

Account Types, and Why You Don't Do Daily Work as Admin

Fri, Aug 7, 4:00 PM EDT · <https://scottslab.io/posts/account-types-and-privilege-elevation>



TL;DR

Account types matter because the blast radius differs: standard accounts for daily work, privileged accounts that log every action and only get elevated to when needed, guest accounts that must expire, and service accounts that run processes with no interactive login and a password no human knows. Shared/generic accounts are the one type to avoid entirely, because they destroy accountability.

The account types are worth keeping straight because the damage each can do is wildly different, and the single most important habit in all of security operations falls right out of that: you do not do your daily work in a privileged account.

A standard user account is the default: routine access, monitored like everything else, subject to the usual lifecycle. A privileged account is the one that can actually hurt you, so it gets treated differently: every action logged, and anything suspicious escalated immediately rather than triaged next week. The discipline that keeps privileged accounts safe is separating your everyday and elevated identities. You live in a standard account and elevate only for the task that needs it, whether that's signing into a separate admin account, assuming a role, or a scoped `sudo`. An admin who

reads email and browses the web from a domain-admin account is one phishing link away from handing that account to someone else.



Guest accounts are temporary by definition and should be tied to a specific person and set to expire on their own. A guest account that outlives its reason is just an unowned door standing open. Service accounts are the quiet ones: they run processes and daemons, never log in interactively, and ideally carry a password no human actually knows, system-managed and rotated automatically. A service credential living in someone's head or a config file is a favorite target.



Then the anti-pattern that ties this series together: shared or generic accounts. Everything above exists to make actions attributable; a shared account throws that away and hands plausible deniability to everyone who holds it. If you take one rule from account management, it's that the right number of shared accounts is zero. Role-based security groups give you the exact convenience people reach for shared accounts to get: everyone on the team can get in. You just don't pay the cost of nobody knowing who did what.