Engineer

Öppna en dApp i Solanas mobilplånböcker

10 augusti 2026

En React-dApp byggd på `@solana/wallet-adapter-react` ansluter till plånböcker på datorn utan problem, men i en telefon faller samma flöde isär: plånboken måste öppna dAppen i sin egen inbyggda webbläsare, och djuplänkarna som ska få det att hända misslyckas i det tysta. Backpack hamnar på en sida med ”ladda ned appen”; Solflare öppnar appen men aldrig webbplatsen; varje variant verkar misslyckas. Vi plockade isär problemet, och det visade sig vara fyra separata problem bakom ett och samma symptom.

Använd erfarenheten i ert system

Arbetet bakom ”Öppna en dApp i Solanas mobilplånböcker” kan bidra till er nästa förändring. Vi kan undersöka liknande beteenden i er programvara, era data eller er infrastruktur. Gränssnitt som laddar snabbt, fungerar i mobilen och beter sig likadant i alla webbläsare dina kunder råkar använda.

Se våra tjänster Hör av er →

Följ resonemanget och dess gränser

Läs resonemanget och dess begränsningar i anteckningen. Vi kan testa om samma diagnos håller i er miljö. Styrde full stack-utvecklingen av GIS-kartor — kartutveckling i full stack, från databas till webbläsare

Styrde full stack-utvecklingen av GIS-kartor Hör av er →

De fyra problemen

- **Backpack-länken var felaktigt utformad.** Det enda dokumenterade formatet är `https://backpack.app/ul/v1/browse/<url>?ref=<ref>` — en universell länk med mål-URL:en i sökvägen och en obligatorisk `ref`. En gissning med ett eget URL-schema, som `backpack://ul/v1/browse?url=...`, matchar ingen rutt som appen registrerar, så användaren hamnar på plånbokens installationssida.

- **Solflare behöver också sin universella länk:** `https://solflare.com/ul/v1/browse/<url>?ref=<ref>`, inte det nakna schemat `solflare://`. Ett naket schema kan starta appen utan att dirigera den vidare — vilket är exakt ”appen öppnas, men webbplatsfliken måste öppnas för hand”.
- **Båda parametrarna måste kodas.** `url` är dAppens fullständiga, absoluta adress och `ref` är ursprunget som gör begäran, och båda skickas genom `encodeURIComponent`. Ett okodat `?` eller `&` i målet förstör tolkningen, och plånboken öppnas på sin startskärm i stället för i webbläsarfliken.
- **Hur länken utlöses spelar lika stor roll som länken.** Universella länkar byter bara app vid en navigering som operativsystemet litar på — och de gör avsiktligt ingenting när de klistras in i adressfältet, vilket också är hur en helt korrekt länk ”misslyckas” under testning.

De dokumenterade formaten

- Phantom: `https://phantom.app/ul/browse/<url>?ref=<ref>` — utan `/v1` i den här.
- Solflare: `https://solflare.com/ul/v1/browse/<url>?ref=<ref>`
- Backpack: `https://backpack.app/ul/v1/browse/<url>?ref=<ref>`

Ett och samma mönster fungerar för alla tre:

```
const WALLET_BROWSE = {
  phantom: (url, ref) =>
    `https://phantom.app/ul/browse/${url}?ref=${ref}`,
  solflare: (url, ref) =>
    `https://solflare.com/ul/v1/browse/${url}?ref=${ref}`,
  backpack: (url, ref) =>
    `https://backpack.app/ul/v1/browse/${url}?ref=${ref}`,
};

function walletBrowseLink(
  walletName,
  targetUrl = window.location.href,
) {
  const build = WALLET_BROWSE[walletName.toLowerCase()];
  if (!build) return null;
  return build(
    encodeURIComponent(targetUrl),
    encodeURIComponent(window.location.origin),
  );
}
```

Utlös länken så att iOS och Android accepterar den

- **Rendera ett riktigt ankarelement, beräknat i förväg.** En vanlig `<a href={walletBrowseLink('phantom')}>` är den mest pålitliga utlösaren på båda plattformarna.
- **Om det måste ske programmatiskt**, sätt `window.location.href` synkront inuti hanteraren för trycket — ingen `await`, ingen `fetch`, ingen `setTimeout` först. Efter asynkront arbete är gestens kontext borta, och iOS faller tillbaka på plånbokens webbplats. Aldrig `window.open`.
- **Testa aldrig genom att klistra in i adressfältet.** Universella länkar utlöses avsiktligt inte där; testa med en länk som du trycker på eller en QR-kod som skannas med kameran.
- **Se upp med meddelandeappars webbvyer.** I Telegrams eller Instagrams inbyggda webbläsare sväljs universella länkar ofta, och plånbokens vanliga webbplats laddas i stället. Att känna igen dem på user agent är i bästa fall en heuristik, så ge också användarna en synlig nödutgång: ”öppna i Safari eller Chrome och anslut sedan”.

Den större lösningen på Android

Handbyggda djuplänkar är iOS-sidan av saken. På Android låter Mobile Wallet Adapter från Solana Mobile en dApp som körs i mobilens webbläsare ansluta direkt till den installerade plånboksappen, helt utan omvägen via en inbyggd webbläsare. Nyare versioner av `@solana/wallet-adapter-react` registrerar mobiladaptern automatiskt, så en uppgradering av wallet-adapter-paketen kan på egen hand lösa problemet på Android. Målarkitekturen: Mobile Wallet Adapter på Android, universella browse-länkar på iOS, där Apple inte tillåter någon motsvarighet.

Verifiera på en enhet

- Riktig enhet, plånboken installerad, länken öppnad från systemets webbläsare — inte från en meddelandeapp.
- Tryck på en renderad länk eller skanna en QR-kod; klistra aldrig in i adressfältet.
- Bekräfta att plånboken öppnas och att dAppen laddas i dess inbyggda webbläsarflik — den andra halvan är den del som misslyckas.

- Upprepa utan plånboken installerad: den universella länken ska då falla tillbaka på plånbokens webbplats. Visas den sidan medan appen är installerad är länken eller utlösningen fortfarande fel.
- Testa sedan vägen via meddelandeappar, och lägg till tipset ”öppna i webbläsaren” om det misslyckas där.

Källor

- Phantom: djuplänkar på iOS och Android
- Solflare: Browse-djuplänken
- Backpack: Browse-djuplänken
- Solana Mobile: Mobile Wallet Adapter

Alla blogginlägg

Mät innan ni ändrar

Beskriv arbetsflödet, befintliga gränssnitt och användarna som behöver förändringen. Vi kan definiera en testbar leverans och en överlämning som gör att teamet kan underhålla lösningen.

Se vad det kostar Hör av er →

Diskutera era observationer

Diskutera ”Öppna en dApp i Solanas mobilplånböcker” i samband med ert eget projekt. Berätta vad ni observerade och vad ett användbart resultat skulle vara.

Se våra tjänster

welcome@engineer.company +45 50 10 61 84 Köpenhamn, Danmark

Ta med dig den här sidan

Visitkort

Omsorgsfullt skapad med kärlek av D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Upphovsrätt © 2025 – nu · Engineer ApS · Alla rättigheter förbehållna.

Utskrivet från https://engineer.company/sv/notes/solana-mobile-wallet-deeplinks/