APPLICATION WHITELISTING (AWL): STRATEGIC PLANNING GUIDE

Purpose

There are many options and facets to deploying Application Whitelisting (AWL) in an existing operational environment. This document highlights and summarizes the types of choices, and the related decisions, that need to be made prior to starting the planning process. It also provides a high-level diagram for an incremental deployment process. This document is designed to assist in the development of a plan for implementing AWL that has a higher chance for success.

Background

It seems that the extended security community has come to a consensus that AWL is one of the most important security technologies/techniques an organization can and should implement.

There are plenty of commercial tools and vendors that advertise their ability to perform AWL. A few of the products in widespread use have built-in AWL capabilities, and there are plenty of informative and easy to understand guidance documents on how to use those capabilities. Still there appear to be fewer successful AWL implementations than horror stories or failed attempts.

One of the main reasons for this is because there are multiple drivers that can constrain implementation (e.g., resources, mission, technology), multiple dimensions of an implementation (e.g., ease of adoption, resource investment, deployment options, issue resolution processes), and multiple "glide path" options for each dimension that need to be considered ***before*** any planning activity can begin. These dimensions and options are not independent, but work together to define an implementation strategy for an organization.

Wading through the experiences of both successful and unsuccessful AWL implementations, studying what worked and why, we have developed a general "plan" for success. This plan identifies areas where an organization needs to understand their operational activities and environment prior to making implementation decisions for AWL. These areas include:

- Which AWL methodology is most appropriate
- Whether to use free or commercial products/services
- What staff skills are available
- How long it will take for users to get used to restrictions
- Which organizations will be easier to migrate

- What types of devices will be easier to migrate
- What the existing activities/plans are for upgrading devices and software

Methodologies

There are multiple methodologies for AWL, and they each have different requirements with respect to resource investment, address different types of threats, and have different success rates depending on organizational factors. In general it is believed you should do any that are "easy" for your organization and then build to a hash-based methodology. That is the methodology that mitigates the most risk by addressing the most advanced threat.

Location-Based (or Path-Based)

The software path, or location on the system, is an attribute commonly used to fingerprint and identify software. In this case, whitelisting specifies a path or multiple paths where software must be located. Anything outside of those paths are not allowed to execute. In addition, those paths are then protected so only administrators can install software in them.

Certificate-Based

Also known as "signing," certificates are created to certify that the applications come from a trusted source. Applications that have been digitally signed by a software vendor's trusted certificate can be assessed by the operating platform and the whitelisting software. Many whitelisting tools refer back to the central server for verifying digitally signed applications due to a higher degree of trust associated with a centrally managed whitelist.

Reputation-Based (or Service-Based)

Many applications have installed or affiliated services associated with them. States or values (e.g., file hash, URL, IP address) are defined per policy, and then the software assesses, identifies and/or compares the state or value at a given time. The most common of these are Reputation Services where vendors have large amounts of information about files and whether they are known good or known bad.

Behavior-Based

Users define specific user and system behavior sequences that the whitelisting program allows. For example, if a particular application routinely (and legitimately) spawns new processes or writes to the hard disk.

Hash-Based

A cryptographic hash can be created for a file or groups of files affiliated with an application using commonly accepted protocols like MD5 or SHA-1. Some of these hashes could be pre-generated by a vendor and available publicly, such as those distributed by Sun Microsystems for Solaris application executables. Others can be generated by whitelisting software at the time of policy generation and enforcement. These hashes are periodically compared to new hashes generated "on the fly" to ensure the software is the same and has not changed.

Affected File Types

A second part of the methodology choice, is also which types of software files will be affected by the application whitelisting implementation. In each of the methodologies above, the implementation will be easier at first by affecting only main process executables, while expanding gradually to affect software libraries, installers, and then scripts as well.

Implementation Stages

For any AWL methodology, there is typically the same 5-stage implementation approach. You choose how long to stay in any stage and how to implement these stages across different user communities in your environment as appropriate for your organization.

Train

In this stage you produce alerts for analysis but do not actually block anything. The result of the analysis is what is used to initially configure the tool. The more refined you want your rules to be after this stage is directly related to the skill set of the resources applied to analyze the alerts.

Alert & Refine

Initial Deployment

In this stage you produce alerts to go to the appropriate IT personnel to make sure the tool is working as expected in the operational environment.

User-Enabled Bypass

In this stage you block according to the rules produced in previous stages, but you allow the user to confirm execution. Alerts related to every User bypass go to the appropriate IT personnel for rule refinement.

IT Support-Enabled Bypass
In this stage you block according to the rules produced in previous stages, but you allow the user to contact IT support personnel to enable execution to be allowed. Tickets or summaries related to all IT support requests of this type are sent to the appropriate IT personnel for rule refinement.

Block
In this stage you block according to the rules produced in previous stages. The only option to enable execution is an 'Appeal' process that is established for the organization.

Key Implementation Factors

Review of successful and unsuccessful AWL implementations identified that there are two key factors directly associated with success: resources and organizational readiness. Without the appropriate investment or application of the correct skill at the necessary time, most AWL implementations will not be sustainable from an operational perspective. In these cases, the initial deployment will be all that gets implemented. More often than not, the initial deployment provides minimal protection against external threats. Organizational readiness determines if the organization has the software policies, management support, and operational understanding necessary to establish and enforce an AWL policy that both mitigates risk and minimizes operational impact.

This section describes certain aspects of each factor, while Appendix A provides a summary of each methodology and its potential impact on resources and operations.

Resources
Quality AWL implementations require investments in both people and products. They also require investments of time and money for planning, testing, deployment, and maintenance of the AWL policies and technologies. You can often minimize investment in products by using methodologies that are already supported in your environment, but these often need to be developed and supported by the individuals with more advanced knowledge about the technology and your environment.

There are typically three different skill sets required to deploy, refine, and maintain an AWL solution. First you need the resources to configure, test, and manage the technology that implements the solution. These resources must be dedicated or permanently assigned to this

activity. Second, you need the resources to support ongoing analysis of alerts and issues to identify more appropriate policies. You will need more of these resources available at times when you are changing the implementation or deploying to new parts of the organization. Third, you need the highly coveted mission critical systems and applications experts. You will only need these resources when deploying an AWL solution to the devices associated with one of these systems or applications.

The use of existing personnel to support AWL implementations and deployments can be optimized if you take advantage of other related operational activities. For example, if there has been a breach and the response is to reimage devices and enforce more restrictive policies, incorporate an AWL solution into this reimaging activity. There will already be personnel assigned to test the more restrictive policies that have the skill set necessary to train and analyze alerts from the AWL solution. Someone will already be assigned to define what software is to be installed to include installation and configuration options, so this work can just feed the AWL policy development process.

Organizational Readiness

AWL solutions require a certain amount of cultural readiness and process maturity to be successful. There has to be a culture that supports the enforcement of restrictive policies, from user acceptance to management support. There needs to be ongoing engagement between: users and leadership; users and AWL solution implementers; leadership and AWL implementers; as well as AWL implementers and IT personnel supporting other IT deployments/implementations. Two way communication channels need to be established, maintained, and used throughout the implementation/deployment activity.

AWL policies are built off of existing policies related to authorized software, installers, and installation options. One of the major concerns organizations have with respect to implementing AWL solutions is the potential for operational impact and disruption to the mission. To perform an analysis of the risk potential for any implementation/deployment option, there must be an accurate view of the operational environment. This includes what types of devices, technologies, and connections are deployed and supported. It also includes information like: what devices are associated with what missions and organizations; what policies apply to what devices; what are the mission requirements for hardware and software; and what limitations or constraints will apply to any AWL solution.

Implementation Dimensions

There are so many related dimensions associated with AWL implementation that decisions surrounding AWL solutions can be very complex. The key drivers for what type of AWL implementation you can deploy in your organization are the same as practically any other IT investment/deployment decision: technology, resources, and mission. Implementation solutions that will match the objectives and constraints defined by these drivers need to be based on an understanding of the different options for adoption, resource availability, and deployment. All of these options have a "glide path" that allows an organization to start small with a high likelihood of success, and then incrementally improve AWL capabilities over time. Some of these options are mutually supportive, meaning that improvement in one area can be naturally linked with improvement in another area. Some of these options are at odds with each other, meaning that a lower starting point in one of them implies you have to start at the higher end of another one.

The following lists questions to consider for the dimensions and options that seem to have the most impact on planning successful AWL solutions.

Adoption – Technology

Prevalence

What technologies are most prevalent in your environment? Are there a set that span a large set of users and systems that already have AWL support or software policies defined?

Consider implementing AWL solutions for the most prevalent technologies and then incrementally improve to include lesser-used technologies.

Device Type

Are the software policies for certain device types already defined and managed? Do some devices already have more restrictive policies applied to them? Is there less disruption to users if AWL solutions are applied to certain servers or infrastructure devices instead of workstations?

Consider implementing AWL solutions for the device types that already have well defined or restrictive policies and then incrementally improve to include more device types.

Consider implementing AWL solutions for device types that have less impact on the end users and then incrementally improve to include user devices and workstations.

Native Support
What operating systems and products already deployed in your environment can implement AWL solutions? How prevalent are they? What systems and devices do they cover? What does it take to use the native capability?

Consider implementing AWL solutions that are supported by technologies already deployed in your environment and incrementally improve to include deployment of new technologies or moving to a single AWL methodology.

Adoption – User Base

User Acceptance
Are there certain organizations that are used to being exempt from security policies? Are there certain organizations that are more likely to accept software restrictions? Are there organizations where there is already executive managerial support to help facilitate any potential pushback on implementation?

Consider implementing AWL solutions in organizations that are more accepting or less resistant to strict software policies and incrementally improve to include less accepting/more resistant organizations.

User Environment
Are there some systems where every device has the same image? Are there some systems where all user needs for software are the same? Are there some systems that change only with technology upgrades and not frequently changing mission needs? How many devices are covered by these systems?

Consider implementing AWL solutions on images/devices that change infrequently or do not have multiple variations and incrementally improve to include more diverse images/devices.

Adoption – Technology and User Base

Options in the adoption dimension tend to be mutually supportive. Consider deploying first on the most prevalent technology with native support for AWL on select device types in the organizations that are most accepting of software restriction policies. Incremental improvements to each of these can occur at different times based on operational readiness – so you decide if

you want to first include more technologies before including more organizations or device types.

Resources – Products

Cost

How much money do you have to buy AWL products? How many of your devices are covered if you use native or "free" capabilities? Are you planning a major deployment of a product that natively supports AWL or can be configured to support AWL?

Consider implementing AWL solutions using capabilities that are free, natively supported, or already purchased for your organization and then incrementally improve to include deployment of new capabilities or a single AWL methodology.

Resources – Administrative

Staff Skills

What staff do you have available to support development, testing, deployment, and management of an AWL solution? What are their skill levels? Do you have personnel who can analyze alerts? Do you have personnel that are subject matter experts in mission critical systems and applications?

Consider implementing AWL solutions in stages that require the least specialized skills to deploy and initially support and incrementally improve to include refinements that require specialized knowledge or skills.

Staff Availability

Do you have staff identified to provide dedicated support to the AWL solution or do you need to share personnel with other IT activities? What is the availability of the different personnel required for refinement of AWL implementations? What is the availability of personnel to collect and study lessons learned from incremental deployments? What is the availability of personnel to study the readiness of any implementation dimension to be improved across some segment of the organization? What backup and succession personnel will be prepared to take over when the current personnel are not available or switch to different positions?

Consider implementing AWL solutions in stages that require minimal support to administer and then incrementally improve to include refinements, analysis, and upgrades when the necessary resources will be available.

Resources – Products versus Staff

Unlike adoption, the options in the resource dimension are usually inverses of each other. If you want to spend less money on products and use native capabilities, then you need the people who know the technology and the operational environment to spend more time developing, testing, deploying, and refining policies. If you do not have the staff, then you need an integrated single AWL solution that is more intuitive for refining and configuring policies, which is usually more expensive.

Also note that there is a dependency between the resource and adoption dimensions. The relationship between native support and resources has been discussed above, but there are other factors to consider. There will probably be more personnel with the desired skills available for common device types, prevalent technologies, or stable environments. There will be fewer personnel with the requisite skills for unique, custom, or highly dynamic devices, applications, and technologies. Lastly, the more different technologies or products in use, the more resources that may be needed to address operations and maintenance (O&M) related complexities.

Deployment – Methodology

Ease of Implementation

Is it easier to change settings for existing technologies and products or is it the same as deploying new ones? How many devices or systems have existing AWL capabilities? How many require new capabilities or technologies to be deployed? Are the required processes in place or is the necessary information available to support the AWL solution?

Consider implementing AWL solutions in stages based on how easy it is to deploy or implement the solution and then incrementally improve to include more advanced options or capabilities.

Effect on Mission

What are the operational requirements for different missions and associated devices and connectivity? What could happen if software is inappropriately blocked for some systems? What is the maximum time window allowed between being inappropriately blocked the first time and having the policies updated to prevent it from happening again? What missions are 24/7 operations and what types of IT support is available during off hours?

Consider implementing AWL methodologies that will have less potential for operational impact and incrementally improve to include more advanced methodologies.

Consider implementing AWL solutions in a manner that keeps them in training and alerting stages for a longer period of time and then incrementally improve by including well tested refinements before blocking.

Consider implementing AWL solutions on devices and systems where the potential for impact is the least and then incrementally improve to include more mission devices and systems.

Effect on Threat
What is the threat your organization is most concerned about? What attacks must you address at a minimum? What attacks do you want to be able to address in the long run?

Consider implementing AWL methodologies that address common and routine threats initially and then incrementally improve to include more advanced methodologies that address more sophisticated threats.

Deployment – Operational Environment

Ease of Adoption
After reviewing all of the previous dimensions/options, determine if for a subset of devices there is an AWL solution that is easy to implement, does not take too many resources, and is likely to be acceptable to the end users.

Consider initially implementing different AWL solutions for different sections of the organization based on what would be easiest for those sections and then incrementally improve to include more devices, more advanced methodologies, or a singular methodology.

Risk Tolerance
What needs to be protected now and against what threats? What AWL methodologies are options? What sections of your organization can deploy them with what resources? What level of potential impact is acceptable for different systems and devices and with what likelihood? What AWL methodologies and stages are options? What sections of your organization can deploy them with what resources?

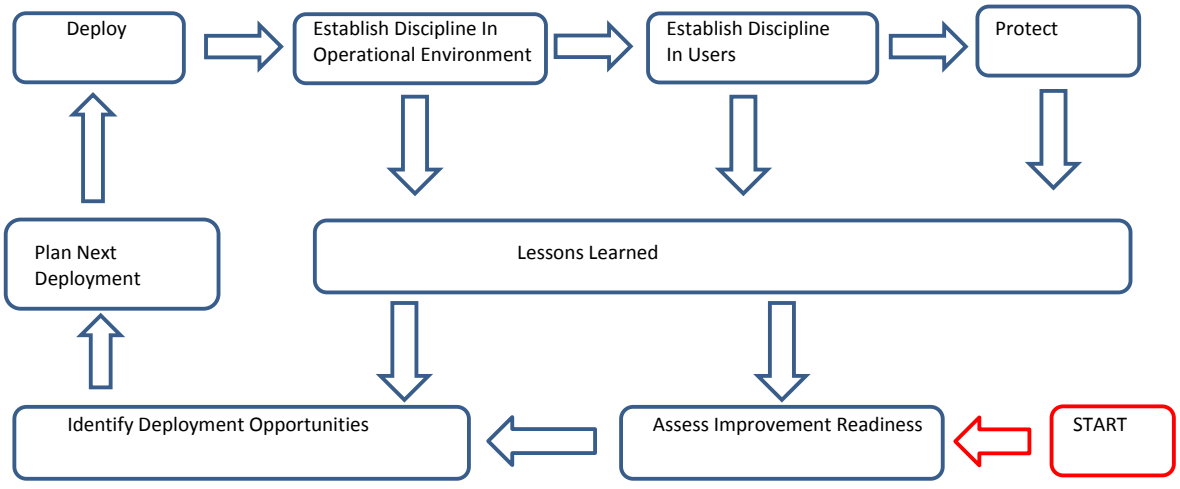
Consider initially implementing different AWL solutions for different sections of the organization based on what you can afford to protect now while mitigating operational impact and threat to acceptable levels. Incrementally improve to include more devices, advanced stages, advanced methodologies, or a singular methodology.

Existing Operations
What activities are already planned to deploy supporting technologies? Make changes to existing technologies? Perform a technology upgrade? Shape a more structured environment? How can you work with those processes to deploy different AWL solutions or refine existing solutions?

Consider implementing AWL solutions as an extension of existing deployment activities and include incremental improvements in the planning, testing, and execution processes of these other activities.

The Incremental Implementation Process

Most organizations cannot just deploy a single complete AWL solution across their entire enterprise and start blocking the execution of software files. There are so many related factors and so much potential for damage if you treat every device/system the same that most do not want to try do this all at once. The idea of implementing different solutions and different stages over different parts of your organization lends itself to a more incremental approach. The following incremental implementation process is representative of how all the different aspects can come together and be used to build an iterative plan that is scoped, understood, and more likely to succeed for each deployment.



Assess Improvement Readiness

Using the characteristics identified from lessons learned and the incremental improvements you are interested in making, identify those devices or organizations that are ready to have an AWL solution deployed or move to the next stage, a more restrictive policy, or a new/enhanced methodology with their existing solution.

Identify Deployment Opportunities

Find out what deployments, upgrades, or changes are scheduled for the near term. Determine if they can be used to support a new deployment, address an issue with a previous deployment, or support an improvement.

Plan Next Deployment

Work out the details for the next increment. This includes:

- Identifying a Deployment Schedule
- Vetting deployment through change management processes
- Any back out plans or necessary mitigation strategies

Deploy

Implement something in the operational environment to log events for initial rule refinement without blocking. This also includes monitoring of the deployment to identify errors or problems that need to be addressed and feeding any additional insight into lessons learned.

Establish Discipline in Operational Environment

Use the Training and Initial Deployment stages of the implementation to correct issues in the environment or the information that was provided to the tools.

Establish Discipline in Users

Use the User-Enabled and IT Support-Enabled Bypass stages of the implementation to adjust user expectations and get them used to working in a more disciplined environment.

Protect

Use the Blocking stage to prevent execution of unauthorized software and monitor ongoing software usage and restrictions.

Lessons Learned

Identify all the things that went right or wrong during the deployment and figuring out why something happened and/or had a particular impact. The point is to determine characteristics that indicate that a set of devices or an organization is or is not ready for an incremental improvement.

Tips for Success

Start Small and Incrementally Improve

It is too hard to do everything all at once, especially for large organizations. If you work incrementally, you can advance different parts of the organization at different times in alignment with resources and risk tolerance.

Piggy Back onto Existing Operations

Most of the success stories are when AWL was implemented as part of a major technology upgrade or recovery from a major breach. If there is already an existing deployment activity that requires many of the same people and processes, then resources are optimized, deployment conflicts are fewer, and the necessary up front work gets done.

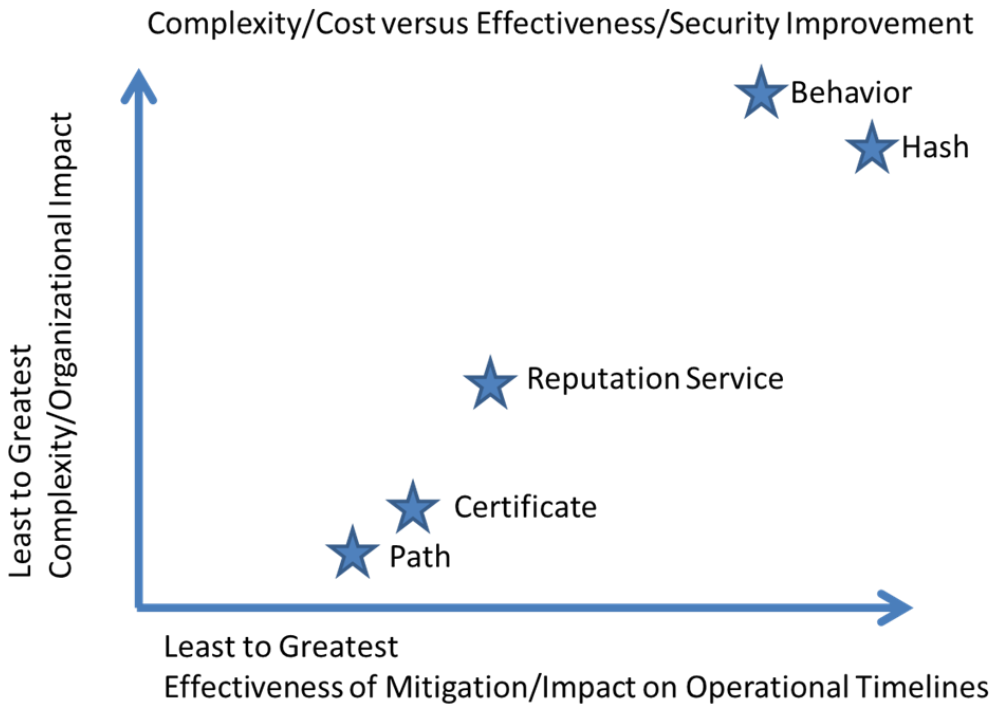
Communicate, Engage, and Demonstrate Management Support

AWL seems to have a large "mythology" surrounding it, and many declare its failure even before implementation begins. Users are worried that they will not be able to do their job and that the IT staff or management isn't interested in hearing their fears or valid complaints. It is important to keep everyone in the loop on an ongoing basis. Don't surprise users with incremental improvements, give them time to adjust. Make sure managers ***show*** that AWL is important, and that there is understanding and support for both the users and the implementers.

Appendix A

Summary

After review of existing threat and technical information related to application whitelisting methodologies, the following chart was created to show the relative relationship between mitigation effectiveness and complexity.



The following table summarizes the relative resource requirements and potential impact for the different AWL methodologies.

Table 1: AWL Methodologies and Their Potential Impact on Resources and Operations

Type	Minimum Resources	Moderate Resources	Maximum Resources	Minimum Operational Impact Potential	Moderate Operational Impact Potential	Maximum Operational Impact Potential
Path	X				X	
Certificate	X[1]			X		
Reputation Service		X		X		
Behavior		To Train	To Refine		On Refined Rules	On Trained Rules
Hash		To Train	To Refine		On Refined Rules	On Trained Rules

Resource Definitions

Minimum: Default tool settings after training on the operational environment. Main resources needed are those that will configure and operate the tool in the environment. Minimal troubleshooting that can be handled by these same individuals after training with vendor. Main cost is for the tool and implementation.

Moderate: Some custom configuration and rule settings after training on the operational environment. Resources needed are those that will configure and operate the tool and also those

[1] The resources required to implement and manage a default certificate-based AWL is minimal. If you intend to apply it to all software installed in the organization, then you need to invest resources is standing up a certificate infrastructure and signing all software that is not part of the default capability. Signing more software increases the effectiveness of the solution, but also increases the resources needed and the potential operational impact.

that know the current operational environment to assist during the training activity. Most troubleshooting can be handled by the individuals tasked to run the tool after training with the vendor, but occasionally they will need assistance from operations personnel more familiar with the deployed environment and associated missions. These operations personnel are usually individuals with advanced understanding, making them important resources for the organization that you cannot dedicate to this activity. Cost includes tools and implementation as well as some required service that contains the content for the tool to use to make decisions.

Maximum: Specialized configuration and rules settings based on specific operational requirements associated with the operational environment. Resources needed are those that will configure and operate the tool; those that know the current operational environment to assist during the training activity; and those that know the technical details about the software products allowed in the environment and how they must be installed for successful operation for the refinement process. Troubleshooting during the alerting only phase will be handled by individuals tasked to run the tools after training with the vendor with support from both operations personnel more familiar with the deployed environment and those with detailed technical knowledge. As the deployment advances through the stages, the skills needed move away from unique and advanced knowledge of the environment and more to help-desk and tool specific knowledge. Some of the individuals required to refine the training and troubleshoot in the alerting phase are probably in high demand and short supply in the environment.

Operational Impact Definitions

Minimum: The false positive rate for the tool is minimal. Most software installations and actions that are not malware related are acceptable and not blocked.

Moderate: The false positive rate for the tool is low once the training and alerting phases have been completed. A majority of the software installations and actions that are not malware related are represented in rules that make them acceptable and not blocked.

Maximum: The false positive rate for the tool is low to moderate after the training and alerting phases, but can be brought down to low during the enabled bypass phases. A majority of the software installations and actions that are not malware related are represented in rules that make them acceptable and not blocked, but new installations and operational needs will require the ongoing development (and potentially phasing) of new rules.