What is LSASS?

LSASS (Local Security Authority Subsystem Service) is the Windows process that enforces the security policy on a machine: it verifies logons, handles password changes, and issues access tokens. To do its job it caches the credentials of every user currently signed in, as hashes, and sometimes Kerberos tickets, in memory. Attackers who gain administrator rights on a host dump the LSASS process to harvest those credentials, then reuse them through Pass-the-Hash or Pass-the-Ticket to move to the next machine. LSASS dumping is the engine of lateral movement, which is why protecting it is a core endpoint-hardening step.

HOW IT WORKS

01 The attack: dumping credentials

An attacker who has local administrator rights on a machine can read the memory of the LSASS process and extract every cached credential: NTLM hashes, Kerberos tickets, and in some configurations plaintext passwords.

Those credentials drive lateral movement. The attacker dumps LSASS on the first machine, reuses a harvested hash to authenticate to the next machine where that account has access (Pass-the-Hash), dumps that machine's LSASS for fresh and more privileged credentials, and repeats until a Domain Admin's credentials appear in memory somewhere. One privileged user logging in to a compromised host is enough to lose the domain.

02 How the attack runs

The documented techniques include reading LSASS live or dumping it for offline parsing:

- Live extraction: `sekurlsa::logonpasswords` (Mimikatz) reads hashes, tickets, and cached secrets
- Create a memory dump to parse elsewhere: `procdump -ma lsass.exe lsass.dmp`, then process it offline
- Reuse a harvested hash without cracking: `wmiexec.py -hashes :<nt-hash> corp.local/admin@host` (Impacket Pass-the-Hash)

These are well-known methods shown so defenders can detect and block them.

HOW TO DEFEND

- Enable Credential Guard, which isolates LSASS secrets in a virtualised container that ordinary admin rights cannot read.
- Add privileged accounts to the Protected Users group so their credentials are not cached in a reusable form.
- Turn on LSASS protection (RunAsPPL) so non-protected processes cannot open it.
- Deploy LAPS so a unique local-admin password on every machine stops one harvested hash from unlocking the next.
- Keep Domain Admins off ordinary workstations, and detect tools that open a handle to `lsass.exe`.

SOURCES

[1] MITRE ATT&CK Enterprise Matrix

[2] Microsoft: Kerberos Authentication Overview

[3] NIST SP 800-115 Technical Guide to Security Testing

WHY ONE ADMIN LOGIN MATTERS

A Domain Admin who logs in to an ordinary, compromised workstation leaves their credentials in that machine's LSASS memory. Keeping privileged accounts off lower-tier machines is the single biggest reduction in LSASS-dumping risk.

Test your Active Directory before an attacker does.

securelayer7.net/learn/active-directory/what-is-lsass

Open online