Application Whitelisting & System Lockdown

Protecting OT Endpoints Through Allowlisting and Lockdown

OT Security Learning Series

Document 810 | January 2026

Contributors: Matthias Niedermaier

Created with AI assistance

LEARNING RESOURCE

Contents

1 Introduction

In Operational Technology environments, endpoints such as HMIs, engineering workstations, and SCADA servers are critical assets that directly interact with industrial processes. Unlike traditional IT environments where regular patching and antivirus updates are standard practice, OT systems often run for years without updates due to availability requirements and vendor certification constraints.

Information

Application Whitelisting (also called allowlisting) is a security approach that permits only pre-approved applications to execute, blocking all others by default. Combined with **System Lockdown**, it provides a robust defense-in-depth strategy for OT endpoints.

1.1 Why Traditional Antivirus Falls Short in OT

- Requires frequent signature updates (network connectivity, maintenance windows)
- Cannot detect zero-day threats or targeted attacks
- May flag legitimate industrial software as suspicious
- Resource-intensive scanning can impact real-time performance
- Reactive approach: detects known threats after they exist

Key Point

Application whitelisting takes a **proactive** approach: instead of trying to identify malicious software, it only allows known-good applications to run. This is particularly effective in OT environments where the software inventory is relatively static.

2 Core Concepts

2.1 Application Whitelisting

Application whitelisting works by maintaining a list of approved executables, scripts, and libraries that are permitted to run on a system. Any attempt to execute unlisted software is blocked.

Whitelisting Methods

Hash-based	Cryptographic hash (SHA-256) of each approved file
Path-based	Allow execution from specific directories only
Certificate-based	Trust applications signed by specific publishers
Publisher-based	Allow all software from trusted vendors

2.1.1 Hash-Based Whitelisting

- Most secure method – exact file matching
- Requires updating hashes after every legitimate software change
- Best for static OT environments with infrequent changes

2.1.2 Certificate-Based Whitelisting

- Trusts all executables signed by approved certificates
- Easier maintenance – new versions automatically trusted
- Risk: compromised signing certificates could allow malware

2.2 System Lockdown

System lockdown goes beyond application whitelisting to restrict system configuration and prevent unauthorized changes.

System Lockdown Components

- **Write Protection:** Prevent modifications to system files and directories
- **Registry Protection:** Lock critical registry keys (Windows)
- **Device Control:** Block unauthorized USB devices and removable media
- **Network Lockdown:** Restrict network connections to approved endpoints
- **User Restrictions:** Limit user privileges and disable unnecessary accounts

3 OT-Specific Considerations

3.1 Challenges in OT Environments

Warning

Implementing application whitelisting in OT requires careful planning. Incorrect configuration can block critical process control software and cause operational disruptions.

- **Legacy Systems:** Older Windows versions (XP, 7) may have limited whitelisting support
- **Vendor Software:** Industrial applications may use unusual execution patterns
- **Change Management:** Software updates require whitelist maintenance
- **Real-Time Requirements:** Whitelisting overhead must not impact process timing
- **Availability:** Cannot risk blocking legitimate control system software

3.2 OT Endpoint Categories

Different endpoint types require tailored approaches:

Endpoint Type	Change Frequency	Recommended Approach
HMI Stations	Low	Hash-based, full lockdown
Engineering Workstations	Medium	Certificate-based, partial lockdown
SCADA Servers	Low	Hash-based, full lockdown
Historian Servers	Low-Medium	Certificate-based, write protection
Jump Servers (DMZ)	Medium	Certificate-based, strict device control

3.3 Purdue Model Integration

Tip

Apply stricter lockdown policies at lower Purdue levels where systems are more critical and changes are less frequent.

- Level 0 Level 1 – Maximum lockdown, hash-based whitelisting
- Level 2 – Full lockdown with controlled exceptions for engineering tools
- Level 3 – Certificate-based whitelisting, device control
- DMZ – Strict whitelisting on jump servers and data transfer systems

4 Implementation Guide

4.1 Phase 1: Discovery and Baselining

Before enabling enforcement, inventory all legitimate software:

1. **Audit Mode:** Run whitelisting solution in monitor-only mode
2. **Baseline Creation:** Capture all running executables and DLLs
3. **Vendor Coordination:** Identify all vendor-required software components
4. **Documentation:** Record software inventory with business justification

Information

Run the discovery phase for at least 2-4 weeks to capture all operational scenarios, including startup sequences, scheduled tasks, and maintenance activities.

4.2 Phase 2: Policy Development

1. **Define Trust Sources:**
   - Operating system components

- Industrial software vendors (Siemens, Rockwell, ABB, etc.)
- Approved security tools
- Internal utilities (signed with corporate certificate)

2. **Create Execution Rules:**
   - Allow by hash for critical control software
   - Allow by certificate for vendor applications
   - Allow by path for specific directories (with caution)
3. **Define Exceptions Process:** Document how to request new software approval

4.3 Phase 3: Staged Rollout

Critical

Never enable enforcement on all systems simultaneously. A phased approach prevents widespread disruption if policies are incomplete.

1. Start with non-critical systems (test environments, non-production)
2. Monitor for false positives and policy violations
3. Refine policies based on findings
4. Gradually expand to production systems
5. Maintain audit logging for troubleshooting

4.4 Phase 4: Maintenance

Ongoing activities to keep whitelisting effective:

- **Change Management Integration:** Update whitelist before software deployments
- **Regular Reviews:** Audit whitelist entries for obsolete software
- **Incident Response:** Investigate all blocked execution attempts
- **Vendor Coordination:** Obtain hashes/certificates before updates

5 Commercial Solutions

Several vendors offer application whitelisting solutions suitable for OT:

OT-Focused Solutions

Carbon Black App Control Enterprise solution with OT support
McAfee Application Control Wide OS support including legacy Windows
Symantec Critical System Protection Designed for fixed-function devices
Honeywell Secure Connection ICS-specific whitelisting
Claroty Edge OT-native endpoint protection

5.1 Windows Built-in Options

Microsoft Technologies

AppLocker	Available in Enterprise editions (Win 7+)
WDAC	Windows Defender Application Control (Win 10+)
SRP	Software Restriction Policies (legacy, all editions)

Warning

Built-in Windows solutions may lack centralized management and OT-specific features. Evaluate carefully for production OT use.

6 Best Practices

6.1 Do's

- [x] Start in audit/monitor mode before enforcement
- [x] Involve OT operators and vendors in policy development
- [x] Document all approved software and business justification
- [x] Integrate with change management processes
- [x] Maintain offline backup of whitelist configuration
- [x] Test policies in non-production environments first
- [x] Plan for emergency bypass procedures

6.2 Don'ts

- [ ] Don't enable enforcement without thorough baselining
- [ ] Don't rely solely on path-based rules (easily bypassed)
- [ ] Don't forget about scripts (PowerShell, VBScript, batch files)
- [ ] Don't ignore DLLs and libraries (DLL hijacking attacks)
- [ ] Don't deploy during critical production periods

- Don't assume vendor software is static (background updaters)

6.3 Emergency Procedures

Critical

Always have a documented procedure to disable whitelisting in emergencies. Process safety must take priority over security controls.

Prepare for scenarios where whitelisting must be bypassed:

- Local administrator override capability
- Emergency boot media with whitelisting disabled
- Out-of-band management access
- Documented escalation procedures

7 IEC 62443 Alignment

Application whitelisting supports multiple IEC 62443 requirements:

Requirement	How Whitelisting Helps
SR 3.4	Software and information integrity – prevents unauthorized code execution
SR 7.6	Network and security configuration settings – system lockdown protects configurations
SR 2.1	Authorization enforcement – restricts what software users can run
CR 7.2	Resource availability – prevents resource exhaustion from malware

7.1 Security Levels

- SL 1 – Basic: Antivirus may be sufficient
- SL 2 – Standard: Application whitelisting recommended
- SL 3 – Enhanced: Whitelisting with full system lockdown required
- SL 4 – Critical: Hardware-enforced integrity, strict whitelisting

8 Summary

Key Takeaways

- Application whitelisting is more effective than antivirus for static OT environments
- Combine with system lockdown for comprehensive endpoint protection
- Thorough baselining and phased rollout are critical for success
- Maintain emergency bypass procedures for operational safety
- Integrate with change management to keep whitelist current
- Stricter policies for lower Purdue levels, more flexibility higher up

Tip

Application whitelisting transforms endpoint security from a reactive "find the bad" approach to a proactive "allow only good" model – perfectly suited for the predictable, stable nature of OT systems.

9 Further Reading

Standards and Guidelines

- **NIST SP 800-167** – Guide to Application Whitelisting
  https://csrc.nist.gov/publications/detail/sp/800-167/final
- **NIST SP 800-82 Rev. 3** – Guide to OT Security
  https://csrc.nist.gov/pubs/sp/800/82/r3/final
- **IEC 62443-3-3** – System Security Requirements and Security Levels
  https://www.isa.org/standards-and-publications/isa-standards/isa-iec-62443-series-of-standards

Government Resources

- **ASD Essential Eight** – Application Control Guidance
  https://www.cyber.gov.au/business-government/asds-cyber-security-frameworks/essential-eight
- **CISA** – ICS Defense-in-Depth Strategies
  https://www.cisa.gov/resources-tools/resources/recommended-practice-improving-industrial-control-system-cybersecurity-defense-depth-strategies
- **NSA/CISA** – Top 10 Cybersecurity Misconfigurations
  https://www.cisa.gov/news-events/cybersecurity-advisories/aa23-278a

Technical Documentation

- **Windows Defender Application Control (WDAC)**
  https://learn.microsoft.com/en-us/windows/security/application-security/application-control/windows-defender-application-control/
- **AppLocker Documentation**
  https://learn.microsoft.com/en-us/windows/security/application-security/application-control/windows-defender-application-control/applocker/applocker-overview

Books

- Knapp, E. & Langill, J. – *Industrial Network Security* (Syngress)
- Macaulay, T. & Singer, B. – *Cybersecurity for Industrial Control Systems* (CRC Press)

Part of the OT Security Learning Series