Breach Check Tool - Bug Fixes

Date: November 6, 2025

Issues Reported

Issue #1: Incorrect Breach Detection Results

Problem: Testing the same email on both our webpage and XposedOrNot's website showed different results. Our webpage said "no breach" while XposedOrNot showed multiple breaches.

Root Cause: The XposedOrNot API response structure was being parsed incorrectly. The actual API returns:

```
{
  "BreachMetrics": { ... },
  "BreachesSummary": { "site": "..." },
  "ExposedBreaches": {
    "breaches_details": [...]
  }
}
```

But our code was looking for `breaches_details` at the root level instead of inside `ExposedBreaches`.

Fix Applied: Updated `/app/api/breach-check/route.ts` to correctly parse the XposedOrNot API response:
- Now accessing `data.ExposedBreaches.breaches_details` instead of `data.breaches_details`
- Also checking `BreachesSummary.site` for a complete count of breaches
- Using the higher count from both sources for accuracy
- Extracting risk score from `BreachMetrics.risk[0]`
- Properly parsing breach dates from `xposed_date` field
- Correctly splitting `xposed_data` field to get compromised data types

Issue #2: Rate Limiting Display Bug

Problem: After one breach check attempt, the display showed "9 out of 4 chances remaining" which doesn't make sense.

Root Cause: Two mismatches between frontend and backend:
1. Backend API route uses a limit of **10 checks per hour** per IP
2. Frontend was hardcoded to show limit of **4 checks**
3. Frontend initialized `checksRemaining` to 4 instead of 10

Fix Applied: Updated `/app/cyber-intelligence/page.tsx`:
- Changed initial state from `useState(4)` to `useState(10)`
- Updated display text from `{checksRemaining}/4` to `{checksRemaining}/10 per hour`

Technical Details

Files Modified

1. `/app/api/breach-check/route.ts`
   - Fixed XposedOrNot API response parsing
   - Added proper handling for nested data structure
   - Improved breach count accuracy by checking multiple data sources
   - Added risk score and risk label to response
   - Better error handling for missing data

2. `/app/cyber-intelligence/page.tsx`
   - Fixed rate limit initial state (4 → 10)
   - Updated rate limit display text (4 → 10 per hour)

API Response Enhancement

The API now returns more detailed information:


```
{
  "exposed": true,
  "breachCount": 434,
  "breachNames": ["Collection-1", "LinkedIn", "Adobe", ...],
  "riskScore": 83,
  "riskLabel": "High",
  "mostRecentBreach": {
    "name": "MyVidster-2025",
    "date": "2025",
    "dataClasses": ["Email addresses", "Usernames"],
    "records": 4300888
  },
  "message": "Warning! This email appeared in 434 known data breaches.",
  "remaining": 9
}
```

Testing Results

✅ **Breach Detection**: Now accurately detects breaches matching XposedOrNot's website
✅ **Rate Limiting**: Correctly shows "X/10 per hour" format
✅ **API Response**: Properly parses all breach data including risk scores
✅ **Build**: Application compiles and builds successfully
✅ **Deployment**: Live at https://quantumleapai.abacusai.app

How Rate Limiting Works

- **Limit**: 10 checks per hour per IP address
- **Reset**: Automatically resets after 1 hour from first check
- **Display**: Shows remaining checks out of 10
- **Backend**: Uses simple in-memory storage (recommended to upgrade to Redis for production)

Next Steps (Optional Improvements)

1. **Redis Integration**: Replace in-memory rate limiting with Redis for better scalability
2. **Enhanced Display**: Show breach risk score on the frontend (currently returned by API but not displayed)
3. **Detailed Breach Info**: Add modal to show detailed information about each breach
4. **Export Feature**: Allow users to download their breach report as PDF

Status: ✅ All Issues Fixed and Deployed
Deployed: November 6, 2025
Live URL: https://quantumleapai.abacusai.app/cyber-intelligence