiyon: devnet-6 | Platform: iOS (WKWebView) | Simulator: iPhone 17 Pro, iOS 26.2

1. Genel Bakış

Celari Wallet, Aztec Network üzerinde çalışan privacy-first bir iOS cüzdan uygulamasıdır. Aztec PXE (Private eXecution Environment) motoru, WKWebView içinde tek thread'li JavaScript olarak çalışmaktadır.

Mevcut Durum: Hesap deploy işlemi, `contractDataProvider.addContractArtifact()` adımında takılmaktadır. Bu fonksiyon WKWebView'ın IndexedDB implementasyonunda büyük buffer yazma işleminde sonsuza kadar bloke olmaktadır.

2. Mimari Özet

```
Swift Layer (Native iOS)
├─ WalletStore.swift — Hesap yönetimi, PXE bridge koordinasyonu
├─ PXEBridge.swift — WKWebView yaşam döngüsü, JS↔Swift mesajlaşma
└─ Passkey Authentication — WebAuthn tabanlı hesap oluşturma

JavaScript Layer (WKWebView)
├─ pxe-bridge.html — Shim'ler yükler, offscreen.js'i başlatır
├─ offscreen.js (73.8 MB IIFE) — Tüm Aztec SDK, PXE client, wallet logic
└─ PXE-Shim — Chrome API polyfill, fetch polyfill, Worker no-op

Aztec SDK Bileşenleri
├─ @aztec/pxe (client/lazy) — PXE istemci, ContractDataProvider
├─ @aztec/kv-store/indexeddb — IndexedDB üzerinde KV store
├─ @aztec/bb.js (BarretenbergSync) — WASM kriptografik işlemler
└─ @aztec/test-wallet — TestWallet, fakeProofs modu
```

3. Deploy Akışı ve Zamanlama Analizi

Hesap deploy süreci 6 ana adımdan oluşur. Aşağıda 3 farklı test çalışmasından (v4, v5, v7) elde edilen zamanlama verileri yer almaktadır.

3.1 Başarılı Adımlar

Adım	İşlem	Süre	Durum
PXE_INIT	Node bağlantısı + WASM yükleme	~3s	**OK**
Step 1/6	deriveKeys(secretKey)	1–6ms	**OK**
Step 2/6	getInitializationFunctionAndArgs	0ms	**OK**
Step 3/6	getContractArtifact	0ms	**OK**
Step 4/6	getContractInstanceFromInstantiationParams	15–29ms	**OK**
Step 5a	AccountManager constructed	0ms	**OK**
Step 5b	getContractMetadata (network)	456–464ms	**OK**
Step 5c.1	getContractClassFromArtifact (WASM)	43–48ms	**OK**
Step 5c.2	computeContractAddressFromInstance	1ms	**OK**

3.2 Takılan Adım

Adım	İşlem	Süre	Durum
Step 5c.3	**contractDataProvider.addContractArtifact()**	**∞**	**HANG**

Bu adımdan sonra hiçbir log gelmemektedir. Uygulama yanıt vermez hale gelmektedir.

4. Kök Neden Analizi

4.1 addContractArtifact İç Yapısı

```
addContractArtifact(classId, artifact)
├─ 1. Filter private functions (artifact.functions)
├─ 2. For each private function:
│   └─ FunctionSelector.fromNameAndParameters()
│       └─ poseidon2HashBytes() → BarretenbergSync WASM
├─ 3. Duplicate selector check
└─ 4. this.#contractArtifacts.set(id, contractArtifactToBuffer(artifact))
```

```
    └─ IndexedDBAztecMap.set(key, val)
        ├─ ohash.hash(val) ← Büyük buffer hash'leme
        └─ db.put({...})   ← IndexedDB yazma
```

4.2 Olası Neden: IndexedDB + Büyük Buffer

CelariPasskeyAccount artifact'i 6 fonksiyon içerir. Serialize edilmiş buffer boyutu yüzlerce KB–MB arasındadır.

IndexedDBAztecMap.set() metodu iki kritik işlem yapar:

- `ohash.hash(val)` — Tüm buffer'ı hash'ler (CPU-bound)
- `db.put(...)` — IndexedDB'ye yazar (I/O-bound)

WKWebView'ın IndexedDB implementasyonu, büyük binary blob'larda bilinen performans sorunlarına sahiptir. Bu işlemlerden biri (muhtemelen ohash veya db.put) sonsuz döngüye girmekte veya deadlock oluşturmaktadır.

4.3 Kanıt Zinciri

v4 logu: Step 5c (registerContract) → HANG (decompose edilmemiş)
v5 logu: Step 5c.1 OK (48ms), Step 5c.2 OK (1ms), Step 5c.3 (addContractArtifact) → HANG
v7 logu: Aynı Step 5c.3 → HANG (decompose fix build'e yansımamış)
3 bağımsız testin tamamında aynı noktada takılma — kesin root cause.

5. Uygulanan Fix ve Durumu

5.1 Strateji: Timeout + In-Memory Monkey-Patch

offscreen.js'te Step 5c.3 decompose edildi:

- **5c.3a:** Her private function için FunctionSelector hesaplama (timing ile)
- **5c.3b:** contractArtifactToBuffer ile serialize etme (timing ile)
- **5c.3c:** addContractArtifact çağrısı, 30 saniye timeout ile Promise.race
  - Timeout olursa: `getContractArtifact` monkey-patch ile memory'den serve

5.2 Build Durumu

Komut	Sonuç
`node extension/build.mjs --dev`	**BUILD OK (offscreen.js 73.8 MB)**
`xcodebuild`	**BUILD SUCCEEDED**
Simulator install + launch	**OK**

5.3 Test Durumu

SON DURUM: v7 testinde decompose edilmiş sub-step logları (5c.3a, 5c.3b, 5c.3c) görünmedi. Hâlâ eski format geliyor.

Olası nedenler:
1. Xcode build cache — eski bundle kullanılmış olabilir
2. Kontrol akışı — Kod beklenen dalına girmemiş olabilir

Sonuç: Fix henüz doğrulanmadı. Clean build + cache temizleme gerekli.

6. Tamamlanan İyileştirmeler (Önceki Oturumlar)

#	Sorun	Çözüm	Durum
1	Double setupWebView() race condition	Tekrarlanan çağrı engellendi	**ÇÖZÜLDÜ**
2	Stale build output, JS_HANDLER_READY eksik	Build pipeline düzeltildi	**ÇÖZÜLDÜ**
3	PXE_INIT sadece hesap varken gönderiliyordu	Koşul düzeltildi	**ÇÖZÜLDÜ**

4	proverEnabled: true (WASM Worker gerekli)	iOS'ta false yapıldı (fakeProofs)	**ÇÖZÜLDÜ**
5	WASM dosyaları bundle'da eksik	Build config'e eklendi	**ÇÖZÜLDÜ**

7. Bekleyen İşler ve Öneriler

7.1 Kritik (Deploy Blocker)

addContractArtifact hang fix'i doğrulanmalı:
1. Xcode clean build (DerivedData temizle)
2. Simulator'da uygulamayı sil + yeniden yükle
3. v7+ loglarında 5c.3a/b/c sub-step'lerin görünmesini doğrula
4. Timeout tetiklenirse monkey-patch'in çalıştığını doğrula

Alternatif yaklaşımlar (fix çalışmazsa):
• IndexedDB yerine in-memory Map kullanmak (KV store bypass)
• `ohash.hash()` çağrısını skip etmek (monkey-patch IndexedDBAztecMap.set)
• ContractDataProvider'ı tamamen memory-only yapmak

7.2 Sonraki Adımlar (Deploy Başarılı Olursa)

#	İş	Açıklama
1	Step 6/6 testi	wallet.deploy() — sponsored fee ile TX gönderme
2	Transfer testi	Token transferi fonksiyonelliği
3	Passkey + Associated Domains	celariwallet.com domain yapılandırması
4	Gerçek cihaz testi	Fiziksel iPhone'da test

7.3 Bilinen Kısıtlamalar

Kısıtlama	Açıklama
Web Worker yok	WKWebView Worker desteklemiyor → proverEnabled: false
WASM performansı	BarretenbergSync WKWebView'da native'den yavaş
IndexedDB limitleri	Büyük blob yazma sorunlu (bu raporun konusu)
Bundle boyutu	offscreen.js 73.8 MB (tree-shaking sınırlı)

8. Test Logları Özeti

v4 — registerContract seviyesinde hang

```
22:05:55.831  Step 5/6: registerContract()...
22:05:55.831  Step 5a: AccountManager constructed (0ms)
22:05:55.831  Step 5b: getContractMetadata...
22:05:56.287  Step 5b: OK (456ms) — existing: false
22:05:56.287  Step 5c: pxe.registerContract... ← SON LOG
22:05:57.748  [heartbeat 5002ms]
              ↓↓↓ HANG — sonsuza kadar bekliyor ↓↓↓
```

v5 — addContractArtifact seviyesinde hang (daha detaylı)

```
22:10:24.311  Step 5b: getContractMetadata OK (456ms)
22:10:24.311  Step 5c.1: getContractClassFromArtifact...
22:10:24.358  Step 5c.1: OK (48ms)
22:10:24.358  Step 5c.2: computeContractAddressFromInstance...
22:10:24.359  Step 5c.2: OK (1ms)
22:10:24.359  Step 5c.3: contractDataProvider.addContractArtifact... ← SON LOG
22:10:25.869  [heartbeat 5098ms]
              ↓↓↓ HANG — sonsuza kadar bekliyor ↓↓↓
```

v7 — Fix uygulandı ama sub-step logları yok

```
02:27:18.928  Step 5c.2: OK (1ms) — match: true
02:27:18.928  Step 5c.3: contractDataProvider.addContractArtifact... ← ESKİ FORMAT
02:27:19.815  [heartbeat 5002ms]
02:27:29.917  [heartbeat 15102ms]
              ↓↓↓ HANG — fix kod'a yansımamış ↓↓↓
```

9. Sonuç

Celari Wallet iOS uygulamasının PXE bridge mimarisi büyük ölçüde çalışır durumdadır. Node bağlantısı, WASM kriptografi, hesap türetme ve contract instance hesaplama adımları başarıyla tamamlanmaktadır.

Tek kalan blocker: `contractDataProvider.addContractArtifact()` fonksiyonunun WKWebView IndexedDB'de hang etmesi.

Bu sorun, timeout + in-memory fallback stratejisi ile çözülecektir. Fix'in clean build ile doğrulanması gerekmektedir. Doğrulama sonrası deploy akışının tamamlanması ve transfer testlerine geçilmesi planlanmaktadır.

Önerilen Sonuç Adımları

Öncelik	İş	Tahmini Süre
P0	addContractArtifact fix'ini clean build ile doğrula	30 dk
P0	Monkey-patch çalışmazsa: IndexedDB bypass (in-memory KV)	1–2 saat
P1	Deploy tx gönderimini test et (Step 6/6)	1 saat
P1	Token transfer fonksiyonelliği	2–3 saat
P2	Passkey + Associated Domains (gerçek cihaz)	2 saat
P2	Fiziksel iPhone testi	1 saat