📊 Late API Rate Limit Tracking System

Date: November 22, 2025

Executive Summary

✅ **COMPREHENSIVE RATE LIMIT TRACKING IMPLEMENTED!**

Your automated social media posting system now tracks and displays rate limits for all Late API platforms (Instagram, Facebook, LinkedIn, Threads, TikTok, Bluesky) across multiple business profiles (Basketball Factory, Rise As One).

Features Implemented

✅ What's New

1. **Per-Platform Rate Limit Tracking** - Tracks 8 posts/day limit for each platform
2. **Per-Profile Tracking** - Separate counters for Basketball Factory and Rise As One
3. **Real-Time Warning System** - Visual alerts when approaching or reaching limits
4. **Automatic Reset** - Clears counters after 24 hours
5. **UI Integration** - Dashboard banner showing current status

Rate Limit Rules

Late API Limits (Per Profile, Per Platform)

- **Instagram**: 8 posts/day
- **Facebook**: 8 posts/day
- **LinkedIn**: 8 posts/day
- **Threads**: 8 posts/day
- **TikTok**: 8 posts/day
- **Bluesky**: 8 posts/day

Twitter Limits (Separate Tracking)

- **Twitter**: 17 tweets/day (already implemented, separate from Late API)

Warning Thresholds

- 🟢 **Good**: 3+ posts remaining
- 🟡 **Warning**: 1-2 posts remaining
- 🔴 **Critical**: 0 posts remaining (limit reached)

Files Created

1. Rate Limit Library

`lib/late-rate-limit.ts`

- Tracks posts by profile ID + platform
- Stores data in `/tmp/late-rate-limit.json`
- Auto-cleans up posts older than 24 hours
- Functions:
- `recordLatePost()` - Record a successful post
- `getLateRateLimitStatus()` - Get current status
- `canPostToLatePlatform()` - Check if posting is allowed
- `getProfilesWithPlatforms()` - Get all profiles with tracked platforms

2. API Endpoint

`app/api/late/rate-limit/route.ts`

- Provides rate limit status to UI
- Groups data by profile
- Calculates overall status (good/warning/critical)
- Returns counts for platforms at limit

3. UI Component

`components/late-rate-limit-banner.tsx`

- Collapsible alert banner
- Shows per-profile breakdown
- Color-coded status badges
- Auto-refreshes every 2 minutes
- Only displays when issues detected

Integration Points

Modified Files

1. Late API Post Route

`app/api/late/post/route.ts`

- Added import: `recordLatePost`, `canPostToLatePlatform`
- Records successful posts after Late API response
- Tracks each platform separately

```
// Record successful posts for rate limit tracking
if (successPlatforms.length > 0 && profileLateId && profileName) {
  for (const platform of successPlatforms) {
    if (platform.platform && platform.platform.toLowerCase() !== 'twitter') {
      recordLatePost(platform.platform, profileLateId, profileName)
    }
  }
}
```

2. Cloud Storage Series Processor

`lib/cloud-storage-series-processor.ts`

- Added import: `recordLatePost`
- Records posts after successful Late API posting
- Uses profile ID and name from series

```ts
// Record rate limit tracking for successful posts
if (series.profile?.id && series.profile?.name) {
  for (const config of platformConfigs) {
    recordLatePost(config.platform, series.profile.id, series.profile.name);
  }
}
```

3. Dashboard Page

`app/dashboard/page.tsx`

- Added import: `LateRateLimitBanner`
- Displays banner below Twitter rate limit banner
- Auto-refreshes status

How It Works

1. Post Recording

When a post is successfully published to a Late API platform:

```
User creates post → Post succeeds → recordLatePost() called →
Data saved to /tmp/late-rate-limit.json → Console log shows status
```

2. Status Checking

```
UI component → Fetches /api/late/rate-limit →
Reads /tmp/late-rate-limit.json → Calculates stats →
Returns per-profile breakdown
```

3. Data Structure

```json
{
  "posts": [
    {
      "platform": "instagram",
      "profileId": "profile-id-123",
      "profileName": "Basketball Factory",
      "timestamp": 1700000000000
    }
  ],
  "lastCleanup": 1700000000000
}
```

4. Automatic Cleanup

- Runs when data is loaded (if >1 hour since last cleanup)

- Removes posts older than 24 hours
- Keeps file size minimal

UI Display Examples

Collapsed View (Warning)

⚠️   Late API Rate Limit Warning (2 platforms)
You're approaching the daily posting limit (8 posts/day) for 2 platforms.
Consider reducing posting frequency.
(click to expand)

Expanded View

⚠️   Late API Rate **Limit** Warning (2 platforms)

Basketball Factory [warning]
  Instagram: 7/8 [1]
  Facebook: 6/8
  LinkedIn: 5/8

Rise **As** One [good]
  Instagram: 3/8
  TikTok: 2/8

💡 Tip: **Each** platform allows 8 posts per day per profile.
Use different profiles **to** post more frequently, **or** wait 24 hours.

Console Output Examples

When Posting

📊 Late API Rate Limit Status (Basketball Factory - instagram):
  Posts today: 6/8
  Remaining: 2

⚠️   WARNING: Only 2 posts remaining **for** instagram (Basketball Factory) before rate limit!
  Consider reducing posting frequency **for** this platform.

When Limit Reached

📊 Late API Rate Limit Status (Basketball Factory - facebook):
  Posts today: 8/8
  Remaining: 0

🚫 RATE LIMIT REACHED: Cannot post more to facebook (Basketball Factory) **until** reset.
  Limit resets 24 hours after the first post of the day.

Testing Performed

✅ Verified

1. Rate limit tracking for Late API posts
2. Separate tracking per profile and platform
3. Automatic cleanup of old posts (24-hour window)
4. UI banner displays correctly
5. Warning thresholds work as expected
6. Build compiles without errors

Usage Tips

For Users

1. **Monitor the Dashboard**: Check the rate limit banner at the top of the Content Journey page

2. **Plan Your Posts**: If you see warnings, consider:
   - Spacing out posts throughout the day
   - Using different profiles (Basketball Factory vs Rise As One)
   - Scheduling posts for the next day

3. **Multiple Profiles**: Each profile has its own 8-post limit per platform, so switching profiles effectively doubles your daily capacity

4. **24-Hour Reset**: Limits reset 24 hours after the first post of the day to each platform

Comparison: Before vs After

BEFORE

```
❌ No visibility into Late API limits
❌ Could hit limit without warning
❌ No way to know how many posts remain
❌ Had to manually count posts
❌ Couldn't distinguish between profiles
```

AFTER

```
✅ Real-time tracking per platform per profile
✅ Visual warnings when approaching limits
✅ Exact counts of remaining posts
✅ Auto-cleanup of old data
✅ Separate tracking for Basketball Factory & Rise As One
✅ Collapsible UI for detailed breakdown
```

API Endpoints

GET `/api/late/rate-limit`

Returns current rate limit status for all user profiles

Response Example:

```json
{
  "hasData": true,
  "profiles": [
    {
      "profileId": "profile-123",
      "profileName": "Basketball Factory",
      "platforms": [
        {
          "platform": "instagram",
          "count": 7,
          "limit": 8,
          "remaining": 1,
          "statusLevel": "warning",
          "percentageUsed": 88
        }
      ],
      "overallStatus": "warning"
    }
  ],
  "overallStatus": "warning",
  "platformsAtLimit": 0,
  "platformsWithWarning": 2,
  "totalPlatformsTracked": 6
}
```

Technical Details

Storage Location

- File: `/tmp/late-rate-limit.json`
- Format: JSON
- Persistence: Survives server restarts (in /tmp directory)
- Size: Minimal (auto-cleaned)

Performance

- Lightweight tracking (< 1KB per profile)
- Efficient cleanup (runs max once per hour)
- Fast lookups (in-memory after load)
- No database required

Accuracy

- Based on successful Late API responses
- Tracks actual posted content (not attempts)
- Per-profile, per-platform granularity
- 24-hour rolling window

Future Enhancements (Optional)

Potential Features

1. Email/SMS notifications when approaching limits
2. Auto-scheduling to avoid hitting limits
3. Historical usage graphs
4. Export usage reports
5. Custom limit thresholds per platform
6. Integration with Late API's native rate limit info (if/when available)

Troubleshooting

"Banner not showing"

1. Make sure you've posted at least once to a Late API platform
2. Refresh the dashboard page
3. Check browser console for errors
4. Verify `/api/late/rate-limit` endpoint returns data

"Counts seem wrong"

1. Rate limits reset 24 hours after first post
2. Only successful posts are counted
3. Failed posts don't count toward limit
4. Twitter is tracked separately (17/day limit)

"Rate limit file missing"

1. File is created on first successful post
2. Located at `/tmp/late-rate-limit.json`
3. Auto-created by system when needed
4. Will recreate if deleted

Related Systems

Twitter Rate Limit Tracking

- Separate system at `/lib/twitter-api.ts`
- File: `/tmp/twitter-rate-limit.json`
- Limit: 17 tweets/day
- Already implemented and working

Bluesky Connection

- User confirmed: ✅ **FIXED**
- Reconnected via Late API dashboard
- Should now post successfully

Summary

🎉 **ALL RATE LIMIT TRACKING OPERATIONAL!**

✅ **Late API**: 8 posts/day per platform per profile
✅ **Twitter**: 17 tweets/day (separate tracking)
✅ **Real-time warnings**: Visual alerts in dashboard
✅ **Per-profile tracking**: Basketball Factory & Rise As One
✅ **Auto-cleanup**: 24-hour rolling window

Status: Production Ready
Location: Dashboard → Content Journey (top of page)
Refresh: Every 2 minutes (automatic)

Last Updated: November 22, 2025 at 3:15 PM
Checkpoint: "Late API rate limit tracking system"