**🧠 SPLUNK FORWARDER SETUP & WINDOWS 10 PRO+ WINDOWS SERVER 2019 (AD)**

**(Windows 10 + Domain Controller → Ubuntu Splunk Enterprise)**

**End-to-End | Theory → Pre-Checks → Audit Policy → Splunk Config → Validation**

**Platform:** Splunk  
**Scope:**

* Windows 10 pro (Domain-Joined Endpoint)
* Windows Server 2019 (Active Directory / Domain Controller)
* Ubuntu Splunk Enterprise

**Goal:**  
Perfect log generation, clean ingestion, and trustworthy **identity-centric SOC investigation data**

**PART 1 — CORE THEORY (MINIMUM, MUST-KNOW)**

**1️⃣ What Splunk Is (in this architecture)**

* Ubuntu Server runs **Splunk Enterprise**
  + Receives logs
  + Indexes logs
  + Makes logs searchable
* Windows systems are **log sources only**

⚠️ Splunk does **not** pull logs from Windows by default.

**2️⃣ What the Splunk Universal Forwarder Is**

The **Splunk Universal Forwarder (UF)** is a lightweight agent that:

* Reads logs **locally** on Windows
* Sends them to Splunk over the network
* Does **NO indexing**
* Does **NO searching**

**Forwarder = log shipper**  
**Splunk Enterprise = analysis engine**

**3️⃣ Why BOTH Windows 10 PRO and AD Need a Forwarder**

|  |  |
| --- | --- |
| System | Why Forwarder Is Required |
| Windows 10 PRO | Captures **logon attempts & sessions** |
| Domain Controller | Captures **credential validation & identity decisions** |

Architecture:

Windows 10 PRO─┐

├─ Splunk Universal Forwarder ──> Ubuntu Splunk

AD (DC) ───┘

❌ No forwarder on DC = **no AD investigation**

**4️⃣ Logs We Ingest (INTENTIONALLY LIMITED)**

✅ **Windows Security log ONLY**

Why:

* Authentication (logon / logoff)
* Failed & successful logons
* Privileged access
* Identity evidence (endpoint + DC)

🚫 Sysmon  
🚫 PowerShell logging  
🚫 Script block logging  
🚫 Registry auditing  
🚫 Object access everywhere

➡️ These are **not missing** — they are intentionally excluded.

**PART 2 — WINDOWS PRE-CHECKS**

**(FOR PERFECT LOG GENERATION — BEFORE SPLUNK)**

🔒 **Golden rule:**  
Splunk can only ingest what Windows actually generates.

**🔴 1️⃣ Windows Event Log Service (BOTH SYSTEMS)**

**Applies to:**

* Windows 10 PRO
* Windows Server 2019 (DC)

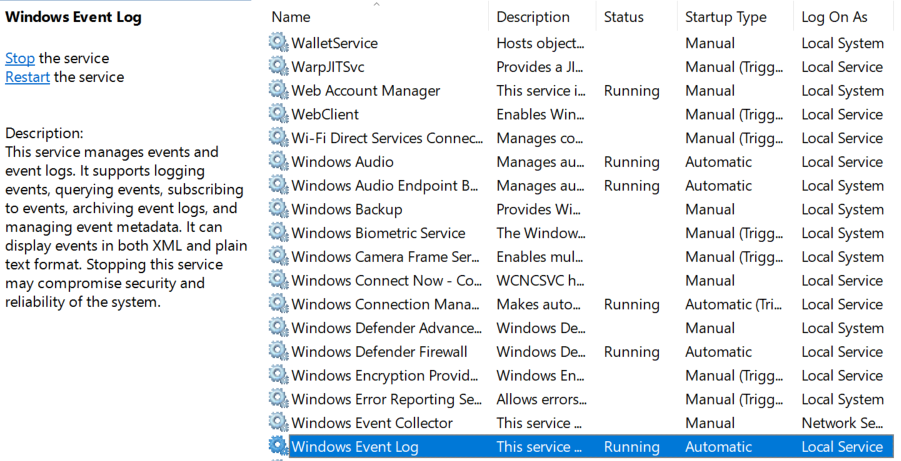
GUI:

1. **Win + R → services.msc**
2. Find **Windows Event Log**

Must be:

* Status: **Running**
* Startup type: **Automatic**

❌ If this fails → stop everything.



**🔴 2️⃣ Security Log Must Exist & Update**

**Windows 10 PRO (Endpoint)**

GUI:

1. Open **Event Viewer**
2. Go to:
3. Windows Logs → Security
4. Confirm:
   * Events already exist
   * New events appear when you:
     + Lock / unlock screen
     + Log in / log out

**Windows Server 2019 (Domain Controller)**

GUI:

1. Open **Event Viewer**
2. Go to:
3. Windows Logs → Security
4. Perform:
   * Log in as domain user
   * Refresh log

If Security log is empty → **AD auditing is broken**

**🔴 3️⃣ Advanced Audit Policy — AUTHENTICATION (CRITICAL)**

This answers the most important SOC question:

**Did Windows reliably record every successful and failed authentication attempt?**

If this is wrong → **everything lies**.

**🎯 Objective**

Ensure logs exist for:

* Successful logons
* Failed logons
* Credential validation attempts

Without this:

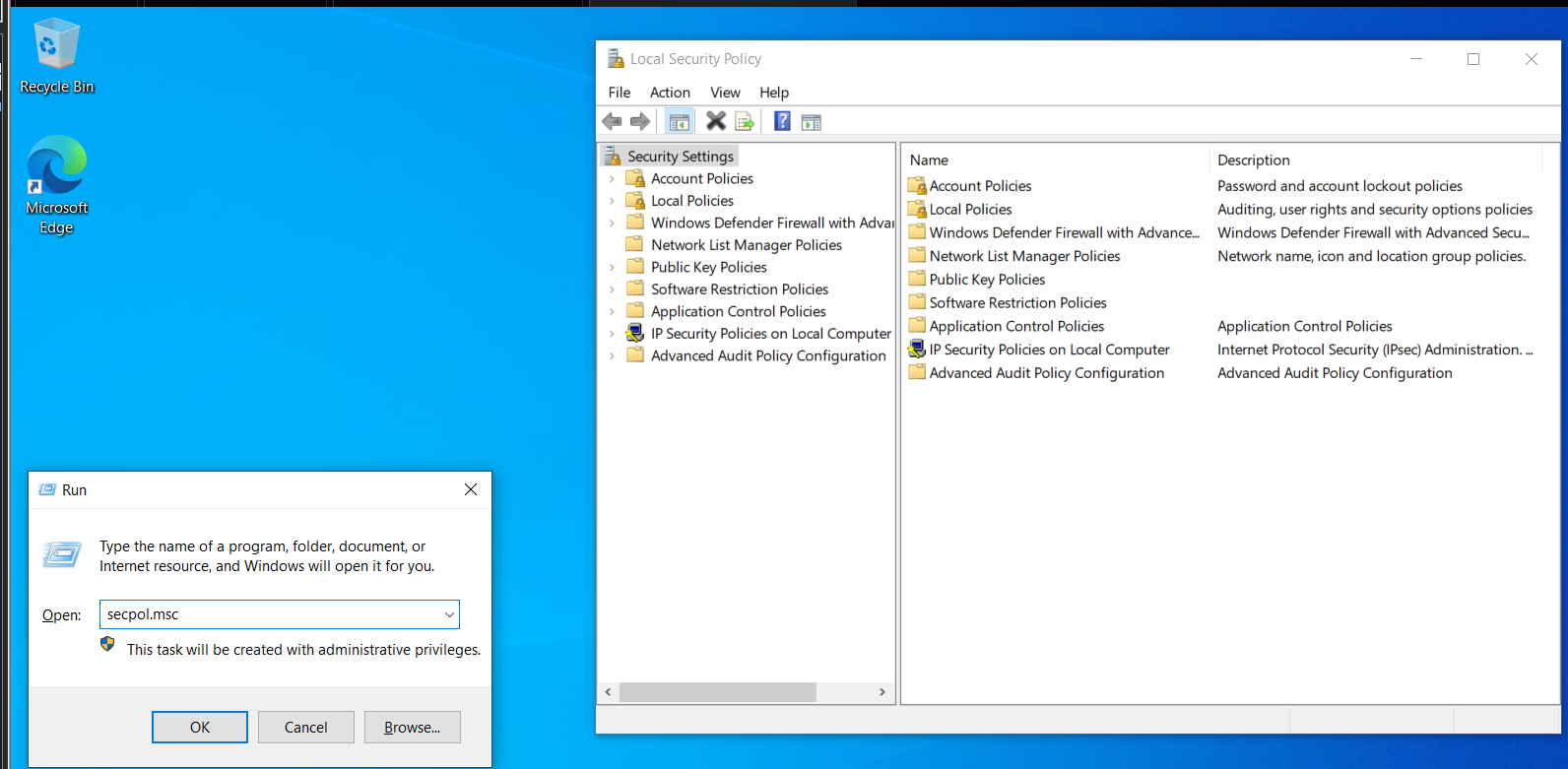
* Password spraying is invisible
* Brute force looks like “nothing”
* AD appears quiet during attacks

**📍 GUI Navigation (IMPORTANT)**

**Windows 10 PRO (Local Policy)**

Win + R → secpol.msc

→ Security Settings



→ Advanced Audit Policy Configuration

→ Audit Policies

**Windows Server 2019 (Domain Controller — GPO)**

Win + R **→ gpmc.msc**

→ Default Domain Controllers Policy

→ Edit

→ Computer Configuration

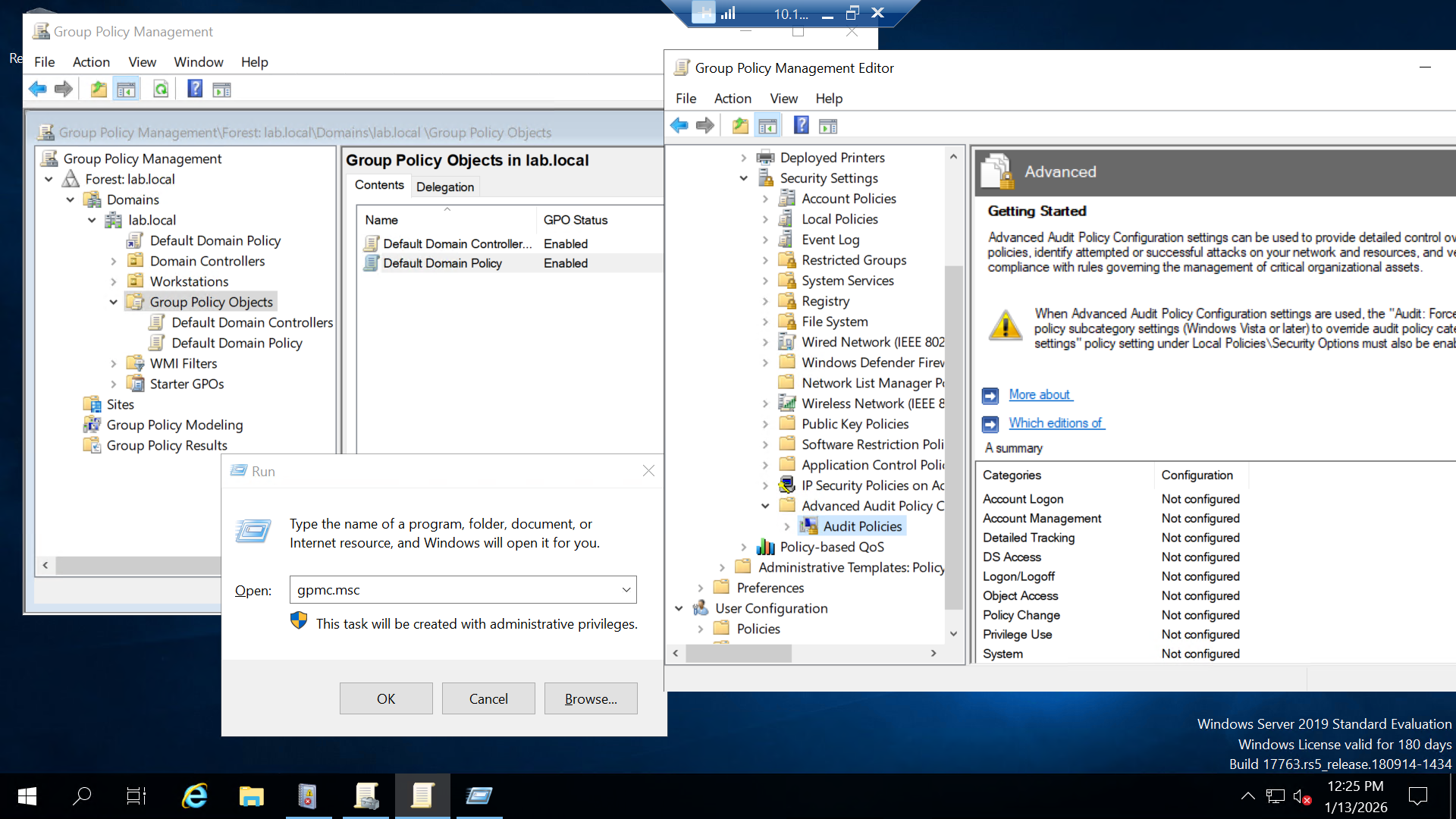
→ Windows Settings

→ Security Settings

→ Advanced Audit Policy Configuration

→ Audit Policies

⚠️ **Do NOT use Basic Audit Policy**  
Advanced Audit Policy overrides it



**🔐 3.1 Account Logon — Credential Validation**

**Path:**

Audit Policies → Account Logon

Enable:

* ✅ Audit Credential Validation
  + ✔ Success
  + ✔ Failure

Why this matters:

* Records password checks
* Generates:
  + 4625 (failed logon)
  + 4776 (NTLM credential validation)

❌ Without this:

* Attackers guess passwords silently

**🔑 3.2 Logon / Logoff — Session Activity**

**Path:**

Audit Policies → Logon/Logoff

Enable:

✅ **Audit Logon**

* ✔ Success
* ✔ Failure
* Events: 4624, 4625

✅ **Audit Logoff**

* ✔ Success
* Event: 4634

✅ **Audit Special Logon**

* ✔ Success
* Event: 4672 (privileged sessions)

SOC value:

* Timeline reconstruction
* Privilege awareness

**🧠 Outcome of Step 3**

After this, BOTH systems must record:

* Successful logons
* Failed logons
* Privileged logons
* Logoffs

Missing any → authentication visibility broken.

**🔴 4️⃣ Privilege Use — Post-Logon Evidence**

**Path:**

Advanced Audit Policy Configuration

→ Audit Policies

→ Privilege Use

Enable:

* ✅ Audit Sensitive Privilege Use
  + ✔ Success
  + ✔ Failure

Why this matters:

* Detects abuse **after access**
* Shows attacker behavior post-login

SOC value: **HIGH**

**🔴 5️⃣ Process Creation — Execution Visibility (NO SYSMON)**

**Path:**

Advanced Audit Policy Configuration

→ Audit Policies

→ Detailed Tracking

Enable:

* ✅ Audit Process Creation
  + ✔ Success

Generates:

* 4688 (new process created)

Why SOCs care:

* Links **logon → execution**
* Enables basic parent/child analysis
* Teaches correlation before advanced tooling

**Additional policy setting for AD (NOT OPTIONAL)**

**STEP 1 — Enable Account Management (CRITICAL)**

**Click path**

Computer Configuration

└ Policies

└ Windows Settings

└ Security Settings

└ Advanced Audit Policy Configuration

└ Audit Policies

└ Account Management

**Configure ALL of these**

Open each one → set **Configure the following audit events** → check **Success + Failure**

* ✅ Audit User Account Management
* ✅ Audit Computer Account Management
* ✅ Audit Security Group Management
* ✅ Audit Distribution Group Management (optional but fine)

Click **Apply → OK** on each.

**Why this matters (facts)**

This generates:

* Event IDs 4720–4735
* Detects:
  + User creation/deletion
  + Group membership abuse
  + Domain Admin escalation

Without this → **AD compromise = invisible**

**STEP 2 — Enable Directory Service Access (AD telemetry)**

**Click path**

Audit Policies

└ Directory Service Access

**Configure**

* ✅ **Audit Directory Service Access** → Success + Failure
* ✅ **Audit Directory Service Changes** → Success

Click **Apply → OK**

**Why this matters**

This produces:

* 5136, 5137, 5141
* Tracks:
  + AD object modification
  + ACL abuse
  + Persistence via permissions

This is **not optional** for SOC work.

**STEP 3 — Enable Policy Change (anti-tampering)**

**Click path**

Audit Policies

└ Policy Change

**Configure**

* ✅ Audit Audit Policy Change → **Success**
* ✅ Audit Authentication Policy Change → **Success**

(Do NOT enable failure here — noise, no value)

**Why this matters**

* Event ID 4719
* Detects attackers disabling logging
* Mandatory for compliance + incident response

**STEP 4 — FORCE Advanced Audit Policy (DO NOT SKIP)**

**Click path**

Computer Configuration

└ Policies

└ Windows Settings

└ Security Settings

└ Local Policies

└ Security Options

**Set this EXACTLY**

**Audit: Force audit policy subcategory settings (Windows Vista or later) to override audit policy category settings**  
➡ **Enabled**

Click **Apply → OK**

**If you skip this**

* Advanced Audit Policy gets ignored
* auditpol lies
* Splunk data becomes inconsistent

This setting is **mandatory**, not advisory.

**STEP 5 — Apply & verify on the DC (proof, not trust)**

**Force policy update**

gpupdate /force

**Verify auditing is active**

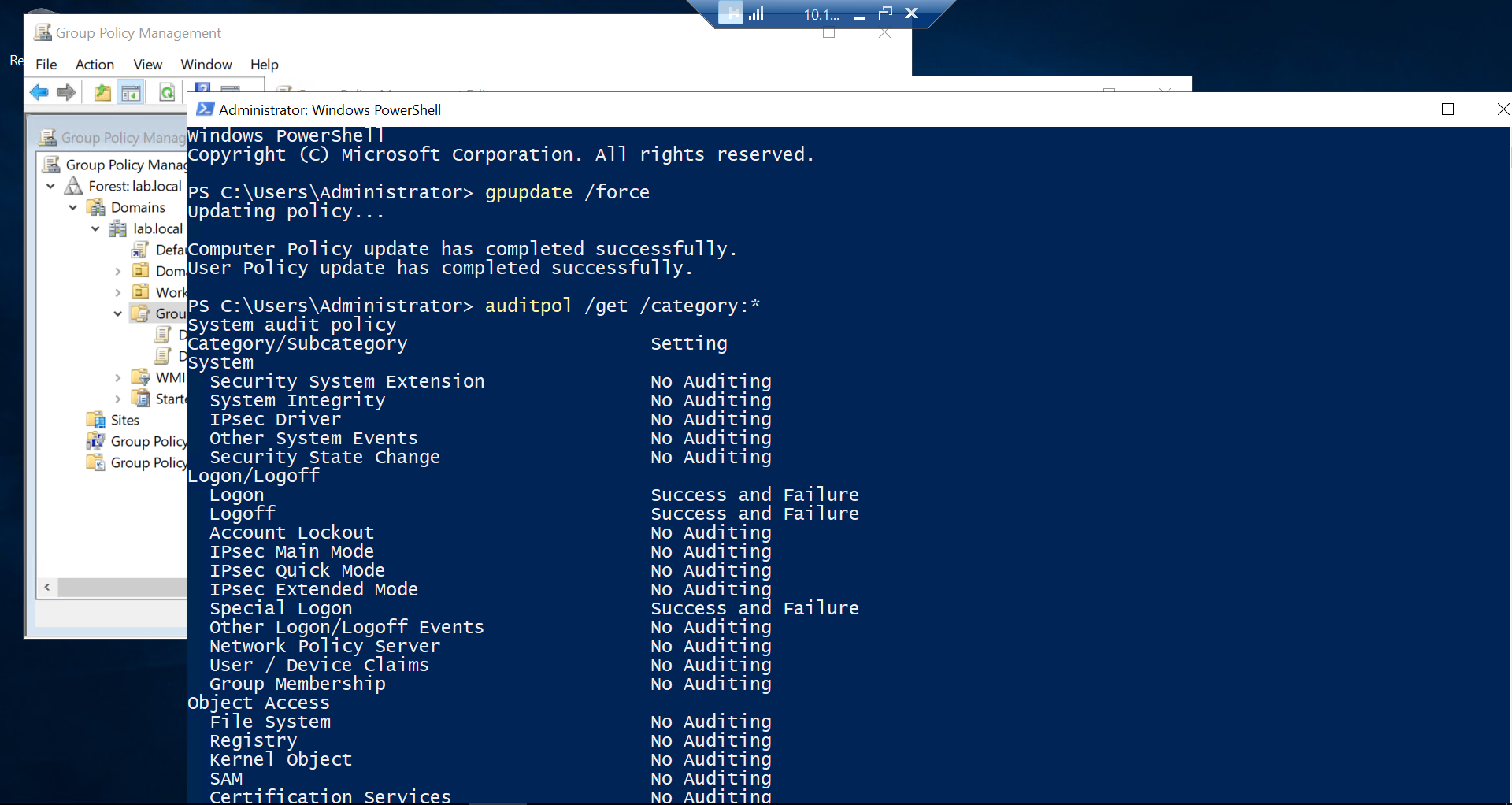
auditpol /get /category:\*

You MUST see:

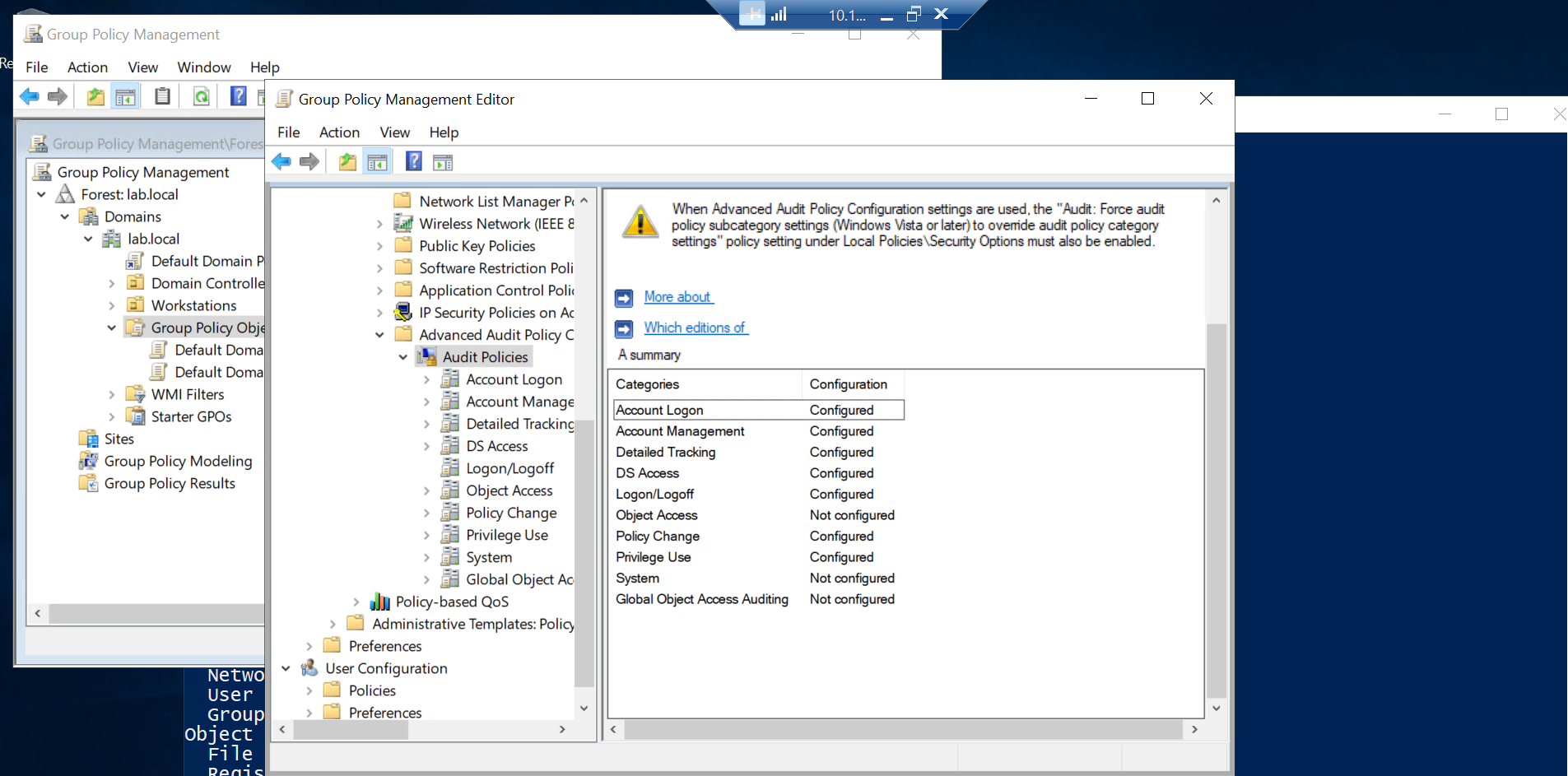
* Account Management → Success and Failure
* DS Access → Enabled
* Policy Change → Enabled

If you don’t → you missed a step.

**WINDOWS SERVER 2019**



**WINDOWS SERVER 2019**



**Additional policy setting for Windows 10 Pro**

You should configure **exactly 5 things**.  
Nothing more. Nothing less.

* 1. **Force Advanced Audit Policy (MANDATORY)**

This applies to **Windows 10 Pro**, not just servers.

**Open**

**Win + R → secpol.msc**

**Go to**

Security Settings

└ Local Policies

└ Security Options

**Set this**

**Audit: Force audit policy subcategory settings (Windows Vista or later) to override audit policy category settings**

➡ **Enabled**

**Why (hard fact):**  
If this is not enabled, Windows may ignore your advanced audit settings.  
That yellow warning you saw? That’s Windows telling you this is required.

**2️. Process Creation WITH command line (CRITICAL)**

This is endpoint gold. Without it, SOC detection is weak.

**Open**

Win + R → gpedit.msc

**Go to**

Computer Configuration

└ Administrative Templates

└ System

└ Audit Process Creation

**Enable**

**Include command line in process creation events**

➡ **Enabled**

This affects **Event ID 4688**.

1. **Account Logon (client-side auth)**

**Path**

Account Logon

Enable:

* **Audit Credential Validation** → Success + Failure

This gives:

* 4776 (NTLM)  
  Useful for lateral movement detection.

**5️⃣ Local Account Management (ONLY local)**

This is the only Account Management setting you need on Windows 10.

**Path**

Account Management

Enable:

* **Audit User Account Management** → Success + Failure

This gives:

* 4720, 4722, 4726  
  (Local user abuse, local admin tricks)

❌ Do NOT enable group management here  
❌ Do NOT enable directory service access

**Apply & verify (don’t trust the GUI)**

**Run as Administrator**

**gpupdate /force**

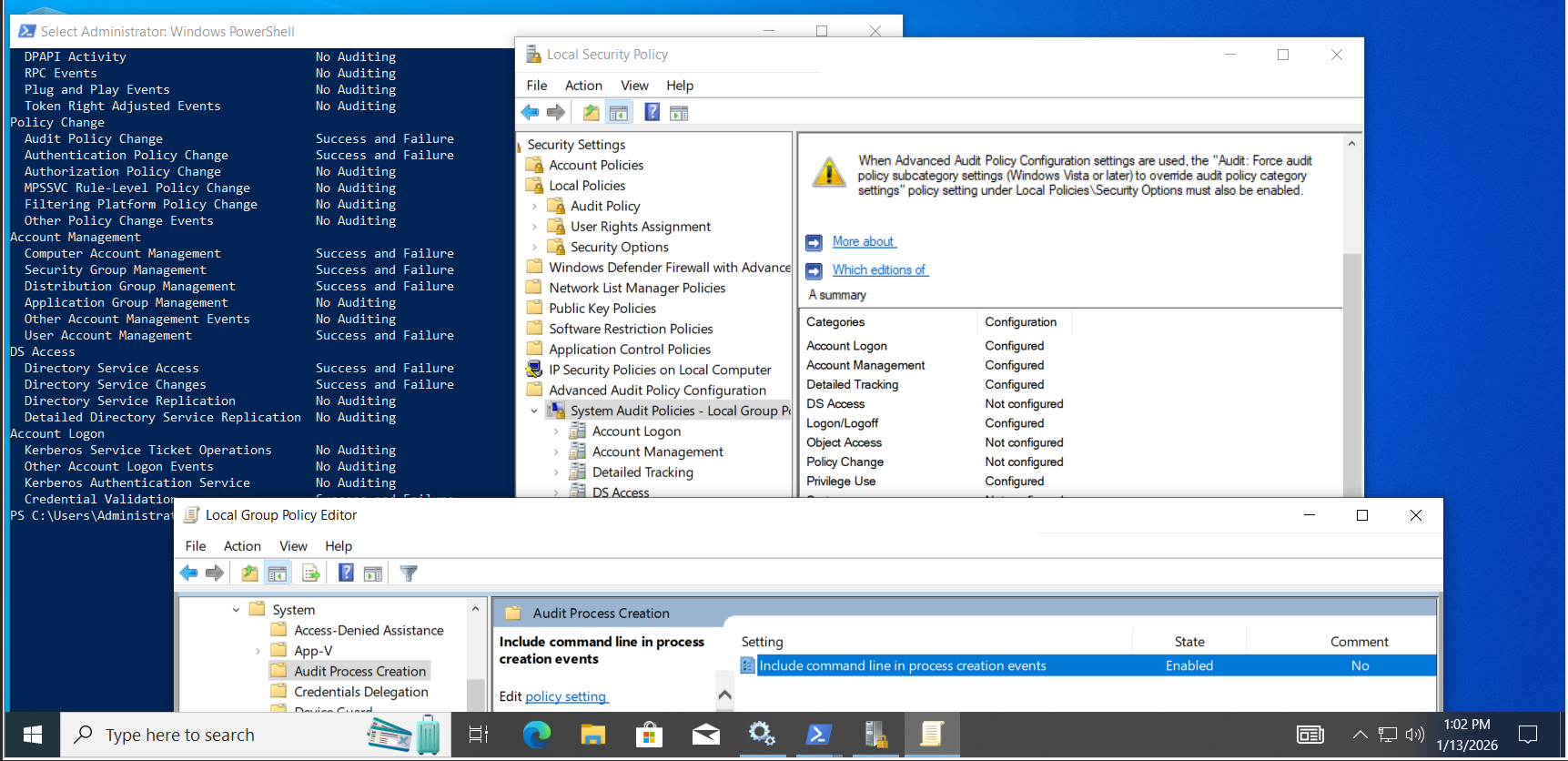
**Verify audit state**

**auditpol /get /category:\***

You should see:

* Logon/Logoff → Enabled
* Account Logon → Enabled
* Detailed Tracking → Process Creation Enabled
* Account Management → User Account Management Enabled
* Policy Change → Enabled

**WINDOWS 10 PRO**



**🔁 Mandatory Reboot (BOTH SYSTEMS)**

After configuring Steps 3–5:  
👉 **Reboot Windows 10 PRO and the Domain Controller**

Reason:

* Audit policies are cached
* Skipping reboot causes missing events

**🔴 6️⃣ Security Log Size & Retention**

**Both systems**

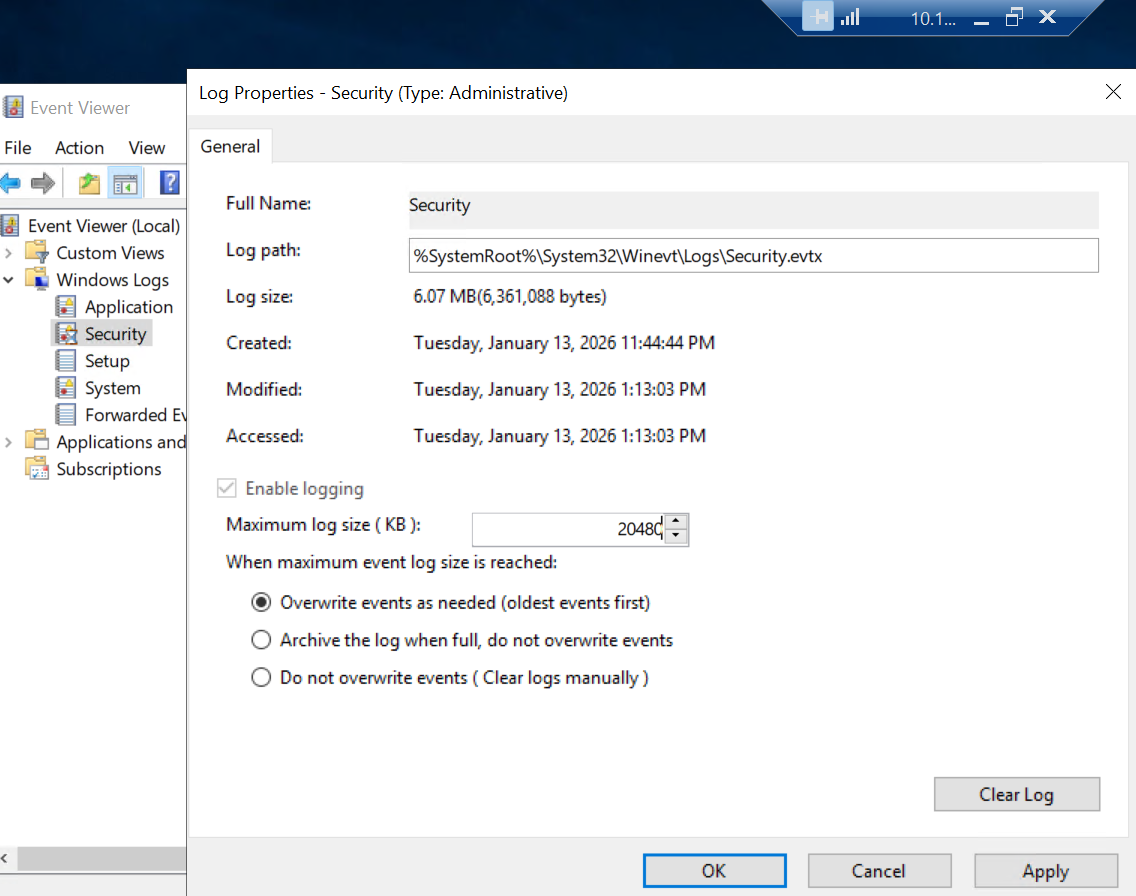
GUI:

1. Event Viewer → Right-click **Security**
2. Properties

Set:

* Log size: **≥ 200 MB**
* Retention: **Overwrite as needed (lab)**

Small logs = missing early attack stages.



**🔴 7️⃣ Time & Time Zone**

**Domain Controller (Windows Server 2019)**

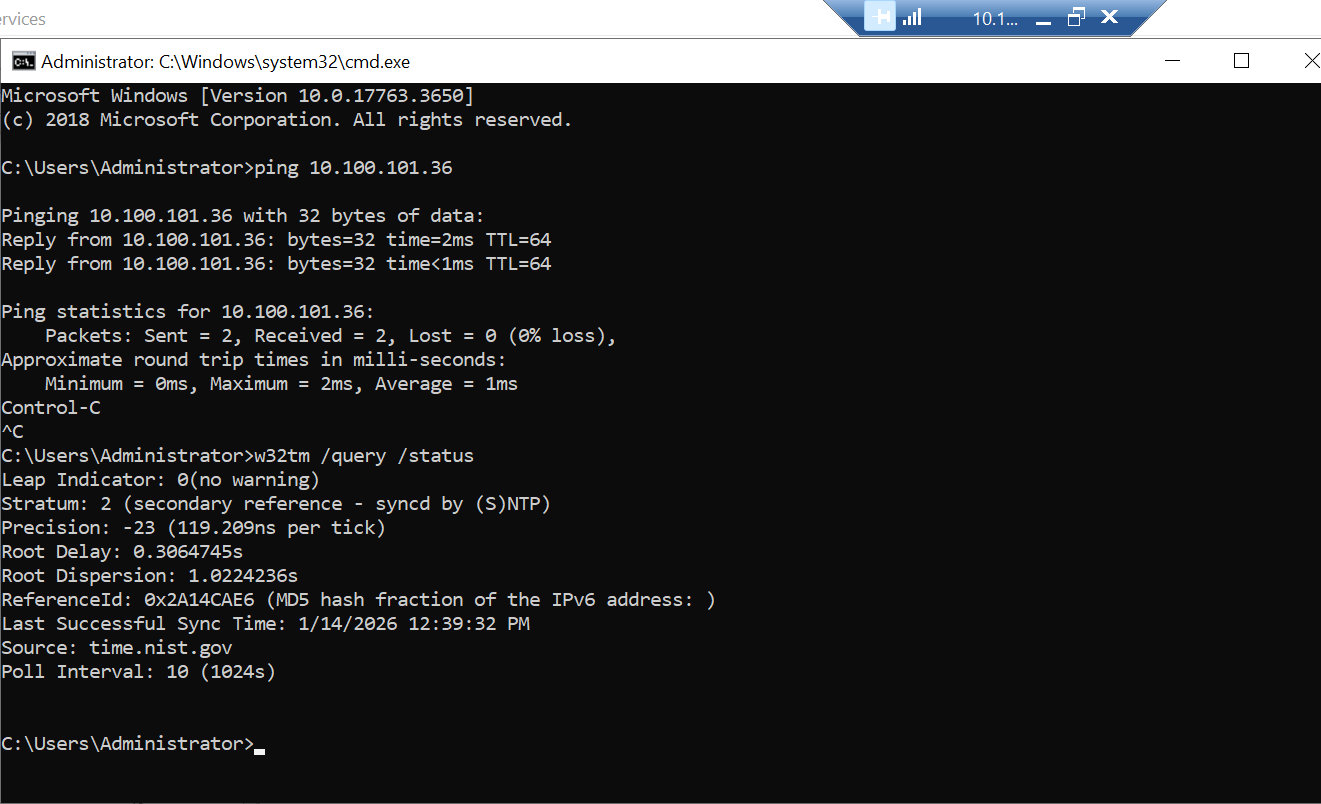
* DC acts as **authoritative time source**
* Windows Time service (w32time) must be running

Verify:

w32tm /query /status

You should see:

* Source: Local CMOS Clock **or** configured NTP
* Stratum present



(Optional but clean lab practice)

w32tm /config /syncfromflags:domhier /update

net stop w32time

net start w32time

**Windows 10 Pro (Domain-Joined)**

* Must sync time from **Domain Controller**, not internet

Verify:

w32tm /query /source

Expected:

<your-DC-hostname>

❌ If it shows:

time.windows.com

→ misconfigured for AD.

Force resync:

**w32tm /resync**

**🔴 8️⃣ Firewall (Outbound)**

**Both systems**

GUI:

1. Windows Defender Firewall
2. Advanced settings → Outbound Rules

Ensure:

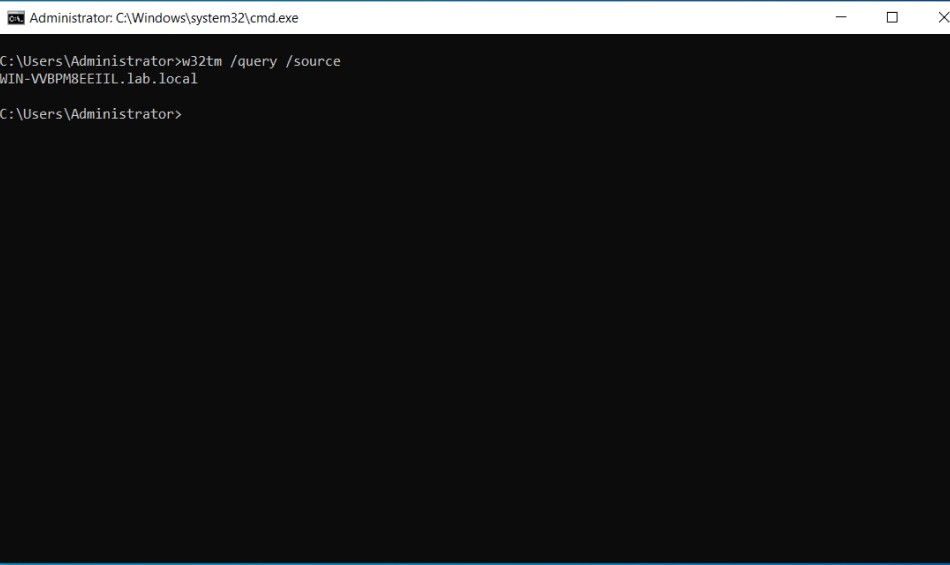
* TCP **9997** is not blocked
* Optional explicit allow rule

Firewall rule added

**🔴 9️⃣ Permissions**

Ensure:

* You are **Local Administrator**
* Splunk Forwarder runs as **Local System**
* Local System can read **Security** logs

****

**✅ WINDOWS + AD PRE-CHECK CHECKLIST**

All must be **YES**:

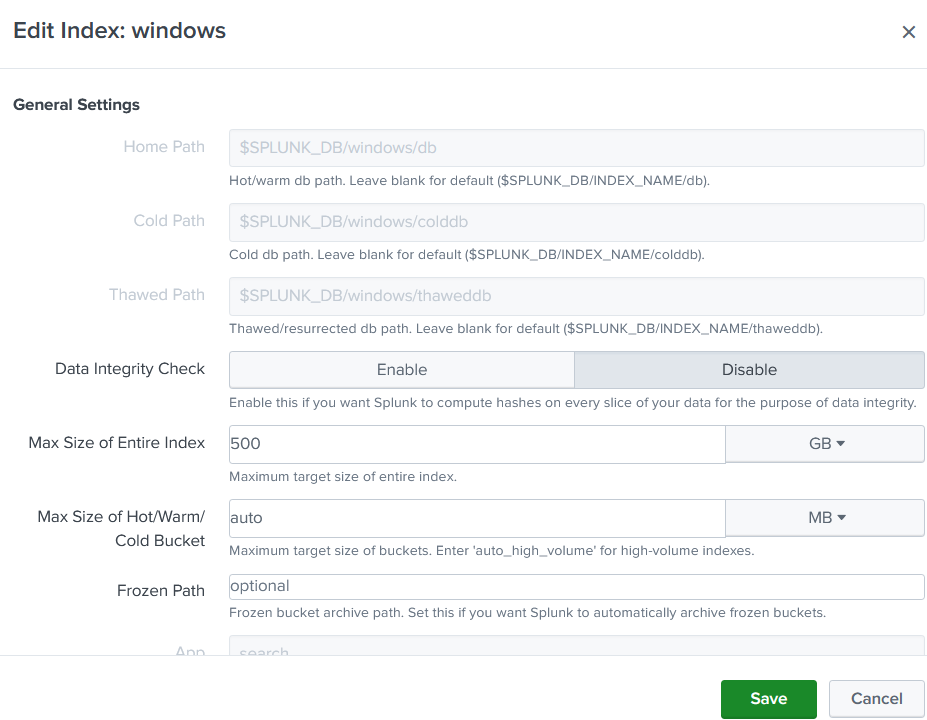
☑ Windows Event Log running  
☑ Security log populated & updating  
☑ Audit Credential Validation enabled  
☑ Audit Logon / Logoff enabled  
☑ Privilege Use auditing enabled  
☑ Process Creation auditing enabled  
☑ Security log size increased  
☑ Correct time & timezone  
☑ Firewall allows outbound 9997  
☑ Admin permissions confirmed

**PART 3 — SPLUNK-SIDE PREPARATION**

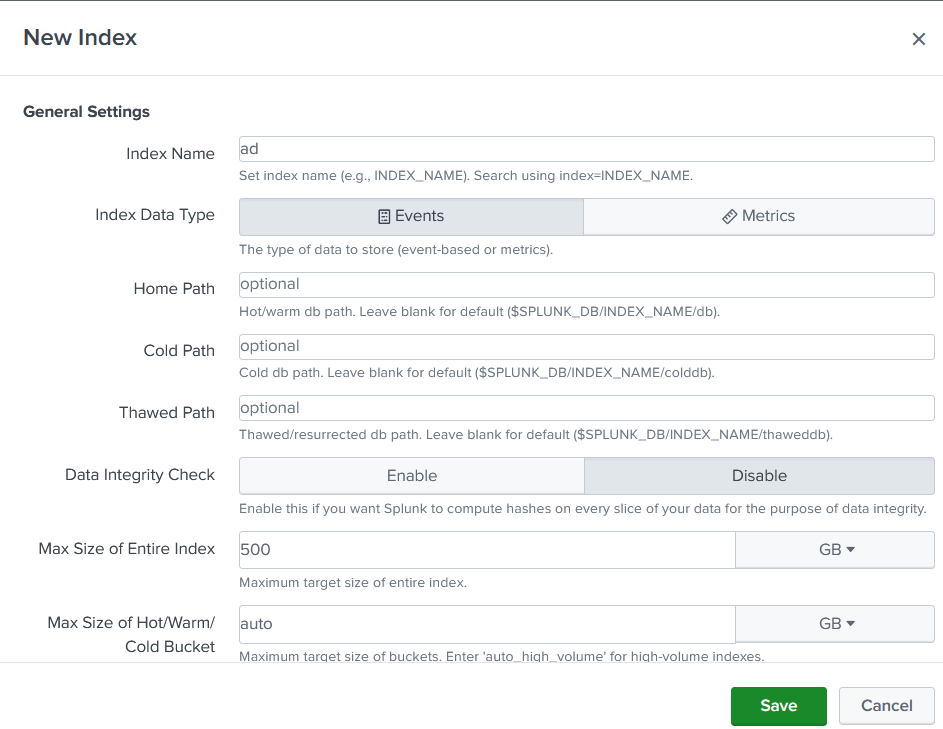
**🔴 STEP 1 — INDEXES (MANDATORY)**

Create:

* windows → Windows 10 PRO logs



* ad → Domain Controller logs



🚫 Using main = beginner mistake.

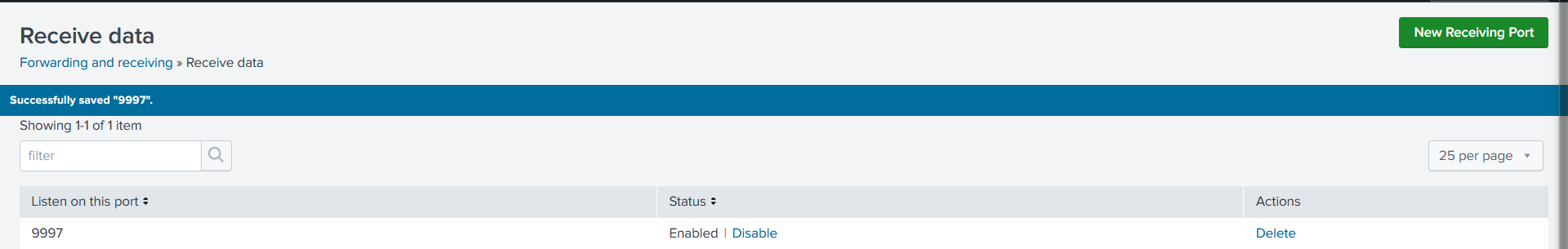
**🔴 STEP 2 — ENABLE RECEIVING**

Splunk Web:

Settings → Forwarding and Receiving

→ Configure receiving

→ Port 9997



**PART 4 — SPLUNK FORWARDER (BOTH SYSTEMS)**

Install **Splunk Universal Forwarder** on:

* **Windows 10 PRO**

# 🧭 GOAL (BE CLEAR)

You want:

Windows 10 Pro

↓

Splunk Universal Forwarder

↓ (TCP 9997)

Splunk Enterprise (10.100.101.36)

↓

Indexes: windows / sysmon (later)

If this pipeline works, **everything else builds on it**

## 🔴 PRE-CHECKS (DON’T SKIP)

### On Splunk Server (Ubuntu)

Confirm receiver is listening:

ss -tulnp | grep 9997

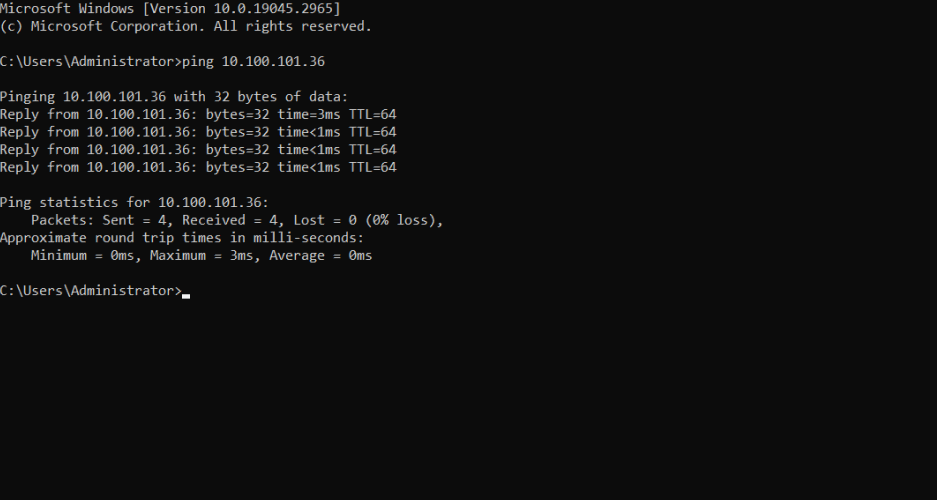
You **must** see:

LISTEN ... :9997 splunkd

If not → stop. Don’t continue.

Confirm:

* Windows 10 **Pro** (not Home)
* Admin access
* Can ping Splunk server:



## 1️⃣ DOWNLOAD SPLUNK UNIVERSAL FORWARDER (WINDOWS)

On **Windows 10 Pro**, open browser and download:

* **Splunk Universal Forwarder for Windows (64-bit)**
* Version should match Splunk Enterprise (10.x is fine)

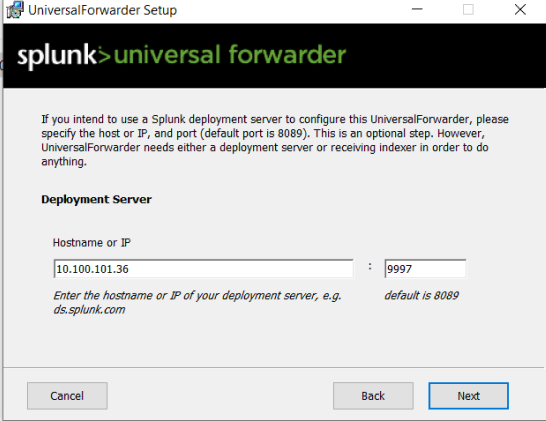
👉 Use **MSI installer**, not ZIP.

**2️⃣ INSTALL UNIVERSAL FORWARDER (IMPORTANT OPTIONS)**

Run the .msi **as Administrator**

**Installation steps (DO EXACTLY THIS):**

1. **Accept License** ✅
2. **Installation type**
   * Choose: **On-premises Splunk Enterprise instance**
3. **Deployment Server**
   * ❌ Leave BLANK (you don’t have one yet)
4. **Receiving Indexer**
   * IP:
   * 10.100.101.36
   * Port:
   * 9997



1. **Finish installation**

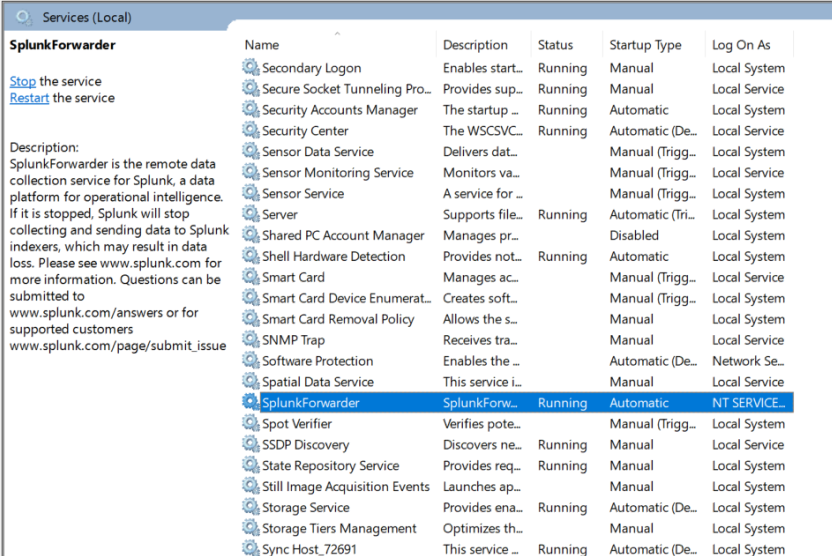
🚫 **Common mistake:**  
Do NOT configure TCP inputs here.  
Do NOT select “Light Forwarder” (deprecated).

**3️⃣ VERIFY FORWARDER SERVICE (WINDOWS)**

Open **Services** (services.msc)

Check:

* **SplunkForwarder**
  + Status: **Running**
  + Startup type: **Automatic**



**4️⃣ CONFIGURE OUTPUTS.CONF (VERIFY, DON’T TRUST GUI)**

Go to:

**C:\Program Files\SplunkUniversalForwarder\etc\system\local\**

Open (or create) **outputs.conf**

**It MUST look like this:**

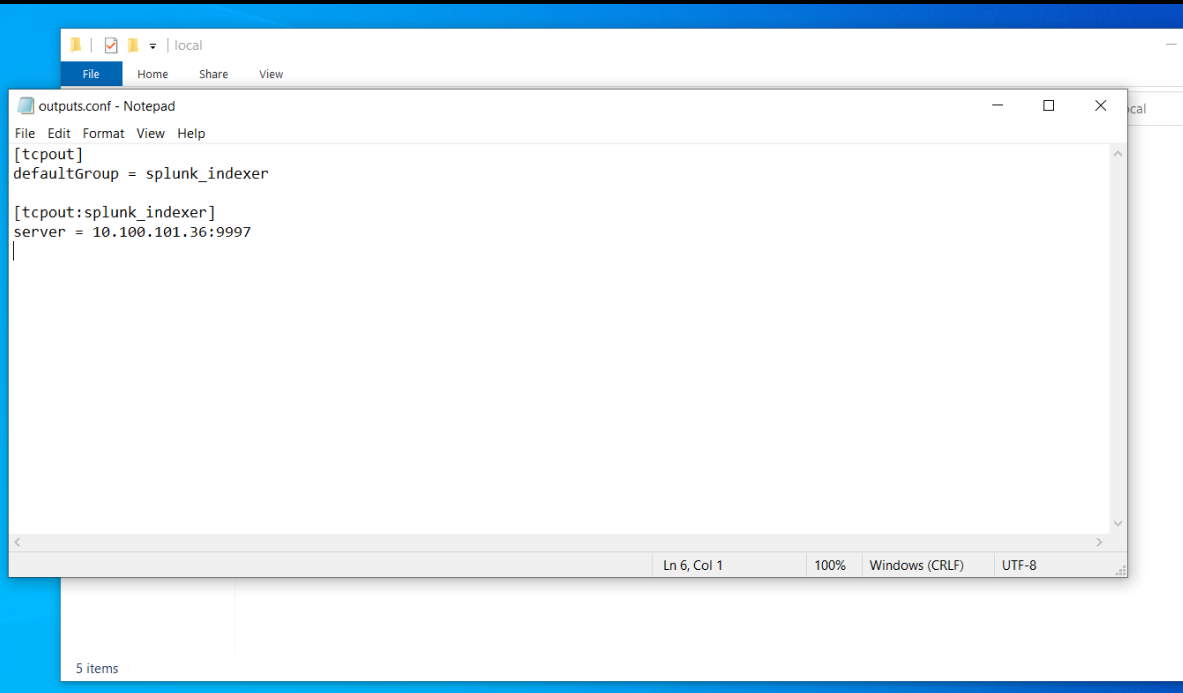
**[tcpout]**

**defaultGroup = splunk\_indexer**

**[tcpout:splunk\_indexer]**

**server = 10.100.101.36:9997**

Save.



Restart forwarder:

Restart-Service SplunkForwarder

**5️⃣ ENABLE WINDOWS EVENT LOGS (THIS IS THE POINT)**

Now we tell the forwarder **what to send**.

Create:

**C:\Program Files\SplunkUniversalForwarder\etc\system\local\inputs.conf**

**Minimal but correct inputs.conf**

**[WinEventLog:Security]**

**index = windows**

**disabled = 0**

**[WinEventLog:System]**

**index = windows**

**disabled = 0**

**[WinEventLog:Application]**

**index = windows**

**disabled = 0**



Restart again:

Restart-Service SplunkForwarder

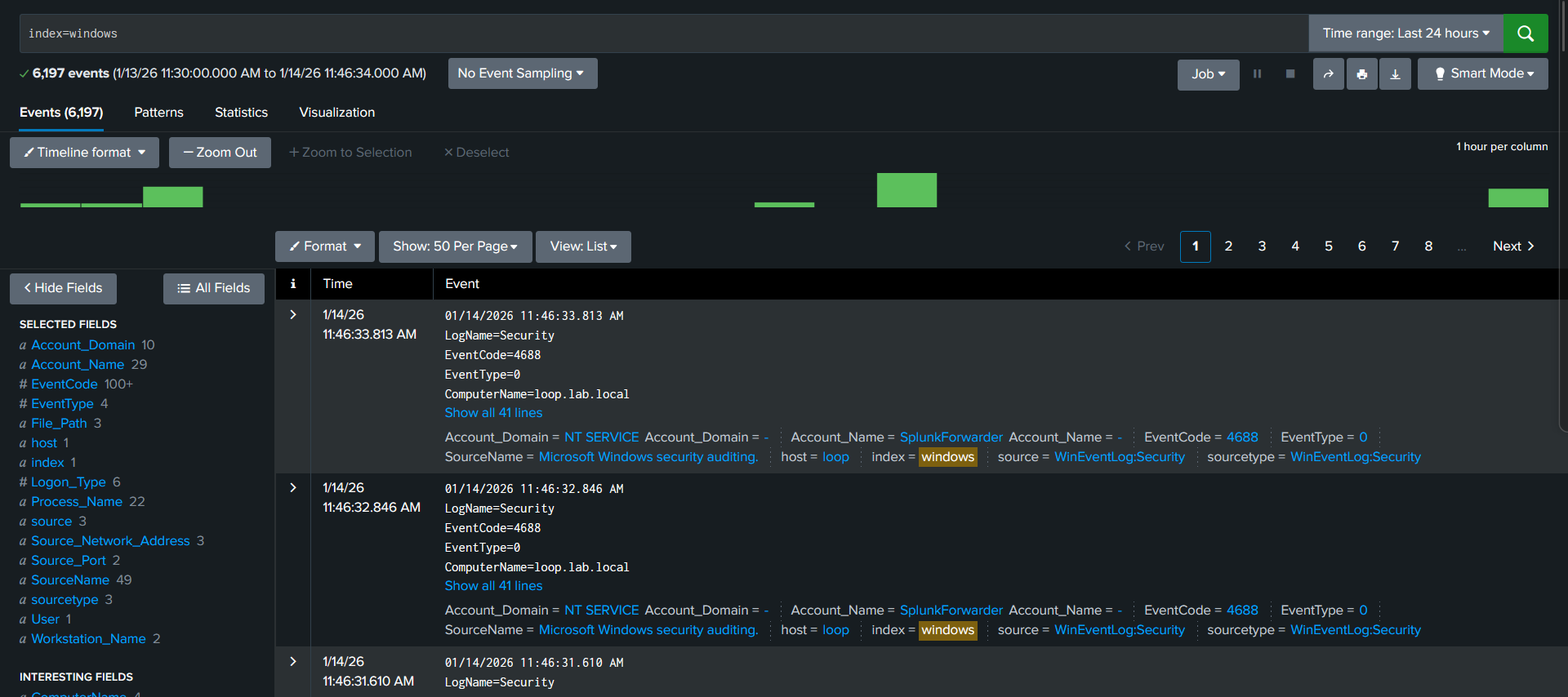
**6️⃣ VERIFY ON SPLUNK SERVER (NON-NEGOTIABLE)**

In Splunk Search (Web UI):

**Basic check**

index=windows

You **must** see events.



**Windows Server 2019 (DC)**

**🧭 GOAL (BE VERY CLEAR)**

You want this pipeline:

Windows Server 2019 (Domain Controller)

↓

Splunk Universal Forwarder

↓ TCP 9997

Splunk Enterprise (10.100.101.36)

↓

Index: ad (NOT windows)

If DC logs go into windows → **wrong architecture**.  
DC logs **must be isolated**.

**🔴 PRE-CHECKS (DON’T PROCEED IF ANY FAIL)**

**On Splunk Server**

Receiver must already be listening:

ss -tulnp | grep 9997

You already confirmed this earlier — good.

**On Windows Server 2019 (DC)**

Confirm:

* Role installed: **Active Directory Domain Services**
* Server is **promoted** to DC
* You have **Domain Admin** access
* DC can reach Splunk server:

**ping 10.100.101.36**

If ping fails → fix network/firewall first.

**1️⃣ INSTALL SPLUNK UNIVERSAL FORWARDER (DC)**

On **Windows Server 2019**, download:

* **Splunk Universal Forwarder – Windows 64-bit (MSI)**

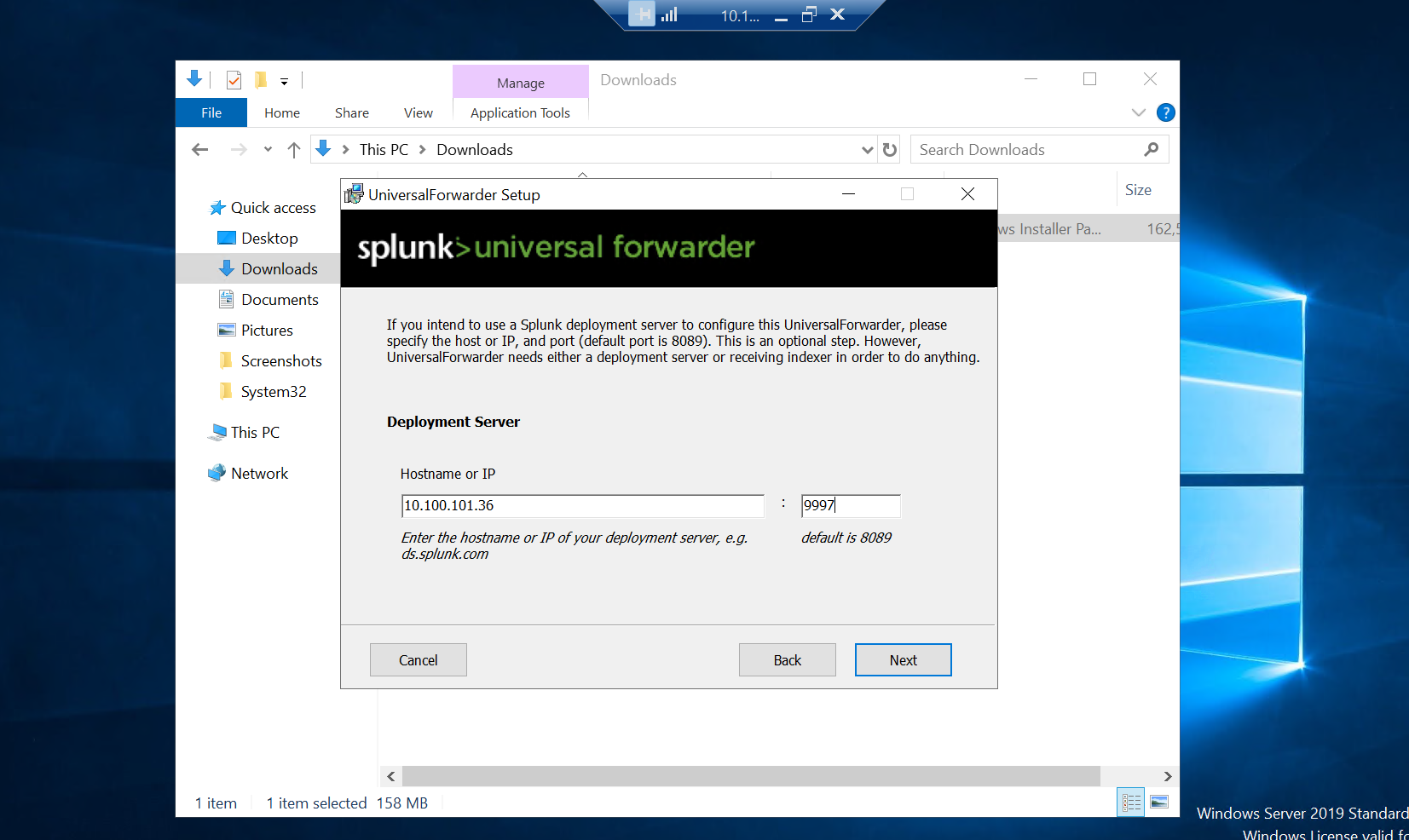
🚨 **IMPORTANT RULE**

Install UF on DC **ONLY AFTER** promotion to DC  
Never before

**Installation steps (DO EXACTLY THIS)**

Run installer **as Administrator**.

1. **Accept license**
2. Installation type:
   * ✅ **On-premises Splunk Enterprise**
3. **Deployment server**
   * ❌ Leave blank
4. **Receiving Indexer**
   * **IP:**
   * **10.100.101.36**
   * **Port:**
   * **9997**



1. Finish installation

**2️⃣ VERIFY FORWARDER SERVICE (DC)**

Open **Services** (services.msc)

Check:

* **SplunkForwarder**
  + Status: Running
  + Startup: Automatic

**3️⃣ CONFIGURE OUTPUTS.CONF (VERIFY, NOT OPTIONAL)**

Go to:

**C:\Program Files\SplunkUniversalForwarder\etc\system\local\**

Open **outputs.conf**

**It MUST look like this:**

[tcpout]

defaultGroup = splunk\_indexer

[tcpout:splunk\_indexer]

server = 10.100.101.36:9997

Restart:

Restart-Service SplunkForwarder

**4️⃣ CONFIGURE DC EVENT LOG COLLECTION (CRITICAL PART)**

Create:

C:\Program Files\SplunkUniversalForwarder\etc\system\local\inputs.conf

**CORRECT DC inputs.conf**

**[WinEventLog:Security]**

**index = ad**

**disabled = 0**

**[WinEventLog:System]**

**index = ad**

**disabled = 0**

**[WinEventLog:Directory Service]**

**index = ad**

**disabled = 0**

**[WinEventLog:DNS Server]**

**index = ad**

**disabled = 0**

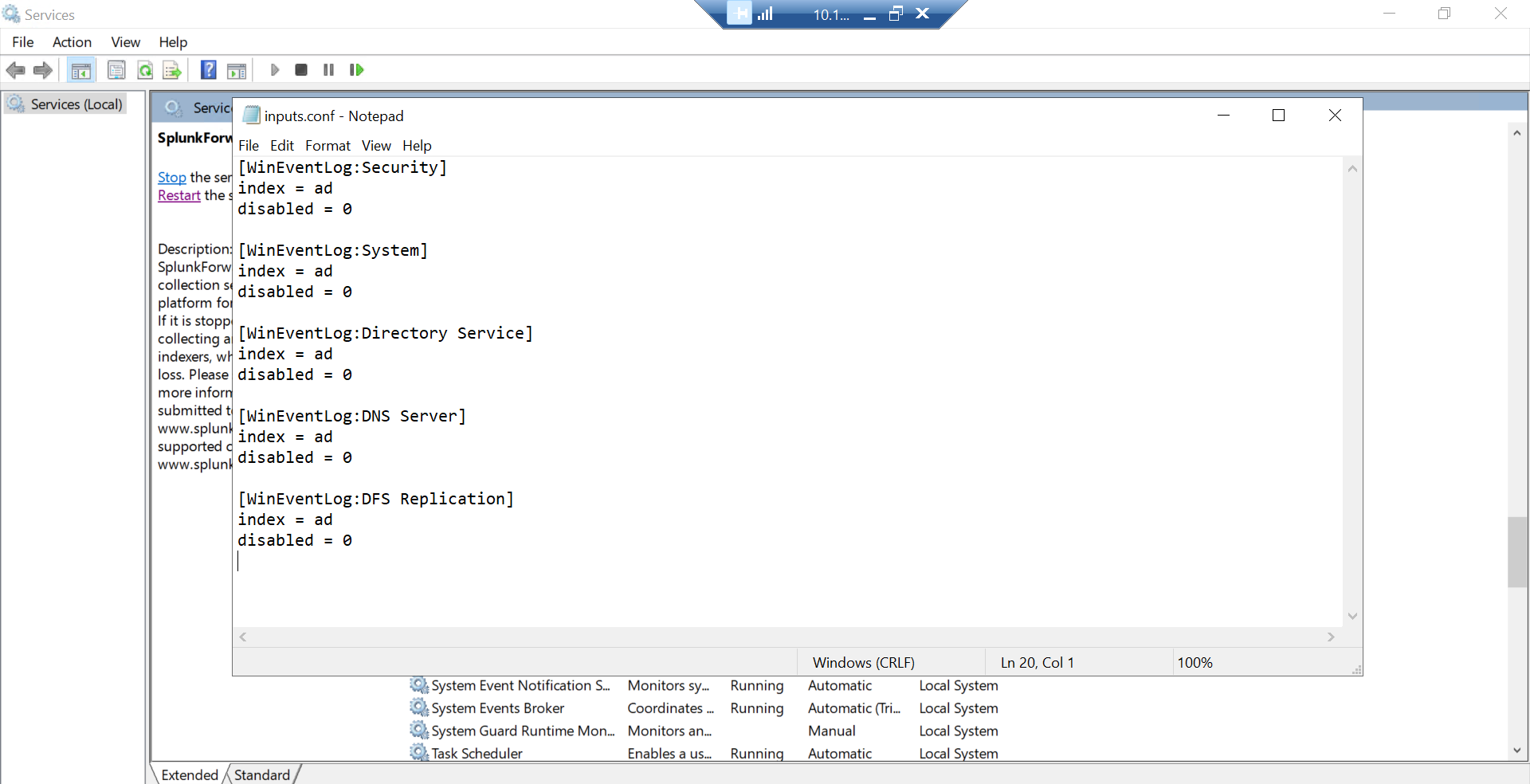
**[WinEventLog:DFS Replication]**

**index = ad**

**disabled = 0**

🚨 **WHY THESE MATTER**

* Security → authentication, Kerberos, NTLM
* Directory Service → AD object changes
* DFS Replication → SYSVOL abuse detection



Restart:

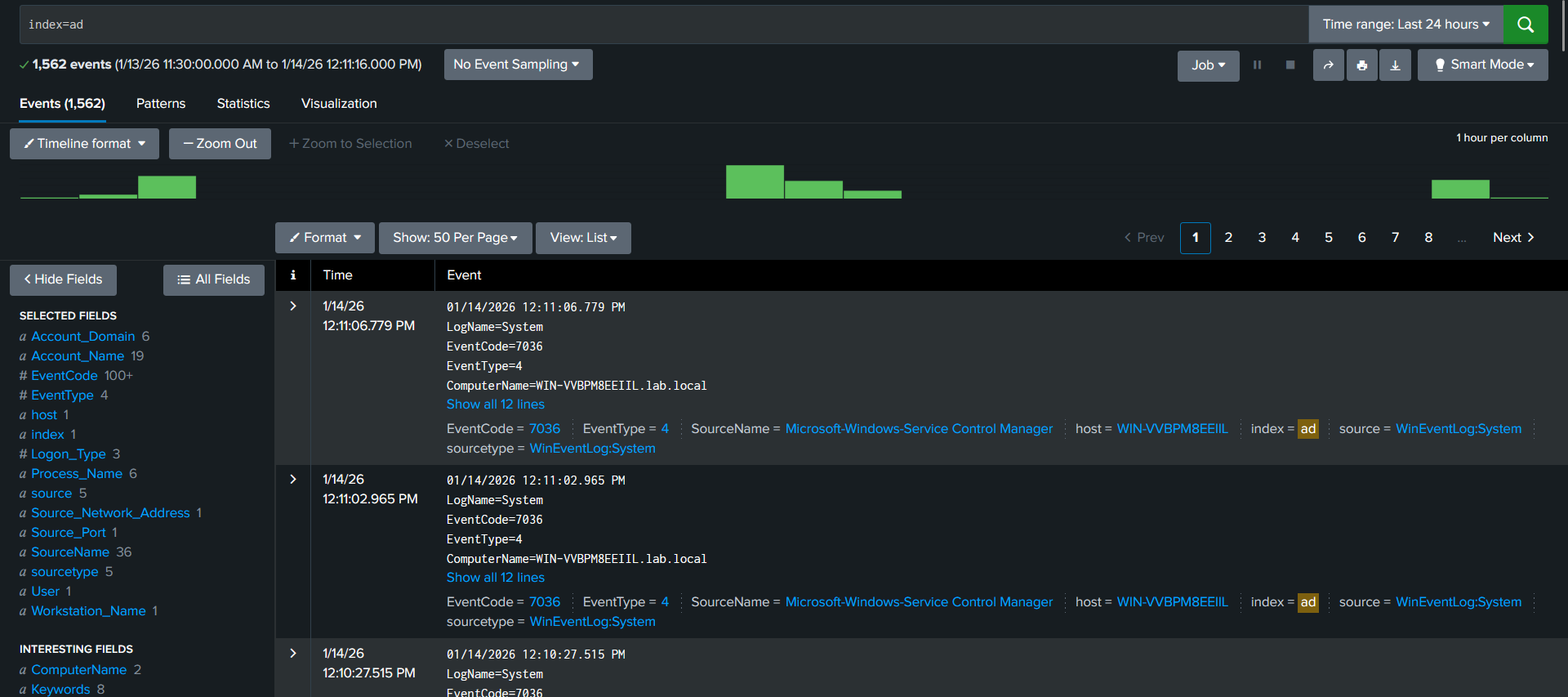
Restart-Service SplunkForwarder

**5️⃣ VERIFY ON SPLUNK (NON-NEGOTIABLE)**

**Basic check**

**index=ad**

If nothing appears → stop and troubleshoot



**PART 5 — VALIDATION (PHASE 0)**

Search:

**index=windows**

**index=ad**

Verify:

* **sourcetype = WinEventLog:Security**
* One Windows event = one Splunk event
* **\_time** matches Event Viewer time

**✅ FINAL RESULT**

You now have:

* Endpoint authentication attempts
* AD credential validation truth
* Privilege & execution visibility
* Clean Splunk ingestion
* SOC-ready identity telemetry