





# **Implementing AppLocker Whitelisting in an Enterprise**

*- Sami Laiho*

WHOAMI /ALL

# Sami Laiho

Senior Technical Fellow

adminize.com

- IT Admin since 1996
- MCT since 2001 (MCT Regional Lead – Finland)
- MVP in Windows OS since 2011
- Specializes in and trains:
  - Troubleshooting
  - Security
  - Centralized Management
  - Active Directory
  - Hacking
  - Penetration testing
  - Social Engineering
- Trophies:
  - Best External Speaker (non-Microsoft) at Ignite 2017
  - NIC 2016, 2017 - Best Speaker
  - Best Sessions (#1 and #2) at TechTalks 2017, Helsinki
  - Best Session at AppManagEvent 2017, Utrecht
  - TechDays Sweden 2016 – Best Speaker
  - Ignite 2015 – Best male presenter ;) (#2 out of 1000 speakers)
  - TechEd Europe and North America 2014 - Best session, Best speaker
  - TechEd Australia 2013 - Best session, Best speaker





I got Certs





1,2 kg of  
them

TODISTUS



**Certificate of  
Excellence  
2009**

**Microsoft**  
CERTIFIED  
Trainer

**SAMI LAIHO**

Has successfully completed the requirements to be recognized  
as a Microsoft Certified Trainer

Certified since 2000

Signed by

[Signature]

Chief Executive Officer, Microsoft Corporation

SALTER

1164

Max 5000g x 1g/Max 11lb x 0.1oz

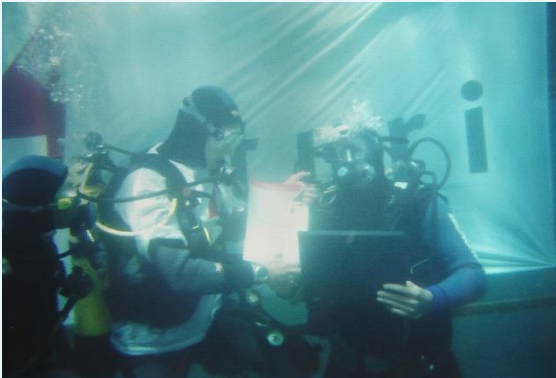
ON-OFF

ML-FL.OZ

ZERO

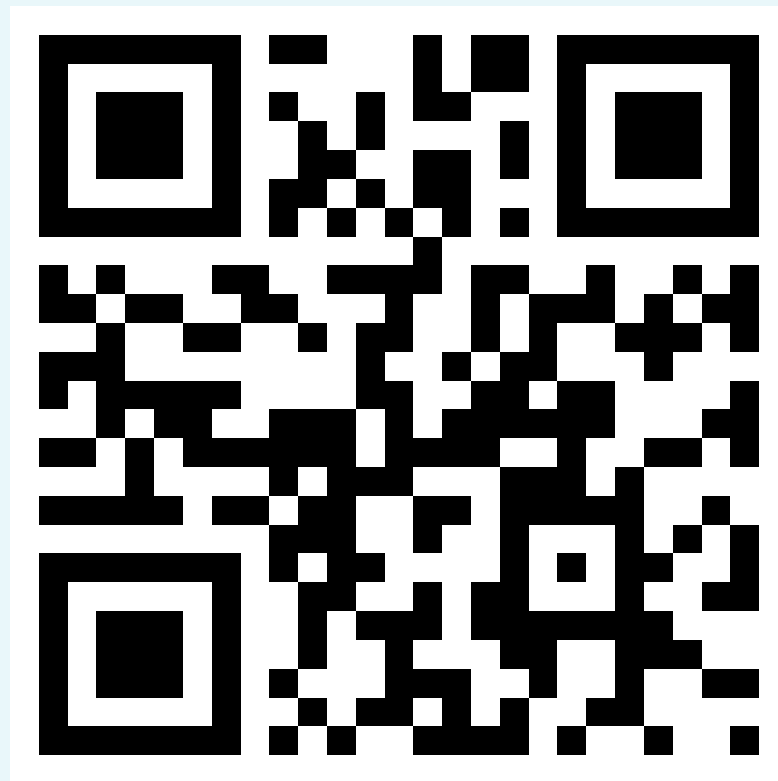
G-OZ

# Windows XP Deep Dive in 2001



# Contact

- [sami@adminize.com](mailto:sami@adminize.com)
- Twitter: @samilaiho
- Blog: <http://blog.win-fu.com/>
- Free newsletter:  
<http://eepurl.com/F-GOj>
- My trainings:
  - <https://win-fu.com/ilt/>
  - <https://win-fu.com/dojo/>



**NIC**



<https://www.finnishcrazygames.com/>

- <https://www.finnishcrazygames.com/>

The image features three bright green, rectangular pillows stacked on a dark, textured background. The pillows are arranged in a slightly overlapping manner, with the topmost pillow being the most prominent. The lighting is soft, highlighting the texture of the pillows and the background. The overall composition is simple and clean, focusing on the vibrant color of the pillows.

# Hyppyynytydytys

'Bouncy cushion  
satisfaction'

IE



Why are we here?

Global ransomware damage costs predicted to exceed \$5 billion in 2017, up from \$325 million in 2015.

Ransomware damages up 15X in 2 years, expected to worsen; Ransomware attacks on healthcare organizations will quadruple by 2020.

– [Steve Morgan](#), Editor-In-Chief

Menlo Park, Calif. – May 18, 2017

[Ransomware](#) — a malware that infects computers and restricts their access to files, often threatening permanent data destruction unless a ransom is paid — has reached epidemic proportions globally.



In 2015, there were numerous media and researchers stating that [ransomware was not all it's cracked up to be](#). Ransomware accounted for roughly [\\$325 million in damages in 2015](#), according to Microsoft.

After a surge of attacks the following year, [Cybersecurity Ventures](#) predicted that ransomware damages and related costs would reach [\\$1 billion annually in 2016](#).

According to the Cisco 2017 Annual Cybersecurity Report, [ransomware is growing at a yearly rate of 350%](#).



## Gartner says so

- “The endpoint is the centre of the malware universe. The endpoint is the only constant for malware. The endpoint is where all attack vectors rejoin to execute. However, with the complexity of modern malware, no organization should rely on a single layer of malware defense.” – Mario de Boer
- Gartner recommends Whitelisting as the most important security feature for 2018

# NATO says so

- NATO requires that systems relating to their's or controlled by them need to use AppLocker (if it's Windows)



# Basics

# Basic Protections

- Firewall Everything!
- BitLocker Everything!
- Remove Admin rights!
- Use Anti-Malware, just not as your main protection!





# Implementing AppLocker

# Basic Rule of Application Design

- In a location that houses executable code there should be no write access to users. In a user writable location there should be no executable code.

# Common Rules

- Whitelisting is a security barrier – BlackListing is not!
- Effective Whitelisting works only when combined with the Principle of Least Privilege

# Whitelisting-project

My way



# Normal Project Run (AppLocker)

- Implement log forwarding
- Implement AppLocker in Audit mode
  - Choose a pilot group
- Collect logs for at least a month
- Create rules based on logs
- Create an escape plan
- Educate Admins
- Turn on AppLocker Auditing for the rest of the environment
- Turn on AppLocker Enforcement for the pilot group
  - You can start with EXE and MSI, and then later turn on DLL and Scripts if you want to balance
- Monitor for sometime
- Turn on everywhere!
- Harden!

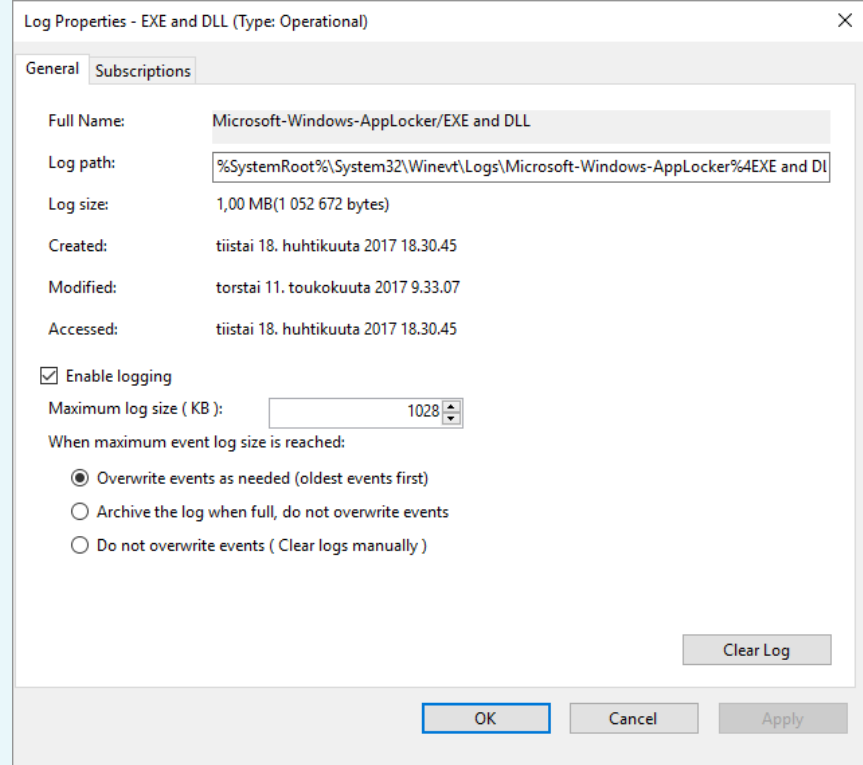
# Log Forwarding

# Log Forwarding

- Available since Windows Vista
- Requires a server or workstations to collect logs
- Gives one place to see all audit or error entries from your whitelisting on all of your computers
- Super important during Auditing phase

# AppLocker

- Log size should be increased!
  - At least 30MB





# DEMO

Enabling Log Forwarding



# AppLocker

SRPv2

# AppLocker

- Blacklisting and Whitelisting
- Can target computers, users or groups
- All software needs to be preapproved in some way
  - Location, hash or signature based
- Is based on a native function of the Windows OS since Windows 7
- Requires Enterprise version of Windows
- Requires the AppIDSvc

# DEMO

Enabling AppLocker for Auditing

# DEMO

Looking at the AppLocker logs

# File/Folder Rules

- You can allow a Folder as `c:\folder\*`
- You can allow a certain file like `c:\folder\file.exe`
  - You can also use wildcard `*` like `c:\users\*\appdata\local\Software1\*`
- AppLocker doesn't support Windows variables
  - Sysvol or NETLOGON require all DC's to be added separately
    - \\dc1\SYSVOL\\*
    - \\dc2\SYSVOL\\*
    - \\dc3\SYSVOL\\*
- UNC-paths might need to be added in three different formats
  - \\Server1\Share\\*
  - \\server1.domain.local\Share\\*
  - \\172.16.0.21\Share\\*

# Publisher-rules

- Best option after Path-rules
- Try to stick to Company-level instead of certain filenames or versions
- \*-rule says that any file signed by a trusted signer is OK to run
  - Trust your own certificate or buy an externally trusted certificate



# Hash-rules

- Don't use unless you can't use Path-rules or Publisher-rules
- Usable exception if the binary doesn't change often

# DEMO

Different Kind of AppLocker Rules

# AppLocker HOW TO





- Keep to containers not items – Folders vs Files, Publishers vs Hashes
- Remember to audit your installation with AccessChk!
- Remember NO ADMIN RIGHTS!!

# DEMO

Enforcing AppLocker

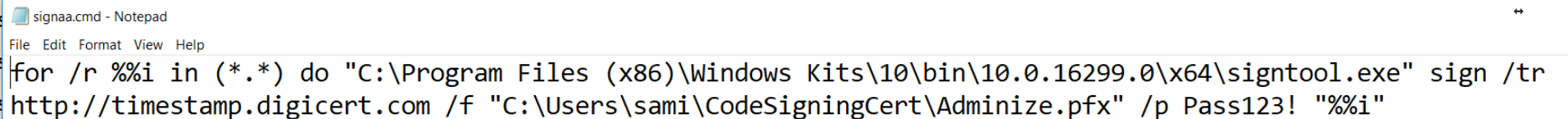
# Simplest AppLocker

- Relies on the knowledge of the user

Action	User	Name	Condition	Exceptions
 Allow	Everyone	Signed by *	Publisher	
 Allow	Everyone	All files located in the Program Files folder	Path	Yes
 Allow	Everyone	All files located in the Windows folder	Path	Yes
 Allow	BUILTIN\Ad...	(Default Rule) All files	Path	

# Signing

- 95% of Malware is not signed – just something to think about
- You can sign apps yourself
  - Use Timestamp if possible!
- If you have the cert on your computer installed:
  - **Signtool sign /v /s MY /n MyPrivateCert**  
**/t <http://timestamp.verisign.com/scripts/timestamp.dll> FileToSign.exe**
- If not:



```
signaa.cmd - Notepad
File Edit Format View Help
for /r %%i in (*.*) do "C:\Program Files (x86)\Windows Kits\10\bin\10.0.16299.0\x64\signtool.exe" sign /tr
http://timestamp.digicert.com /f "C:\Users\sami\CodeSigningCert\Adminize.pfx" /p Pass123! "%%i"
```

- Guide: <https://blogs.msdn.microsoft.com/winsdk/2009/11/13/steps-to-sign-a-file-using-signtool-exe/>



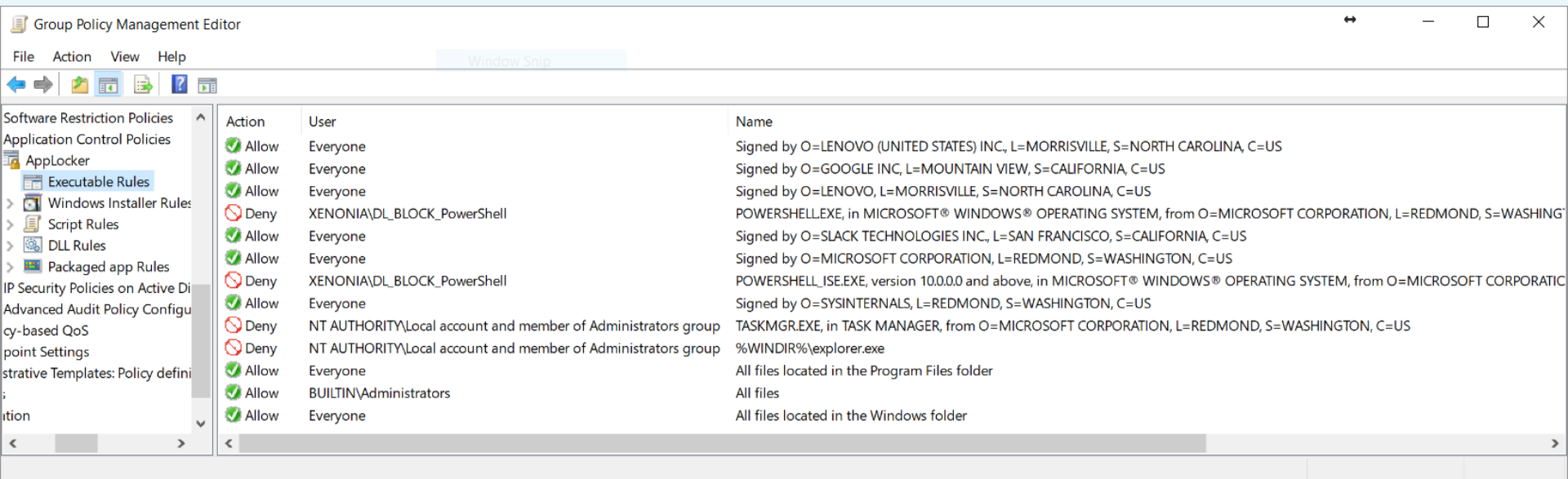
# DEMO

Signing Apps



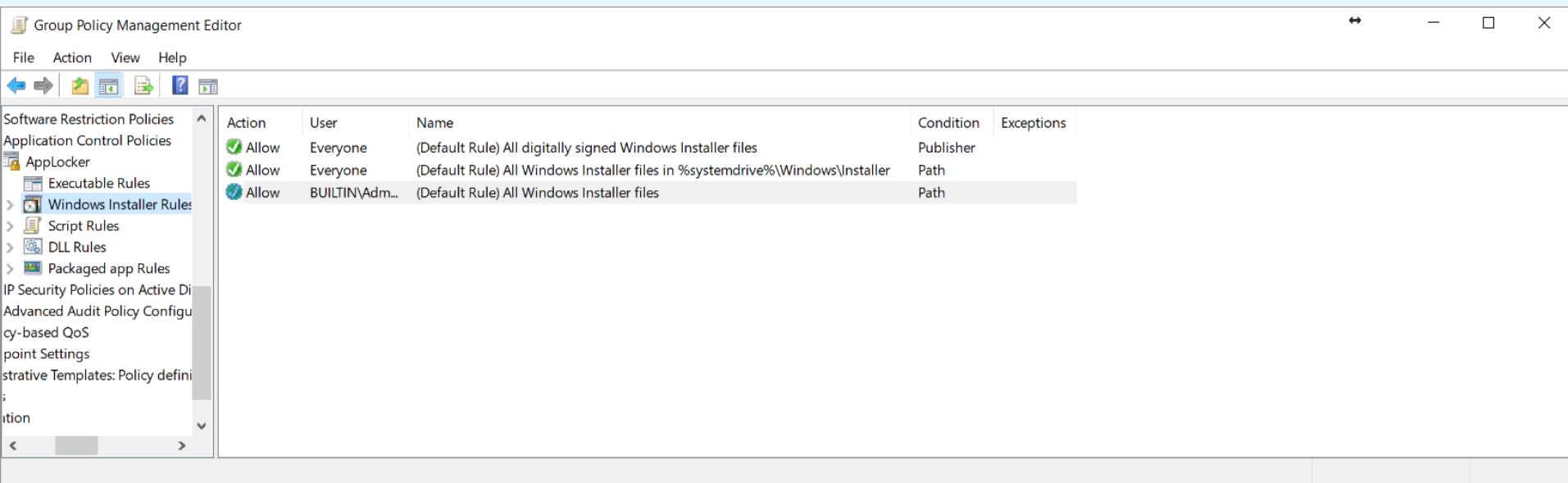
# AppLocker example

- My own → January 2017



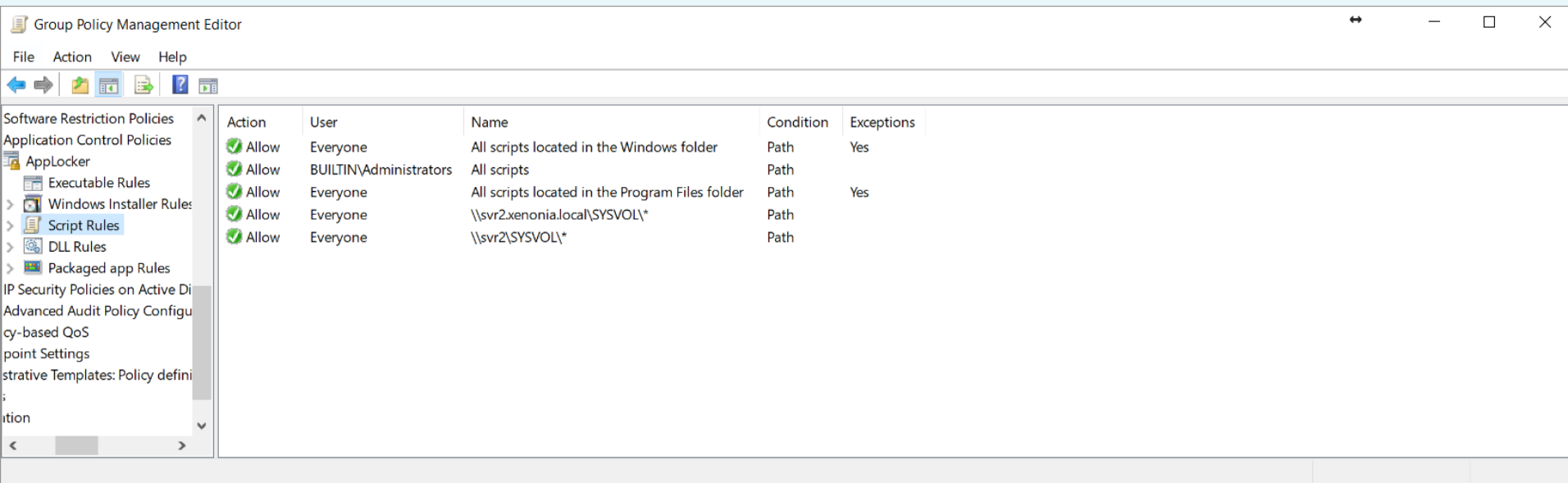
# AppLocker example

- My own



# AppLocker example

- My own



The screenshot shows the Group Policy Management Editor window. The left pane displays the tree structure with 'AppLocker' expanded under 'Software Restriction Policies'. The right pane shows a list of five rules, all set to 'Allow'.

Action	User	Name	Condition	Exceptions
✓ Allow	Everyone	All scripts located in the Windows folder	Path	Yes
✓ Allow	BUILTIN\Administrators	All scripts	Path	
✓ Allow	Everyone	All scripts located in the Program Files folder	Path	Yes
✓ Allow	Everyone	\\svr2.xenonia.local\SYSVOL\*	Path	
✓ Allow	Everyone	\\svr2\SYSVOL\*	Path	

## Weird PS1 and PSM1 scripts – thousands of them?

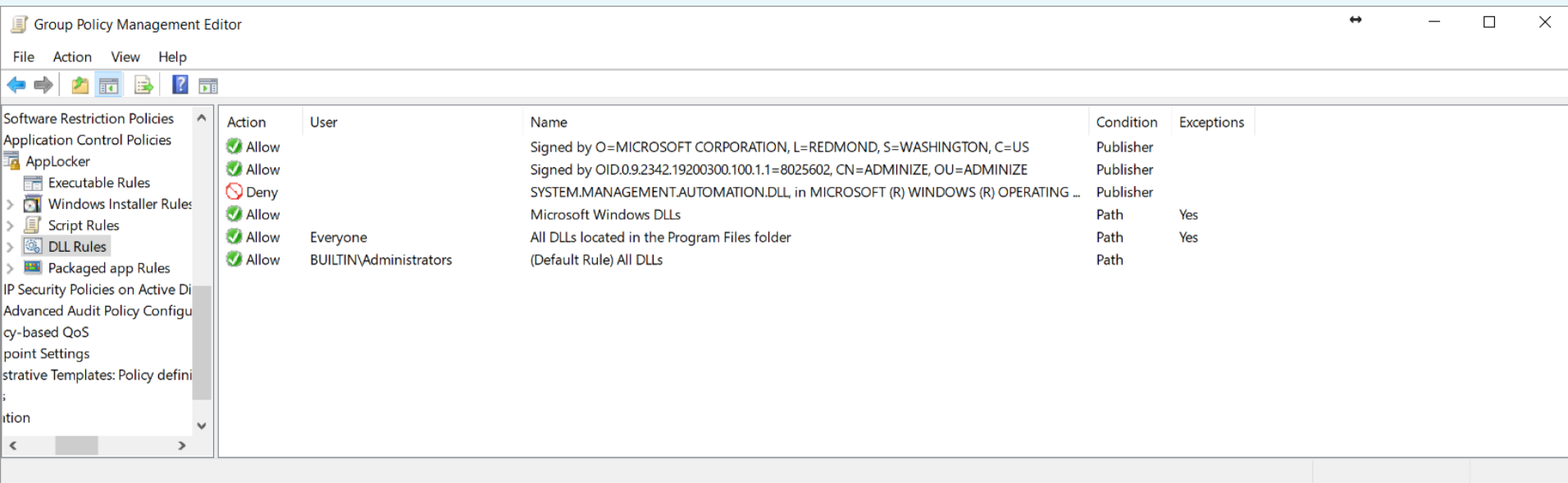
- PowerShell 5 has this weird feature that when ever it runs a script it will create two random named PS-scripts in users TEMP folder → This causes a lot of alerts in logs (I MEAN A LOT!!)
  - Microsoft says: "This is just how it works"
  - They really haven't thought about AppLocker at all 😞

## Weird PS1 and PSM1 scripts – thousands of them?

- Luckily the script Hash is universal (WORLDWIDE) 😊
  - You can allow these with the hash-value below:
  - 6B86B273FF34FCE19D6B804EFF5A3F5747ADA4EAA22F1D49C01E52DDB7875B4B
- Don't know how to put a Hash in a rule without the file?
  - Read further and use PowerShell
  - Create a script with the contents of “1” without the quotes ;)

# AppLocker example

- My own

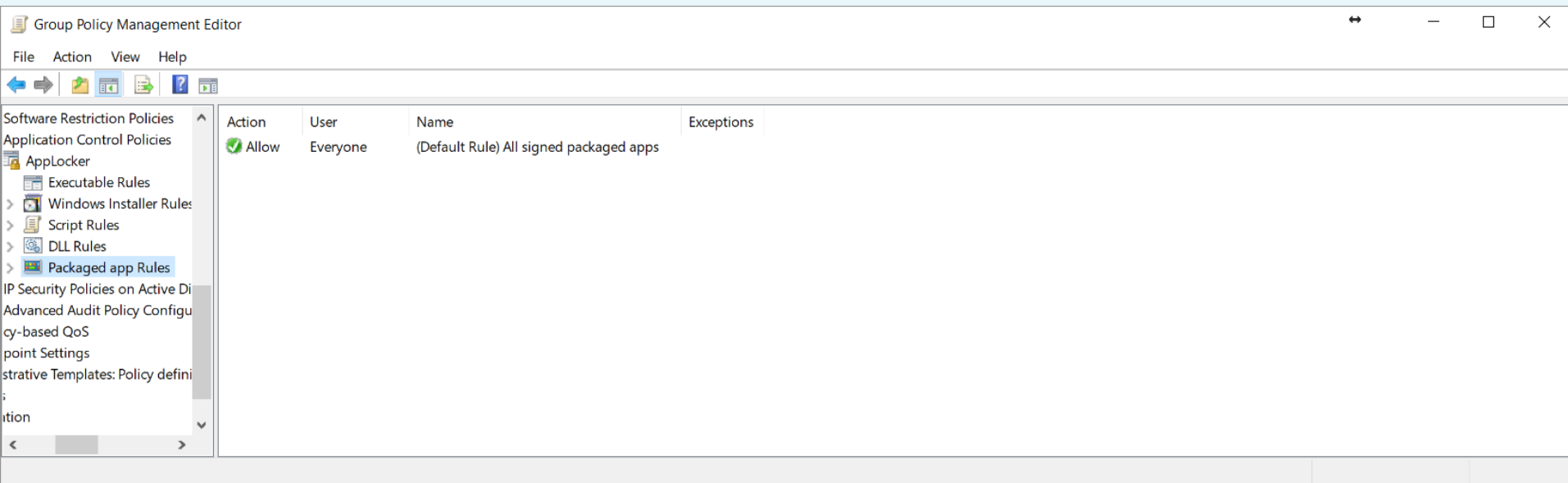


The screenshot shows the Group Policy Management Editor window. The left pane displays the hierarchy: Software Restriction Policies > Application Control Policies > AppLocker > DLL Rules. The right pane shows a table of rules.

Action	User	Name	Condition	Exceptions
Allow		Signed by O=MICROSOFT CORPORATION, L=REDMOND, S=WASHINGTON, C=US	Publisher	
Allow		Signed by OID.0.9.2342.19200300.100.1.1=8025602, CN=ADMINIZE, OU=ADMINIZE	Publisher	
Deny		SYSTEM.MANAGEMENT.AUTOMATION.DLL, in MICROSOFT (R) WINDOWS (R) OPERATING ...	Publisher	
Allow		Microsoft Windows DLLs	Path	Yes
Allow	Everyone	All DLLs located in the Program Files folder	Path	Yes
Allow	BUILTIN\Administrators	(Default Rule) All DLLs	Path	

# AppLocker example

- My own





# Disclaimer

- I don't use PowerShell in logon scripts!
- I have to use a Deny rule as I allow all signed executables → For customers that Deny rule would not exist as would not the “Signed \*” either
- What if let's say “OneDrive.exe” needs to run a powershell script with limited user rights?
  - I deal with this with Avecto DefendPoint
  - You could just remove the users from the DL\_BLOCK\_PowerShell group temporarily

# DEMO

Current AppLocker on my own

# My customer devices

- Basic rules + AccessChk revealed exceptions
- Use certificates if you can (and trust the company)
- **Then add required network locations with**
  - UNC
  - IP
  - FQDN
  - Sometimes also with the drive letter: P:, \\SVR\Share, \\SVR.dom.com\share, and \\192.1.1.2\Share
- Then add local applications outside of the default folders with Certs, Folders (if they can be blocked from writing to by limited users)
- Problematic ones
  - Self-updating, not signed and stored in users profile
  - TIP! File/Folder rules allow \* at any point!
    - Use with caution – but usually need some! Try to use HASH rather if possible!

# TIPS

- Don't touch the
  - `__PSScriptPolicyTest_bavjba32.xjg.ps1`
    - *This tests weather the SRP/DeviceGuard is on so it knows that PS is running in Constrained Mode*
  - `SDIAG*.ps1`
    - These are diagnostics scripts for Windows
- If you are running SCCM software distribution then you might need to allow MSI from(this depends on who installs USER/SYSTEM):
  - `C:\Windows\CCMCache`
  - `ProgramData`

# PowerShell for AppLocker in production

```
PS C:\Windows\system32> Get-AppLockerFileInformation -EventLog -Logpath "Microsoft-Windows-AppLocker/EXE and DLL" -Event
Type Denied -Statistics

FilePath      : %OSDRIVE%\USERS\SAMI\APPDATA\LOCAL\TEMP\NSR473A.TMP\SYSTEM.DLL
FilePublisher :
FileHash      : SHA256 0xC74313FE9AE95005F41CF9CF11E8BE8B87CF79D01354156F906B47CD9CB85E92
PolicyDecision : Denied
Counter       : 37

PS C:\Windows\system32>
```

# Getting the Effective AppLocker policy (XML)

- `Get-AppLockerPolicy -Effective -Xml`

# Testing

The following shows example input for **Test-AppLockerPolicy**:

```
PS C:\ Get-AppLockerPolicy -Effective -XML > C:\Effective.xml
```

```
PS C:\ Get-ChildItem 'C:\Program Files\Microsoft Office\' -filter *.exe -Recurse | Convert-Path | Test-AppLockerPolicy  
-XMLPolicy C:\Effective.xml -User contoso\zwie -Filter Denied,DeniedByDefault | Export-CSV C:\BlockedFiles.csv
```

In the example, the effective AppLocker policy is exported to the file C:\Effective.xml. The **Get-ChildItem** cmdlet is used to recursively gather path names for the .exe files in C:\Program Files\Microsoft Office\. The XMLPolicy parameter specifies that the C:\Effective.xml file is an XML AppLocker policy file. By specifying the User parameter, you can test the rules for specific users, and the **Export-CSV** cmdlet allows the results to be exported to a comma-separated file. In the example, **-Filter Denied,DeniedByDefault** displays only those files that will be blocked for the user under the policy.

## Merge Local policy with one in a GPO

- `Get-AppLockerPolicy -Local | Set-AppLockerPolicy -LDAP "LDAP://DC13.Contoso.com/CN={31B2F340-016D-11D2-945F-00C044FB984F9},CN=Policies,CN=System,DC=Contoso,DC=com" -Merge`



# PowerShell

- `Get-ChildItem C:\Windows\System32\*.exe | Get-AppLockerFileInformation | New-AppLockerPolicy -RuleType Publisher, Hash -User Everyone -RuleNamePrefix System32`
  - Creates an AppLocker policy containing allow rules for all of the executable files in C:\Windows\System32. The policy contains publisher rules for those files with publisher information and hash rules for those that do not. The rules are prefixed with "System32:" and the rules apply to the Everyone group.
- `Get-ChildItem C:\Windows\System32\*.exe | Get-AppLockerFileInformation | New-AppLockerPolicy -RuleType Path -User Everyone -Optimize -XML`
  - Creates an XML-formatted AppLocker policy for all of the executable files in C:\Windows\System32. The policy contains only path rules, the rules are applied to the Everyone group, and the Optimize parameter indicates that similar rules are grouped together where possible.

# No need for the example file!

- The most important example!!
- `Get-AppLockerFileInformation -EventLog -LogPath "Microsoft-Windows-AppLocker/EXE and DLL" -EventType Audited | New-AppLockerPolicy -RuleType Publisher,Hash -User domain\FinanceGroup -IgnoreMissingFileInformation | Set-AppLockerPolicy -LDAP "LDAP://DC13.TailspinToys.com/CN={31B2F340-016D-11D2-945F-00C04FB984F9},CN=Policies,CN=System,DC=WingTipToys,DC=com"`

## Same after auditing!

- `Get-AppLockerFileInformation -EventLog -Logpath "Microsoft-Windows-AppLocker/EXE and DLL" -EventType Denied | New-AppLockerPolicy -RuleType Publisher,Hash -User Everyone -IgnoreMissingFileInformation | Set-AppLockerPolicy -LDAP "LDAP://SVR2.xenonia.local/CN={04EEA143-7133-428C-B56B-E07BDC368E59},CN=Policies,CN=System,DC=XENONIA,DC=local"`

# Create a Publisher rule from the log

- This is the easiest/fastest way!
  - Take a publisher info from the log entry, like:
    - O=TEAMVIEWER, L=GOEPPINGEN, S=BADEN WUERTTEMBERG, C=DE
  - Create a Publisher rule and use ANY file as a reference → Change to Custom and change to your own publisher, \*, \* and 0.0.0.0

Reference file:

C:\Program Files\Avecto\Privilege Guard Client\PG\ Browse...

Any publisher

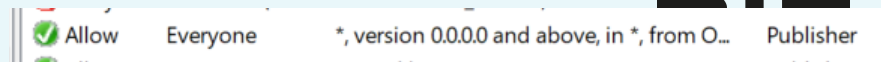
Publisher: L=GOEPPINGEN, S=BADEN WUERTTEMBERG, C=DE

Product name: \*

File name: \*

File version: 0.0.0.0 And above

☒ Use custom values



## Locally EFFECTIVE policy MERGED with AD one

- `Get-AppLockerPolicy -Effective | Set-AppLockerPolicy -ldap "LDAP://SVR2.xenonia.local/CN={04EEA143-7133-428C-B56B-E07BDC368E59},CN=Policies,CN=System,DC=XENONIA,DC=local" -Merge`

# Statistics

- `Get-AppLockerFileInformation –EventLog –Logpath "Microsoft-Windows-AppLocker/EXE and DLL" -EventType Denied –Statistics`
- On the collector server these are very handy!
  - `Get-AppLockerFileInformation –EventLog –Logpath "ForwardedEvents" -EventType Audited –Statistics | Out-GridView`
    - No need for Excel ;)
  - `Get-AppLockerFileInformation –EventLog –Logpath "ForwardedEvents" -EventType Audited –Statistics | Export-CSV C:\Temp\AppLocker.csv`
    - Import in Excel!

# My Own

- Get all the Denied entries from the log to an AppLocker policy
  - `Get-AppLockerFileInformation -EventLog -Logpath "ForwardedEvents" -EventType Denied | New-AppLockerPolicy -RuleType Publisher,Hash -User Everyone -IgnoreMissingFileInformation | Set-AppLockerPolicy -Merge`
- Filtering to one DLL, getting the HASH from it and merging it to the local AppLocker policy
  - `Get-AppLockerFileInformation -EventLog -Logpath "Microsoft-Windows-AppLocker/EXE and DLL" -EventType Denied | Where-Object "Path" -like *SQLITE* | New-AppLockerPolicy -RuleType Hash -User Everyone -IgnoreMissingFileInformation | Set-AppLockerPolicy -Merge`

# DEMO

PowerShellify AppLocker



# Manually Merge (XML)

- **To merge two or more AppLocker policies**
  - Open an XML policy file in a text editor or XML editor, such as Notepad.
  - Select the rule collection where you want to copy rules from.
  - Select the rules that you want to add to another policy file, and then copy the text.
  - Open the policy where you want to add the copied rules.
  - Select and expand the rule collection where you want to add the rules.
  - At the bottom of the rule list for the collection, after the closing element, paste the rules that you copied from the first policy file. Verify that the opening and closing elements are intact, and then save the policy.
  - Upload the policy to a reference computer to ensure that it is functioning properly within the GPO.
- *From <<https://docs.microsoft.com/en-us/windows/device-security/applocker/merge-applocker-policies-manually>>*

# Troubleshooting

- If you can't logon etc.
- Remove network cable, boot with another media and destroy  
C:\Windows\AppLocker\ folder

# Recommendations for HelpDesk

1. Find the ForwardedEvents or AppLocker log to find the denied binary
2. Identify if it has a signature
3. If it has and you can trust the company, please build a Publisher Rule to trust the company
4. If not, then if the file is not expected to change create a File Hash rule
5. If nothing else works, build a Path-rule like:
  1. C:\Users\\*\AppData\Local\Slack\\*

# Hardening

# Hardening Whitelisting

Make sure your containers don't leak

- `ACCESSCHK -s -w Users C:\Program Files\`
- `ACCESSCHK -s -w Users C:\Program Files (x86)\`
- `ACCESSCHK -s -w Users C:\Windows\`

# DEMO

ACCESSCHK.exe

**NIC**

# Firewall needed as well

- Firewall needed to Block
  - At least
    - Rundll32 & regsvr32 (32 and 64bit)
  - Maybe also
    - PowerShell
    - Scripting host
- If you only have one subnet you can block just INTERNET
  - Otherwise you need to block outbound connections by default, and only allow correct destinations
    - Azure etc.

# Firewall

Windows Firewall with Advanced Security

File Action View Help



Windows Firewall with Advanced Security

- Inbound Rules
- Outbound Rules
- Connection Security Rules
- Monitoring

## Outbound Rules

Name	Group	Profile	Enabled	Action	Override	Program	Local Address	Remote Address	Protocol	Local Port	Remote Port	Authorized Computers
WScript 64bit		All	Yes	Block	No	c:\windo...	Any	Any	Any	Any	Any	Any
WScript 32bit		All	Yes	Block	No	c:\windo...	Any	Any	Any	Any	Any	Any
RUNDLL32.exe 64-bit		All	Yes	Block	No	%System...	Any	Any	Any	Any	Any	Any
RUNDLL32.exe 32bit		All	Yes	Block	No	%System...	Any	Any	Any	Any	Any	Any
Regsvr32 64-bit		All	Yes	Block	No	c:\windo...	Any	Any	Any	Any	Any	Any
Regsvr32 32-bit		All	Yes	Block	No	%System...	Any	Any	Any	Any	Any	Any
PS ISE 64-bit		All	Yes	Block	No	%System...	Any	Internet	Any	Any	Any	Any
PS ISE 32-bit		All	Yes	Block	No	%System...	Any	Internet	Any	Any	Any	Any
PowerShell 64-bit		All	Yes	Block	No	%System...	Any	Internet	Any	Any	Any	Any
PowerShell 32-Bit		All	Yes	Block	No	%System...	Any	Internet	Any	Any	Any	Any
CScript 64bit		All	Yes	Block	No	%System...	Any	Any	Any	Any	Any	Any
CScript 32bit		All	Yes	Block	No	c:\windo...	Any	Any	Any	Any	Any	Any

NIC



Well... I have PRO only??

# Training, Training, Training

Training employees how to recognize and defend against cyber attacks is the most under spent sector of the cybersecurity industry — and yet it holds out the greatest hope for combating ransomware attacks.

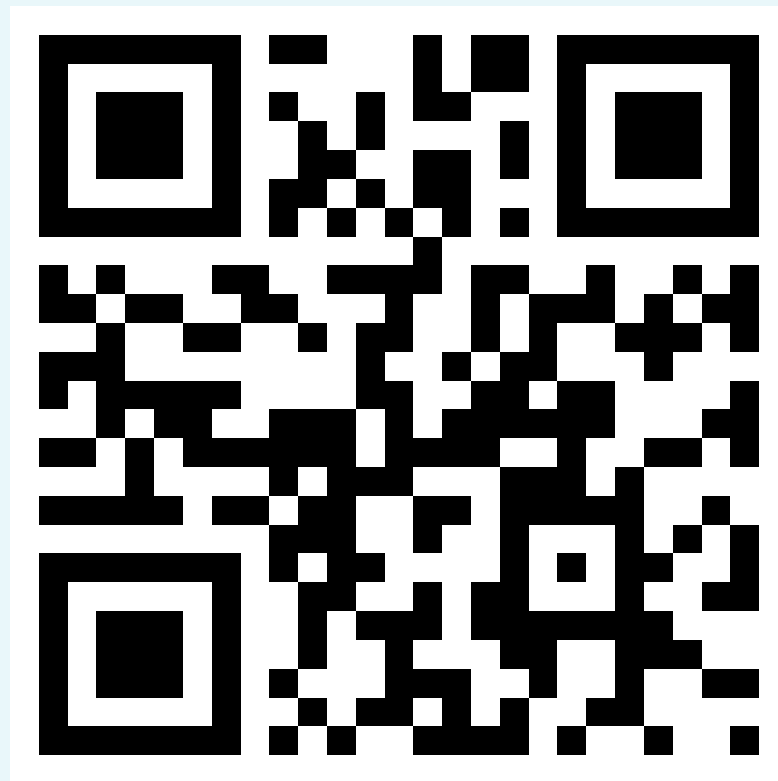
91 percent of attacks by sophisticated cybercriminals start through email, according to Mimecast, a leading email security firm. Spear phishing attacks on employees are commonly used to infect organizations with ransomware.

“Training employees on security will immediately bolster the cyber defenses at most companies,” says Lawrence Pingree, a research director at Gartner, because most data breaches are based on “exploiting common user knowledge gaps to social engineer them to install malware or give away their credentials.”

Phishing identification training definitely bolstered Wells Fargo’s cyber defenses, notes Chief Information Security Officer Rich Baich. Through the use of various security awareness techniques, he says, workforce susceptibility to phishing declined by more than 40 percent.

# Contact

- [sami@adminize.com](mailto:sami@adminize.com)
- Twitter: @samilaiho
- Blog: <http://blog.win-fu.com/>
- Free newsletter:  
<http://eepurl.com/F-GOj>
- My trainings:
  - <https://win-fu.com/ilt/>
  - <https://win-fu.com/dojo/>



**NIC**