# Fraud Prevention

## Bot Detection, Rate Limiting & Spam Prevention

---

## 1. Bot Detection

### 1.1 reCAPTCHA Implementation

**Recommended: reCAPTCHA v3 (invisible)**

| Trigger Point | Risk Score Threshold | Action |
|---|---|---|
| Registration | < 0.3 | Block + show v2 challenge |
| Login | < 0.3 | Require additional verification |
| Message send | < 0.5 | Rate limit + flag for review |
| Profile update | < 0.5 | Flag for review |

**Implementation:**

```
// Example: reCAPTCHA v3 verification const verifyRecaptcha = async (token, action) => {
const response = await fetch( https://www.google.com/recaptcha/api/siteverify , {
method: POST , body: new URLSearchParams({ secret: RECAPTCHA_SECRET, response: token }) }
); const data = await response.json(); return data.success && data.score >= 0.5 &&
data.action === action; }
```

### 1.2 Device Fingerprinting

**Signals to collect:**

- ■ Device ID (IDFV/Android ID)
- ■ Screen resolution
- ■ Timezone
- ■ Language settings
- ■ Installed fonts (web)
- ■ Canvas fingerprint (web)

**Suspicious patterns:**
- Multiple accounts from same device
- Device ID changes frequently
- Emulator detection signals

### 1.3 Behavioral Analysis

| Behavior | Normal | Suspicious |
|---|---|---|
| Swipes per minute | < 30 | > 60 |
| Messages per hour | < 50 | > 100 |
| Profile views per hour | < 100 | > 300 |
| Account age for first message | > 5 min | < 1 min |
| Time between matches and message | Variable | Always identical |

---

## 2. Rate Limiting

### 2.1 Rate Limit Configuration

| Endpoint/Action | Limit | Window | Action on Exceed |
|---|---|---|---|
| Registration | 3 | 1 hour / IP | Block IP temporarily |
| Login attempts | 5 | 15 minutes / user | Lockout + CAPTCHA |
| Password reset | 3 | 1 hour / email | Delay response |
| Messages sent | 100 | 1 hour / user | Soft block |
| Swipes | 100 | 1 hour / user | Require wait |
| Profile updates | 10 | 1 hour / user | Soft block |
| Photo uploads | 20 | 1 hour / user | Soft block |

| | | | |
|---|---|---|---|
| Reports submitted | 10 | 24 hours / user | Review user |
| API calls (general) | 1000 | 1 hour / user | 429 response |

### 2.2 Implementation Strategy

**Token bucket algorithm recommended:**

```
// Pseudocode: Token bucket rate limiter class RateLimiter { constructor(capacity, refillRate) { this.capacity = capacity; this.tokens = capacity; this.refillRate = refillRate; // tokens per second this.lastRefill = Date.now(); } tryConsume(tokens = 1) { this.refill(); if (this.tokens >= tokens) { this.tokens -= tokens; return true; } return false; } refill() { const now = Date.now(); const elapsed = (now - this.lastRefill) / 1000; this.tokens = Math.min( this.capacity, this.tokens + elapsed * this.refillRate ); this.lastRefill = now; } }
```

### 2.3 Response to Rate Limiting

| Action | User Feedback | Backend Action |
|---|---|---|
| Soft limit | Warning message | Log + continue |
| Hard limit | Temporary block message | Block + log |
| Abuse pattern | Account review | Flag + notify admin |

---

## 3. Spam Prevention

### 3.1 Message Spam Detection

**Patterns to detect:**
- ■ Identical messages to multiple users
- ■ Messages containing URLs
- ■ Messages with phone numbers
- ■ Messages with email addresses
- ■ Copy-paste detected (timing analysis)
- ■ High message volume to new matches

### 3.2 Profile Spam Detection

**Red flags:**
- ■ Profile created and immediately active
- ■ Generic/stock photo appearance
- ■ Bio contains URLs or contact info
- ■ Bio matches known spam patterns
- ■ Profile changes frequently

### 3.3 Spam Detection Rules

```
// Example: Simple spam detection rules const spamRules = { // URL detection containsURL: /https?:\/\/|www\./i, // Contact info in bio containsPhone: /\b\d{3}[-.]?\d{3}[-.]?\d{4}\b/, containsEmail: /[a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}/, // Social media handles containsSocial: /@\w+|instagram|snapchat|telegram|whatsapp/i, // Crypto/financial spam containsCrypto: /bitcoin|crypto|invest|profit|trading/i, // Duplicate detection isDuplicate: (message, recentMessages) => recentMessages.some(m => similarity(m, message) > 0.9) };
```

### 3.4 Action Matrix

| Detection | Confidence | Action |
|---|---|---|
| URL in first message | High | Block + warn user |
| Contact info in bio | Medium | Flag for review |
| Duplicate messages | High | Block messages + warn |
| High volume activity | Medium | Rate limit + monitor |
| Multiple spam signals | High | Suspend account |

---

## 4. Account Fraud Prevention

### 4.1 Fake Account Indicators

| Indicator | Weight | Detection Method |
|---|---|---|
| No profile photo | Medium | Automated check |
| Stock photo | High | Reverse image search |

| | | |
|---|---|---|
| Stolen photo | High | Reverse image search |
| Generic bio | Low | Pattern matching |
| Immediate aggressive matching | High | Behavioral analysis |
| Device previously banned | High | Device fingerprint |
| VPN/proxy detected | Medium | IP analysis |

### 4.2 Account Verification Layers

| Layer | Implementation | User Friction |
|---|---|---|
| Email verification | Required at signup | Low |
| Phone verification | Required at signup | Low |
| Photo verification | Selfie match (optional) | Medium |
| ID verification | Third-party service (optional) | High |
| Social verification | Link social accounts (optional) | Medium |

### 4.3 Trust Score System

Implement an internal trust score:

```
const calculateTrustScore = (user) => { let score = 50; // Start at neutral // Positive signals if (user.emailVerified) score += 10; if (user.phoneVerified) score += 10; if (user.photoVerified) score += 15; if (user.linkedSocial) score += 10; if (user.accountAge > 30) score += 5; if (user.hasMatches && user.noReports) score += 10; // Negative signals if (user.reportsReceived > 0) score -= 10 * user.reportsReceived; if (user.shadowBanned) score = 20; if (user.suspiciousDevice) score -= 15; if (user.vpnDetected) score -= 5; return Math.max(0, Math.min(100, score)); };
```

---

## 5. Fraud Prevention Checklist

### Immediate Implementation

- ■ reCAPTCHA v3 on registration

- ■ Email verification required
- ■ Phone verification required
- ■ Basic rate limiting on all endpoints
- ■ URL detection in messages

### Short-term (30 days)

- ■ Device fingerprinting
- ■ Message spam detection
- ■ Behavioral rate limiting
- ■ Trust score system (basic)

### Medium-term (90 days)

- ■ Photo verification option
- ■ Advanced behavioral analysis
- ■ Machine learning spam detection
- ■ Real-time fraud monitoring dashboard

### Ongoing

- ■ Weekly fraud review
- ■ Pattern analysis updates
- ■ False positive review
- ■ Rule tuning based on new patterns

---

## 6. Fraud Monitoring Dashboard

### Key Metrics to Track

| Metric | Target | Alert Threshold |
|---|---|---|
| Fake account rate | < 5% | > 10% |
| Spam message rate | < 1% | > 3% |

| | | |
|---|---|---|
| User reports rate | < 0.5% | > 2% |
| Bot detection rate | < 1% | > 5% |
| Account suspension rate | < 2% | > 5% |

## Weekly Review Checklist

- ■ Review suspended accounts
- ■ Analyze new spam patterns
- ■ Check false positive reports
- ■ Update detection rules
- ■ Review rate limit effectiveness

---

*Last Updated: December 2024*
*Review Due: March 2025*