Homeland Security

NCCIC

National Cybersecurity and
Communications Integration Center

INTRODUCTION

This document serves as an appendix to the “Seven Steps to Defend Industrial Control Systems”[a] document, providing additional conceptual-level guidance on implementing application whitelisting.

Application Whitelisting (AWL) can detect and prevent attempted execution of malware uploaded by adversaries. The static nature of some industrial control system (ICS) components, such as database servers and human-machine interfaces, makes these ideal candidates to run AWL. Operators are thus encouraged to work with vendors to baseline and calibrate AWL deployments.

AWL BENEFITS

AWL is a security technology that is a key part of an effective layered defense. AWL allows only selected authorized programs to run, while all other programs are blocked from running by default. AWL is implemented by creating and maintaining a list of approved or “whitelisted” applications or file locations. AWL enforcement mechanisms may include verification of file hashes (e.g., SHA2), software signatures, trusted paths, and/or file names. Note that AWL based on trusted paths is recommended only for corporate environments, and AWL based solely on file names is comparatively weak and thus is not recommended.

When an ICS device is secured with AWL, security software checks each application that attempts to execute against a locally stored approved list (“policy”). Since execution is permitted only for items on the approved list, the control system device is thus able to run legitimate software applications, but malware or other unauthorized programs are blocked. Adversaries attempting to upload and execute malicious software are thus confronted with an environment resistant to execution of malware or offensive cyber tool software.

[a] https://ics-cert.us-cert.gov/sites/default/files/documents/Seven%20Steps%20to%20Effectively%20Defend%20Industrial%20Control%20Systems_S508C.pdf

HOW AWL DIFFERS FROM AND COMPLEMENTS ANTI-VIRUS

While similar in purpose to anti-virus (AV) products, there is an important difference between AWL and AV solutions. AV products identify malware based mostly by using large “blacklists” of signatures. If an executable program matches a list entry, AV software prevents execution and quarantines files. This approach works well against known malware, thus AV earns a place in an ICS defense in depth strategy. However, AV still leaves systems vulnerable to malware that has not been pre-identified and malware whose signatures have not yet been loaded into the AV system. In addition, even unsophisticated adversaries intentionally circumvent AV by creating “one off” unique variants of malware that avoid AV blacklists, or by using techniques such as self-modifying “polymorphic” code that avoids signature-based detection by constantly changing. Studies show the rate of new malware creation vastly outpaces the agility of standard traditional AV[b]. Considering these challenges, AWL complements AV by limiting the programs that can run on a particular system to only those that are pre-approved.

HOW AWL OPERATES

Most AWL solutions come with an initial set of policy rules and support a “learning mode”. As the system operates during the “learning mode”, changes to the policy are accumulated. The initial policy may then be updated based on the system-unique applications seen during “learning mode”. The policy can then be tested before the AWL solution is placed in “blocking mode”. Once placed in “blocking mode” the AWL solution enforces protections, and only items on the whitelist are allowed to execute. AWL can also log blocking actions and/or report blocking actions to a Security Information and Event Management (SIEM) solution.

Many AWL packages also support an “alert mode”. When in “alert mode”, execution of non-whitelisted programs is not entirely blocked - the user is prompted and may choose to allow execution or not.

CREATING WHITELISTS

Whitelists generated during the “learning mode” should be checked carefully to ensure that no pre-existing malware is added into the whitelist. It is advised to start the learning mode on a known good “gold image” of the system. Once AWL is operating effectively, maintenance should be minimal, with the only required ongoing activity being occasional review of blocking logs to determine if any unauthorized applications attempted to execute or if any new legitimate software has been installed that should be added to the whitelist.

[b] Verizon Data Breach 2015 Report

During whitelist creation, ensure that software that may be needed gets included. If there is no local expert who has a detailed understanding about which software programs will get executed under all possible conditions, work with your ICS vendor to get a complete accounting. For example, particular programs may only execute during degraded or emergency situations and therefore might not be seen during the “learning mode”. Ensure such programs are identified and included in the whitelist.

Creating an effective whitelist of necessary applications for more complicated systems is sometimes not a trivial matter. Luckily, many devices within ICS environments do not suffer from constantly changing requirements like corporate hosts and instead operate in a very deterministic manner with a relatively static set of applications. This consistency allows for a very tailored and precise whitelist to be configured and applied to each of the supported systems within the control system environment, with much greater success and fewer problems than on a traditional corporate system.

Some AWL packages support trusting all applications that are signed by specified vendors. Using an AWL with this feature can simplify whitelist creation and maintenance. Once a vendor is trusted, all software signed by that vendor will be whitelisted.

AWL AS A CHANGE CONTROL PROCESS VERIFICATION TOOL

AWL may be leveraged to help ensure installed applications have been vetted properly, gone through the proper security checks, and are approved. Comparing the lists of whitelisted programs approved to execute under AWL to records of vetted programs can provide such validation.

AWL LIMITATIONS

While AWL lends itself very well to defending ICSs, it is not a “silver bullet” solution for everything that may afflict a host within the control system network. Just like a physical insider that can penetrate some of the most elaborate security mechanisms, AWL does not prevent “approved” programs from being compromised in the supply chain or from being exploited and turned against a system. Such subverted applications are not flagged by AWL because such exploitation takes place by misuse of legitimately whitelisted programs. Acknowledging this short-coming, some AWL vendors have developed products to address some of these weaknesses through optional memory-protection features. AWL will also continue to be challenged by malware that exploits applications that run in higher-level execution environments, such as Java, .NET Framework, and scripting languages. Finally, if an adversary has already gained privileged credentials despite the whitelisting, the use of these tools will often not be flagged by AWL.

CHOOSING A COMPATIBLE AWL SOLUTION

The chosen AWL software must be compatible with the operating system on the ICS device. Some ICS devices have legacy operating systems that may not support any or all AWL solutions. AWL software that is compatible with the legacy operating system must therefore be selected and installed. There are AWL solutions on the market which operate on many different platforms. In cases where no AWL solutions are available, it is important to note that in your cybersecurity framework[c], and use compensating controls.

Some AWL solutions are certified by certain ICS vendors to operate with certain ICS components. Some ICS vendors offer managed security services. Owner-operators should be aware of the products and services available for their particular devices. If no products or services are available, customers should begin advocating to their vendors to make these available and integrate AWL support into future contracts.

CHALLENGE OF RUNNING AWL IN SOME SPECIALIZED ENVIRONMENTS

Getting AWL to run on some ICS systems may be a challenge. Systems that use a large set of custom drivers and services may require tweaking to get AWL successfully operating. For example, some programmable logic controller engineering work station environments have dozens of custom services and drivers that support legacy networking functions and custom hardware devices. In these situations it is vital to work with vendors to ensure identification and encapsulation of all needed services in the whitelist. Sometimes getting AWL successfully running on such a system may require finding and working around conflicts between the services and drivers used to support the system. Fortunately, the majority of PC-architecture based ICS systems are simpler and thus easier places to implement AWL. Leveraging the "learning modes" supported by most AWL solutions can be critical to getting a solution working.

It is critical to test each AWL solution in an appropriate test bed before implementing it on an operational ICS. Such testing should specifically include operating the ICS in situations where any infrequently used modules get executed, thus ensuring all relevant programs are included in the whitelist.

In addition to all vendor-provided executable code, custom developed code should also be included in the application whitelist.

In critical process environments, it may be advisable to configure the AWL in an alert mode, rather than blocking mode. A false positive could negatively impact a critical process. In any

[c] http://www.nist.gov/cyberframework

case, all vendors of an ICS should be consulted in the design and implementation of the AWL solution.

PROTECT ADMINISTRATOR ACCESS

Administrative access should be carefully controlled, since most AWL solutions allow administrators to modify and/or bypass protections of the whitelist. Any modifications to the whitelist should alert all of the administrators that a change has occurred.

ICS Owner-Operators should also be aware of any policy compliance implications (e.g., NERC/CIP) and safety-related implications throughout the entirety of the system's lifecycle.

MANAGING AN AWL SYSTEM

If local expertise in ICS is limited, AWL supported as a service by ICS vendors may be preferable. The organization and ICS equipment provider(s) and/or maintainer(s) should communicate and fully document if the AWL solution will impact the way the system is operated, including in emergency/contingency situations so that technicians will be able to use any management tools not executed on a daily basis when necessary.

If a hash-based whitelisting approach is used, then as patches and updates are installed one must give permission for the new software to operate under the AWL. Thus, the extra step of adding the patched software to the whitelist must be added as part of the update installation process. However, note that as discussed previously, AWL systems that support acceptance of certificate-signed code will not require this extra step for software signed by trusted vendors. Testing updates is still recommended as part of typical configuration management.

It is important that any technicians maintaining the system be fully aware that AWL is operating so that any tools they may bring to the system can be allowed to run based on the whitelisting policy. This may mean having certain directories or publishers whitelisted for diagnostic or maintenance tools.

SUMMARY

While not a cure-all, properly configured AWL should be an integral component of a defense-in-depth solution.

ACKNOWLEDGEMENT

This document "Guidelines for Application Whitelisting in Industrial Control Systems" was written collaboratively, with contributions from Subject Matter Experts working at the Department of Homeland Security (DHS) and the National Security Agency (NSA).

ADDITIONAL RESOURCES

DHS article “Application Whitelisting in an ICS Environment,”[d] DHS ICS-CERT Monitor July, August, September 2013.

NIST Special Publication 800-167; “Guide to Application Whitelisting.”[e]

NSA Publication, “Application Whitelisting Using Microsoft AppLocker,”[f] August 2014.

NSA Publication “Application Whitelisting Using Software Restriction Policies,”[g] Version 1.1, August 2010.

NSA/IAD Publication MIT-006FS-2013 “Application Whitelisting.”[h]

DISCLAIMER

The information and opinions contained in this document are provided "as is" and without any warranties or guarantees.

CONTACT INFORMATION

POC	Phone	e-Mail
Department of Homeland Security ICS-CERT	877-776-7585	ICS-CERT@HQ.DHS.GOV
National Security Agency (Industry) Industry Inquiries	410-854-6091	bao@nsa.gov
National Security Agency (Government) IAD Client Contact Center	410-854-4200	IAD_CCC@nsa.gov

[d] https://ics-cert.us-cert.gov/sites/default/files/Monitors/ICS-CERT_Monitor_Jul-Sep2013.pdf

[e] http://nvlpubs nist.gov/nistpubs/SpecialPublications/NIST.SP.800-167.pdf

[f] https://www.nsa.gov/ia/_files/app/Application_Whitelisting_Using_Microsoft_AppLocker_FINAL.pdf

[g] http://www.nsa.gov/ia/_files/os/win2k/application_whitelisting_using_srp.pdf

[h] https://www.nsa.gov/ia/_files/factsheets/i43v_slick_sheets/slicksheet_applicationwhitelisting_standard.pdf