REPORT

File Ownership Recovery

v1.0.0

Author:

Eldon Gabriel

August 31, 2025

TABLE OF CONTENTS

REVISION HISTORY

Version	Date	👤 Author	Description of Changes
v1.0.0	08/31/2025	Eldon G.	Initial draft.

1.0 FILE OWNERSHIP & REGISTRY

1.1 Project Description

This report documents an experiment to determine whether the Windows Registry can be used to recover file ownership. The intent was to test the boundaries of Windows administration tools and to record the outcome clearly for future reference.

Disclaimer: This guide is based on independent testing and research. It is not affiliated with, endorsed by, or representative of any institution, training provider, or employer.

1.2 Initial Task: Registry-Based Ownership Recovery

The starting assumption was that ownership information might be retrievable from registry keys. The task was to explore registry paths related to security identifiers. This includes (SIDs), policies, and file permissions, and then attempt to restore ownership through edits.

1.3 Findings and Testing Results

Testing confirmed that NTFS (New Technology File System) file ownership is stored as metadata on the file system itself, not in the Registry. Attempts to locate ownership attributes in registry locations did not produce usable results. File recovery or reassignment of ownership must be done using supported tools such as `takeown`, `icacls`, or PowerShell commands.

1.4 Special Case Consideration

While the Registry influences global security settings and policies. It does not provide a direct way to recover file ownership. Modifying registry keys for this purpose may lead to instability and is not a recommended practice.

1.5 Supporting Work

I needed to check my results. I changed who owned the files. I used Windows tools like (`takeown` and `icacls`). This proved the right way to do it. It also showed a clear difference. Registry settings and file data (NTFS) did not match.

2.0 CONCLUSION

2.1 Key Takeaways

- File ownership is managed by NTFS, not the Windows Registry.
- Registry modification is not a valid method for recovering file ownership.
- Built-in tools such as `takeown`, `icacls`, and PowerShell provide supported solutions.
- Documenting unsuccessful methods adds value by clarifying system boundaries.

2.2 Security Implications and Recommendations

Improper handling of file ownership can expose sensitive data or lead to privilege escalation. Attempting registry modifications to recover ownership introduces additional risks, such as corrupting security policies or destabilizing the operating system.

Recommendations

Technical Controls

- Use only supported tools (`takeown`, `icacls`, PowerShell) for ownership management.
- Enforce least privilege by restricting administrative rights to only those who require them.
- Monitor ownership changes through event logging and SIEM alerts.

Procedural Controls

- Establish clear incident response steps for handling inaccessible files.
- Train administrators to avoid unsupported registry modifications for security tasks.
- Document and regularly review file permission/ownership procedures.

Best Practices and Frameworks

- Aligns with **NIST SP 800-53 AC-3 (Access Enforcement)** and **AC-6 (Least Privilege)**.
- Supports compliance efforts under **ISO 27001 A.9.2 (User Access Management)**.
