REPORT
Hardening Windows User Rights via Local Group Policy

v1.0.0

Author:

Eldon Gabriel

October 16, 2025

TABLE OF CONTENTS

Disclaimer: This report documents my personal work completing an MCSI lab exercise. It reflects my independent understanding and configuration of Windows 10 Local Group Policy settings in a controlled environment. No MCSI video content, lab materials, or proprietary instructions have been shared or distributed. All information presented follows MCSI's disclosure and academic integrity policies.

REVISION HISTORY

Version	Date	Author	Description of Changes
v1.0.0	10/16/2025	Eldon G.	Initial draft.

1.0 CONFIGURING USER RIGHTS VIA LOCAL GPO

1.1 Project Description

This project involved securing a Windows 10 Virtual Machine (VM). It involved using a **Local Group Policy Object (GPO)** to set and limit **User Rights Assignment** policies. If user rights are not set correctly, unauthorized users may exploit the system to gain access or perform privileged actions. By setting these rights, administrators ensure system security by restricting sensitive functions to approved users

This exercise aims to demonstrate local system hardening through Group Policy, enforcing the principle of least privilege.

1.2 User Rights Policy Configuration

I opened the **Local Group Policy Editor (`gpedit.msc`)** and navigated to **Computer Configuration → Windows Settings → Security Settings → Local Policies → User Rights Assignment**.

From there, I configured the following settings:

Policy Path	Assigned To	Purpose
Allow log on locally	Administrators, Users	Restrict local login privileges to authorized roles
Create a pagefile	Administrators	Limit virtual memory control to administrators
Create symbolic links	Administrators	Prevent privilege escalation via symlink abuse
Debug programs	Administrators	Restrict debugging tools to administrators
Force shutdown from a remote system	Administrators	Prevent remote shutdown misuse
Load and unload device drivers	Administrators	Limit driver management to authorized admins
Profile single process	Administrators	Restrict performance profiling tools
Take ownership of files or other objects	Administrators	Ensure only administrators can take file ownership

After applying the settings, I closed the Group Policy Editor and executed `gpupdate /force` to apply the configuration.

1.3 Verification Steps

1. The Local Security Policy (`secpol.msc`) was opened to cross-check each setting.
2. All changes under **User Rights Assignment were verified**.

1.4 Observations and Challenges

- `gpedit.msc` enables policy control; yet, some changes require administrative privileges.
- Policies for **symbolic links** and **debugging programs** prevent privilege escalation attacks.
- Precautions must be taken to avoid locking out necessary local accounts from basic operations (e.g., logging in)

1.5 Additional or Supporting Work

The configuration was tested in a **virtualized lab environment** to prevent **unintentional restrictions** on production machines.

2.0 CONCLUSION

2.1 Key Takeaways

This exercise demonstrated how **User Rights Assignment** provides detailed control over important actions on Windows systems. These limits help ensure that only approved administrators can perform high-risk tasks.

2.2 Security Implications and Recommendations

Setting clear user rights policies makes computers secure. It prevents unauthorized access and the misuse of privileges. For enterprise environments, it is best to use **Active Directory Group Policy Objects**. Thus, the rules are the same everywhere and easier to manage.
