AUSTRIAN INSTITUTE OF TECHNOLOGY
TOMORROW TODAY

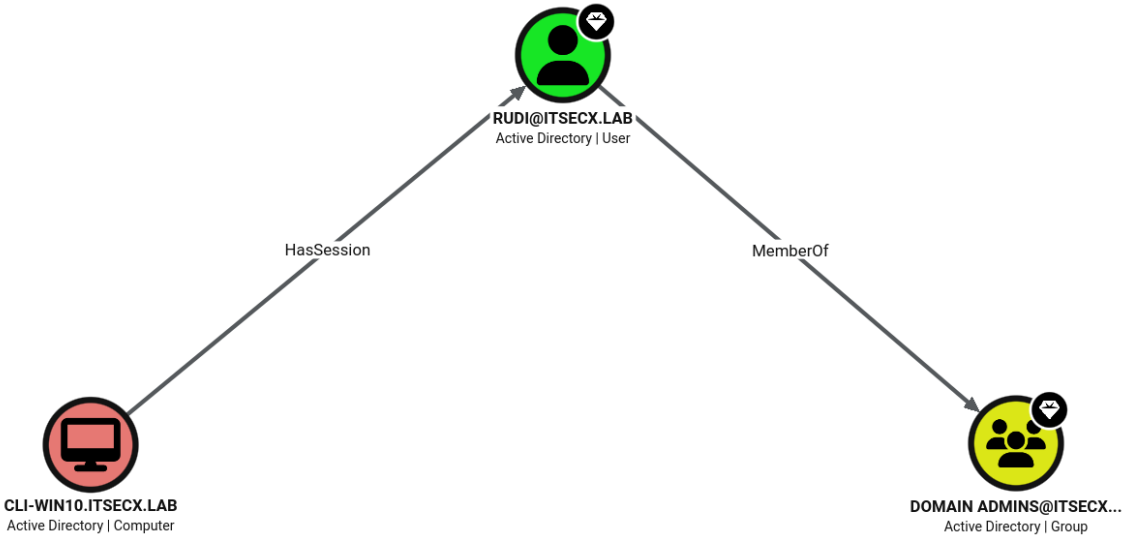
# LSASS IN DER POST-PPL-ÄRA

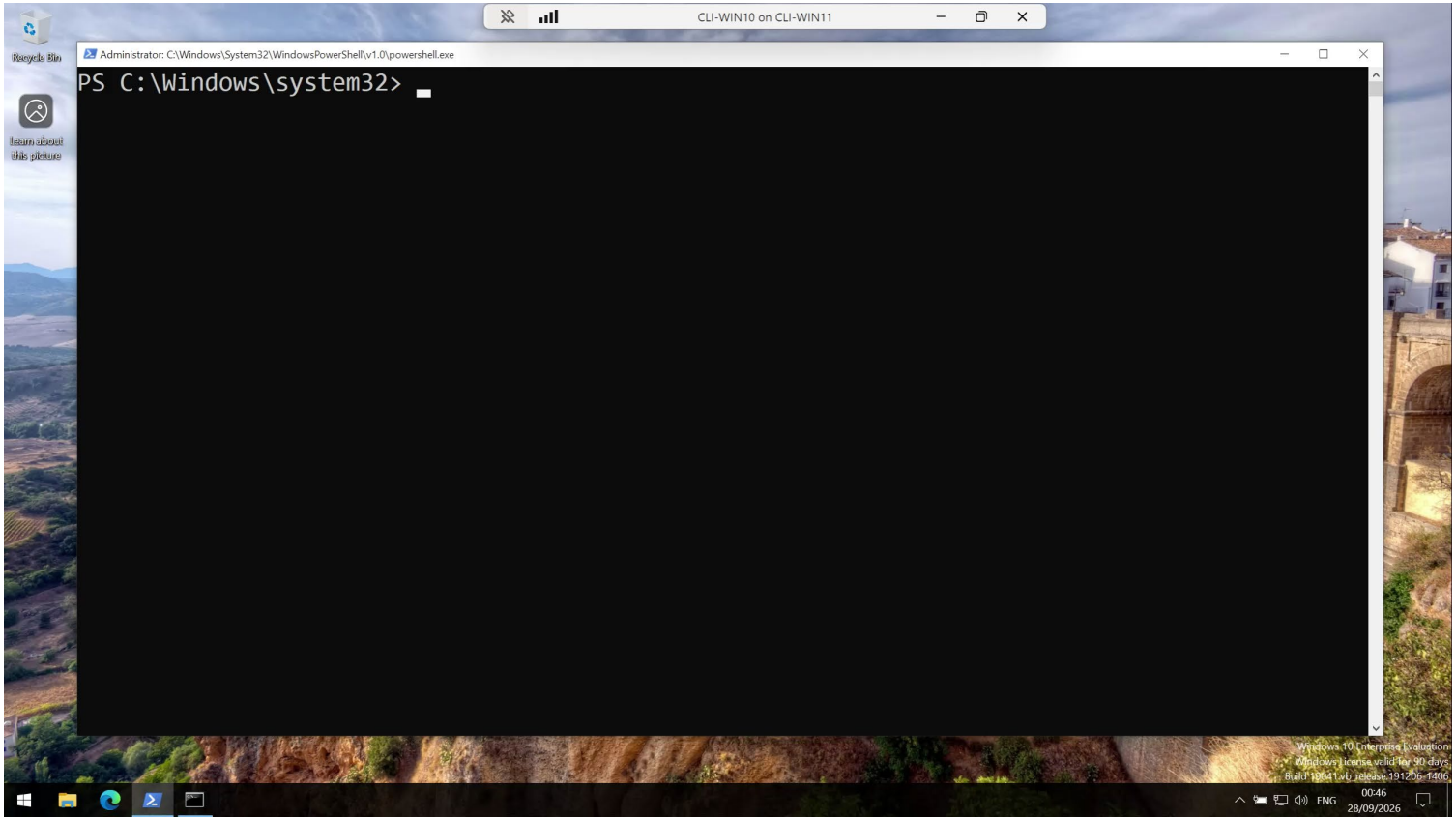
## Credential Access auf Windows 11 22H2+

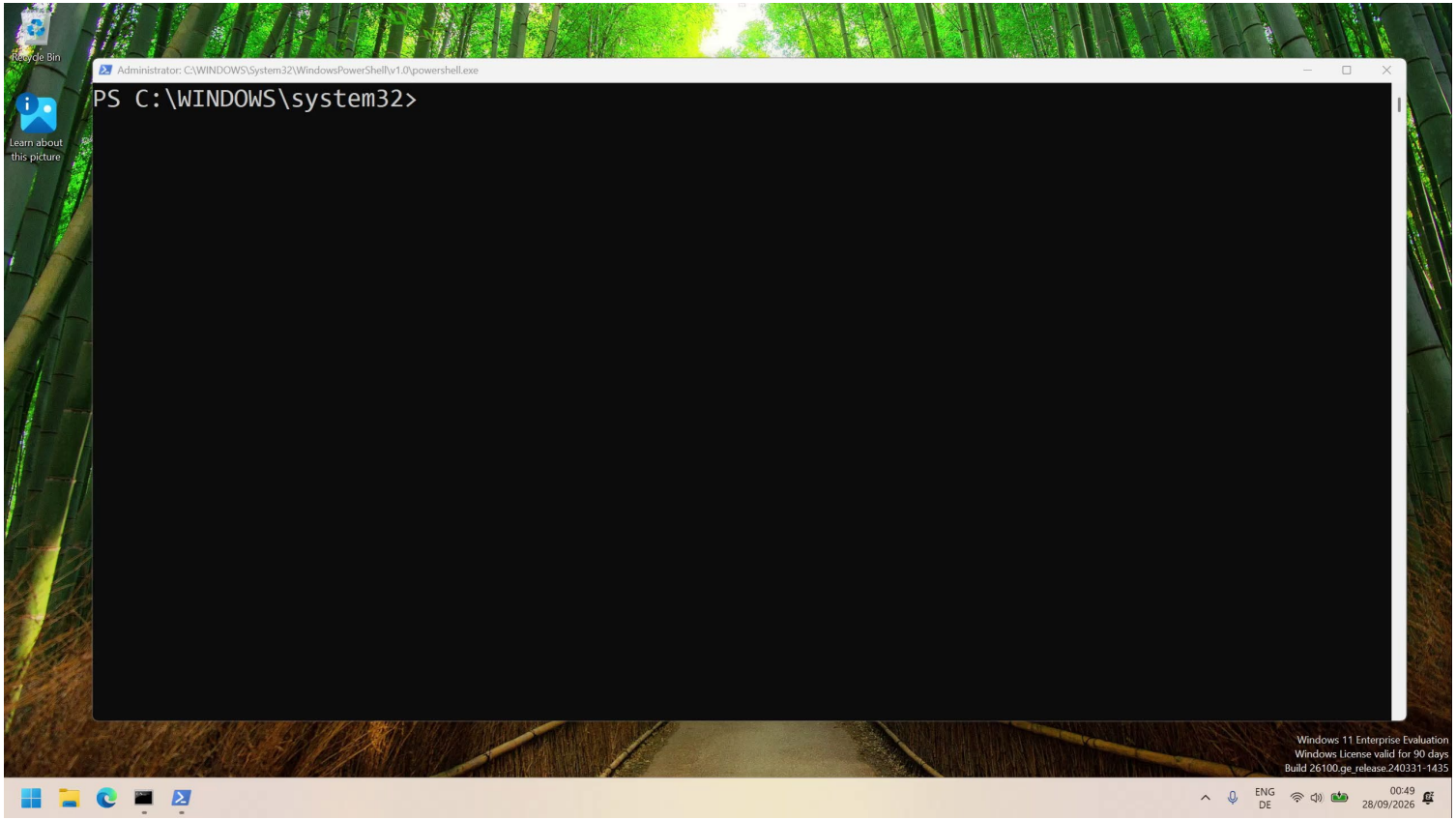
Dominik Steffan · Felix Schuster
Research Engineers · AIT Austrian Institute of Technology
IT-SECX 2026 · 2. Oktober 2026



# MOTIVATION


RUDI@ITSECX.LAB
Active Directory | User
HasSession
MemberOf
CLI-WIN10.ITSECX.LAB
Active Directory | Computer
DOMAIN ADMINS@ITSECX...
Active Directory | Group


CLI-WIN10 on CLI-WIN11
Administrator: C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe
PS C:\Windows\system32>
Recycle Bin
Learn about this picture
Windows 10 Enterprise Evaluation
Windows license valid for 90 days
Build 19041.vb_release.191206-1406
ENG
00:46
28/09/2026


Recycle Bin
Learn about this picture
Administrator: C:\WINDOWS\System32\WindowsPowerShell\v1.0\powershell.exe
PS C:\WINDOWS\system32>
Windows 11 Enterprise Evaluation
Windows License valid for 90 days
Build 26100.ge_release.240331-1435
ENG
DE
00:49
28/09/2026

# WAS MICROSOFT GEBAUT HAT: 22H2 → 25H2

- **22H2:** LSA Protection (PPL) Default auf domain-joined
- Treiber-Blockliste: mimidrv.sys & Co.
- Hypervisor-Protected Code Integrity (HVCI) / Memory Integrity
- Core Isolation als übergeordneter Rahmen
- **25H2:** = eKB auf 24H2 → PPL-Default geerbt (kein belegter Zusatz-Schritt)

# WANN GREIFT DER DEFAULT? - DIE DOKU WIDERSPRICHT SICH

- **„Automatic enablement":** Default nur bei Neuinstallation 22H2+ + enterprise-joined + HVCI-fähig (=VBS-fähig)

- **„Enabled without UEFI lock":** „applied by default … with a new installation of Windows 11 22H2 or later" - kein Domain-Join genannt

- **Unser Messwert:**
  - 24H2-Neuinstallation, standalone (kein Domain-Join)
  - LSA Protection ist aktiviert

*Quelle: Microsoft Learn - Configuring Additional LSA Protection*

# 25H2 = ENABLEMENT PACKAGE AUF 24H2

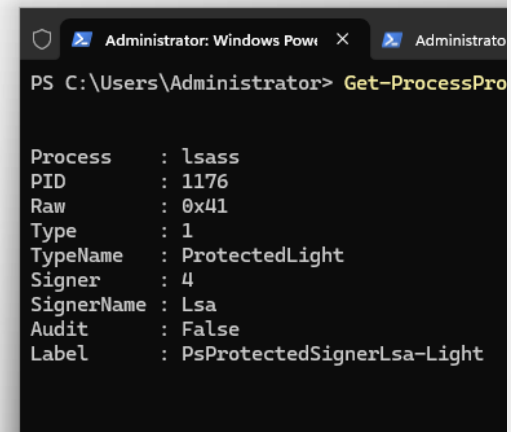
- 25H2 = eKB (KB5054156) auf 24H2-Basis (Build 26100 → 26200)
- Media Creation Tool liefert aktuell noch 24H2 (26100)
- PPL-Default fällt zur 24H2-Install-Zeit und wird vererbt
- **Wir testen den repräsentativen Zustand: 25H2 via eKB**
  - Hyper-V VM + Bare Metal
  - Secure Boot aktiv
  - Standalone Installation
- Research begann auf Hyper-V-VM → Bare-Metal + Frisch-VM bestätigt

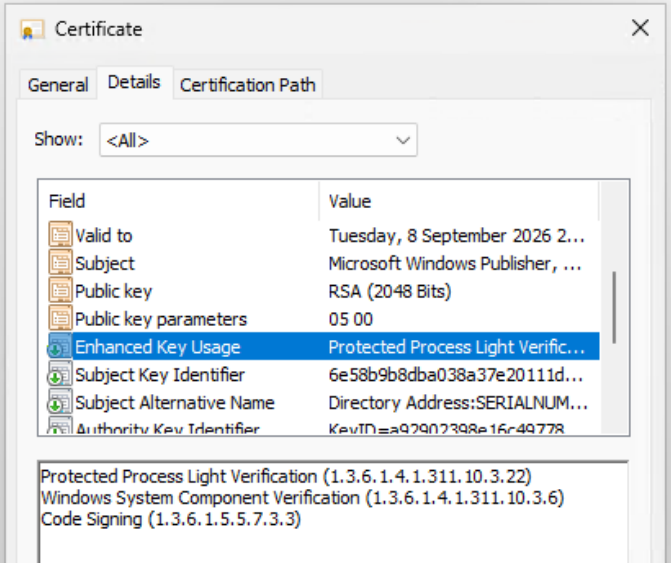
# PROTECTED PROCESS LIGHT IN 60 SEKUNDEN

- lsass läuft als geschützter Prozess (Signer-Level)
- Kein Handle-Zugriff mit vollen Rechten für normale Admin-Prozesse
- Label: **PsProtectedSignerLsa-Light**
- Type 1 (ProtectedLight) · Signer 4 (Lsa) · Raw 0x41




| Protection level | Value | Signer | Type |
|---|---|---|---|
| PS_PROTECTED_SYSTEM | 0x72 | WinSystem (7) | Protected (2) |
| PS_PROTECTED_WINTCB | 0x62 | WinTcb (6) | Protected (2) |
| PS_PROTECTED_WINDOWS | 0x52 | Windows (5) | Protected (2) |
| PS_PROTECTED_AUTHENTICODE | 0x12 | Authenticode (1) | Protected (2) |
| PS_PROTECTED_WINTCB_LIGHT | 0x61 | WinTcb (6) | Protected Light (1) |
| PS_PROTECTED_WINDOWS_LIGHT | 0x51 | Windows (5) | Protected Light (1) |
| PS_PROTECTED_LSA_LIGHT | 0x41 | Lsa (4) | Protected Light (1) |
| PS_PROTECTED_ANTIMALWARE_LIGHT | 0x31 | Antimalware (3) | Protected Light (1) |
| PS_PROTECTED_AUTHENTICODE_LIGHT | 0x11 | Authenticode (1) | Protected Light (1) |



AUSTRIAN INSTITUTE OF TECHNOLOGY
TOMORROW TODAY

# KURZE FRAGE: RUNASPPL - 1 ODER 2?

Ihr aktiviert LSA Protection auf einer Test-Maschine per Registry.

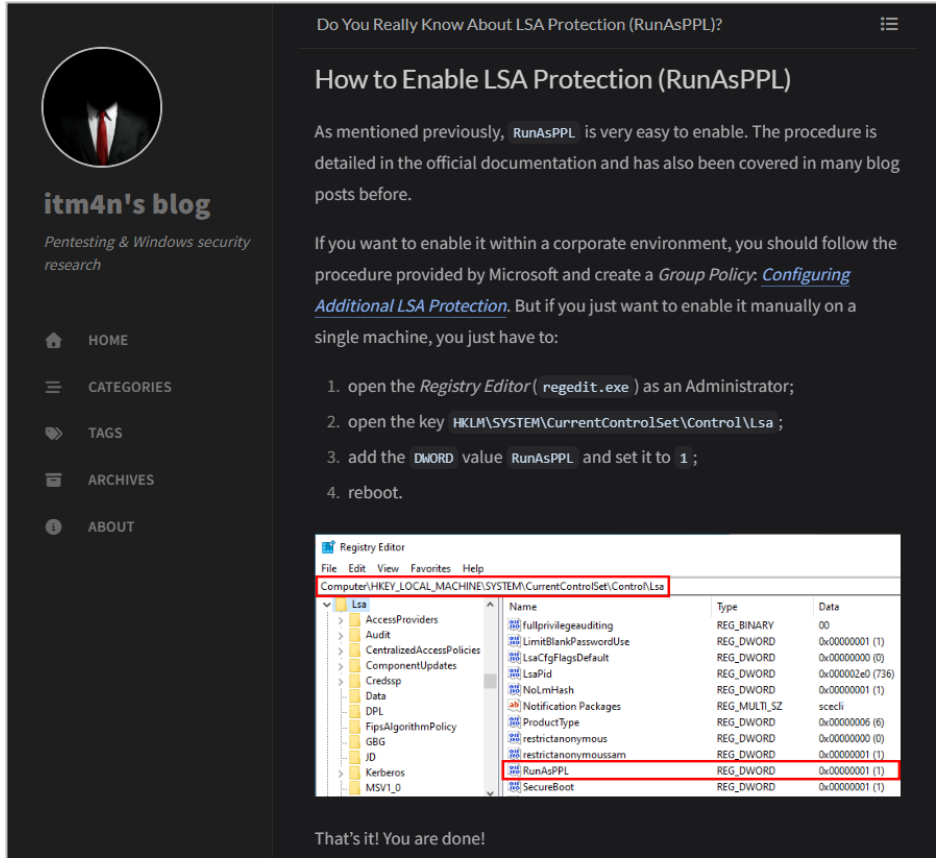
Welchen Wert setzt ihr für RunAsPPL? - Hand hoch: 1 oder 2?



# WER „1" GESAGT HAT: WILLKOMMEN IM CLUB

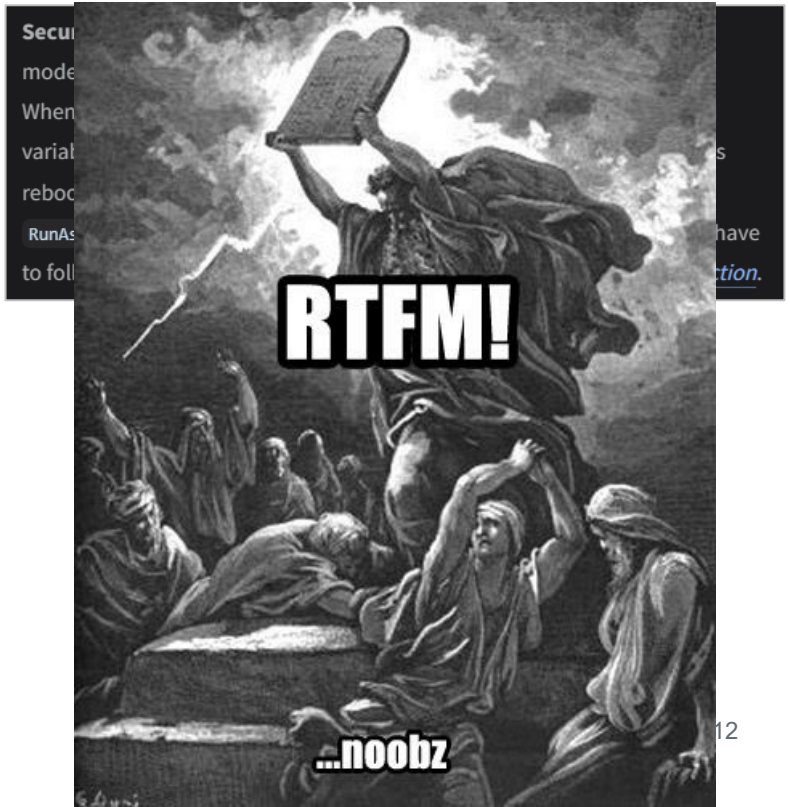
itm4n's vielgelesener Artikel empfiehlt genau das: RunAsPPL = 1.

- **Aber: 1 = „with UEFI lock"** → Wert landet in einer UEFI-Firmware-Variable
- Ab da ist PPL verriegelt: RunAsPPL = 0 wirkt nicht mehr



# „DISABLED" HEISST NICHT DISABLED

- Erste Test-VM: RunAsPPL = 0 gesetzt, UI-Toggle aus
- Trotzdem: lsass = **PsProtectedSignerLsa-Light**
- WinInit Event ID 12 bestätigt PPL beim Boot



# LSA PROTECTION ≠ CREDENTIAL GUARD

| | LSA Protection (PPL) | Credential Guard |
|---|---|---|
| **Schützt** | lsass.exe als Protected Process | Secrets im VBS-isolierten LsaIso.exe |
| **Grundlage** | PPL / Signer-Level | Virtualization-Based Security |
| **Registry** | …\Control\Lsa\RunAsPPL | …\DeviceGuard\LsaCfgFlags |
| **Werte** | 1 = mit UEFI-Lock<br>2 = ohne<br>0 = aus | 1 = mit UEFI-Lock<br>2 = ohne<br>0 = aus |
| **Opt-Out (Lock)** | **LsaPplConfig.efi** | **SecConfig.efi** (DISABLE-LSA-ISO) |

# ZWEI MODI: DEFAULT VS. UEFI-LOCK

| Modus | Wert | Speicher | Lokal abschaltbar? |
|---|---|---|---|
| **Default** | RunAsPPL = 2 | Registry | ✓ **Ja: RunAsPPL=0 + Reboot** |
| **UEFI-Lock** | RunAsPPL = 1 | UEFI-Firmware-Variable | ✗ **Nein: nur Firmware-Weg** |

# BEI AKTIVEM UEFI-LOCK HILFT NUR DER FIRMWARE-WEG

| Ebene / Methode | Ergebnis |
|---|---|
| Registry RunAsPPL = 0 | ✗ **wirkungslos** |
| UI-Toggle / GPO | ✗ **wirkungslos** |
| MS Opt-Out (LsaPplConfig.efi) | ✓ **löst den Lock (VM wie HW)** |
| Hyper-V: Secure Boot deaktivieren | ✓ **PsProtectedSignerNone** |

# MAN SIEHT DER MASCHINE DEN MODUS NICHT AN

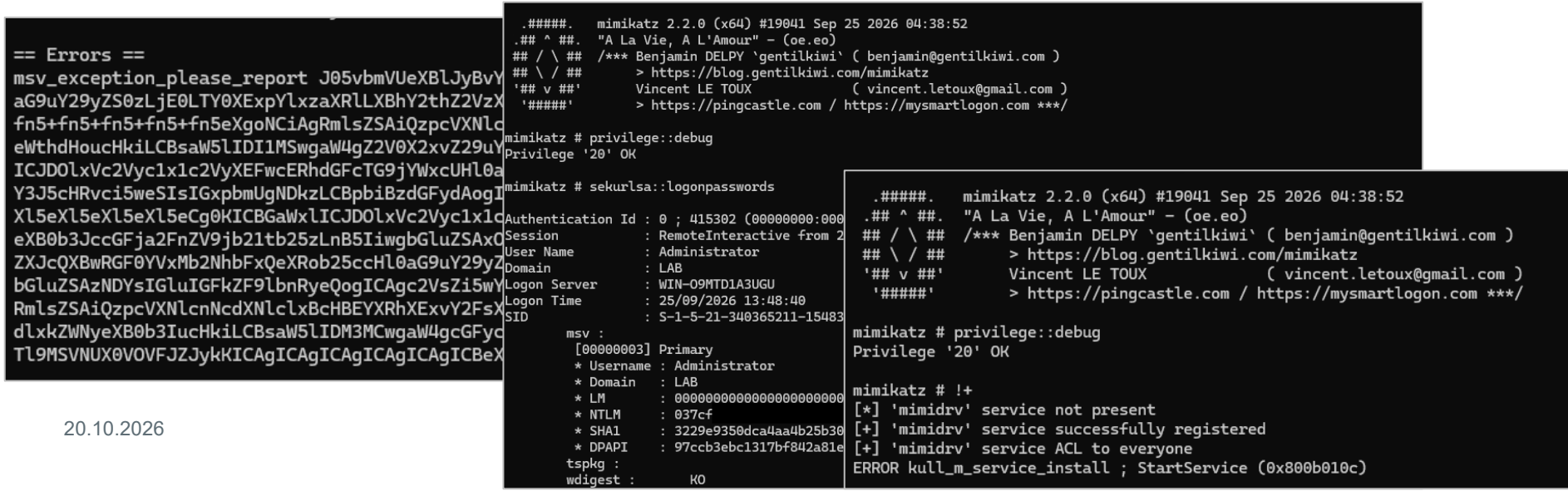
- procexp + Event ID 12 → in beiden Modi identisch
- GUI / Registry können sogar „aus" zeigen, während der Lock erzwingt
- Einziger verlässlicher Test: **RunAsPPL = 0 + Reboot → bleibt PPL = verriegelt**
- Angreifer: erkennt Modus nicht per Enumeration → probieren oder Bypass
- Defender: Härtung nicht per Blick verifizierbar

# ZWEI MAUERN: HANDLE (PPL) + STRUCT-DRIFT

| Tool | PPL aktiv | PPL aus |
|---|---|---|
| **kslkatz** | ✓ Bypass | — |
| **mimikatz (aus Source)** | ✗ Handle | ✓ Dump gelingt |
| **mimikatz 2022-Release** | ✗ Handle | ✗ Struct-Drift |
| **pypykatz** | ✗ Handle | ✗ Struct-Drift |
| **mimidrv** | ✗ Blockliste | ✗ Blockliste |
| **PPLdump** | ✗ Technik dicht seit 21H2 | — |
| **Rubeus (Kerberos)** | ✓ API-Call statt Dump | — |

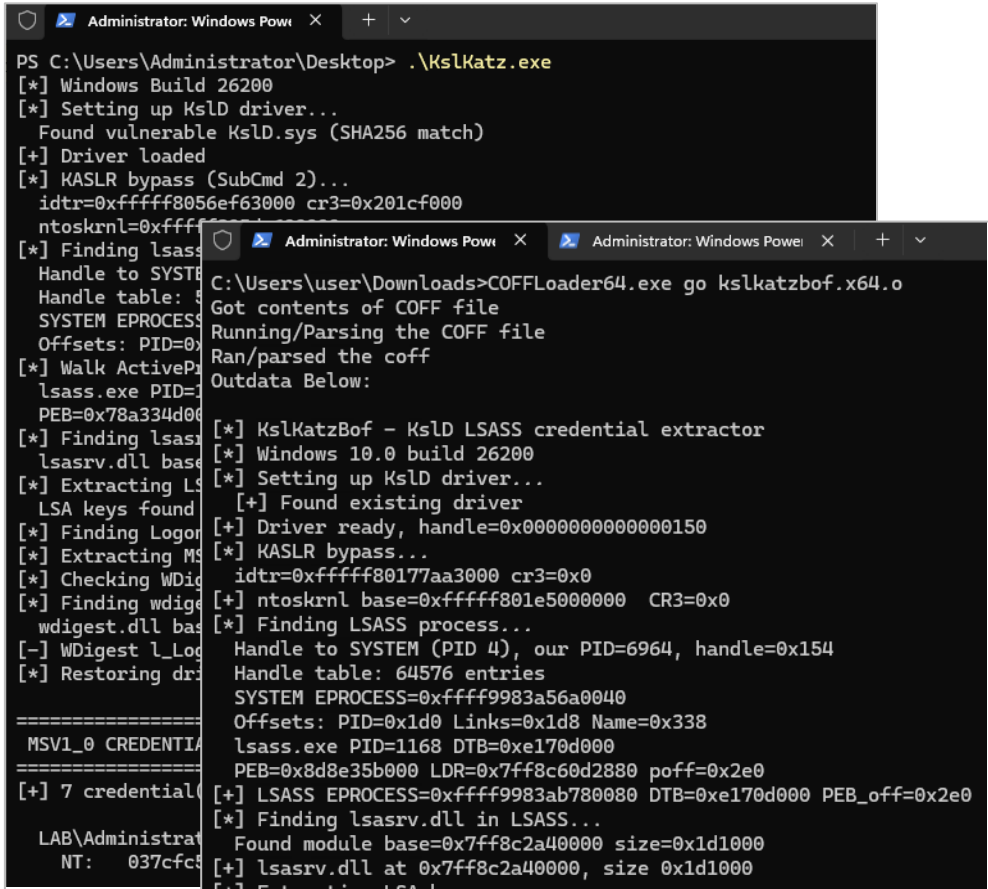
# ZWEI VERSCHIEDENE FEHLERSTELLEN

- **PPL aus:**
  - pypykatz → msv_exception (Struct-Parsing)
  - mimikatz aus Source → Dump
- **PPL an:**
  - Access Denied schon am Handle
  - mimidrv → vermutlich Treiber-Blockliste



# KSLKATZ

- Umgeht PPL (statt es abzuschalten) - dumpt lsass bei aktivem PPL
- **Original (S1lkys/KslKatz, M. Barz)**
- **BOF-Port (KslKatzBof, S. Ackermann)**



# DER DEFAULT SCHÜTZT NICHT GEGEN DEN LOKALEN ADMIN

- Default (RunAsPPL=2) ist per Registry abschaltbar → aktiv härten
- **LSA Protection explizit „with UEFI Lock" ausrollen (GPO)**
  - Ausrollen ohne UEFI-Lock gibt Angreifern ein PPL-freies Fenster nach Reboot
- UEFI-Firmware-Passwort → schließt den Secure-Boot-Reset-Ausweg
- Verifikation nur behavioral / Event-Logs (kein visueller Cue!)

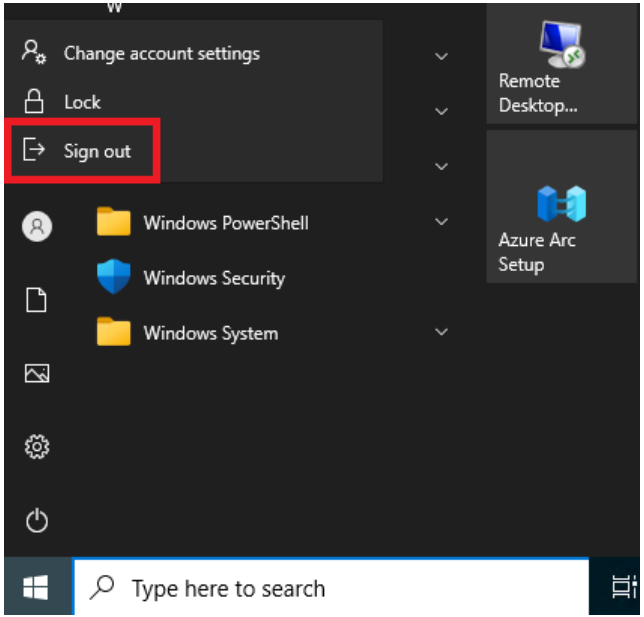
# DOMAIN ADMIN OHNE LSASS

## Access Token Impersonation



# ACCESS TOKEN BASICS

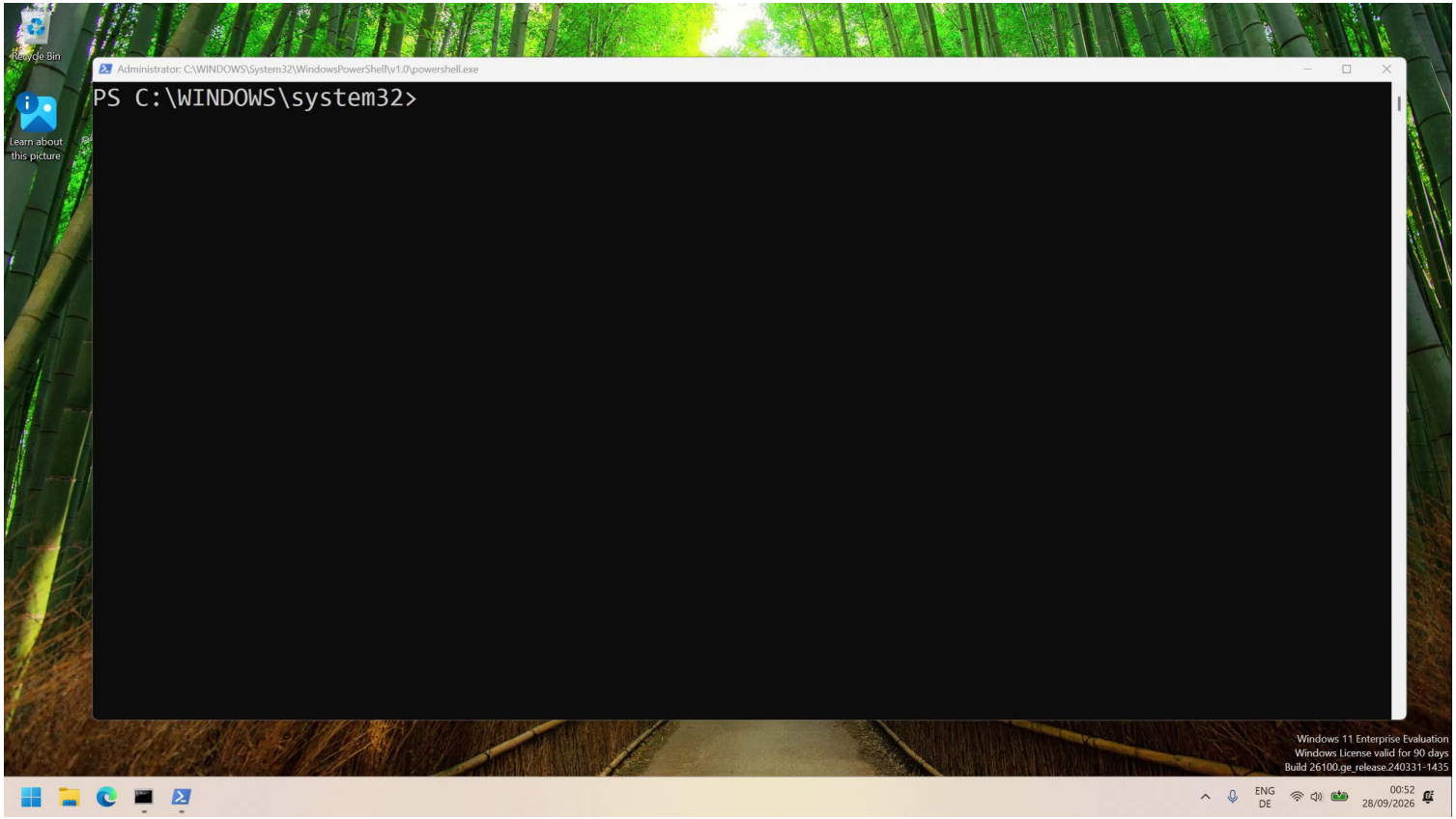
- **Access Token:** Beschreibt den Security Kontext eines Prozesses
  - Security Identifier (SID) für den aktuellen User
  - SIDs für Gruppenmitgliedschaften
  - Liste der Privilegien (SeBackupPrivilege, SeDebugPrivilege, SeImpersonatePrivilege, ...)
- Token bleibt im Speicher, so lange eine Session aktiv ist



# ACCESS TOKEN IMPERSONATION

- **Voraussetzung:**
  - SeDebugPrivilege (Token stehlen)
  - SeImpersonatePrivilege (Token nutzen)

- **Windows API Calls:**
  - OpenProcess() → Prozess des Domain Admins öffnen
  - OpenProcessToken() → Token auslesen
  - DuplicateTokenEx() → Token duplizieren
  - CreateProcessWithTokenW() → Neuen Prozess mit dupliziertem Token starten


Recycle Bin
Learn about this picture
Administrator: C:\WINDOWS\System32\WindowsPowerShell\v1.0\powershell.exe
PS C:\WINDOWS\system32>
Windows 11 Enterprise Evaluation
Windows License valid for 90 days
Build 26100.ge_release.240331-1435
ENG
DE
00:52
28/09/2026

AUSTRIAN INSTITUTE OF TECHNOLOGY
TOMORROW TODAY

# FRAGEN?

Dominik Steffan · Felix Schuster
AIT Austrian Institute of Technology · IT-SECX 2026