Blender Security Scanner

A security addon for Blender that detects and prevents malicious Python scripts in blend files. Protects against malware campaigns targeting Blender users through malicious addons and embedded scripts.

⚠️ Security Notice

This addon provides security improvements but does not guarantee complete protection against all malware. As an open-source tool, its detection methods are publicly visible and can potentially be bypassed by sophisticated attackers. However, it significantly improves security compared to having no protection and serves as an important first line of defense against common malware patterns.

Features

Threat Detection

- Real-time scanning of Python scripts in blend files
- Obfuscation detection for hex, unicode, and character substitution
- Pattern matching for 60+ malicious code signatures
- Multi-severity classification (Critical, High, Medium)
- Auto-scan on file load with configurable settings

Protection Mechanisms

- Timer blocking prevents auto-execution of malicious timers
- Handler protection blocks suspicious event handler registration
- Quarantine system safely disables malicious code
- Whitelist management for trusted scripts and addons

Detects Common Threats

- **Subprocess execution** (`subprocess.call`, `os.system`)
- **Code injection** (`exec()`, `eval()`, dynamic imports)
- **Network operations** (requests, urllib, data exfiltration)
- **PowerShell execution** (hidden windows, command injection)
- **Auto-execution** (timers, handlers, persistence mechanisms)
- **Encoding operations** (base64, payload decoding)
- **File operations** (suspicious writes, persistence files)

Obfuscation Detection

- **Hex encoding**: `\x65\x78\x65\x63` (encoded 'exec')
- **Unicode encoding**: `\u0065\u0078\u0065\x63` (encoded 'exec')
- **Character substitution**: `p0w3rsh3ll` instead of 'powershell'
- **Dynamic access**: `getattr(__builtins__, 'exec')`
- **String concatenation**: `'ex' + 'ec'` obfuscation

Installation

Method 1: Download Release

1. Download the latest `blender_security_scanner.py` from Releases
2. Open Blender → Edit → Preferences → Add-ons
3. Click "Install..." and select the downloaded file
4. Enable "System: Blender Security Scanner"

Method 2: Manual Installation

1. Clone this repository or download the source code
2. Copy `blender_security_scanner.py` to your Blender addons directory:
   - **Windows**: `%APPDATA%\Blender Foundation\Blender\4.4\scripts\addons\`
   - **macOS**: `~/Library/Application Support/Blender/4.4/scripts/addons/`
   - **Linux**: `~/.config/blender/4.4/scripts/addons/`
3. Restart Blender and enable the addon in Preferences

Usage

Quick Start

1. Install and enable the addon
2. Open any blend file - auto-scan will run automatically
3. View results in the security panel or threat dialog
4. Take action by quarantining threats or whitelisting safe scripts

Access Points

- **File Menu**: `File → Security Scanner`
- **3D Viewport**: Press `N` → Security tab
- **Text Editor**: Press `N` → Security tab

- **Python Console**: `bpy.ops.security.scan_blend_file()`

Security Panel Features

- **Scan Current File**: Manual security scan
- **Security Status**: Real-time threat overview
- **Whitelist Management**: Manage trusted scripts
- **Tools & Testing**: Debug and configuration options

Threat Management

- **View Threats**: Detailed threat information with context
- **Show Context**: See 3 lines before/after suspicious code
- **Quarantine**: Safely disable malicious scripts
- **Whitelist**: Mark trusted scripts to skip future scans

How It Works

Detection Engine

The security scanner uses regex pattern matching to identify malicious code patterns:

```python
# Example patterns detected:
subprocess.call(['malicious_command'])    # Direct subprocess
e\x78ec('malicious_code')                 # Hex obfuscated 'exec'
getattr(__builtins__, 'eval')             # Dynamic function access
bpy.app.timers.register(malicious_func)  # Auto-execution
```

Scanning Process

1. **File Load Detection**: Monitors blend file opening
2. **Script Enumeration**: Finds all Python text blocks
3. **Pattern Analysis**: Scans for malicious signatures
4. **Threat Classification**: Assigns severity levels
5. **User Notification**: Displays results and options

Protection Mechanisms

- **Timer Override**: Replaces `bpy.app.timers.register` with secure version
- **Whitelist System**: Allows legitimate Blender functions while blocking suspicious ones
- **Auto-Scan**: Automatically scans files on load (configurable)

Configuration

Auto-Scan Settings

- Enable/disable automatic scanning on file load
- Configure scan sensitivity levels
- Manage whitelist for trusted scripts

Whitelist Management

- Add scripts to whitelist during threat detection
- Bulk whitelist management interface
- Reason tracking for whitelisted items

Security Limitations

Important Disclaimers

- **Not foolproof**: Sophisticated malware may evade detection
- **Open source**: Detection methods are publicly visible
- **False positives**: Legitimate code may trigger alerts
- **No real-time protection**: Only scans when explicitly triggered
- **Python-only**: Does not detect non-Python threats

Best Practices

- Keep updated: Install latest versions for newest threat signatures
- Verify sources: Only download blend files from trusted sources
- Regular scans: Manually scan files from unknown sources
- Backup important work: Maintain clean backups of projects
- Report issues: Help improve detection by reporting false positives/negatives

Testing

Test Script

Use the included `comprehensive_security_test.py` to verify all detection methods:

```python
# Creates 60+ test threats across all categories
# Safe for testing - no actual malicious code executes
# Tests: subprocess, exec/eval, network, obfuscation, etc.
```

Expected Results

- **Critical threats**: Timer registration, code execution, subprocess calls
- **High severity**: Network requests, PowerShell execution
- **Medium severity**: File operations, suspicious strings, encoding
- **Obfuscation**: Hex/unicode encoding, character substitution

Changelog

v1.0.0 (2025)

- Initial release
- Core threat detection engine
- Auto-scan functionality
- Whitelist management system
- UI with threat context
- Obfuscation detection (hex, unicode, substitution)
- Quarantine system
- Real-time timer protection

License

This project is licensed under the MIT License - see the LICENSE file for details.

Support

- **Issues**: GitHub Issues for bugs, feature requests, and minor issues
- **Discussions**: GitHub Discussions for general questions and feedback
- **Security Reports**: Please report severe security vulnerabilities privately

Responsible Disclosure

If you discover security vulnerabilities in this addon:

1. **Severe vulnerabilities**: Contact the maintainers privately, do not create public issues
2. **Minor security issues**: Can be reported via GitHub Issues
3. Provide detailed information about the vulnerability
4. Allow reasonable time for fixes before public disclosure

Please note: This addon is maintained in my free time and I cannot guarantee future updates, bug fixes, or immediate responses to security reports.

Remember: This tool improves security but cannot guarantee complete protection. Always practice safe computing habits and verify the sources of blend files you open.

Stay safe, stay creative!