**PRODUCT SECURITY OVERVIEW**

Metal Finishing Calculator v3.0.0

|  |  |
| --- | --- |
| **Document Version:** | 1.0 |
| **Date:** | January 30, 2026 |
| **Prepared By:** | Psyrcuit LLC |
| **Classification:** | Business Confidential |

# 1. Executive Summary

Metal Finishing Calculator is a 100% offline desktop application designed for generating professional industrial finishing estimates. The application has zero network connectivity, no cloud integration, no user accounts, no telemetry, and no data transmission capabilities of any kind. All data remains exclusively on the customer's local machine within their existing security boundary. The application inherits and operates within the security controls of the host workstation, making it inherently compatible with ITAR, CUI, and CMMC-controlled environments.

# 2. Architecture Overview

Metal Finishing Calculator is built on Electron, a desktop application framework that packages web technologies for offline use. All dependencies are bundled locally within the application—there are no CDN calls, external API connections, or runtime downloads.

## Core Architecture

|  |  |
| --- | --- |
| **Attribute** | **Value** |
| **Application Name** | Metal Finishing Calculator |
| **Version** | 3.0.0 |
| **Vendor** | Psyrcuit LLC |
| **Platform** | Windows Desktop (Electron 40.x) |
| **Network Connectivity** | **NONE - 100% Offline** |
| **Data Transmission** | **NONE - All data remains local** |
| **Cloud Services** | **NONE - No cloud integration** |
| **User Accounts** | **NONE - License key only** |
| **Telemetry/Analytics** | **NONE - No data collection** |

# 3. Security Posture

The application's offline architecture eliminates entire categories of security risks common to modern software:

|  |  |
| --- | --- |
| **Security Aspect** | **Implementation** |
| **Network Attack Surface** | Eliminated - No network sockets, APIs, or external connections |
| **Data in Transit** | N/A - No data transmission occurs |
| **Authentication** | N/A - No user accounts to compromise |
| **Session Management** | N/A - No persistent connections to intercept |
| **Context Isolation** | Enabled - Renderer cannot access Node.js APIs directly |
| **Node Integration** | Disabled - Prevents code injection attacks |
| **Sandbox Mode** | Disabled - Required for offline MAC-based license validation |
| **DevTools Access** | Available (IP protection warnings in demo mode only) |
| **Window Security** | Multi-monitor lock prevents accidental data exposure |

# 4. Data Handling Summary

All application data is stored locally in JSON format within directories controlled by the customer:

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Storage Location** | **Customer Control** |
| **Quote Files** | Customer-specified directory | Full control - location, retention, deletion |
| **Settings** | %APPDATA% or Shared Hub folder | Full control - configurable location |
| **PDF Exports** | Customer-specified directory | Full control - location, retention, deletion |
| **License Key** | Within settings file | Stored locally, never transmitted |

# 5. What We Do NOT Collect

Psyrcuit LLC has no mechanism to collect, receive, access, or view any customer data. The application contains no code for:

|  |  |
| --- | --- |
| ✗ Telemetry or usage analytics | ✗ Crash reporting |
| ✗ User behavior tracking | ✗ Feature usage metrics |
| ✗ Customer information collection | ✗ Quote or pricing data transmission |
| ✗ License server communication | ✗ Auto-update mechanisms |
| ✗ Background network requests | ✗ Third-party service integrations |

# 6. License System

The licensing system is designed for complete offline operation:

|  |  |
| --- | --- |
| **License Aspect** | **Implementation** |
| **Validation Method** | Offline cryptographic validation using MAC address hash |
| **License Server** | None - No server communication required |
| **Key Storage** | Local settings file on customer machine |
| **Hardware Binding** | Tied to computer's MAC address for single-machine use |
| **Activation Process** | One-time key entry; validated locally without network |

# 7. Development Disclosure

In the interest of transparency, Psyrcuit LLC discloses that AI coding assistants were utilized during the development of this application. Important clarifications:

|  |
| --- |
| • AI tools were used solely for code generation and debugging during development |
| • AI tools never had access to customer data, protected information, or production systems |
| • All generated code was reviewed and compiled locally by Psyrcuit developers |
| • The production application contains NO AI integration whatsoever |
| • The application operates entirely offline with no AI or cloud dependencies |

# 8. Add-On System Security

Metal Finishing Calculator v3.0.0 supports optional add-on modules that extend core functionality. All add-ons are subject to mandatory cryptographic signature verification before execution.

## 8.1 Add-On Architecture

Add-ons are sandboxed HTML/JavaScript modules that run within the application’s renderer process. The add-on system enforces strict capability boundaries:

|  |  |
| --- | --- |
| **Capability** | Status |
| **Read quote data** | PERMITTED – Read access to active quote fields |
| **Write quote data** | PERMITTED – Can populate quote fields via defined API |
| **Read application settings** | PERMITTED – Read-only access to non-sensitive settings |
| **Write application settings** | BLOCKED – Cannot modify application configuration |
| **Filesystem access** | BLOCKED – No direct filesystem read/write |
| **Network access** | BLOCKED – No outbound HTTP, WebSocket, or fetch |
| **Node.js integration** | BLOCKED – nodeIntegration disabled in renderer |
| **Execute system commands** | BLOCKED – No shell or child\_process access |

## 8.2 Cryptographic Signature Verification

All add-ons must include a valid cryptographic signature file (addon.sig) before the application will permit execution. The signing infrastructure uses the Ed25519 elliptic curve algorithm:

|  |  |
| --- | --- |
| **Component** | Detail |
| **Algorithm** | Ed25519 (Curve25519 elliptic curve) |
| **Hash Function** | SHA-256 (content digest before signing) |
| **Public Key Storage** | Hardcoded in application binary (main.js) |
| **Private Key Storage** | Offline, held exclusively by Psyrcuit LLC |
| **Signature File** | addon.sig (per add-on directory) |
| **Signing Tool** | admin\_tools/Addon\_Signer.js (not distributed) |
| **Key Generator** | admin\_tools/Addon\_Key\_Generator.js (not distributed) |

## 8.3 Signature Enforcement Points

Signature verification occurs at five distinct enforcement points, creating a defense-in-depth chain:

|  |  |
| --- | --- |
| **Enforcement Point** | Behavior on Failure |
| **1. Installation (file copy)** | Add-on files are copied but flagged as unverified |
| **2. Post-copy verification** | Immediate signature check after installation completes |
| **3. Application launch** | All installed add-ons re-verified at every startup |
| **4. On-demand re-verification** | Manual re-check available via Add-On Manager UI |
| **5. Renderer load gate** | Unverified add-ons are visible in UI but inert – all interactive controls are disabled |

## 8.4 Manifest Validation

Each add-on must include an addon\_manifest.json file containing required metadata fields (id, name, version, description, author, entryPoint). The manifest is validated at load time. Missing or malformed manifests prevent add-on registration regardless of signature status.

## 8.5 Authorization Policy and Liability

**WARNING: Only add-ons cryptographically signed by Psyrcuit LLC are authorized for execution. Psyrcuit LLC accepts no responsibility for any add-on whose signature has been tampered with, bypassed, or removed through modification of the application binary. Unauthorized modification of signature verification logic voids all warranties and support agreements.**

## 8.6 ITAR Compliance

The add-on system operates entirely offline. No add-on code, data, or metadata is transmitted externally. Add-on installation is performed via local file copy only. This architecture is compatible with air-gapped and ITAR-controlled environments.

## 8.7 Future Enhancements

Planned improvements include add-on permission declarations in manifests, granular capability grants per add-on, and administrative controls for enterprise deployment policies.

For security inquiries, contact: security@psyrcuit.com

Psyrcuit LLC • Tucson, Arizona • psyrcuit.com