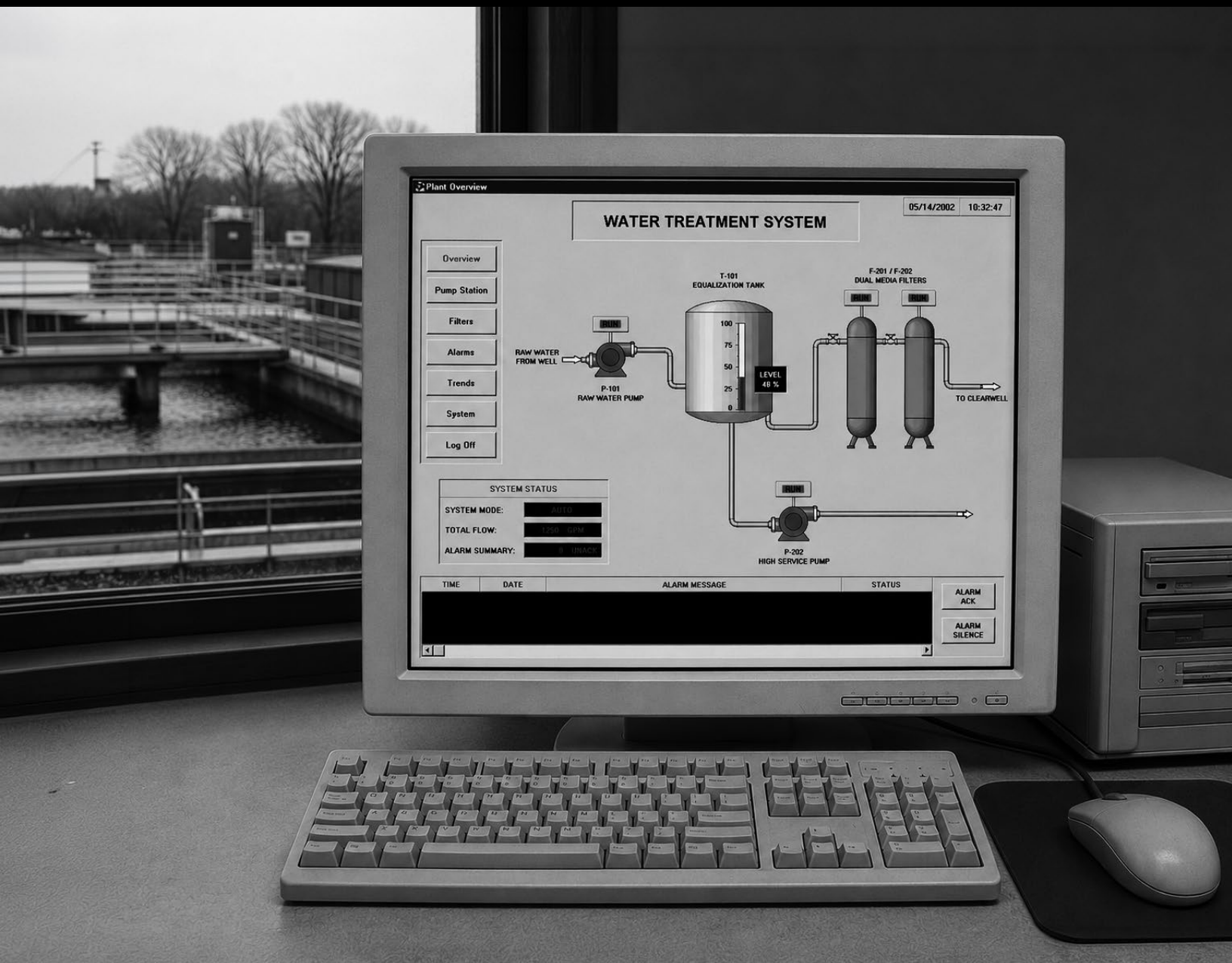
DRAGOS

WHITE PAPER

Practical Considerations for Legacy Commodity Malware Cleanup in OT

Lesley Carhart | Principal Industrial Incident Responder, Dragos Australia

Jan Hoff | Digital Forensics and Incident Response Lead, Dragos Germany



Executive Summary

Industrial organizations across every sector and geography are making an unpleasant and unexpected discovery as they deploy detection capability for the first time: their operational technology (OT) networks are significantly infected. Legacy commodity malware, in some cases dozens of concurrent infections, have been quietly persisting in industrial environments for years - in some cases for decades. The Dragos Incident Response Team is receiving a growing volume of activations and enquiries from organizations that are overwhelmed by these infections, often as a byproduct of their first formal cybersecurity assessment or threat hunt. Conficker, WannaCry, Sality, and a broader ecosystem of worms are still actively propagating across networks that were never cleaned because they were never visible, and because cleaning and preventing infection poses an increasing challenge and operational risk.

When an end user plugs a USB drive infected with Conficker into a typical modern Windows enterprise system, its anti-virus or endpoint detection and response agent will almost instantly block and quarantine the malware and generate an alert message. Unfortunately, older OT environments may contain a variety of end-of-life operating systems, lack effective antimalware solutions, and have very limited security detection. Some of their systems are decades old, must stay in production, and have limited support tooling. Vendor contracts that forbid modification of systems complicate remediation further. Remediation of malware infections in these environments must be performed with significant risk management and planning, or it may fail repeatedly, or cause consequences worse than the infection itself.

Conversely, it is also inadvisable to ignore these infections. Dragos has observed commodity malware causing increasing system and process instability as computers age, networks are further interconnected, and bandwidth becomes a premium in older network environments. Regulatory and legal complications may arise from maintaining an infected network.

This paper exists because no dedicated guidance does. Enterprise incident response playbooks are not designed for legacy OT environments; applying them without adaptation has caused operational incidents. However, leaving infections in place can also have unwanted side effects, which increase as systems age. This paper provides security leaders with a foundation for

responding to legacy commodity malware in OT environments by choosing between three different risk-based remediation strategies: immediate, planned project, or documented deferral, as well as guidance on how to execute and validate remediation. Unlike standard enterprise playbooks, this paper considers OT-specific problems, and it is based on methodology currently put into practice by Dragos's IR team.

When handling a legacy malware infection, three paths are available. This whitepaper will cover each potential approach as well as its risks and benefits.

PATH 01

Immediate Cleanup

Remove now. Accept the operational disruption to eliminate the threat completely.

PATH 02

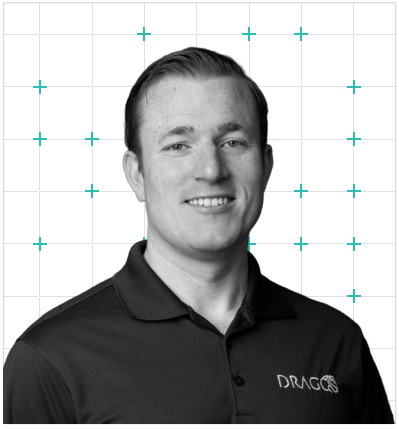
Planned Cleanup

Schedule remediation during a downtime window. Balance thoroughness with operational continuity.

PATH 03

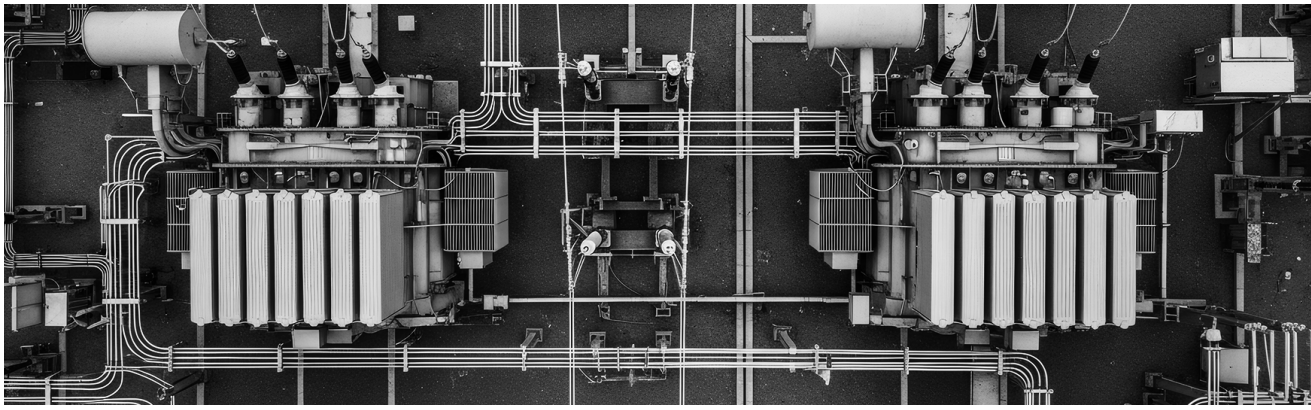
Leave in Place

Accept the risk formally. Understand the consequences and monitor accordingly.



> Conficker has done more damage to industrial orgs than China, Russia, and Iran combined. When you do not maintain your OT networks, do not have backups, and cannot do anything beyond trying to keep people out – Conficker will catch up to you eventually.
>
> **– Robert M. Lee, CEO, Dragos**

Table of Contents



SECTION 01:

Introduction

We live in a transformative era of Operational Technology (OT) cybersecurity. Industrial organizations are investing more in cybersecurity maturity than at any point in the history of the discipline. Organizations are implementing detection, assessments, and threat hunting programs.

Alongside the state threat groups and ransomware operators that dominate industry reporting, a quieter and more pervasive problem is present in industrial environments across every sector and geography: legacy commodity malware, in some cases, dozens of concurrent old infections, quietly persisting in networks where it has gone undetected and uncleaned for years. In some cases, for decades. Organizations are doing more, and they are surprised to find that legacy malware is widespread.

CUSTOM OT MALWARE

TRISIS • Stuxnet • Industroyer

ORIGIN
Nation-state / APT
State-sponsored, targeted, long-term

TARGET
Specific OT systems
PLCs, RTUs, Safety Instrumented Systems

DELIVERY
Supply chain / targeted intrusion
Spear-phishing, hardware implants

DETECTION
Rare — no AV signatures
Custom protocols, long dwell time

OT IMPACT
Designed to cause operational failure. Rare, but consequences can be severe.

ELECTRUM
Active since 2016

XENOTIME
Active since 2014

COMMODITY LEGACY MALWARE

Conficker • Sality • Ramnit

ORIGIN
Organised criminal groups
Financially motivated, opportunistic

TARGET
Any reachable Windows host
No OT knowledge required

DELIVERY
Opportunistic vectors
USB, network scanning, removable media

DETECTION
Signatures exist
But most OT hosts run no AV

OT IMPACT
Not designed for OT —but causes instability, spreads across networks, degrades systems over time.

The Conficker worm in action screenshot on Windows 7

Figure 1 | **Comparison: Custom OT Malware vs. Commodity Legacy Malware**

Dragos Incident Response has seen an uptick in volume and severity of activations and inquiries from organizations that have discovered these infections, often as a byproduct of deploying detection capabilities or undertaking their first formal cybersecurity assessment. The infections are not new, but they are suddenly apparent. So is the unpleasant realization that they are not at all easy to eradicate. Dragos sometimes is engaged after an expensive, failed second or third attempt and the realization that "whack-a-mole" does not result in any improvements.

The culprit malware families involved are well-known. Conficker, first identified in 2008, continues to spread across unpatched networks via the same SMB vulnerability it exploited 18 years ago. WannaCry, which caused global disruption in 2017, continues to propagate within isolated industrial networks where its kill switch domain is unreachable. Sality, a peer-to-peer botnet and file infector active since 2003, enters OT environments through trojanized tools and spreads quietly through shared drives and removable media. These families are joined by a broader ecosystem of headless worms, including Gamarue, Palevo, and Virut, that spread via removable media and network shares, requiring no operator interaction and generating minimal visible activity.

Security researchers and vendors understand these variants thoroughly. What makes the situation particularly challenging is the critical and legacy environment that is infected, the near-total absence of guidance on how to respond to the situation appropriately, the significant atrophy of preventative controls intended for those computers, as well as a dwindling number of humans still capable of analyzing old computers that cannot immediately (or ever) be taken out of production.

Enterprise incident response playbooks and best practices are written for reasonably modern endpoints running supported operating systems, with functioning backup infrastructure, available vendor support, and the operational flexibility to take systems offline. None of these assumptions hold reliably in legacy OT environments. Applying standard IT remediation procedures to a facility running Windows XP-era control systems, with no supported antimalware tooling, no tested backups, and production processes that cannot tolerate unexpected downtime, is not just ineffective, but may cause consequences worse than the infection itself.

However, there are a multitude of reasons that organizations cannot simply ignore commodity infections, even if they have been present for years. As systems and infrastructure age and expand, they become more susceptible to degradation and failures due to resident unpredictable and unstable code and network traffic. Even headless, "zombie" worms can reduce system resources, cause network bottlenecks, and increase crashes in sensitive process environments. Maintaining an infected network can have legal and regulatory consequences.

As of this publication, no dedicated guidance exists for practitioners and security leaders navigating legacy commodity malware cleanup in OT environments. This paper is written to address that gap directly.

Technical profiles of the primary malware families discussed in this paper are provided in Appendix B: The Most Common Threat Families for readers who want background on their propagation mechanisms and OT-specific behavior.

Who and What This Paper Is For

This paper is written for security leaders and practitioners developing incident response programs and playbooks for OT environments; ideally before an incident occurs, but it can also help in the decisions during response activity. It assumes familiarity with industrial control systems concepts and with general cybersecurity incident response principles, but does not assume prior experience with legacy malware cleanup in OT.

This paper addresses commodity legacy malware: self-propagating worms and file infectors that spread opportunistically, do not require operator interaction to propagate, and are not the product of a targeted intrusion. It does not address targeted ICS-capable malware such as PIPEDREAM, FrostyGoop, or TRISIS, which require distinct response approaches. It does not address modern ransomware in OT environments, which is covered extensively elsewhere.

SECTION 02:

Why Legacy Malware Persists in OT Environments

Some readers may be shocked or bemused that Conficker is still an issue in 2026, and simply blame lack of patching or security best practices. That is not entirely wrong, but reality is far more complex. It is essential to consider what OT environments are really like, and why assumptions underpinning standard enterprise security practices simply do not apply to them. These infections are not a joke, require substantial strategy today to remove, and both their presence and removal can have real life consequences in critical infrastructure.

The Reality of OT Lifecycles

IT environments typically operate on hardware and software tech refresh cycles of three to seven years. OT environments do not work this way. Industrial equipment is designed and procured to stay in place reliably for decades. A control system installed in a refinery in 2003 may still be running the same Windows XP-based HMI today, not because nobody noticed or cares, but because replacing it requires vendor certification, process validation, extended downtime, and capital expenditure that may run into the millions. In many cases, the original vendor no longer exists, or no longer supports the product. The engineering knowledge to safely modify or replace it may have retired (or passed away) with the person who originally commissioned it.

This is not negligence. It is the structural reality of industrial operations, where the priority ordering is safety first, operational uptime second, and everything else after that. Security retrofits into this environment on whatever limited terms the process will allow, based on operational risk.

The result is that extremely out-of-date Windows and Linux hosts are not an anomaly in OT. They are a feature of the landscape. Legacy commodity malware does not persist in these environments because defenders do not care or are not trying. It persists because the vulnerabilities it exploits are, in many cases, genuinely unremediated and in some cases unremediable.

Why These Specific Vulnerabilities Cannot Simply Be Closed

Conficker exploits MS08-067, a vulnerability in the Windows Server service, and propagates via SMB and NetBIOS. WannaCry exploits EternalBlue (MS17-010), a vulnerability in SMBv1. Both families rely on protocols that are frequently deeply embedded in how OT networks function. SMB is used for file sharing between higher Purdue level Windows systems. NetBIOS supports name resolution in flat network segments that were never designed with DNS in mind. Disabling these protocols on a live OT network may not be a simple configuration change. It may break historian connections, disrupt HMI communications, and cause a loss of control or visibility of the process. Changes require evaluation and certification. Even where patches exist and are vendor-approved, applying them requires downtime that may only be available once or twice a year, if at all.

The practical result is that many OT networks contain systems that are structurally vulnerable to these families, not as a temporary gap but as a semi-permanent condition that must be managed rather than eliminated until a major or complete industrial system replacement occurs.

The Air Gap That Was Never Really There

For years, there was a common cybersecurity practitioner misconception that OT networks were air-gapped and therefore largely insulated from the threats circulating in IT environments. Even then, that assumption was always more theoretical than real, and it has become less real over time. Today, everyone from vendors to senior leaders wants or mandates real-time telemetry and remote support capability.

Removable media has always been an infection vector. USB drives move between corporate laptops and engineering workstations constantly. Vendor technicians arrive on site with devices that have visited many other facilities. Contractors plug in media that were last used months ago in a different region. None of this requires a direct internet connection. An infection can enter a completely isolated OT network on a single USB drive and propagate to every accessible host before anyone notices.

Transient assets compound the problem significantly. A vendor laptop used to commission a variable frequency drive controller may carry a Sality infection from a previous engagement. A programming device used by a maintenance contractor may reintroduce Conficker to a network that was just cleaned but remains vulnerable. These vectors are not hypothetical. The Dragos Incident Response Team sees them regularly, and they are one of the primary reasons that commodity malware cleanups fail and must be repeated.

File shares add a third dimension. OT networks frequently rely on shared network drives for storing configuration backups, historian exports, and engineering project files. For example, Sality spreads by infecting executable files wherever it finds them, including on mapped network drives. A single infected file left on a share can reinfect every host that touches it, indefinitely.

Beyond removable media and transient assets, OT networks are increasingly connected to other networks. Digital transformation initiatives, remote access requirements, and IT/OT integration projects have eroded the isolation that once provided a degree of passive protection. Many environments that were genuinely air-gapped a decade ago now have multiple remote access paths, cloud historian connections, and demilitarized zone bridging infrastructure shared with enterprise networks. WannaCry's kill switch mechanism illustrates an ironic consequence of this: the ransomware's propagation was slowed in internet-connected networks when a researcher registered the kill switch domain. In relatively isolated OT networks with no outbound internet access, the kill switch never fires, and WannaCry continues to spread indefinitely within the network perimeter.

Passive Control Atrophy

There is a less structural but equally important factor: in environments where infections have been present for years without causing obvious operational impact, the security controls that should detect and prevent them quietly degrade. Antivirus definition updates cease because a subscription lapsed or an update server was decommissioned. AV agents crash on aging hardware and are never restarted because nobody is monitoring their health. Firewall rulesets accumulate exceptions year after year as engineers request access for new applications and integrations, until the ruleset is effectively permissive for internal traffic. Group policies drift as systems are added outside the normal provisioning process. Nobody notices any of this because the process keeps running, and performing major changes is costly and challenging.

The problem typically is not as simple as just updating or replacing host antivirus. Antimalware vendors eventually cease support for older agents, and do not provide backwards compatibility for extremely old operating systems in their newer agents. Older agents cannot necessarily connect to new consoles or definition servers. So even installing an antivirus from an old CD will likely not be particularly effective. Additionally, industrial OEMs typically only permit the use of a single vendor and major version.

Human alert fatigue adds to the problem. A Conficker infection that has been quietly propagating across a historian and a handful of HMIs for seven years without causing a process upset is easy to deprioritize.

This passive control atrophy is particularly common in environments where cybersecurity ownership is unclear, where IT and OT teams operate in separate organizational silos, or where security investment has historically been minimal. It is also common in environments undergoing ownership changes, where institutional knowledge of the environment's history and prior incidents has been lost.

The combination of these structural problems, operational limitations, legacy systems, and limited cybersecurity tooling are why we are still responding to Conficker infections in 2026[1]. It is still the same malware we dealt with in 2009. The environments and their security controls have simply stayed static, or even degraded over the intervening years.

[1] Dragos, Inc. "OT Threat Landscape 2026: What OT Cybersecurity Defenders Need to Know." Dragos Blog, April 2026. https://www.dragos.com/blog/ot-threat-landscape-2026

SECTION 03:

The Risk of Remediation with IT Playbooks

It is neither effective nor safe to approach response to commodity malware in OT the way you would approach malware remediation on a modern enterprise network. The instinct to follow simple, established Enterprise IR playbooks is understandable, but it can get an organization into serious trouble.

Applying standard IT remediation procedures to a legacy OT environment without carefully adapting them to the specific environment and life/safety can cause process disruptions, unpredictable behaviors, and safety system responses. In some environments, it can put people at risk. The malware you are trying to remove may be less dangerous in the short to medium term than a poorly planned cleanup attempt.

OT Systems Are Different

Modern enterprise endpoints are relatively uniform. They run supported operating systems, they usually have endpoint detection and response (EDR) agents installed, they belong to a Domain, and they can be rebooted, reimaged, and replaced with manageable disruption. The Windows 7-era and older HMIs, historians, and engineering workstations that carry the bulk of legacy OT infections are none of these things.

Industrial networks often host operating systems that have not received security updates in over a decade. They frequently cannot run modern security tooling at all, either because the OS is too old, because safety concerns or vendor contracts prohibit unauthorized software installation, or because the hardware cannot support the resource requirements. They may be running 24 hours a day, seven days a week, with no scheduled downtime window in the near term. They may be controlling or monitoring physical processes where an unexpected reboot or application crash has real consequences.

It is very common that nobody has a truly tested restore procedure for these systems. Original “gold images” for the installation may not exist, or they may carry the infection. Backups, where they exist at all, may not have been tested for restoration. The original installation media may be lost. The vendor who built the system may no longer be in business. We have run into situations where crown jewel systems were built by an engineer who passed away and did not document anything. In this context, “we will just reimage it if something goes wrong” is not a safety net. It is wishful thinking.

Tooling Does Not Exist at Scale

One of the most consistent surprises for enterprise security practitioners encountering OT environments for the first time is the near-total absence of the tools they have trained on and rely on every day. In the case of junior people in the field, this may mean the absence of all the tools they have ever had training or experience with.

EDR is not close to standardized in OT environments. The endpoint visibility and remote telemetry and isolation capability that modern IR practitioners take for granted simply does not exist on most legacy OT hosts. There are no process creation events or network connection telemetry from the endpoint, and no third-party tamper protection or application allowlisting. What forensic visibility exists must be built using traditional collection methods: whatever Windows event logs were retained, prefetch artifacts, filesystem metadata, and network-level observations.

Supported antimalware options for pre-Windows XP SP2 systems are extremely limited. The commercial security industry has (reasonably) moved on, and Windows architecture has changed too much. The options that remain are discussed in detail in Appendix A: The Tooling Reality, but the short version is that the tooling landscape is sparse, and practitioners should not assume that the antimalware products their organization already licenses will run on the systems that need them most. Older installation media may exist, but it will likely no longer properly function, install, or receive definitions.

Agent-based scoping tools face the same constraints. Velociraptor, one of the most capable open source digital forensics and incident response (DFIR) platforms available, does not support Windows XP or Windows Server 2003. Many commercial forensic triage tools have similar floors. This means that scoping a legacy OT environment requires a fundamentally different approach from scoping a modern enterprise environment: more manual, more network-based, and more reliant on careful spot-checking than comprehensive automated collection. This requires tools and training that not every cybersecurity graduate receives in 2026.

DFIR Is Different

In a modern enterprise malware incident, DFIR practitioners focus on comprehensive forensic evidence collection, meticulous preservation of artifacts, malware reversing, detailed timeline reconstruction, and thorough threat intelligence analysis. All of that is appropriate when dealing with a recent or advanced intrusion where understanding the full scope of adversary activity matters enormously.

Legacy commodity malware in OT does not usually require that approach, and applying it wastes time and resources that could be spent on the actual problem. We already know how Conficker works. We already know how WannaCry spreads. We should not spend days performing deep forensic analysis and malware reverse engineering of a Conficker infection to understand what it did to the computer. It is very unlikely that adequate forensic artifacts will remain after a decade or more that support identifying the initial intrusion. We need to understand how far it has spread, what vectors are sustaining it, what it may impact in the network, and what a safe cleanup looks like for this specific environment.

Spot-check representative and important hosts to confirm which variants and file hashes are present and understand the spread. Assess the environment's topology, visibility, and vulnerability posture. Identify all relevant reinfection vectors. Evaluate what tooling is safe and supported. Make risk-based decisions about remediation approach and timing. The goal is almost always improving operational reliability and security, not building a court-ready evidence package.

Sending a team to perform enterprise-style comprehensive forensic imaging of every host in a large OT environment, before anyone has assessed whether those hosts can safely be touched, is a common mistake. It consumes enormous time, may require taking systems offline unnecessarily, and produces a large volume of data that does not meaningfully inform the remediation decision.

The Risk of Making Things Worse

Standard enterprise malware remediation playbooks frequently include steps that are dangerous in legacy OT environments. Running intrusive vulnerability scans to identify infected hosts can disrupt time-sensitive industrial communications and crash legacy network devices. Deploying antimalware agents without vendor approval can trigger process upsets on systems where any unexpected process or resource consumption is a problem. Attempting to clean an infected file in place on a system running a legacy OS without a tested restore path risks rendering that system unbootable. Isolating a network segment without understanding its dependencies can cause loss of visibility or control of a physical process.

Simultaneously, ignoring these infections can cause industrial consequences as well. Commodity worms and similar infections can cause network and system latency and instability, and this tends to increase as the devices and infrastructure in the environment age and expand.

Remediation may now sound impossible, but it just requires a different way of thinking: careful planning, close coordination with engineering and operations teams, vendor involvement, and a clear understanding of the specific risks in the specific environment. The sections that follow provide a framework for doing exactly that.

The bottom line is simple: the malware we are discussing is old, well understood, and in many cases not actively causing immediate operational harm. Careless response can cause immediate operational harm. Respect that dichotomy and plan accordingly.

SECTION 04:

Understanding Risk Before Acting

We have seen organizations discover legacy malware infections and respond hastily, using their Enterprise malware incident response plans. Risk is gauged by number of infected hosts. Deep dive forensics and malware reversing are performed before restoring operations. Systems are taken out of operation to install EDR and extended detection and response agents without vendor or engineer assessment or sign off. We have even seen professional Enterprise incident response firms perform similar actions in customer environments at great cost. Both have led to operational disasters, sometimes requiring an entirely discrete second response effort.

The Dragos Incident Response Team has responded to cases where well-intentioned but poorly planned forensic or remediation attempts caused more disruption than the original infection. Systems that had been running continuously for years, carrying infections that were stable and effectively headless, failed during cleanup attempts on fragile and low-resourced legacy hardware. Network segments were actively scanned, causing low level device disruption. Antimalware tools were deployed without vendor approval on systems running certified software configurations, voiding support contracts and safety certifications, and triggering unexpected process behavior. In each case, the infection itself had not caused a significant operational incident. The responders did.

The core problem is a mismatch between the urgency malware remediation feels like it deserves (and may deserve in a modern Enterprise environment) and the careful, deliberate pace it requires in process environments. Discovering that your OT network has been infected with Conficker for over eight years can be a nasty surprise. The immediate reaction is understandable, and often baked into boilerplate incident severity matrices. However, acting on it immediately without adequate planning is dangerous.

Risk Is Not Binary

The first shift in thinking required for legacy OT malware is accepting that risk is not binary. The infection is not simply either present and dangerous or absent and safe. The actual risk picture is considerably more nuanced and quantifying that risk is the prerequisite for every decision that follows.

Operational risk, in this context, has several dimensions that must be assessed independently before any remediation decision is made.

Infection risk: What malware variants are present? What are their propagation mechanisms? Are they actively spreading, or have they reached a stable state in the environment? Is there any evidence of interactive command and control activity, or are the infections effectively headless? Is there any indication that a more sophisticated threat group has accessed or is exploiting the existing infection or vulnerability?

Operational risk of the infection: Is the malware currently causing measurable operational impact, such as system instability, network congestion, or process anomalies? Given how the malware functions, does the OEM or system expert posit that there is a credible near-term risk of it doing so? Are there areas of the OT environment where even a small increase in resource consumption or network traffic could cause a problem?

Operational risk of remediation: What systems would need to be touched or taken offline to execute cleaning of the malware? What are the operational dependencies of those systems? Is there a maintenance window available, and when? What is the risk of a system failing to recover after a cleanup attempt? What is the OEM's position and policy on the proposed remediation activities, and what does the broader vendor support ecosystem require?

Reinfection risk: What infection vectors are allowing the infection to persist? Can those vectors be controlled before or during remediation? If reinfection vectors cannot be fully controlled, is remediation likely to have any permanence?

Regulatory and compliance risk: Are there reporting obligations triggered by the discovery of any infection in this environment? Are there compliance frameworks the organization operates under that affect the decision to remediate? It is worth noting that regulatory requirements, while real, should not be the sole driver of a remediation decision that carries significant operational, life and safety risk. Regulators generally do not want organizations to cause process upsets or safety incidents in the name of compliance. Document the risk assessment and the reasoning behind the chosen approach, then talk to appropriate regulatory body contacts.

You cannot exclusively solve this problem from behind your desk. It requires direct engagement with the impacted environment, with engineering and operations teams, and in most cases with the relevant OEMs and support vendors. Cybersecurity practitioners who arrive on site with a remediation plan already formed, before they have assessed the situation, are operating on assumptions that may be entirely wrong.

The Cost of Getting It Wrong

In process industries, losing visibility or control of a physical process can cause equipment damage, environmental incidents, product loss, and personnel safety events. The loss of an HMI that controls a compressor train or a chemical dosing system at the wrong moment in a process cycle is not just an IT incident. It can cause a condition where the process must be shut down to ensure safety. In a chemical or temperature sensitive environment, that can mean damage to multi-million-dollar equipment. These are the realistic consequences of applying enterprise IR reflexes to environments where unplanned changes have huge impacts.

In many cases, these infections have been present for years and are only being discovered in the present due to additional network visibility and cybersecurity maturity. Taking an extra few weeks to plan a safe remediation is typically the right call. The exception, which we address in the next section, is when the infection is actively causing harm, may plausibly cause harm or process disruption soon, or when credible evidence exists of threat group activity leveraging the infection. In those cases, the risk calculus changes, and faster and more drastic action may be warranted. Even then, faster does not mean unplanned or without risk calculus.

Involve the Right People, First

One of the most consistent findings across Dragos legacy malware engagements is that cybersecurity teams make decisions in isolation that should have involved engineering, operations, safety personnel, and vendors from the start. Almost universally, cybersecurity personnel are not subject matter experts in the industrial process.

The control system engineer who has been running that historian for twenty years knows things about its stability and dependencies that will not appear in any asset inventory. The OEM support team may have specific guidance, approved tools, or documented procedures for exactly this situation. Operations leadership needs to understand the risk picture and be part of the decision about timing and approach.

Remediation of legacy OT malware is not a cybersecurity project. It is an operational project with a cybersecurity workstream. Treating it as anything less is one of the most reliable ways to make it go wrong.

Choosing a Path Forward

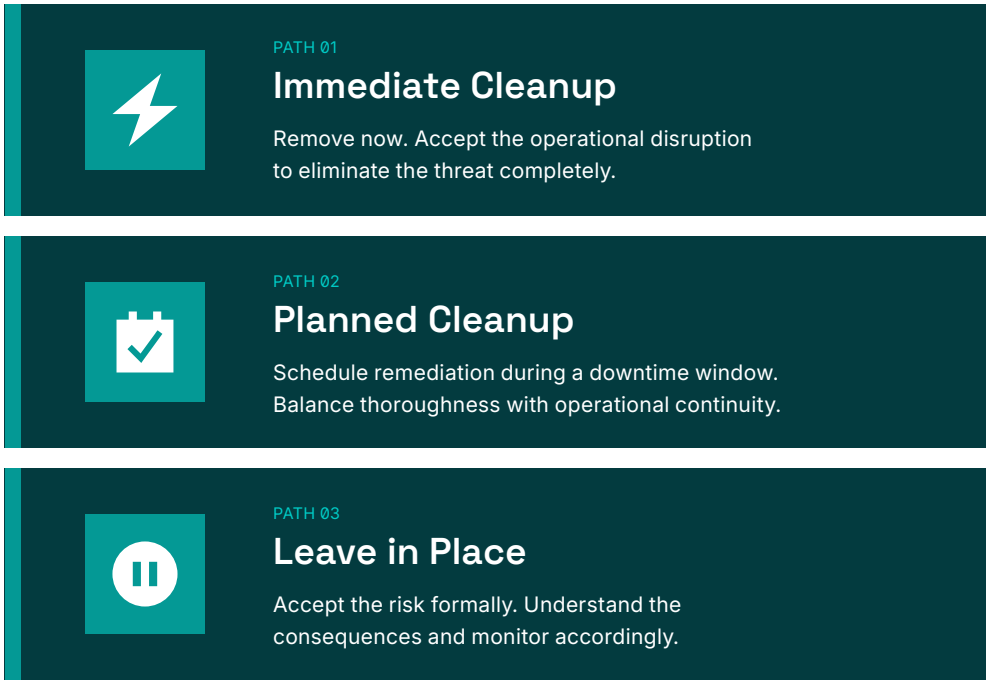
Once the risk picture is understood and the right stakeholders are involved, three realistic paths exist. Each should be evaluated based on the circumstances and expert input. The next section describes them in detail, including the potential risks and benefits of each, so that organizations can make an informed decision rather than defaulting to Enterprise cybersecurity procedures that were never designed for this environment.

SECTION 05:

The Three Options

Once the risk picture is understood, organizations typically face three realistic paths. None of them is universally correct, and none of them is without trade-offs. What matters is choosing deliberately, with full awareness of what each path requires and what it risks.

In our experience, organizations that approach this decision clearly, with the right people in the room and an accurate picture of the environment, make good choices. Organizations that skip the risk assessment and default to one path because it feels like the obvious one tend to run into trouble.



Option 1: Immediate Remediation

Immediate remediation means prioritizing cleaning the infections as promptly as possible, accepting the operational disruption that likely entails and managing it carefully. This may include a full OEM system upgrade or replacement at significant expense.

This option is appropriate when the infection is actively causing operational harm, when it is assessed as a credible near-term risk to process stability or safety, or when evidence exists that a more sophisticated threat group is present in the environment and may be leveraging the existing infection or the vulnerabilities it exploits to cause an unacceptable condition. It may also be appropriate when rigid regulatory obligations require action within a defined timeframe and do not permit a longer planning cycle.

OPTION 01

Immediate Cleanup

Act now, restore fast

FOR

- Compliance met immediately
- Instability removed now
- Fastest resolution timeline

AGAINST

- Significant operational downtime
- High resource demand
- Cleanup failures can worsen impact
- Missed systems means starting over

→ **USE WHEN**

Regulatory deadline is imminent or active threat actor presence is confirmed.

What it requires to succeed:

Immediate does not mean hasty. Even when faster action is warranted, the work outlined in Sections 6 and 7 of this paper must be completed before cleaning begins. The environment must be scoped. All families must be identified. All reinfection vectors must be understood. Engineering, facility management, safety, and vendor approvals must be obtained. A maintenance window must be agreed. Tested tooling must be prepared.

What is different in this option is the overall remediation timeline, not the methodology. The planning cycle is compressed, which increases risk and requires more resources to execute safely in a shorter period. Organizations choosing immediate remediation should expect to dedicate significant personnel to the effort and should strongly consider engaging specialist OT incident response support if it is not already involved.

The risks:

Compressed timelines increase the likelihood of missing infected systems, missing reinfection vectors, or making a technical decision that causes an OT system fault. The pressure to act quickly can usurp the careful coordination with engineering, operations, and OEMs that makes remediation safe. Reinfection is more likely if vectors are not fully controlled before cleaning begins.

A common failure under this option is cleaning visible, online systems without fully controlling reinfection vectors, then declaring success before the infection resurfaces. This can mean performing the whole, costly exercise repeatedly.

Option 2: Planned Remediation Project

A planned remediation project treats the cleanup as a discrete, resourced operational project with defined scope, timeline, and success criteria. It is the most commonly appropriate path for organizations that have discovered stable, headless infections that are not currently causing operational harm (and are not likely to cause harm soon).

Projectizing remediation allows for thorough scoping, careful OEM and vendor engagement, appropriate maintenance window planning, tooling evaluation and procurement, staff training, and architecture and passive control improvements that support both the cleanup and the long-term security posture of the environment. It is the approach most likely to produce a durable outcome, because it addresses the conditions that allowed the infection to persist, not just the infection itself.

OPTION 02

Planned Cleanup / Upgrade

Controlled remediation window

FOR

- No immediate ops disruption
- Resources and risk pre-planned
- Can modernize patch and security posture
- Lower risk of missing infected systems

AGAINST

- Major upgrades are costly
- Infection persists during planning window
- Unforeseen impacts still possible

→ **USE WHEN**

Environment is stable, threat is latent, and planning time is available

What it requires to succeed:

A genuine organizational commitment to completing the project, including the operational disruption it will cause. Planned remediation projects in OT environments frequently stall because the maintenance windows required are difficult to obtain, because competing operational priorities displace the work, or there is extreme internal resistance to making any changes. Organizations should define clear milestones, assign ownership, and treat the project with the same operational discipline as any other major maintenance activity.

The project should include: full environment mapping and asset inventory (including offline and transient systems), identification and control of all reinfection vectors, OEM and vendor engagement on approved tools and procedures, architectural improvements where feasible, system patching where vendor-approved patches exist and can be applied, and a thorough validation phase to confirm the cleanup has held before the project is closed.

The risks:

Planned projects take time, and during that time the infection continues to propagate. If the planning phase is too long, or if the project stalls, the organization is left in a state of known infection without an impetus to act. This can be worse than either acting immediately or making a fully informed decision to defer, because it combines the organizational and regulatory exposure of a known infection with the operational risk of an uncontrolled environment. The attitude becomes “we will get to that sometime”.

Project scope creep is a real risk. Legacy OT malware remediation tends to surface other problems: undocumented systems, unsupported software, lapsed vendor contracts, missing backups. These are important to address but should not be allowed to expand the remediation project to an unmanageable size and timeframes. You cannot reach all of OT cybersecurity maturity in one go. Define the scope clearly and address adjacent problems as separate workstreams.

Option 3: Informed Deferral

The third option is the one most likely to cause our cybersecurity practitioner readers the most discomfort. It is possible it will be the right answer in some specific circumstances: making a fully informed, documented decision to leave the infection(s) in place for the indefinite future and manage them rather than remediate them.

This is not the same as ignoring the infection. It is a deliberate, owned risk acceptance decision, made with full awareness of the infection, a clear understanding of operational and safety risks, active monitoring to detect any change in infection behavior or threat group activity, and a defined set of conditions that would trigger escalation to one of the other two options.

This option may be appropriate when the infection is confirmed headless and stable, when all assessed reinfection vectors are understood and partially or fully controlled, when no safe maintenance window can exist within a feasible timeframe, when OEM engagement has confirmed that remediation carries a higher risk of system failure than the infection itself, or when the systems in question are approaching planned end of life and complete replacement is the more appropriate long-term solution. In some cases, no upgrades or rebuilds may be safe and feasible for the impacted systems, and no effective host security tools may be supported.

OPTION 03

Indefinite Infection

Accept and monitor

FOR

- No disruption to current operations
- Irreplaceable systems unchanged
- No immediate resource expenditure

AGAINST

- Instability grows as hardware ages
- Regulatory and compliance exposure
- Lateral spread risk to partners and segments
- Can be repurposed by threat actors

→ USE WHEN

Systems are truly irreplaceable, threat is headless, and risk is formally accepted.

What it requires to succeed:

- **Documentation.** The decision must be formally recorded, with the supporting risk assessment, the stakeholders who were involved, the final decision authority, and the conditions under which the decision will be revisited. This protects the organization in the event of a regulatory inquiry, and it ensures that the decision does not become permanent or precedent setting when the people who made it move on.
- **Active cyber defense.** Human-driven network monitoring via the Dragos Platform or equivalent capability should be in place to detect any change in infection behavior, new propagation activity, or indicators of more sophisticated threat group presence. A stable infection that was appropriate to defer last year may not be appropriate to defer next year if the environment changes or the threat landscape evolves.
- **Defined escalation conditions.** Before deferring, the organization should agree on what would change the decision: specific indicators of threat group activity, a change in the infection's propagation behavior, a regulatory requirement with a fixed deadline, or the availability of a maintenance window that makes planned remediation feasible.

The risks:

Legacy malware infections do not always stay stable forever. Systems age, hardware degrades, and infections that were benign for years can begin causing instability as the hosts they run on become increasingly resource constrained. The longer an infection persists, the more

opportunities exist for a threat group with access to the network to leverage an existing vulnerability or use a backdoored host as a foothold or cover for their activity. Any unrelated major or minor modification to the network or system can increase these risks.

There is also an organizational risk. A documented deferral decision made carefully by the right people is defensible. An undocumented infection that was informally tolerated for years is not. If the infection eventually contributes to an incident, the distinction matters enormously.

Informed deferral is not a path to avoiding the problem. It is a path to managing it responsibly until circumstances allow a better solution like a major system upgrade. Treat it as such. project to an unmanageable size and timeframes. You cannot reach all of OT cybersecurity maturity in one go. Define the scope clearly and address adjacent problems as separate workstreams.

Choosing Between the Three

The decision between these options is not purely a cybersecurity decision. It belongs to the organization's operational and executive leadership, informed by input from cybersecurity, engineering, operations, safety, legal, and OEM and vendor stakeholders.

Cybersecurity practitioners, including external IR teams, can provide stakeholders with the technical picture. They can scope the infection, characterize the families, assess the reinfection vectors, evaluate the tooling, and outline the risks of each path. They cannot make the operational risk decision on behalf of the operational organization, and they should not try to.

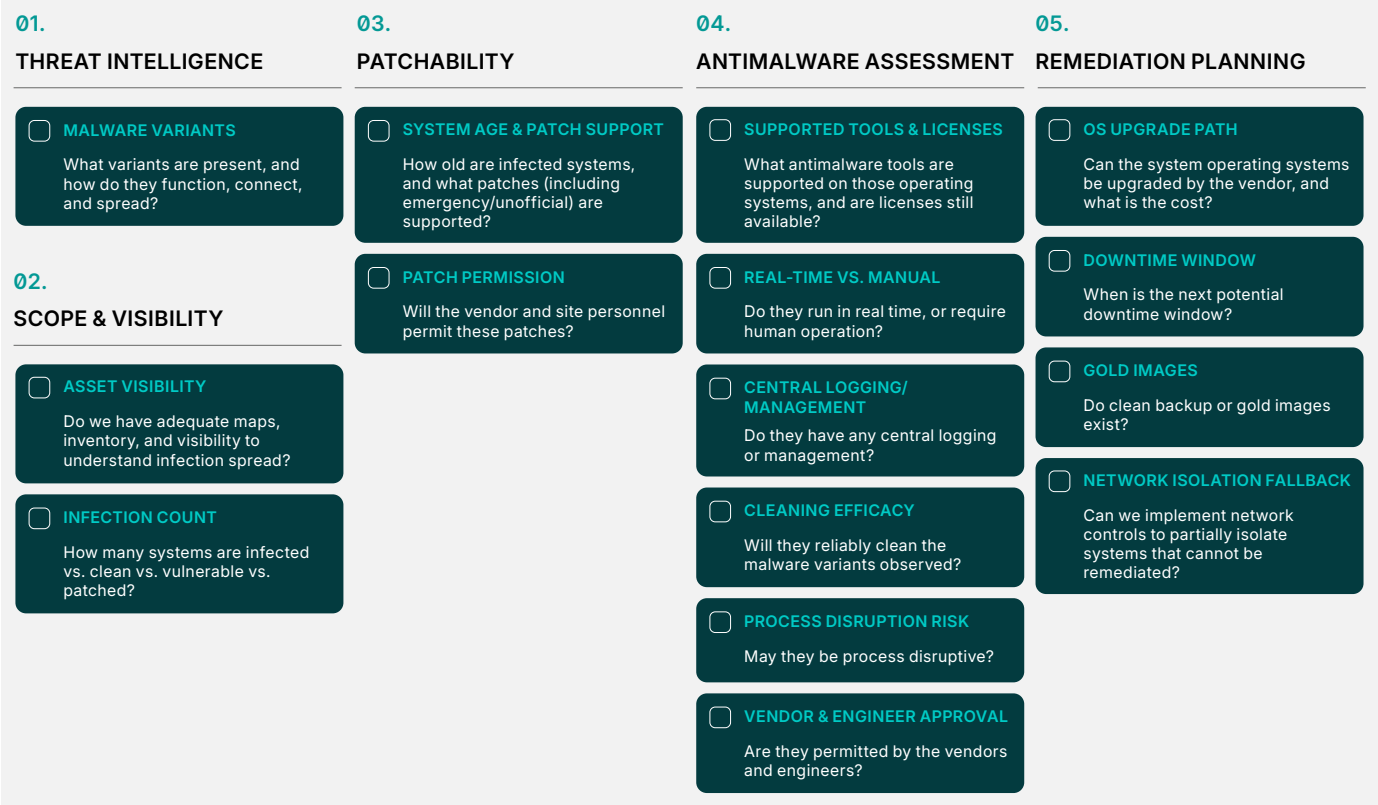
What practitioners can and should do is ensure that the decision is made with accurate information, by the right people, and that it is documented regardless of which path is chosen. A well-documented risk acceptance decision is not a failure. An undocumented infection is.

SECTION 06:

The Pre-Remediation Checklist

Regardless of which option was chosen in the previous section, it is important that any organization proceeding toward active remediation answers a set of questions before touching a single system. Every item on this list represents a category of failure we have seen derail real remediation efforts.

Work through these questions with the full stakeholder group identified in Section 4. Document the answers. Where answers are unknown, find them before proceeding. Skipping steps in this process will likely lead to operational impact or reinfection.



Step 1: Understand the Situation

What malware families are present, and are you confident the list is complete?

Passive network monitoring and manual host forensic spot checks should give you a working family list, but it is rarely complete on the first pass. Plan for the possibility that additional families will be identified during the remediation itself, and have a process for handling that discovery without derailing the broader effort.

How is each family propagating in this specific environment?

The variants of malware we are discussing in this paper are already well documented and understood. Threat intelligence will provide indicators of compromise, signatures, and information about malware infection vectors and persistence. Controlling spread via network exploitation, removable media, file shares, or transient assets and modems will require different controls. Understanding the active propagation mechanisms for each malware family in your specific environment is not the same as knowing how those families propagate in general. The network topology, the removable media practices, the vendor access patterns, and the file share structure of this specific site determine which vectors need to be addressed.

Is there any evidence of interactive threat group activity?

This is really important. Legacy commodity malware and a human-driven intrusion require entirely different responses. Hunt for indicators and check for detections that go beyond the expected behavior of the identified families: unusual outbound connections, credential access activity, lateral movement that does not match the malware's known propagation patterns, or new tools or implants that do not belong to the identified families. If you find any of these, stop and reassess before proceeding with commodity malware cleanup. You may be dealing with something considerably more advanced, and that is the time to begin more typical OT digital forensics and incident response.

Step 2: Understand the Environment

Do you have a complete and current asset inventory, including offline systems? This is still a challenge for a lot of organizations.

Systems that are powered off, in storage, or used intermittently are common reinfection sources. A laptop used by a rotating maintenance contractor that sits in a cabinet between visits. A spare Windows-based HMI kept as a cold standby. A building management system that was taken offline six months ago but is still on the network plan. Every one of these can reintroduce an infection to a freshly cleaned environment. Understanding this may require multiple types of network discovery as well as interviews and site walks.

Do you have network diagrams that reflect the current state of the environment?

Not the diagrams from the 2013 OEM upgrade. The real condition, including any shadow IT connections, temporary modifications, vendor remote access paths, and IT/OT integration points that have likely been added since the last formal documentation. If you do not have current diagrams, produce them before starting. Cleaning a network you do not understand is how you miss reinfection vectors.

Do you know which systems are most sensitive to disruption?

Not every system in an OT environment carries the same operational risk if it becomes unstable or requires a reboot. Identify the systems where any unexpected behavior or failure has real life consequences, and plan the remediation sequence accordingly. These systems may require additional precautions, additional approvals, or exclusion from certain remediation activities.

Are your backups current, tested, and confirmed clean?

If a system fails during remediation and the backup is either outdated, untested, or itself infected, you have a significant problem. This question needs a verified answer, not an assumed

one. If backups do not exist or cannot be confirmed clean due to the age and embedding of the infection, that changes the risk profile of the remediation significantly and should factor into your decision.

Step 3: Define Potential Actions

Do you have OEM approval for the proposed remediation activities on each affected system?

Often, this is not optional from a safety certification or support contract perspective. OEM approval may require formal change requests, specific approved tool lists, and documentation of the proposed activities. In some cases, OEMs will want to be present or will want to perform changes themselves. Start these conversations early. OEM approval processes can take weeks or months, and discovering this after a maintenance window has been scheduled is a painful way to learn it.

Have your vendors been engaged on any implications for support contracts or warranties?

Related but distinct from OEM approval. Support vendors and resellers may have their own requirements or recommendations. Some may void support contracts if certain tools or updates are run on managed systems without prior notification. Check before proceeding.

Do you have tools confirmed and tested for each system you plan to clean?

It is important to confirm that tools will work as expected on the older and non-standard systems. For instance, ClamWin definitions updated. Surge Collect version current and license valid. Wintriage tested on a representative legacy OS build. Any scripts or procedures tested on non-production equivalent hardware before being run on production systems. Tooling failures discovered mid-remediation on production systems are one of the more avoidable problems.

Do you have enough people with the right skills for the planned scope and timeline?

Legacy OT malware remediation is hands-on work. It requires people who are comfortable working in OT environments, who have some knowledge and experience with older operating systems and manual system administration, who understand the tooling constraints, and who can make good judgment calls when something unexpected happens - because it will. Understaffing a remediation project is a reliable way to produce an incomplete one.

Step 4: Control Reinfection

Have you identified every reinfection vector for every family present?

Network-based propagation vectors like removable media, remote access, transient assets, and infected file shares each need a specific control. List them explicitly. For each one, document whether it will be controlled before remediation begins, controlled during remediation, or represents a residual risk that will need to be managed after cleanup.

What is your removable media policy during the remediation, and how will it be enforced?

During active remediation, any unscreened removable media introduced to the environment is a potential reinfection source. This includes vendor and contractor devices. Establish a clear policy for the duration of the project and communicate it to everyone with physical access to the environment.

What is your process for handling transient assets?

Contractor laptops, portable programming devices, and vendor equipment that moves between sites should be screened before connecting to the environment during the remediation window. This is operationally inconvenient. It is considerably less inconvenient than discovering the environment was reinfected on day two by a contractor's programming device.

What will you do about infected file shares?

Sality will reinfect cleaned hosts from infected executables left on shared drives. File shares must be scanned and cleaned as part of the remediation, not after it. Define the process for this, including how you will handle shares that contain files belonging to systems that are not being remediated in this phase.

Step 5: Account for Operations

Is there a confirmed maintenance window, and does it give you enough time?

Ensure that you have operations and engineering leadership sign-off for the downtime, sufficient buffer for things to take longer than planned, and a defined process for what happens if the window runs out before the work is complete. Rushed endings to maintenance windows are where corners get cut and systems get reconnected before they are ready. There will also likely be additional activities going on in those limited windows that impact your response.

What is the process if something goes wrong?

Before starting, agree on who has authority to halt the remediation if a system becomes unstable, what the recovery procedure is for each high-risk system, who is the primary contact for each OEM if emergency support is needed, and at what point the organization would consider the remediation a failure and revert to managed deferral or a major system upgrade. Having these conversations before they are needed is significantly more productive than having them during an incident.

Has the relevant leadership been briefed and does the work have their support?

Remediation projects that lose leadership buy-in mid-execution are very difficult to complete. Ensure that operations and executive leadership understand what the project involves, what disruption to expect, and what success looks like before the work begins.

If you can answer all these questions with confidence, you are ready to plan and execute the malware remediation. If significant gaps remain, close them first. The next section describes how to structure and execute the project itself.

SECTION 07:

Building and Executing the Remediation

At this stage, you should have your stakeholders aligned, a strong understanding of the environment and infections, your maintenance window confirmed, and your tooling tested. This section covers how to structure and execute the actual work.

Step 1: Start With the Network, Not the Hosts

Following safety and process continuity, an essential principle in OT malware remediation is controlling the spread of infection before you clean anything.

Cleaning hosts while active network propagation vectors remain open is pretty futile. Conficker will almost instantly reinfect a patched host if an unpatched neighbor is still on the same flat network segment. Sality will return to a workstation the moment someone runs an infected executable from a shared drive that was not addressed. WannaCry will find a newly cleaned SMBv1 host in minutes if the network allows it. Cleaning first and controlling vectors second is the pattern behind most failed remediations we have encountered.

Practically, this means network segmentation and share lockdown need to happen in the same maintenance window as host cleaning, ideally before host cleaning begins. Where full segmentation is not immediately achievable, partial controls are better than none: disabling unneeded SMB shares, restricting administrative share access, removing or quarantining known infected file share content, and enforcing the removable media policy established during pre-remediation planning.

Step 2: Sequence the Work Deliberately

Solving this problem will take time. A useful approach is to work from the outside in, controlling the perimeter of each network zone before cleaning hosts within it, then validating that zone before moving to the next.

For each zone or segment:

- Implement network controls first. Restrict propagation pathways before touching hosts. This includes share permissions, implementing some network or protocol segmentation where available, and any firewall rule changes that have received operations and vendor approval. Identify any direct internet exposure.
- File shares are an unfortunate constant in many industrial environments. Clean and control file shares before cleaning hosts that access them. An engineering workstation that maps to a share full of Sality-infected executables will be reinfected before the analyst finishes logging off.
- Work through hosts systematically, starting with the systems most likely to act as propagation hubs: historians, file servers, jump hosts, and systems with broad network connectivity. These are the systems whose clean state matters most for preventing spread to everything else.
- Crown jewel systems, identified in the pre-remediation checklist as carrying the highest operational risk, should be addressed last and with the most conservative approach. By that point, the propagation environment around them should be controlled.

Step 3: Perform the Cleanup

For most of the malware families discussed in this paper, cleaning a host involves running a manual scan with up to date ClamWin or another legacy-supported antimalware tool, reviewing detections, removing or quarantining identified malware components, and verifying that the system remains stable and functional after the process. Where vendor-approved patches for the relevant vulnerabilities exist and can be applied, apply them during the same window.

For Sality, cleaning is more involved. Because Sality infects legitimate executable files rather than simply dropping new ones, the infected files themselves need to be addressed. ClamWin will detect and quarantine infected executables, but this can break applications if the infected file was a legitimate system binary or application component. Test the cleaning process on a non-production equivalent system before applying it to production hosts. If the only copy of a critical application binary is infected and no clean backup exists, that is a problem that needs OEM involvement before the cleaning attempt. Ensure there is a mechanism to restore the systems to their previous state in a worst-case scenario.

Systems that cannot be cleaned safely due to tooling constraints, OEM restrictions, or instability risk should be documented explicitly. These become candidates for either the Informed Deferral option discussed in Section 5 or for longer-term replacement planning, depending on their operational criticality. They certainly risk becoming reinfection vectors.

Step 4: Implement Patches, Updates, and Hardening

Cleaning without patching is cleaning without solving the underlying problem. Where vendor-approved patches for MS08-067 and MS17-010 exist and can be applied to eligible operating system versions, they should be part of the remediation plan. For systems where patching is not feasible, compensating controls become critical: network segmentation to limit exposure, disabling SMBv1 where it is not required by OEM-supported applications, and password hygiene improvements to address Conficker's credential brute-forcing vector.

Password hygiene in OT environments deserves specific attention. Default credentials, shared service account passwords, and passwords that have not changed in a decade are pervasive. Device and vendor limitations often make changes to passwords very challenging. Weak and reused passwords are not just a Conficker vector. They represent a broad authentication risk that a malware remediation project is a useful opportunity to start addressing, even if full remediation or mitigation is a longer-term effort.

Disabling Autorun and Autoplay

Disabling Autorun and Autoplay on all removable media is a direct and practical control against the removable media propagation vectors used by Conficker, Sality, Gammima, and similar families. The behavior varies significantly by Windows version, and many legacy OT systems will not have the relevant updates applied, even where Windows Update was once operational.

Windows 7 and later disable AutoPlay on USB drives and non-optical removable media by default. For older systems the situation requires specific attention. Microsoft released KB971029 for Windows XP, Vista, Server 2003, and Server 2008, which restricts AutoPlay functionality to CD and DVD media only. On Windows XP and Server 2003, KB971029 should be preceded by

update KB967715, which corrects a longstanding bug that prevented the NoDriveTypeAutoRun registry key from functioning correctly[2]. Without KB967715 installed first, the Autorun disable may not hold against malware that exploits that registry gap, as Conficker did.

In Domain-joined environments, if these updates cannot be confirmed as applied or cannot be installed, Autorun can also be disabled via Group Policy (Computer Configuration > Administrative Templates > Windows Components > Autoplay Policies > Turn off Autoplay, set to All Drives) or directly via the NoDriveTypeAutoRun registry key. Both require testing on representative hardware before production deployment and OEM approval where certified configurations are involved. Systems running Windows 2000 or earlier have no supported update path for fully disabling Autorun. Physical removable media controls and enforced access policies become the primary mitigation for those hosts.

Firewall Controls for SMB and NetBIOS

Both Conficker and WannaCry propagate via SMB. Blocking or restricting SMB at the network level, both via host-based Windows Firewall rules and at network firewall boundaries between segments, is one of the most effective compensating controls available where patching is not feasible.

United States CISA recommends blocking all versions of SMB at network boundaries by blocking TCP port 445, along with related protocols on UDP ports 137 and 138 and TCP port 139[3]. In OT environments, applying these blocks between network zones rather than globally is often more practical, since SMB is legitimately used within zones for historian connections and file sharing. The goal is to prevent SMB propagation across zone boundaries, not necessarily to eliminate it within a trusted segment.

Windows Firewall has blocked all inbound SMB communications by default since Windows XP SP2 and Windows Server 2003 SP1.[4] This is an important baseline point: systems running Windows XP pre-SP2, or systems where the Windows Firewall has been disabled or has accumulated exceptions over time, may not have this minimal protection in place. Verify the Windows Firewall state on legacy hosts as part of the remediation assessment and restore default inbound SMB blocking where it has been disabled without a documented operational reason. Keep in mind that it is not uncommon for industrial computers to host cybersecurity tools like commercial firewalls and antivirus applications that are years out of date and broken, but their software presence can disable more functional Windows native security features.

Where SMBv1 is definitively not required by process applications, it should be disabled. Disabling SMBv1 removes the attack surface for EternalBlue entirely, independent of whether MS17-010 has been patched. On Windows XP and Server 2003 where SMBv1 cannot be disabled without breaking OEM applications, network-level controls at zone boundaries become the primary mitigation.

[2] Microsoft. “Microsoft Security Advisory 967940: Update for Windows Autorun.” Published February 2009, updated February 2011. https://learn.microsoft.com/en-us/security-updates/securityadvisories/2009/967940

[3] Cybersecurity and Infrastructure Security Agency (CISA). “SMB Security Best Practices.” US-CERT Current Activity, January 2017. https://us-cert.cisa.gov/ncas/current-activity/2017/01/16/SMB-Security-Best-Practices

[4] Microsoft. “Preventing SMB Traffic From Lateral Connections and Entering or Leaving the Network.” Microsoft Support. https://support.microsoft.com/en-us/topic/preventing-smb-traffic-from-lateral-connections-and-entering-or-leaving-the-network-c0541db7-2244-0dce-18fd-14a3ddeb282a

A note of caution applies here as with all OT network changes: modifying firewall rules and disabling protocols in a live OT environment requires engineering review, relevant OEM and vendor consultation, and staged testing. A firewall rule that blocks SMB across a segment boundary may silently break historian replication, file share access, or other operational dependencies that were not visible in the network documentation. Test in a controlled window and have a rollback plan ready.

Removing unnecessary software, unneeded network services, and devices that have no current operational purpose reduces the attack surface for future infections. It also tends to surface undocumented systems and informal connections that were not visible during the pre-remediation assessment. Document and discuss what you find with subject matter experts.

Step 5: Identify and Control Offline and Transient Systems

Every offline and transient system in scope needs to be addressed prior to reconnection, not just networked hosts that passive monitoring identified. This means physically locating and scanning: spare systems in storage, cold standby equipment, contractor programming devices, vendor laptops used for remote access, and any removable media that is used in the environment.

This part of the work is unglamorous and operationally complex. Coordinating with contractors to inspect their devices, tracking down spare hardware in storage rooms, and chasing the maintenance laptop that travels with the rotating shift team requires persistence. It is also where remediation efforts most often fall short. An infection that survived on a contractor's USB drive will be back in the environment the next time that contractor plugs in. A single missed host on a shelf or in a contractor's backpack may force the entire exercise to be repeated.

Policy controls that restrict switchport access to specific machines and contractually require vendors to use authorized devices which meet security standards can be a powerful mitigation.

Step 6: Maintain Documentation During Cleaning

Keep a running timeline of every system touched, every tool run, every detection found, every action taken, and every decision made. This is not primarily for compliance, though it supports that too. Legacy OT environments are complex, people are working under pressure, and the difference between a system that was cleaned and confirmed stable and a system that was cleaned and then quietly started behaving oddly three days later matters enormously for post-remediation validation.

If something goes wrong during execution, the documentation is also what allows the team to understand what happened and make an informed decision about whether to continue, pause, or escalate. Undocumented changes in a legacy OT environment are their own category of risk.

SECTION 08:

Validation and Moving Forward

Cleaning is not the same as clean. This distinction matters more in legacy OT environments than almost anywhere else, because the consequences of false confidence about remediation completion are significant: a network that is believed clean but is not will be reinfected, and the next cleanup will carry the same operational risk as the first. The same risk of instability and lack of regulatory compliance of older systems will continue.

How to Know If It Worked

Validation in a legacy OT malware engagement draws on the same sources used in scoping, because the question being answered is the same one: is the infection still present, and is it still spreading?

Passive network monitoring is the most reliable ongoing validation mechanism. After cleaning and hardening, the propagation behavior that passive monitoring was detecting before remediation should stop. Conficker's MS08-067 exploit attempts should no longer be visible on the network. WannaCry's EternalBlue scanning should cease. Sality (a file-infecting virus - see Appendix B: The Most Common Threat Families) peer-to-peer traffic patterns should disappear. There should not be any unexplained internet-bound connection attempts or domain queries. If any of these signatures persist after remediation, the infection is not gone.

The Dragos Platform provides this capability continuously. Post-remediation, passive monitoring should remain in place and be actively reviewed. Do not just run antivirus and then declare success and stand down monitoring. The monitoring is what tells you whether success was real, and if you reached every relevant device.

Host-based forensic spot checks on a small but representative sample of systems, including the highest-risk propagation hubs cleaned earliest in the project, can confirm that the host-level cleaning held. For example, rerun ClamWin or an equivalent tool on selected systems. Review event logs, registries, and file systems for indicators of compromise. Check memory for malicious processes. Check the file shares that were cleaned for any recurrence of infected executables.

Site walks and stakeholder interviews play an important factor. Network monitoring alone simply cannot catch everything. An engineer who noticed the historian running slowly last week, or the contractor who plugged in a device that has not been through the screening process, or the spare engineering laptop that was brought out of storage and connected without going through the remediation workflow: none of these will appear on a network diagram or a security information and event management alert. Walking the floor and talking to the people who work in the environment is how you find out what happened.

Log review and ticket monitoring should confirm that the remediation activities themselves went as planned and that no unexpected side effects occurred during the process. Look specifically for any system instability events, unexpected reboots, or application errors logged during or immediately after the cleaning work. These may indicate a system that was affected by the remediation process in ways that were not immediately visible.

Declaring Completion

Completion should not be proclaimed by responders on a fixed timeline. It should be declared when the validation evidence supports it. That means passive monitoring showing no propagation activity for a defined observation period, host spot checks returning clean, site walks and interviews raising no new concerns, and all systems touched during the remediation confirmed stable. Detection should be confirmed effective to quickly catch reinfection.

The observation period for passive monitoring before declaring completion should be long enough to capture a realistic sample of the environment's normal operational cycle, including any regular maintenance activities, vendor visits, or process cycles that bring removable media or transient assets into contact with the network. For many environments, a representative sample might be a few days to a few weeks.

Where systems were documented as unable to be cleaned and deferred, those systems should be formally noted in the completion documentation with their deferral rationale and the monitoring and escalation conditions that apply to them. Completion means the defined scope was addressed, not that the environment has zero infections.

Cleanup Is Only the Beginning

It is genuinely a surreal experience to respond to 20-year-old malware. Keep in mind that effectively removing such an embedded infection from a large legacy network is a significant achievement in OT. It typically requires operational disruption, sustained stakeholder coordination, and more hands-on work than anyone anticipated at the start. The temptation to just declare victory and move on is understandable.

Resist it.

A cleaned environment that does not build on the remediation to establish sustainable security controls will be back in the same position within months or years. The infection vectors that allowed Conficker and WannaCry to persist for a decade did not just disappear because that malware was removed. They are structural features of the environment that require ongoing management.

The SANS ICS Five Critical Controls[5], developed through a comprehensive analysis of all known ICS cyber attacks and co-published by SANS and Dragos, provide a practical framework for building toward a defensible OT security posture from exactly this starting point.

The five controls are:

1. **ICS Incident Response Plan.** The remediation project just completed is a forcing function for developing or improving an OT-specific incident response plan. Document what was learned. Formalize the decision frameworks that were used. Establish the stakeholder relationships that made the work possible as a permanent capability rather than a one-time arrangement.

[5] Conway, Tim and Robert M. Lee. "The Five ICS Cybersecurity Critical Controls." SANS Institute White Paper, 2022. https://www.sans.org/white-papers/five-ics-cybersecurity-critical-controls
See also: Dragos, Inc. "The SANS ICS Five Critical Controls: A Practical Framework for OT Cybersecurity." Dragos Blog. https://www.dragos.com/blog/the-sans-ics-five-critical-controls-a-practical-framework-for-ot-cybersecurity/

2. **Defensible Architecture.** The network segmentation and firewall controls put in place during remediation are the beginning of a defensible architecture, not the end. Work with engineering and operations to formalize zone boundaries, document approved traffic flows, and establish a change management process that prevents informal connections and accumulated exceptions from eroding the controls over time.
3. **ICS Network Visibility and Monitoring.** Passive monitoring via the Dragos Platform or equivalent capability, deployed during the remediation for validation, should remain in place permanently. Ongoing visibility into OT network behavior is what makes future infections detectable before they have persisted for a decade. This is the control that transforms the discovery pattern from "we found this during an assessment" to "we detected this within days of introduction."
4. **Secure Remote Access.** Remote access paths were likely identified and addressed during the remediation. Formalize what was put in place. Every remote access path into the OT environment should be documented, authenticated, monitored, and subject to regular review.
5. **Risk-Based Vulnerability Management.** The patching and hardening work done during remediation is a one-time exercise without a vulnerability management process to sustain it. Establish a process for tracking vendor-approved patches, evaluating new vulnerabilities against the OT asset inventory, and making documented risk-based decisions about which vulnerabilities to remediate, mitigate, or accept. This does not require patching everything. It requires knowing what is there and making deliberate choices to patch now, next opportunity, or perhaps in the indefinite future.

None of these controls requires a large budget or a mature security program to begin implementing. Each one builds on work that was done during the remediation. The remediation created the conditions. The five controls are the path to making those conditions durable. They all play a part in preventing commodity malware infections.

A Note on Recurrence

Legacy malware recurrence after a completed remediation is not unusual, despite best efforts. Removing a deeply embedded infection without modern tools is hard. Reinfection may mean a reinfection vector that was believed controlled was not fully controlled, a transient asset introduced the infection from outside the remediation perimeter, or a system that was deferred as part of the informed deferral decision acted as a persistent reservoir.

When recurrence is detected, the response should be proportionate and informed by what passive monitoring and host checks reveal. A single reinfection event on one system, traced to a contractor device that bypassed the screening process, is just a process or policy failure that can be corrected without repeating the entire remediation. Widespread resurgence of propagation traffic across multiple segments is a different situation.

Passive monitoring needs to be in place to detect recurrence quickly, and the stakeholder relationships and decision frameworks established during the original remediation are available to respond to it. A second remediation effort, if needed, will be faster and less disruptive than the first, because the environment is now understood and the groundwork has been laid.

SECTION 09:

Conclusion

We opened this paper by noting that nobody is talking about this pervasive problem as a real, serious issue. Having read this far, we hope you understand why that silence is causing an industry-wide problem.

Legacy commodity malware in OT environments is not a historical problem. It is one that is rising to the top of our activation queue as organizations build the maturity to understand the condition of their older networks. The infections are real, they are widespread, and they are being discovered right now by organizations deploying detection capability for the first time. Conficker is still propagating. WannaCry's worm component is still trying to connect to its command-and-control domains. Sality is still infecting executable files on network shares. As systems age and run out of resources, these infections gradually introduce failures and unexpected behavior in life-critical processes.

What has been missing is honest, practical guidance about how to deal with these infections in a sensible, measured manner.

What We Have Covered

The structural reasons legacy malware persists in OT are not easy to fix. Device lifecycles measured in decades, protocols that cannot be easily disabled, removable media that bypasses network controls, transient assets that reintroduce infections from outside the organization, and security controls that have quietly degraded for years: these are not problems that resolve themselves, and they are not problems that a single remediation project permanently eliminates. They require ongoing management.

The tooling landscape is sparse. Popular tools like Velociraptor (as detailed in Appendix A: The Tooling Reality) do not support Windows XP. Real-time antimalware agents for pre-XP SP2 systems are effectively non-existent. Active network scanning carries life and safety implications that do not exist in enterprise environments. Passive monitoring, careful manual collection, legacy-compatible antimalware and forensic tools are what practitioners who may not even have been teenagers during Windows XP's heyday must work with. Understanding this before arriving on a site matters enormously.

Blindly following enterprise Incident Response Plans and risk measures will get you into trouble. The risk asymmetry between the infection and a poorly planned response is real, and we have seen fairly dire consequences of ignoring it. The malware is old, well-understood, and in many cases headless. The response, if unplanned, can cause an operational incident that the malware never did.

Risk is not binary, and there is no universally correct remediation path. Immediate cleanup, a scheduled project, and informed deferral are all legitimate options in the right circumstances. The right answer depends on the specific environment, the specific infections, the specific operational constraints, and the specific risk tolerance of the organization. What is not acceptable is choosing without the information to choose well.

The Bigger Picture

We want to be direct about something that the technical detail of this paper can obscure. Organizations running industrial environments with 10, 20, or 50 concurrent legacy malware

infections are not unique. They are representative of a large portion of the global industrial base. The conditions that allow these infections to persist are structural, not exceptional. The organizations discovering them now are not admitting old failures. They are, for the first time, building visibility to see the real state of their environment. That should be a point of pride for the OT cybersecurity community.

Understanding the state of OT environments is progress, and should be recognized as such, even when what it reveals is uncomfortable.

The response to discovering these legacy infections should be proportionate, risk-informed, and operationally sensible. It should not be driven by alarm, by enterprise IR reflexes, or by overwhelming regulatory pressure that ignores operational and safety context. Done carelessly, remediation causes the very harm it is trying to prevent.

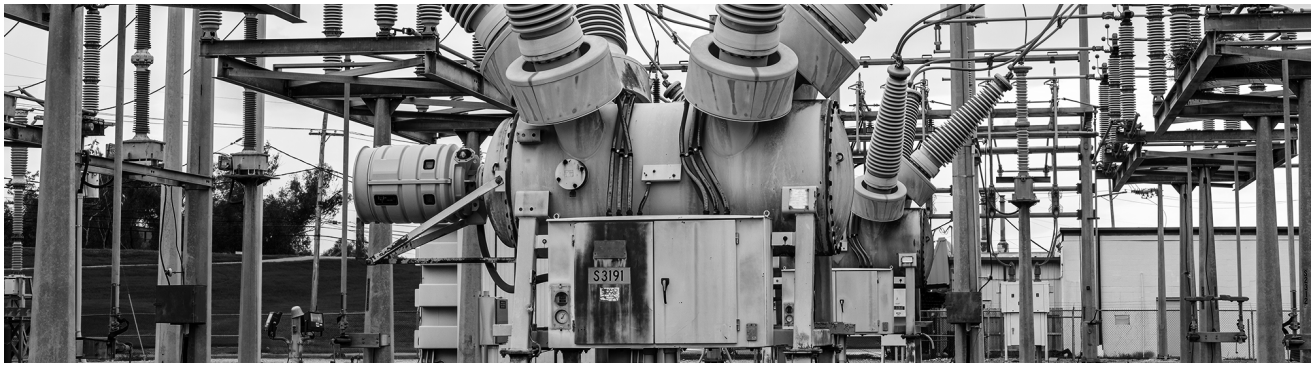
Done well, remediating these infections can be the starting point for something better. For example, network segmentation and protocol restrictions put in place to control Conficker propagation become the foundation of defensible architecture. Passive industrial cybersecurity monitoring deployed to validate cleanup becomes permanent OT visibility. The stakeholder relationships built to navigate the remediation become the backbone of an OT incident response capability. The work is hard and the conditions are challenging, but the path to OT cybersecurity maturity from a successful remediation will be clearer than it was before the infection was found.

A Final Word

If you are a security leader reading this because your organization just discovered a Conficker infection that has apparently been present since 2011, please pause and consider this whitepaper before reacting. You are not alone, and this strange circumstance is not as unusual as it probably feels right now.

Get the right stakeholders in the room. Understand what you are dealing with before you decide what to do about it. Choose a path deliberately and document your reasoning. Engage OEMs, engineers, safety personnel, and vendors early. Seek out external incident response expertise if needed. Control reinfection vectors before you try to clean the hosts. Validate thoroughly before you declare completion. And when the work is done, build on it.

That is the practical approach this paper has described. We hope it is useful.



A1–B1.

Appendix

APPENDIX A:

The Tooling Reality

Practitioners coming from enterprise backgrounds may find the legacy OT tooling landscape distressing. The options are limited, several carry their own risks, and some of the most capable modern DFIR tools simply cannot reach the systems that need them most. Understanding this landscape honestly is essential before any scoping or remediation work begins.

We have organized this section into two parts: scoping tools, which help establish the presence and spread of infections, and remediation tools, which address cleaning and prevention. The two categories require different considerations and carry different risk profiles. This list is not comprehensive and provides examples of popular tools which might be considered.

Scoping Tools

The goal of scoping in a legacy OT malware engagement is to ensure the team has a full picture of which systems are infected, which malware families are present, how infections are spreading, and what reinfection vectors may exist. In a modern enterprise environment, much of this can be accomplished with agent-based tools deployed across the fleet. In legacy OT, the approach must be far more deliberate.

Passive Network Monitoring[6]

The safest and most appropriate first step in any legacy OT malware scoping exercise is passive network monitoring. Network-level observation via a span port or network tap can detect the propagation behavior of Conficker (MS08-067 exploit attempts, NetBIOS traffic anomalies), WannaCry (EternalBlue scanning, DoublePulsar callbacks), and Sality (peer-to-peer traffic patterns) without touching any host. No agents are deployed. No host resources are consumed. No process is disrupted.

The Dragos Platform provides this capability. Passive monitoring via the Dragos Platform is a great starting point for any organization that wants to understand the scope of legacy malware infections in their OT environment. It provides network-level visibility into malware propagation behavior, identifies vulnerable and infected hosts based on observed traffic, and does so continuously without requiring any host agent deployment. For organizations without an existing OT detection capability, this is also the right moment to begin building one, as it supports ongoing detection well beyond the immediate cleanup effort.

A Critical Warning on Active Network Scanning

When familiar DFIR tools fail, many response teams may be tempted to scope the condition of the network by intrusively vulnerability scanning. Active network scanning in OT environments requires extreme caution and should only be considered after careful risk assessment, with explicit approval from engineering, operations, and vendor stakeholders.

This is not a standard security caveat. This can impact people's lives and safety. In legacy industrial environments, active scanning traffic can directly cause process disruption, safety system responses, device lockouts, and communication failures on time-sensitive network segments. Legacy PLCs, RTUs, and network switches were not designed to handle the traffic that modern scanning tools may generate. The consequences of a scanning-induced fault in an

[6] Dragos, Inc. "Dragos Platform." Dragos, Inc. Accessed July 2026. https://www.dragos.com/cybersecurity-platform/

operational environment can include loss of process visibility or control, unplanned shutdowns, equipment damage, and in the worst cases, physical harm to personnel.

Tools like Nmap include NSE scripts capable of detecting Conficker and EternalBlue-vulnerable hosts via network-based checks without requiring host access. These can be useful in controlled conditions on isolated or powered-down network segments. They should never be run against live OT networks without explicit engineering approval, thorough understanding of all connected devices, careful rate limiting, and a clear understanding of what a scanning-induced fault would mean for the process.

If in doubt, passive monitoring, configuration analysis, and physical site inspection are always safer than active scanning.

Volexity Surge Collect[7]

For host-based evidence collection from systems that can safely be accessed, Volexity Surge Collect is Dragos's preferred forensic collection tool. It collects physical memory as its first and most volatile target, followed by a configurable set of disk artifacts defined by a collection file. It is lightweight, does not require installation, and is extremely customizable.

A basic configuration collects a comprehensive set of standard Windows forensic artifacts such as the Master File Table and NTFS journal, system and user registry hives, event logs, WMI repository, prefetch, recycle bin contents, scheduled tasks, USB device connection history, Windows Update and patch logs, Amcache and RecentFileCache (program execution evidence), PowerShell history, remote access tool logs covering a broad range of commercial tools, and antivirus logs and quarantine data for common products.

Critically for OT engagements, Surge Collect can be configured to also collect vendor-specific logs and project files for a wide range of ICS software. In a legacy OT malware engagement, these vendor logs can provide valuable evidence of malware interactions with industrial software and configuration changes that occurred during the infection period.

Surge Collect supports Windows XP to a limited extent. Compatibility for specific legacy OS builds should be confirmed before deployment.

Surge Collect should be run from a clean and dedicated USB drive, using local administrator credentials only wherever possible. Each drive should be used on one system only to avoid cross-contamination. On OT hosts, technical authority approval must be obtained before running any collection tool, as even low-touch command line tools will make some modifications to the system and evidence which analysts must be aware of.

Memory Forensics: Volatility[8] and MemProcFS[9]

Memory forensics is one of the most valuable scoping techniques available in OT, because it can surface running malware, injected code, and live network connections without writing to or

[7] Volexity. "Surge Collect." Volexity. Accessed July 2026. https://www.volexity.com/products-overview/surge/

[8] The Volatility Foundation. "Volatility." GitHub, Volatility Foundation. Accessed July 2026. https://www.volatilityfoundation.org/ and https://github.com/volatilityfoundation/volatility3

[9] Frisk, Ulf. "MemProcFS: The Memory Process File System." GitHub. Accessed July 2026. https://github.com/ufrisk/MemProcFS

otherwise altering the production host. Two open source tools dominate this space. Volatility is a mature memory forensics framework that supports analysis of memory images from Windows XP SP2 and later, across both 32-bit and 64-bit builds, through an extensive library of analysis plugins. MemProcFS takes a different approach, mounting an acquired image or live memory as a browsable virtual file system so that processes, handles, network connections, and registry hives can be examined with ordinary file tools instead of complex command-line arguments. The two are complementary: the plugin depth of Volatility pairs well with the speed and intuitive navigation of MemProcFS, and many responders run both against the same image.

Both tools analyze the acquired image offline rather than running live on the host, which cleanly separates collection from analysis. Memory can be captured from a legacy host using any of several free or commercial acquisition utilities and then analyzed on a forensic workstation with no further interaction with the production system. This suits OT environments, where minimizing time and interaction on a live process host is a priority. Memory images are large, however, so copying them off the device can be difficult on aging systems and low-bandwidth networks.

Velociraptor[10]

Velociraptor is a powerful open source DFIR platform widely used in enterprise IR engagements. Velociraptor does not support Windows XP or Windows Server 2003. Its minimum supported Windows version is Windows 7. Practitioners who rely on Velociraptor for enterprise work should plan for this gap and ensure alternative collection methods are available for legacy OT hosts. In an environment dating back no further than Windows 7, it may be a possible alternative to agent-based analysis. Velociraptor can be run in agentless or standalone CLI modes, which provides more flexibility.

YARA[11], PowerShell, Psexec, and Bash

Where supported, remotely or locally executing custom Windows scripts to check file presence, registry keys, and even running processes can be a last-ditch mechanism for scoping infection. Of course, this requires the ability to run the scripts with appropriate permissions across a wide swathe of potentially infected computers (and receive a result) in an efficient way. PowerShell was installed by default from Windows 2008 onward. Coverage of optional PowerShell and Psexec installations on older systems will vary. If all else fails, bash scripts with supported commands will manually run on Windows systems of all ages. Some of the early forensic triage scripts from the early aughts and 1990s still exist on the internet.

YARA is an open source malware signature language that can be deployed in a variety of open source and commercial network and host-based cybersecurity tools. Unbeknownst to some analysts, YARA allows the creation of small, custom executables to search hosts for a set of indicators of compromise on a computer, then return a “match” or “no match” response. There are free community signatures available for most of the malware variants discussed in this whitepaper, and a savvy analyst may be able to identify a safe and authorized method to efficiently script a YARA executable to run across multiple hosts to screen for specific indicators of a problem variant.

[10] Velocidex Enterprises. “Velociraptor: Endpoint Visibility and Collection Tool.” GitHub. Accessed July 2026. https://github.com/Velocidex/velociraptor

[11] VirusTotal. “YARA: The Pattern Matching Swiss Knife for Malware Researchers.” Accessed July 2026. https://virustotal.github.io/yara/

Traditional Disk Analysis Suites

While collecting hard drive images and performing disk analysis is time consuming and should be used sparingly as required to spot check crown jewel systems and understand infection presence and spread, it is one of the most backwards-compatible analysis methods available to forensic analysts. Tools like OpenText EnCase, Sleuth Kit Labs Autopsy, and Exterro FTK are still capable of analyzing very old file systems and providing useful data on the presence of files and system modifications.

Note that when our modern forensic tools do not support the systems we are responding to, we continue to fall back to older and more traditional mechanisms of answering key investigative questions. Hard drive forensics is one of the oldest forms of digital forensics in existence. Unfortunately, the number of cybersecurity analysts who still understand how to do malware related investigations using only these very traditional methods on old computers is shrinking significantly over time. Educational programs generally are not producing replacements for us.

Remediation Tools

The remediation tooling landscape for legacy OT is, if anything, more constrained than the scoping landscape. The options that exist are worth understanding clearly.

ClamWin[12]

ClamWin is a free, open source antivirus tool for Windows built on the ClamAV engine and maintained by Cisco Talos. It provides on-demand and scheduled scanning, and remediation of infection when malware is detected. It supports Windows 98 and newer, making it one of the very few anti-malware options available for some of the oldest OT Windows hosts. ClamWin provides on-demand and scheduled scanning; it does not include a real-time agent or central management.

In OT environments, the absence of a real-time agent may be an advantage as well as a limitation. Real-time scanning agents consume resources continuously, can interfere unpredictably with time-sensitive industrial applications, and require vendor approval before deployment. ClamWin's on-demand model allows analysts to run controlled, permissioned scans at chosen times without permanently installing a resident process on a production host. Unfortunately, this also requires user initiation on every impacted computer.

ClamWin's virus definitions are actively maintained and cover all three primary families discussed in this paper. It is free to deploy, requires no licensing, and does not introduce an ongoing commercial vendor relationship. For organizations that cannot obtain or afford commercial antimalware options for legacy hosts, ClamWin is a credible and practical first-line scanning tool.

Without a real-time agent, it will not prevent reinfection between scans. It is a detection and reporting tool, not a continuous protection mechanism. Cleanup must be validated and reinfection vectors must be addressed separately, or this very time consuming effort will be futile.

[12] ClamWin Pty Ltd. "ClamWin Free Antivirus." Accessed July 2026. https://www.clamwin.com/

Commercial Extended Support Options

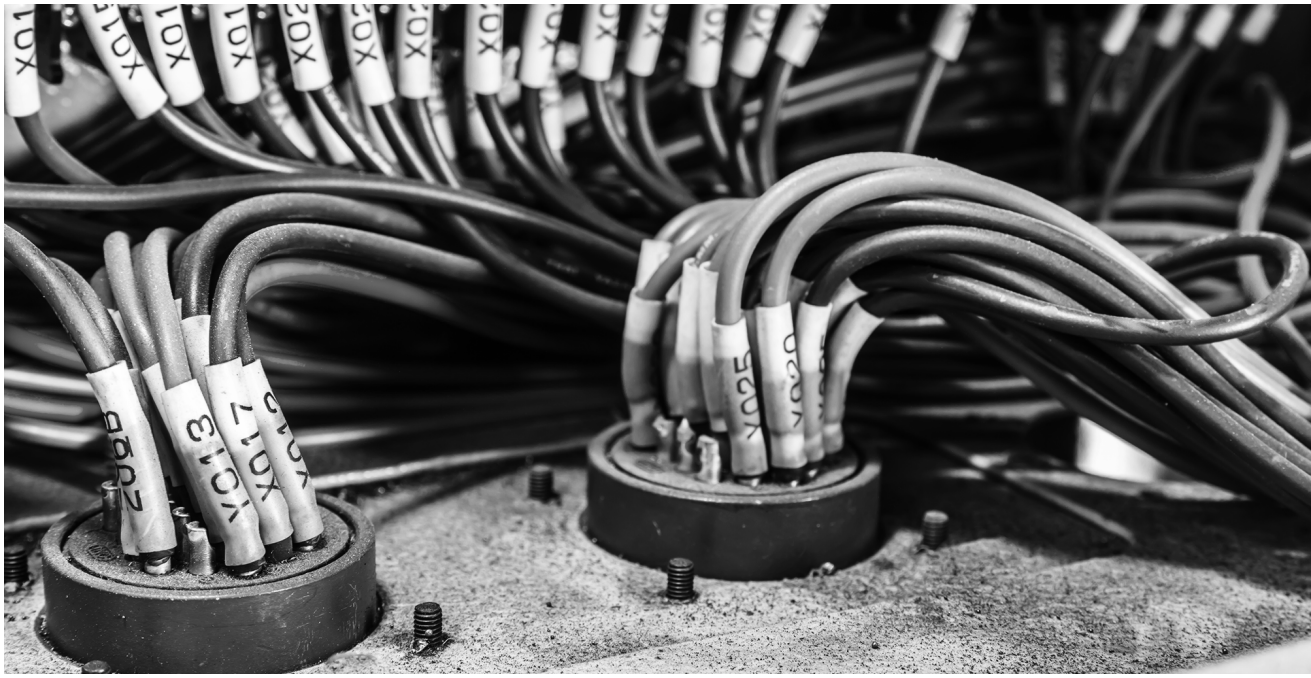
Some commercial antimalware vendors offer legacy OS support on extended or custom terms. Bitdefender has provided limited antimalware signature updates for Windows XP and Windows Server 2003 customers under extended support agreements. Availability depends on existing vendor relationships and contractual arrangements, and organizations should confirm current support status directly with the vendor.

We have deliberately not named certain other commercial options that have been cited in legacy OS antimalware discussions, due to vendor-specific concerns around supply chain integrity and regulatory guidance that some of our clients operate under. Practitioners should evaluate any commercial product against the specific regulatory and risk context of their organization.

The bottom line is, do a thorough evaluation of all the supported and unsupported commercial tools that you have in your toolbox, and do not remove any options from the table without consideration and evaluation. If the only tool you have that will work and be accepted by the vendor is an end of life antivirus, evaluate if you can make it work to solve the immediate problem, temporarily.

The Honest Assessment

There is unfortunately no fully supported, actively developed, commercial antimalware product that provides real-time centralized protection for pre-Windows XP SP2 systems at time of publication. This is a gap that neither practitioners nor vendors can wish away. It is one of the clearest illustrations of why legacy commodity malware in OT cannot just be solved by deploying a tool. It requires a program: network controls, removable media policies, transient asset management, patch evaluation, and where possible system upgrades, sustained over time.



APPENDIX B:

The Most Common Threat Families

What sets apart legacy commodity malware in OT is that we already understand these families thoroughly. The technical behavior of Conficker, WannaCry, and Sality has been documented in exhaustive detail for years, even for analysts too young to have experienced its heyday. What practitioners may lack is not knowledge of the malware itself, but a clear picture of how it behaves specifically in legacy OT environments, how it gets in, and why it is so very challenging to remove completely.

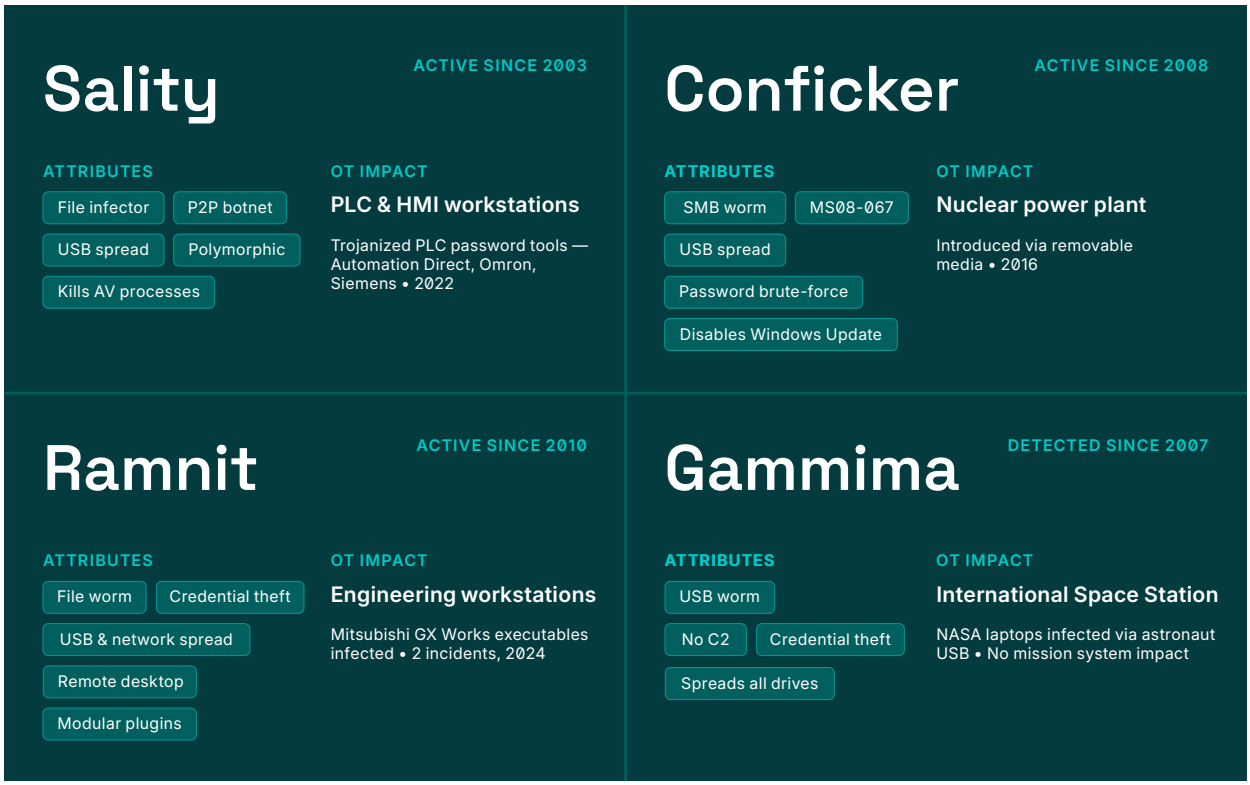


Figure 2 | **Example: Legacy Malware Family Overview**

Let us take a moment to revisit some of the common, pervasive malware variants that Dragos sees in industrial environments, particularly for readers who entered the field post-2010s.

Conficker (Win32/Conficker, Downadup)

Conficker was first identified in November 2008, exploiting MS08-067, a critical and reliably exploitable vulnerability in the Windows Server service. Conficker had a huge impact on the cybersecurity industry in its formative years and remains one of the most prevalent malware families in OT environments globally. This is not because the worm is sophisticated, but because it is especially virulent, and the conditions that allow it to persist in legacy environments have not changed.

Conficker propagates through three primary mechanisms[13]: exploitation of MS08-067 over the network, brute-forcing weak admin credentials on network shares, and infection via removable media using the Windows Autorun mechanism[14]. In OT environments, all three vectors remain viable. Whilst an emergency fix was released even for some end of life Windows versions, MS08-067 is unpatched on many legacy systems because vendor certification has not been obtained or replacing significant system components is not feasible. Administrative and service account credential reuse and sharing in industrial environments is pervasive. Removable media policies are inconsistently enforced, especially against shadow IT and onsite vendors.

Once established on a host, Conficker disables Windows Update and Windows Security Center, blocks access to several security vendor websites, and attempts to download additional payloads from a rotating list of domains generated by a domain generation algorithm. These domains are generally long expired or seized, rendering Conficker "headless", and OT network restrictions typically prevent direct outbound connections, anyway. This creates a deceptive stability: the infection appears benign because it is not receiving commands, but it continues to spread, consumes host resources, and can introduce system instability and vulnerability to aging hardware with limited memory and processing capacity.

Conficker variants continue to be detected on ICS endpoints across multiple sectors. Trend Micro data from 2020 found Conficker still present on hundreds of ICS endpoints[15], with variants actively spreading via brute-forced administrative shares even on newer operating systems.

WannaCry (WannaCrypt, WCry)

WannaCry caused global disruption in May 2017, encrypting systems across more than 150 countries in a matter of hours. The ransomware component received most of the attention at the time, critically impacting enterprises and industrial environments alike. In OT environments in 2026, the worm component is the more persistent concern.

WannaCry exploits EternalBlue (MS17-010), a vulnerability in the Windows SMBv1 implementation, and uses DoublePulsar as a backdoor installer. It spreads autonomously across networks by scanning for and exploiting vulnerable SMBv1 hosts, requiring no user interaction. The original WannaCry campaign was slowed by the accidental discovery and registration of a kill switch domain: the malware checked for the domain's existence before encrypting, and registering it halted the encryption across internet-connected systems.

OT environments present a specific and underappreciated risk here. Networks with no outbound internet access cannot reach the kill switch domain. The kill switch does not fire unless emulated internally. WannaCry's worm component continues to propagate across the internal network indefinitely, scanning and reinfecting vulnerable hosts without ever triggering encryption in some variants. This means that an OT network that followed sound security practice by isolating itself from the internet may, in a specific sense, be more exposed to continued WannaCry propagation than one with outbound connectivity.

[13] Cybersecurity and Infrastructure Security Agency (CISA). "Conficker Worm Targets Microsoft Windows Systems." US-CERT Alert, March 2009. https://www.cisa.gov/news-events/alerts/2009/03/29/conficker-worm-targets-microsoft-windows-systems

[14] Wikipedia contributors. "Conficker." Wikipedia, The Free Encyclopedia. Accessed July 2026. https://en.wikipedia.org/wiki/Conficker

[15] Trend Micro. "2020 Report on Threats Affecting ICS Endpoints." Trend Micro Research, June 2021. https://www.trendmicro.com/vinfo/us/security/news/internet-of-things/2020-report-ics-endpoints-as-starting-points-for-threats

The Microsoft patch for MS17-010 was released in March 2017, two months before the WannaCry campaign. The problem was so severe that emergency patches were later made available by Microsoft for Windows XP and Windows Server 2003, which had already reached end of life[16]. Many OT environments have still not applied these patches, either because vendor certification has not been sought, because the systems cannot be taken offline, or because the emergency patches for legacy systems have not been evaluated and approved by engineering and operations teams. Dragos still routinely sees Windows operating systems older than Windows XP, and they are not eligible for most emergency patches.

WannaCry's worm component scanning behavior can also cause direct operational impact on sensitive OT networks independent of any encryption. The volume of SMB scanning traffic it generates can disrupt legacy network equipment, cause packet loss on low-bandwidth segments, and interfere with time-sensitive industrial communications.

Sality (Win32/Sality)

Before ransomware was the most popular moneymaking tool for cybercriminals, botnets made up of quietly infected computers were pervasive. Infected computers were used to launch distributed denial of service attacks or farm banking and game credentials. Sality is a file-infecting virus and peer-to-peer botnet first detected in 2003. It is one of the longer-lived malware families still encountered in production environments, and it behaves differently from the worm families described above in ways that have direct implications for remediation.

Sality infects portable executable (PE) files like .exe and .scr files it finds on local drives, mapped network drives, and removable media. Every infected file becomes a carrier. When that file is copied to another system and executed, the infection spreads. This mechanism means Sality can propagate across "air-gapped" networks through routine maintenance activity: copying a configuration backup, transferring an installer, or running a utility from a shared drive.

In OT environments, Sality has been observed entering through trojanized tools. In 2022, Dragos identified multiple social media accounts selling PLC password-cracking software that was trojanized with Sality[17]. Engineers searching for tools to recover lost PLC credentials downloaded and ran software that silently installed a Sality variant while appearing to perform the requested function. The infection then spread through the engineering workstation's mapped drives and any removable media subsequently used on the system.

Sality's botnet functionality uses a peer-to-peer architecture rather than centralized command and control. Infected hosts communicate directly with each other to receive tasks and updates, making it resilient to takedown and difficult to block with simple network controls. Tasks have historically included cryptocurrency mining, password cracking, and spam distribution. In OT environments with no outbound internet connectivity, the botnet functionality is typically non-operational, but the file infection and propagation mechanisms continue regardless.

[16] Cybersecurity and Infrastructure Security Agency (CISA). "Indicators Associated With WannaCry Ransomware." Alert TA17-132A, May 2017. https://www.cisa.gov/news-events/alerts/2017/05/12/indicators-associated-wannacry-ransomware

[17] Dragos, Inc. "The Trojan Horse Malware & Password 'Cracking' Ecosystem Targeting Industrial Operators." Dragos Blog, July 2022. https://www.dragos.com/blog/the-trojan-horse-malware-password-cracking-ecosystem-targeting-industrial-operators

The file-infecting nature of Sality creates a specific and serious remediation challenge: it is not sufficient to just clean the infected hosts. Every infected executable file on every network share, backup location, and removable media device must be identified and addressed. Failing to do so means reinfection from a shared drive or a vendor USB drive is essentially guaranteed.

The Broader Category: Removable Media Infectors

Beyond these very common variants, OT environments frequently carry a collection of older malware that spread primarily or exclusively via removable media and file shares[18]. These include Gamarue (Andromeda), a botnet worm with removable media propagation capability; Palevo, a worm that spreads via removable media and instant messaging file transfers; Virut, a file infector with network propagation; and Gammima, a credential-stealing worm notable for being detected on NASA laptops aboard the International Space Station in 2008[19], having arrived via removable media despite a truly exceptionally air-gapped environment.

They spread without network connectivity via USB drives, SD cards, and portable hard drives. They often carry no payload that is operationally relevant to the OT environment, having been designed to steal gaming or financial credentials, or perform distributed computing tasks like distributed denial of service as part of a botnet. Their presence in an OT environment is therefore often incidental rather than targeted: an engineer plugged in an infected personal drive, or a vendor arrived with a device used at a previous site.

Their incidental presence does not make them implicitly safe. Any of these families can consume host resources, cause system instability on aging host and network hardware (particularly embedded devices), spread to additional hosts, and in some cases be repurposed or leveraged by a more sophisticated threat group who subsequently gains access to the environment. An environment with many concurrent legacy infections is also an environment where distinguishing a targeted intrusion from background noise becomes significantly harder. Dragos Incident Response has responded to multiple cases where legacy commodity malware introduced system instability on legacy Windows systems, led to failure of high level systems, and therefore led to operational disruption.

How Multiple Families Coexist

It is common for the Dragos Incident Response Team to encounter not one, but a multitude of concurrent infections in a single OT environment. Discovering 10 to 70 distinct legacy infections is not unusual, particularly in older industrial facilities.

This coexistence is possible because these families do not compete with or displace each other. Each exploits a different combination of vulnerabilities and propagation mechanisms, and they are entirely indifferent to sharing a host. Conficker spreads via network exploitation and removable media. Sality spreads via file infection on shares and media. WannaCry scans for EternalBlue targets. A removable media carrier arrives on a contractor's USB drive. None of these mechanisms interferes with the others, and none of them will clean up after themselves.

[18] Trend Micro. "2020 Report on Threats Affecting ICS Endpoints." See endnote [15].

[19] Wikipedia contributors. "W32.Gammima.AG." Wikipedia, The Free Encyclopedia. Accessed July 2026. https://en.wikipedia.org/wiki/W32.Gammima.AG

Citations

[1] Dragos, Inc. "OT Threat Landscape 2026: What OT Cybersecurity Defenders Need to Know." Dragos Blog, April 2026. https://www.dragos.com/blog/ot-threat-landscape-2026

[2] Microsoft. "Microsoft Security Advisory 967940: Update for Windows Autorun." Published February 2009, updated February 2011. https://learn.microsoft.com/en-us/security-updates/securityadvisories/2009/967940

[3] Cybersecurity and Infrastructure Security Agency (CISA). "SMB Security Best Practices." US-CERT Current Activity, January 2017. https://us-cert.cisa.gov/ncas/current-activity/2017/01/16/SMB-Security-Best-Practices

[4] Microsoft. "Preventing SMB Traffic From Lateral Connections and Entering or Leaving the Network." Microsoft Support. Accessed July 2026. https://support.microsoft.com/en-us/topic/preventing-smb-traffic-from-lateral-connections-and-entering-or-leaving-the-network-c0541db7-2244-0dce-18fd-14a3ddeb282a

[5] Conway, Tim and Robert M. Lee. "The Five ICS Cybersecurity Critical Controls." SANS Institute White Paper, 2022. https://www.sans.org/white-papers/five-ics-cybersecurity-critical-controls. See also: Dragos, Inc. "The SANS ICS Five Critical Controls: A Practical Framework for OT Cybersecurity." Dragos Blog. https://www.dragos.com/blog/the-sans-ics-five-critical-controls-a-practical-framework-for-ot-cybersecurity/

[6] Dragos, Inc. "Dragos Platform." Dragos, Inc. Accessed July 2026. https://www.dragos.com/cybersecurity-platform/

[7] Volexity. "Surge Collect." Volexity. Accessed July 2026. https://www.volexity.com/products-overview/surge/

[8] The Volatility Foundation. "Volatility." GitHub, Volatility Foundation. Accessed July 2026. https://www.volatilityfoundation.org/ and https://github.com/volatilityfoundation/volatility3

[9] Frisk, Ulf. "MemProcFS: The Memory Process File System." GitHub. Accessed July 2026. https://github.com/ufrisk/MemProcFS

[10] Velocidex Enterprises. "Velociraptor: Endpoint Visibility and Collection Tool." GitHub. Accessed July 2026. https://github.com/Velocidex/velociraptor

[11] VirusTotal. "YARA: The Pattern Matching Swiss Knife for Malware Researchers." Accessed July 2026. https://virustotal.github.io/yara/

[12] ClamWin Pty Ltd. "ClamWin Free Antivirus." Accessed July 2026. https://www.clamwin.com/

[13] Cybersecurity and Infrastructure Security Agency (CISA). "Conficker Worm Targets Microsoft Windows Systems." US-CERT Alert, March 2009. https://www.cisa.gov/news-events/alerts/2009/03/29/conficker-worm-targets-microsoft-windows-systems

[14] Wikipedia contributors. "Conficker." Wikipedia, The Free Encyclopedia. Accessed July 2026. https://en.wikipedia.org/wiki/Conficker

[15] Trend Micro. “2020 Report on Threats Affecting ICS Endpoints.” Trend Micro Research, June 2021. https://www.trendmicro.com/vinfo/us/security/news/internet-of-things/2020-report-ics-endpoints-as-starting-points-for-threats

[16] Cybersecurity and Infrastructure Security Agency (CISA). “Indicators Associated With WannaCry Ransomware.” Alert TA17-132A, May 2017. https://www.cisa.gov/news-events/alerts/2017/05/12/indicators-associated-wannacry-ransomware

[17] Dragos, Inc. “The Trojan Horse Malware & Password ‘Cracking’ Ecosystem Targeting Industrial Operators.” Dragos Blog, July 2022. https://www.dragos.com/blog/the-trojan-horse-malware-password-cracking-ecosystem-targeting-industrial-operators

[18] Trend Micro. “2020 Report on Threats Affecting ICS Endpoints.” See endnote [15].

[19] Wikipedia contributors. “W32.Gammima.AG.” Wikipedia, The Free Encyclopedia. Accessed July 2026. https://en.wikipedia.org/wiki/W32.Gammima.AG

About Dragos

Dragos is the world's leading OT cybersecurity firm headquartered in Washington DC, USA area with offices around the world. It provides the most effective OT cybersecurity technology for industrial and critical infrastructure to deliver on our global mission: safeguarding civilization. The Dragos Platform provides visibility and monitoring of OT environments for asset identification, vulnerability management, and threat detection with continuous insights generated by the industry's most experienced OT threat intelligence and services team. Dragos protects customers across the range of operational sectors, including electric, oil & gas, data centers, manufacturing, water, transportation, mining, and government.

Learn more: dragos.com