WinHarden

Program to find misconfigurations on Windows Systems

Description

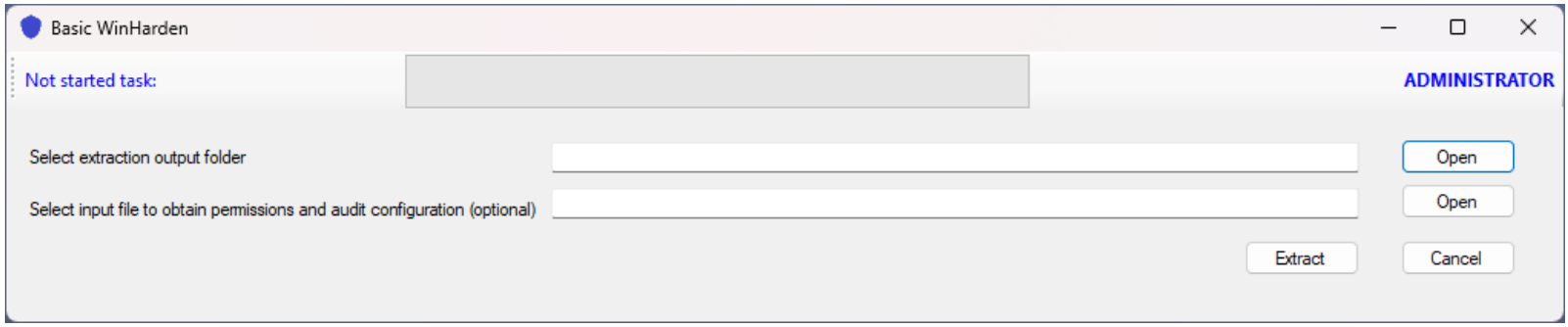
- WinHarden is a C# project that includes programs and libraries to extract the most relevant security configuration files and to analyze them to identify exploitable misconfigurations, including ways to escalate privileges through binary planting or similar attacks.
- WinHarden is mainly composed by:
  - Module to extract Windows security configuration.
  - Module to analyse files extracted by above module.

Key features

- It does not require administrator privileges to perform most of the analysis.
- Exhaustive fine-tuning to remove false positives regarding to exploitable privilege escalation misconfigurations.
- Usage of Windows API to prevent being dependant of the regional configuration and languages of the analysed Windows host.
- There are non-dependencies with third parties software to prevent potential security risks related to consume external packages. All functionalities come from C# source code and Windows API.
- Files output are human readable format to ease manual review for IT experts.

1 Windows security configuration extraction module

- Objective of this module is gathering all information of a Windows system that could be useful and interesting for any security review.
- This module generates output files to be reviewed manually or to be processed in next analysis module with advanced filtering capabilities to detect exploitable misconfigurations.
- Currently it generates more than 75 output files. It is on continue improvement, so new output files are being added. Examples of relevant generated files for security reviews:
  - Access control on paths/registry keys related to binary planting attacks to allow escalating privileges, as startup folders, startup programs, running processes, scheduled tasks, services.
  - Access control on net shared folders.
  - System audit policy.
  - Relevant register keys.
  - System privileges for local users.
  - Password policy.
  - Active directory information for local administrator accounts.
  - Access control and audit configuration in selected list of folders by input file.
- This module may be run from ExtractWinHardenApp program or Extract button from WinHardenApp program.

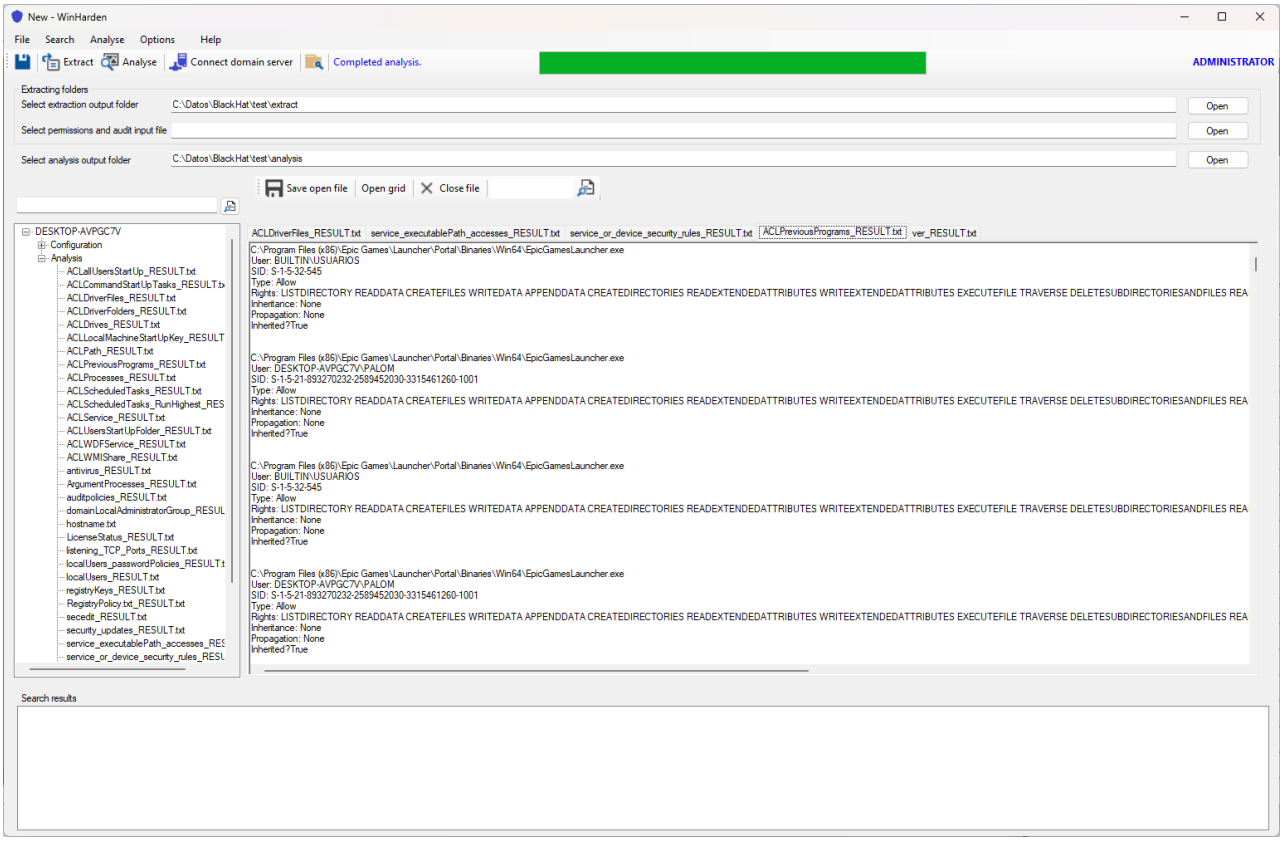


Simple GUI of ExtractWinHardenApp to extract security information of a Windows system

2 Windows security configuration analysis module

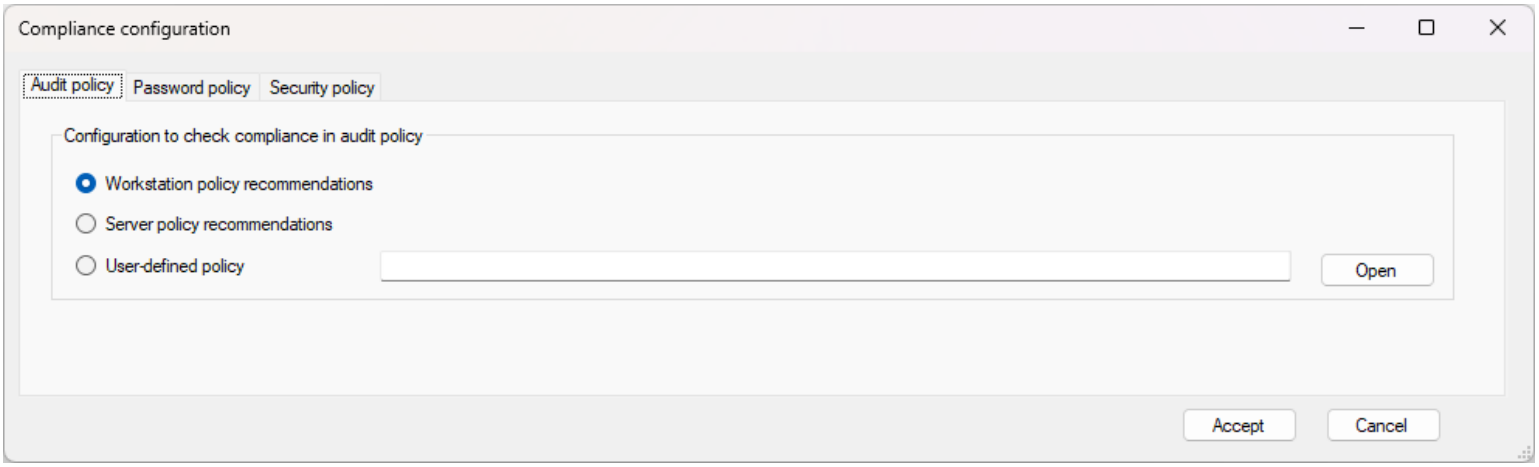
- Objective of this module is filtering relevant information to identify any possible misconfiguration that deteriorates the security of the Windows host or that could be exploitable to escalate privileges.
- Currently it generates more than 30 output files. It is on continue improvement, so new output files are being added. Examples of relevant filtered result files for security reviews:
  - Vulnerable folders, tasks, programs, services and registry keys for binary planting and other techniques to escalate privileges. This feature only filters system and administration accounts. Instead of only filtering by Everyone or similar groups as other public tools do. So, vulnerable objects for domain groups are identified by this tool.
  - Unsecure registry key values.
  - Process arguments to try to find passwords as parameters.
  - Antivirus status.
  - Windows version and latest hotfix to allow assessing the updating status.

- This module is invoked from Analyze button from WinHardenApp program.



3 Additional features

- The analysis module may include compliance checks versus audit, password and security policies.
- It includes string search functionalities to ease manual reviews.



4 Public successful use cases

- As an example, this tool was used to identify an unquoted service path on Visual Studio, generating CVE-2023-36758 with 7.8 base score from CVSS:3.1, high severity. The developer of this tool is one of the acknowledged contributors

https://msrc.microsoft.com/update-guide/vulnerability/CVE-2023-36758

- As an anecdote, NIST rated it as 9.8 base score from CVSS version 3, **critical** severity.

https://nvd.nist.gov/vuln/detail/CVE-2023-36758

- Currently, WinHarden would consider this vulnerability as a false positive due to default Windows security configuration prevents it.

Full source code and Visual Studio solution in

https://github.com/GRGM/WinHarden

Developed by Ángel Palomo

WinHarden