

In MSI (Windows Installer), the context refers to the level of access a process or component has within the Windows operating system. The key difference lies in **whether the installation or action runs under the user's profile (User Context) or with elevated system privileges (System Context)**. Additionally, there are situations where actions might require Admin privileges, even if not directly running in System Context. [\[1, 2, 3, 4\]](#)

User Context:

- **Definition:** Runs under the currently logged-in user's credentials and within their user profile.
- **Access:** Can access files and settings specific to the user profile, but typically doesn't have full system-wide access.
- **Best for:** User-specific applications, customizations, and tasks that don't require system-wide changes. [\[2, 3\]](#)

System Context:

- **Definition:** Runs with elevated privileges, often as the SYSTEM user, with full system-wide access.
- **Access:** Has access to all files and system resources, including those outside the user's profile.
- **Best for:** System-wide installations, critical system policies, and scenarios where full control is needed. [\[2\]](#)

Admin Context (Implicit in some cases):

- **Definition:** Although not a distinct context like User or System, many MSI installations that require system-wide changes (like installing a program that runs as a service) might require Admin privileges.
- **Access:** These installations require the user to have Admin privileges to run the MSI and perform the necessary system changes.
- **Best for:** Installations that modify system files, services, or other resources that require elevated permissions. [\[3, 3, 4, 4, 5, 6, 7, 8\]](#)

In summary:

- **User Context:** Limited access to the user's profile.
- **System Context:** Full system-wide access.
- **Admin Context (Implied):** Requires Admin privileges for system-wide changes. [\[2, 2, 3, 3, 4, 4, 9\]](#)

Understanding these contexts is crucial for correctly deploying software using MSI, ensuring the right level of access for the application and its components. [\[1, 2\]](#)

- [1] <https://www.linkedin.com/pulse/understanding-system-context-vs-user-process-execution-so-ni-rsdgc>
- [2] <https://andrewstaylor.com/2022/11/22/intune-comparing-system-vs-user-for-everything/>
- [3] <https://www.reverera.com/blog/software-installation/just-be-yourself-understanding-windows-installer-msi-custom-action-contexts/>
- [4] <https://www.cyber.gc.ca/en/guidance/managing-and-controlling-administrative-privileges-itsa-p10094>
- [5] <https://docs.cribl.io/edge/deploy-windows-msi-options/>
- [6] <https://community.spiceworks.com/t/running-the-bat-file-with-admin-rights/1015593>
- [7] <https://stackoverflow.com/questions/8085823/windows-installer-with-both-per-user-and-admin-parts>
- [8] <https://www.sciencedirect.com/topics/computer-science/administrative-credential>
- [9] <https://docs.flathub.org/docs/for-users/user-vs-system-install>