

# Endpoint Reconciliation Automation

## Standard Operating Procedure

---

### Tools & Pricing

Tool	Purpose	Monthly Cost
n8n	Automation platform	\$20
Airtable	Data storage and tracking	Free (within limits)
Claude API (optional)	AI-powered report generation	Pay per use (~\$1-5/month)

**Total Additional Cost: \$20-25/month**

---

### Purpose & Value

This automation reconciles all client endpoints across your core MSP tools, running monthly to identify discrepancies that lead to billing gaps, security holes, and service failures. Instead of manually checking spreadsheets from multiple systems, you get a single actionable report.

#### What You'll Catch:

- Endpoints missing from PSA (revenue leakage)
- Unprotected machines without antivirus
- Failed RMM agents (online in antivirus but offline in RMM)
- Undocumented endpoints in IT Glue
- Abandoned machines needing cleanup
- Missing client portal agents

**Time Saved:** 2-4 hours monthly in manual reconciliation work

---

# Architecture Overview

The automation uses a **modular sub-workflow design** where each sub-workflow handles one specific task. This makes it easy to add or remove tools as your stack changes.

## Master Workflow Flow:

1. Setup Database (establish client list)
2. Get Endpoints from RMM (source of truth)
3. Pull Endpoints from SentinelOne
4. Pull Endpoints from PSA
5. Pull Endpoints from CloudRadial
6. Pull Endpoints from IT Glue
7. Generate Report

Each sub-workflow updates a central Airtable database, building a complete picture of endpoint coverage across all systems.

---

## Airtable Database Structure

### Clients Table

Stores all active clients and their system identifiers for cross-platform matching.

#### Fields:

- Client Name (text)
- PSA Client ID (text)
- RMM Site UID (text)
- Last Run Timestamp (date/time)

### Endpoints Table

One record per endpoint, updated by each sub-workflow with coverage status.

#### Fields:

- Device Name (text)
- Client (linked record to Clients table)
- RMM Device ID (text)

- RMM Site ID (text)
  - RMM UID (text) - used for CloudRadial matching
  - Last Seen Date (date/time)
  - Online Status (checkbox) - from RMM
  - Has PSA Config (checkbox)
  - Autotask Contract ID (text)
  - Has IT Glue Config (checkbox)
  - Has Sentinel1 (checkbox)
  - Sentinel1 Online (checkbox)
  - Sentinel1 Infected (checkbox)
  - Sentinel1 Up To Date (checkbox)
  - Sentinel1 Ranger Status (text)
  - Sentinel1 Last Active (date/time)
  - Sentinel1 Active Threats (number)
  - Sentinel1 Exempt (checkbox) - manual field for exceptions
  - Has CloudRadial (checkbox)
- 

## Sub-Workflow Details

### 1. Setup Database

**Purpose:** Establish the foundation by syncing all clients from PSA into Airtable.

**Process:**

- Query Autotask PSA API for all active clients
- Split into batches for processing
- For each client:
  - Check if already exists in Airtable Clients table
  - If exists: Update timestamp
  - If new: Create record with PSA Client ID
- Query Datto RMM API for all sites
- Match RMM sites to PSA clients by name
- Store RMM Site UID in matching client record

**Why PSA is Source of Truth:** MSPs bill from PSA, so this is your most accurate client list.

---

## 2. Get Endpoints from RMM

**Purpose:** Pull all endpoints into Airtable as the baseline for reconciliation.

**Process:**

- Authenticate to Datto RMM API
- Query for all devices across all sites
- For each device, create/update record in Endpoints table:
  - Device Name
  - RMM Device ID
  - RMM Site ID
  - Link to Client (matched via Site ID)
  - Last Seen Date
  - Online Status (boolean)
  - RMM UID (for later matching)

**Why RMM is Source of Truth for Endpoints:** If it's not in RMM, you're not monitoring it.

---

## 3. Pull Endpoints from SentinelOne

**Purpose:** Check antivirus coverage and health across all endpoints.

**Process:**

- Authenticate to SentinelOne API
- Query for all agents/endpoints
- Split into batches and extract relevant data
- Match to Endpoints table by hostname
- Update matched records with:
  - Has Sentinel1 (boolean)
  - Sentinel1 Online (boolean)
  - Sentinel1 Infected (boolean)
  - Sentinel1 Up To Date (boolean)
  - Ranger Status (text)
  - Sentinel1 Last Active (date/time)
  - Sentinel1 Active Threats (number)

**Key Insight:** Compare Sentinel1 Last Active vs RMM Last Seen to identify agent failures.

---

## 4. Pull Endpoints from PSA

**Purpose:** Verify endpoints are configured for billing.

**Process:**

- Authenticate to Autotask PSA API
- Query for all configuration items (workstations/servers)
- Format and normalize data
- Match to Endpoints table by device name
- Update matched records with:
  - Has PSA Config (boolean)
  - Autotask Contract ID (text)

**Revenue Impact:** Missing PSA configs mean you're supporting unbilled endpoints.

---

## 5. Pull Endpoints from CloudRadial

**Purpose:** Check client portal agent installation.

**Process:**

- Authenticate to CloudRadial API
- Query for all endpoints with installed agents
- Match to Endpoints table by hostname
- Update matched records with:
  - Has CloudRadial (boolean)

**Cleanup Note:** CloudRadial often retains decommissioned machines—use this to identify orphaned records.

---

## 6. Pull Endpoints from IT Glue

**Purpose:** Verify endpoint documentation exists.

**Process:**

- Authenticate to IT Glue API
- Query configuration items with filters:

- Organization ID (active clients only)
- Active status (not archived)
- Type ID (endpoints only—exclude printers, switches, routers)
- Match to Endpoints table by hostname
- Update matched records with:
  - Has IT Glue Config (boolean)

**Documentation Matters:** Undocumented endpoints create confusion during incident response.

---

## 7. Generate Report (Optional - AI-Powered)

**Purpose:** Transform raw data into actionable insights.

**Process:**

- Query all Endpoints table records
- Format as JSON or CSV
- Send to Claude API with structured prompt (see example below)
- Receive prioritized report with:
  - Executive summary
  - Critical issues (unprotected machines, billing gaps)
  - Important issues (outdated agents, missing docs)
  - Compliance statistics
  - Cleanup candidates

**Alternative:** Export Airtable to Excel and manually review, or create Airtable views with filters for common issues.

---

## Example Claude API Prompt

*You are analyzing MSP endpoint reconciliation data. Generate a report with these sections:*

*EXECUTIVE SUMMARY*

- *Total endpoints analyzed*
- *Fully compliant count*
- *Endpoints requiring attention*
- *Critical issues needing immediate action*

*CRITICAL ISSUES (Priority 1)*

- Missing SentinelOne (unprotected)
- Online in SentinelOne but offline in RMM (agent failure)
- Active infections/threats
- Missing PSA config (billing leakage)

#### IMPORTANT ISSUES (Priority 2)

- Outdated SentinelOne agents
- Missing CloudRadial
- Missing IT Glue documentation
- Offline >30 days (decommission candidates)

#### COMPLIANCE SUMMARY

- Workstations: X fully compliant, Y missing 1 tool, Z missing 2+ tools
- Servers: [same breakdown]

#### CLEANUP CANDIDATES

- Endpoints offline >60 days in both RMM and SentinelOne

FORMAT: Clear headers, device name + client + issue + recommended action.

Do NOT list fully compliant endpoints. Exclude "Sentinel1\_Exempt" devices.

[Insert JSON data here]

---

## Implementation Tips

**Start Small:** Build one sub-workflow at a time. Test with one client before running across all clients.

**Error Handling:** Add try/catch blocks in n8n. If one API fails, log the error and continue with other checks.

**API Rate Limits:** Add delays between API calls (1-2 seconds) to avoid throttling.

**Matching Logic:** Hostname matching isn't perfect. Standardize naming conventions across systems where possible.

**Scheduling:** Run monthly on the 1st at 2 AM. Adjust based on your change management schedule.

**Adapt to Your Stack:** Don't have CloudRadial? Remove that sub-workflow. Use Hudu instead of IT Glue? Swap the API calls.

---

## Next Steps

1. **Set up Airtable base** with Clients and Endpoints tables
  2. **Get API credentials** for all systems (store securely in n8n)
  3. **Build sub-workflows one at a time** in n8n, testing each independently
  4. **Create master workflow** that calls sub-workflows in sequence
  5. **Test with 1-2 clients** before full deployment
  6. **Schedule monthly runs** and refine based on results
- 

## Need Help?

If you want this built for you, customized to your exact tool stack, or integrated with automated ticketing, contact Q Labs at [dennis@qlabs.dev](mailto:dennis@qlabs.dev) or visit <https://qlabs.dev>

---

*This SOP provides the logical framework and starting point. Implementation details will vary based on your specific tools and API capabilities. Use AI tools like Claude to help write API integration code as you build.*