* **Windows Tools for Debugging**

These tools are like magnifying glasses for your computer. They help you find problems or understand what’s going on behind the scenes.

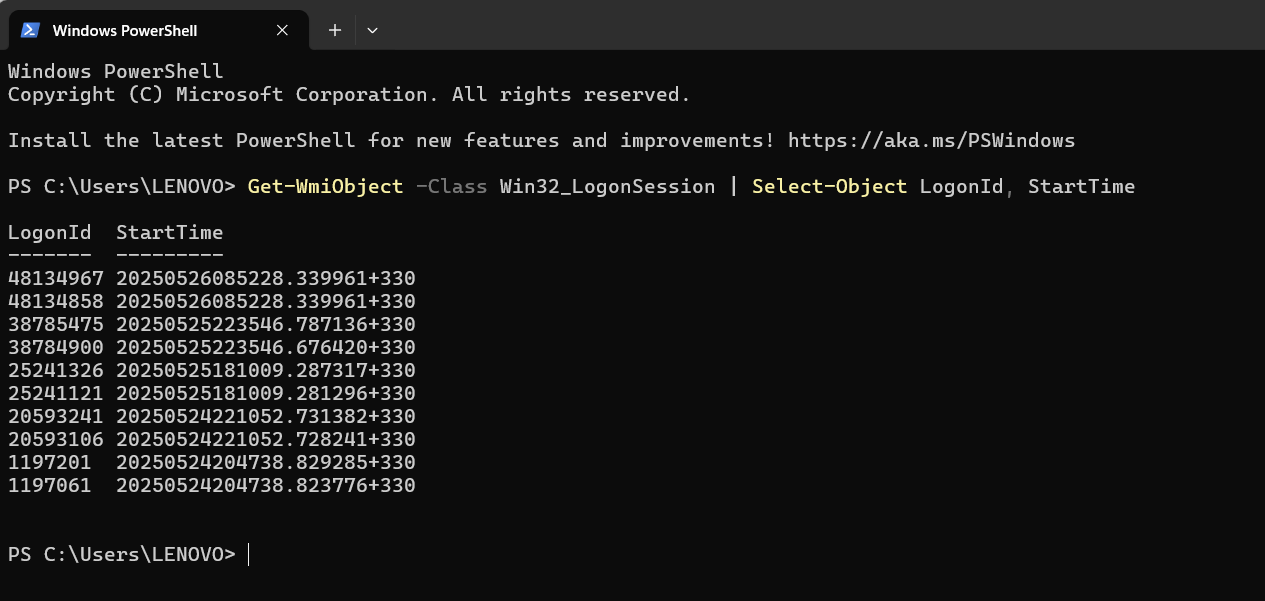
* **LogonSessions**

This tool shows **who is logged into the computer** right now.

It shows **all active user accounts** and **how long they’ve been logged in**.

You can use it to see if a hacker is logged in, or just to check who’s using the computer.

Think of it like a list of names of people in the room with their log-in time.



* **Autologon**

Normally, when Windows starts, you have to **type a password to log in**.

**Autologon** lets you **skip the password screen** and **log in automatically** with a saved username and password.

This is helpful if you don’t want to type a password every time.

Be careful: this makes the computer **less secure** since anyone can access it.

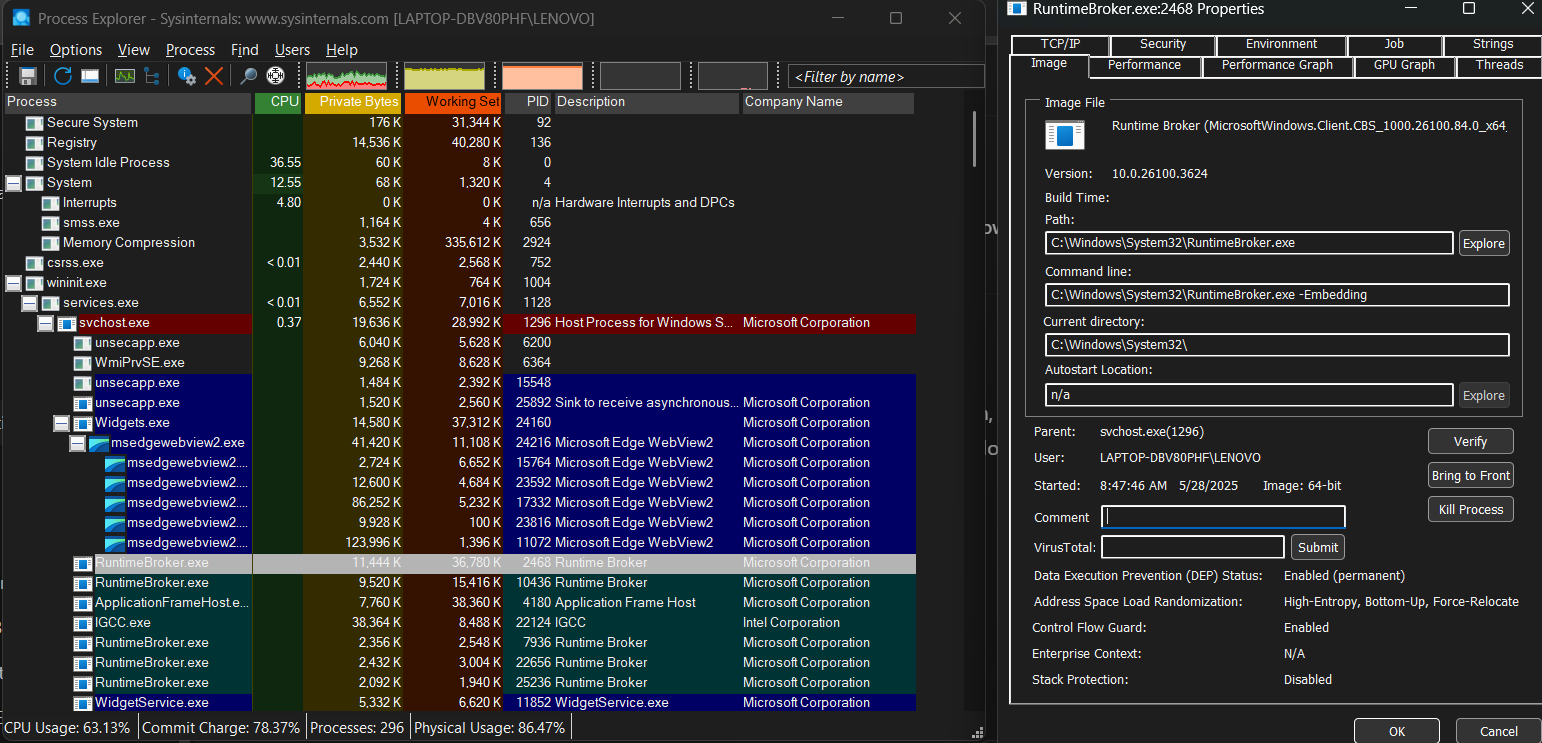
* **Process Explorer**

This tool shows **all the programs and processes running on the computer**.

It tells you how much CPU, memory, and other resources each program is using.

You can **see which program is causing problems** or **which one is using too much power**.

It’s like **Task Manager** but with **more details and power**.



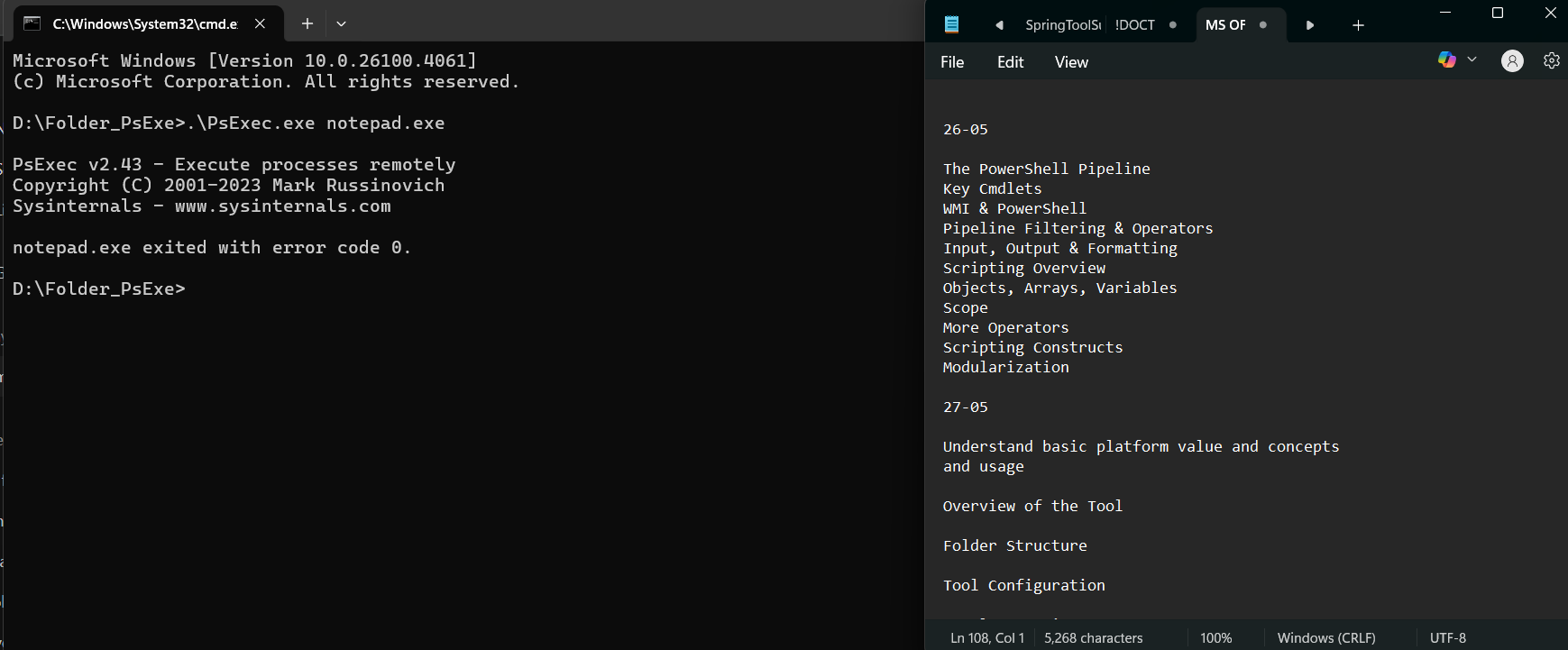
* **PsExec**

This tool lets you **run a program on another computer** (for example, across a network).

You don’t have to be sitting in front of that computer.

It’s helpful if you’re an IT person managing many computers.

Example: You can run updates or fixes on someone’s computer while sitting at your desk.

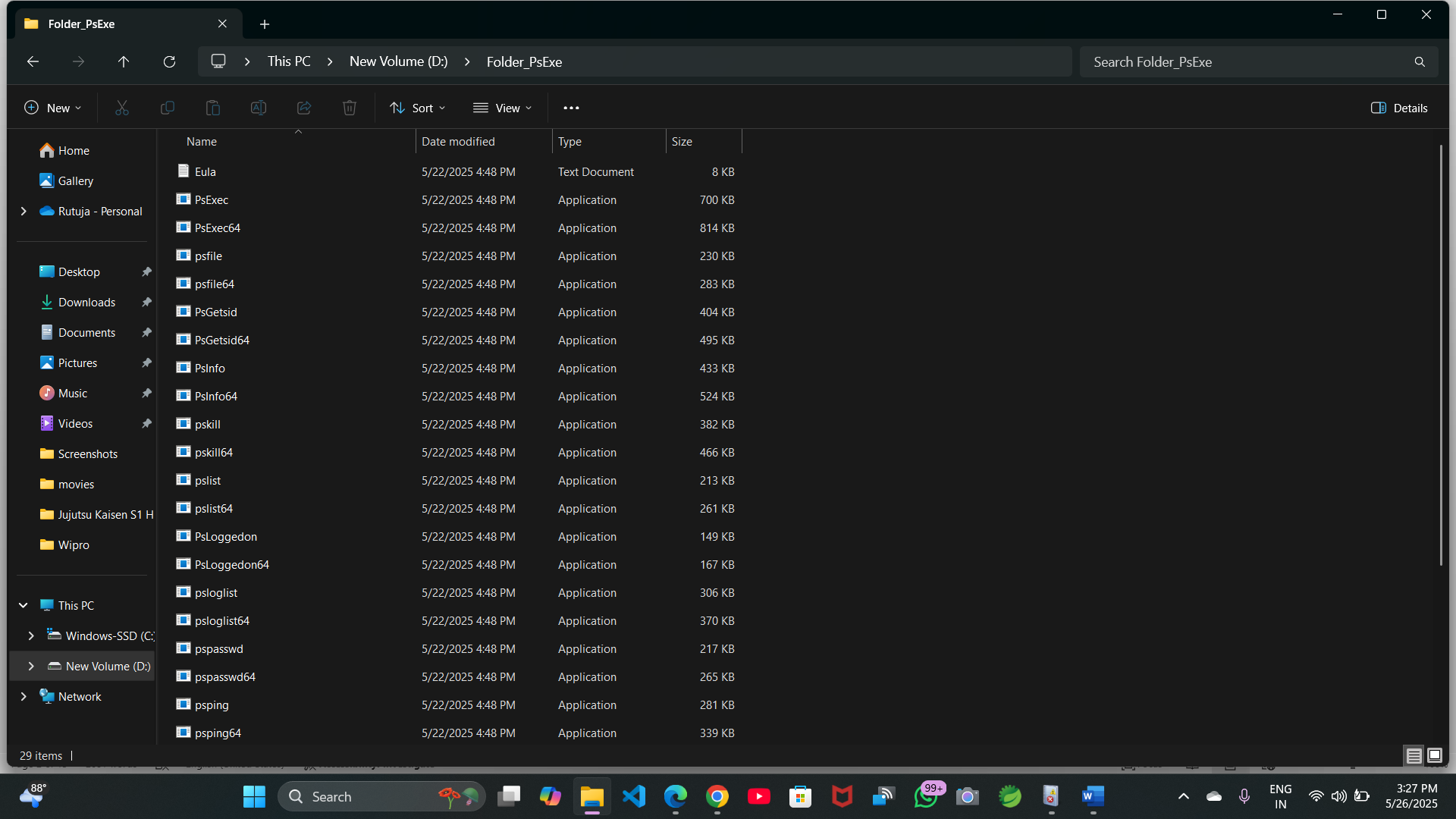


* **PSTools**

**PSTools** is a collection of small tools for **managing Windows computers remotely**.

It includes tools like **PsExec**, **PsList** (shows running processes), and **PsKill** (kills a process).

Think of it as a **Swiss Army knife for Windows remote management**.



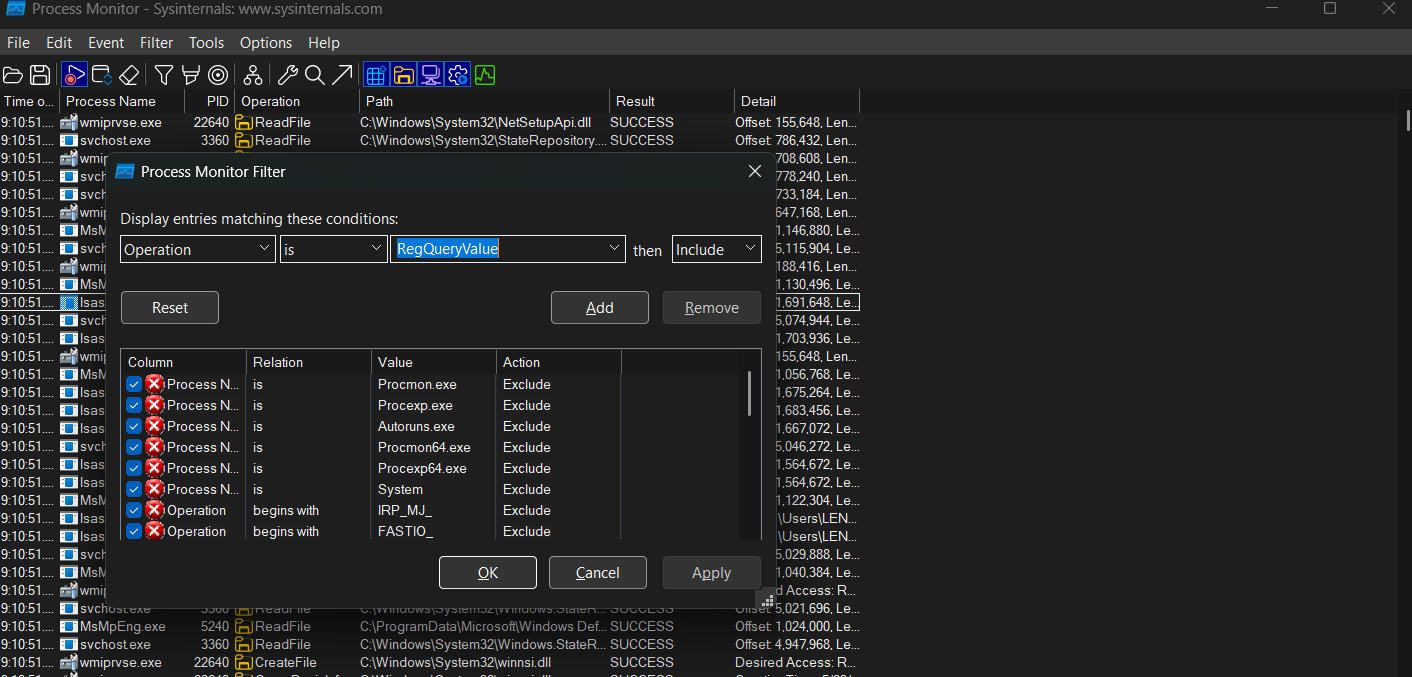
* **RegMon**

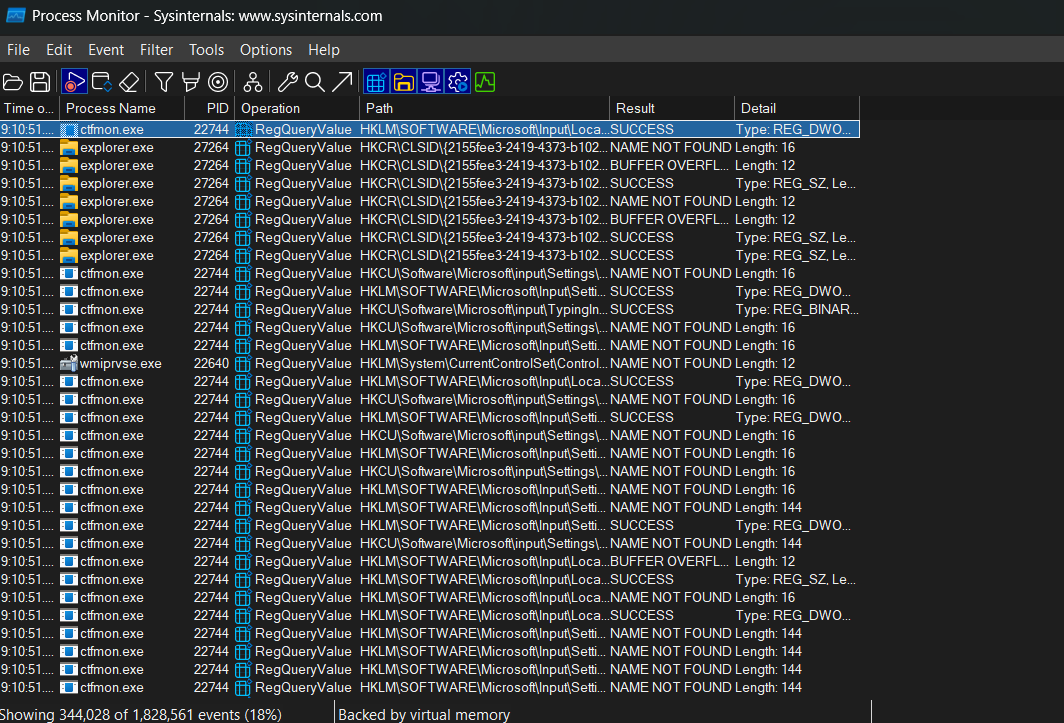
The **registry** is like a **big database of settings** that Windows uses to run the system and apps.

**RegMon** watches what’s happening in the registry **in real time**.

It shows **which programs are changing which settings**.

Useful for debugging when something’s going wrong because of a bad registry change.



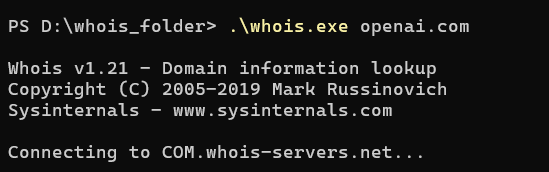


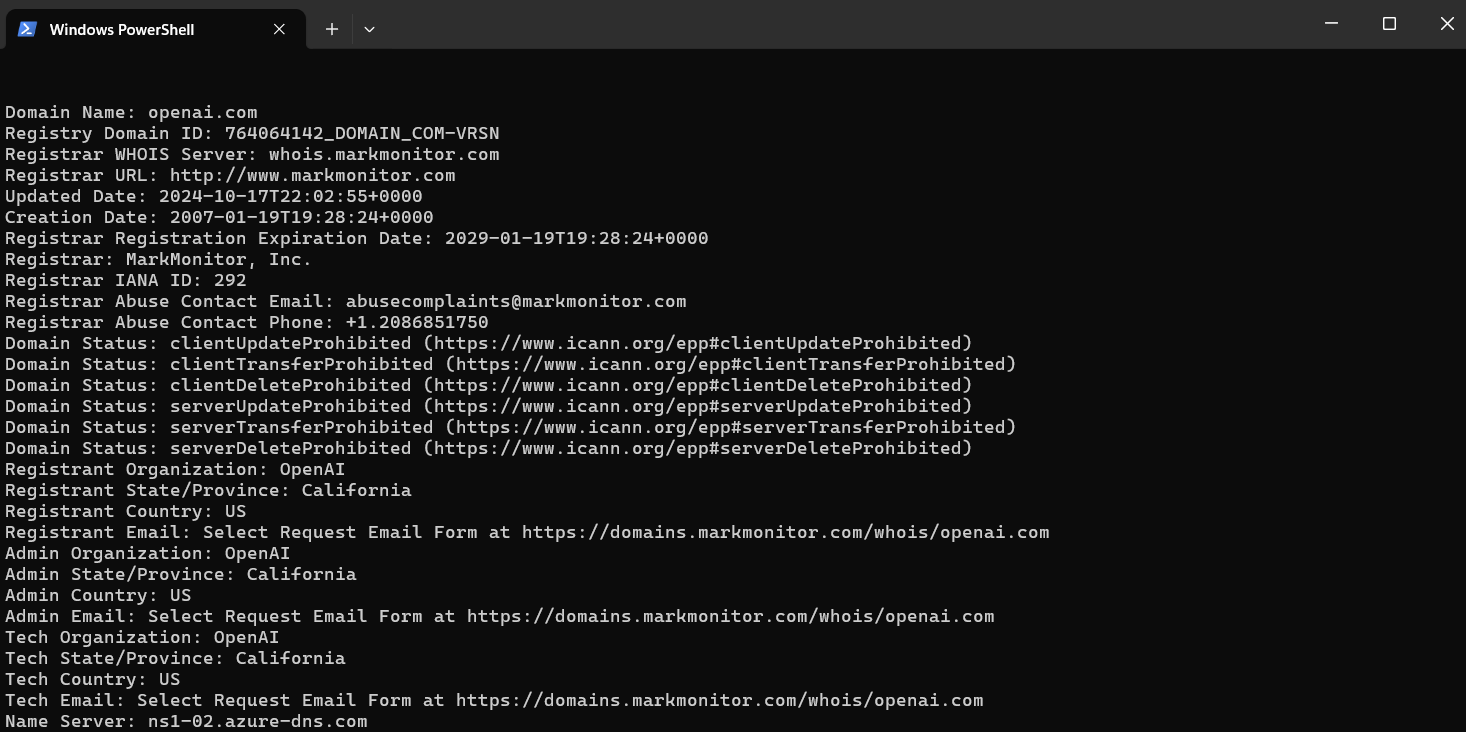
* **Whois**

If you know a **website or IP address** and you want to know **who owns it**, use **Whois**.

It’s like a phonebook lookup for websites.

It gives you **contact information** for the domain owner, which is useful for network troubleshooting or reporting suspicious websites.





* **SysMon**

**SysMon** logs important details about what’s happening on the computer, such as:

* + Programs starting and stopping
  + Network connections
  + File changes

It’s useful for **tracking down malware** or **understanding strange behavior**.

It logs **everything in the background**, so you can look back and see what happened.

* **Why Use These Tools?**

They help you **solve computer problems**.

They give you **extra information** that normal Windows tools don’t show.

They are used by **IT professionals**, **security experts**, and **power users**.

* **Vulnerability Applications Clean Up from a Lockdown Environment**
* **What does this mean?**

Suppose you have a **company computer** (or a group of computers) that’s supposed to be **secure**.  
When we say **lockdown environment**, we mean:

The computer is super-secure (e.g., limited user permissions, strict rules).

You don’t want people to install apps, change settings, or do anything risky.

But sometimes, **extra programs (applications)** sneak in:

They might have **security holes (vulnerabilities)** that hackers could use.

Or they might be **old versions** that aren’t safe.

So, we need to **clean them up**, which means **finding them and removing them**.

* **Removing Bulk Versions of Apps via Scripts**

**Bulk versions** mean there might be **many copies** of apps (like if the same risky app is installed on multiple computers).

Instead of manually uninstalling each app one by one (which takes forever), we can use a **script**.

A **script** is like a **set of instructions** for the computer.  
It tells the computer:

“Find App A, App B, App C on this computer.”

“If you see them, uninstall them.”

“Do this on every computer on the network.”

The **script does the work for you automatically**.

* **Simple Example**

Imagine you have a lot of laptops in an office, and you know there’s an old version of "Zoom" and "Java" on them. These old versions are **not secure**.

You can:  
Write a script to find any old Zoom or Java apps.  
The script will automatically uninstall them.  
Run this script on every laptop.

This way, you **clean up the risky apps** from all the computers without doing it manually.

* **Why is this important?**

**Old apps** can be dangerous because hackers can use them to break into the system.

**Lockdown environments** need to stay secure. Removing these apps **reduces risks**.

Scripts save **time and effort** when you have lots of computers.

* **So, in simple terms:**

**Lockdown environment** = Super secure computer setup.

**Vulnerability apps** = Apps that might make the computer unsafe.

**Clean up** = Get rid of them.

**Scripts** = Instructions that do this automatically, for lots of computers at once.

* **Complex Applications: Office 365 and SAP**

These are **big, powerful software programs** used by many businesses. They can do a lot of things, but that also makes them **complex**.

* **Office 365 (Now called Microsoft 365)**

**What is it?**  
Office 365 is a **set of Microsoft tools** for **email, documents, spreadsheets, collaboration, and communication**.

**What’s included?**

**Word** – for writing documents

**Excel** – for spreadsheets and calculations

**PowerPoint** – for presentations

**Outlook** – for email

**Teams** – for chatting and video calls

**OneDrive** – for saving files in the cloud

**SharePoint** – for team file sharing

**Why is it complex?**

It’s not just one program – it’s a **collection of apps** that work together.

It’s available **online and on desktop** – your files can sync between them.

Businesses use it for **collaboration** (people working together on projects).

You can **manage security, permissions, and sharing settings**.

* **SAP**

**What is it?**  
SAP is a **huge business software** that helps companies manage things like:

**Finances** – accounting, budgets, payments

**Inventory** – tracking products and supplies

**Sales** – managing customer orders and invoices

**HR** – employee records, salaries, time tracking

**Why is it complex?**

It’s like the **brain of a big company** – it connects all departments.

It manages **huge amounts of data** and complex workflows.

It requires **special training** because it has many features.

Every company can **customize SAP** for its needs.

* **Key Differences:**

| **Feature** | **Office 365** | **SAP** |
| --- | --- | --- |
| Main Use | Communication and collaboration | Business operations and management |
| Examples | Word, Excel, Teams | Finance, HR, Inventory, Sales |
| Complexity | Medium (lots of apps, sync) | High (connects entire company systems) |
| Users | Almost everyone in an office | Usually, IT, finance, HR, and operations staff |

* **Simple Summary:**

**Office 365** is like your **everyday tools** for doing work—writing, emailing, calling.

**SAP** is like a **giant system** that helps a company run its business smoothly.

* **Task Sequence to Install Multiple Applications with Multiple Reboots**

**What does it mean?**

When you need to **install a bunch of applications** (like Office, Chrome, SAP) on a computer, and some of them need the computer to **reboot** (restart) in between, you can **automate the process**. This is called a **task sequence**.

* **What is a Task Sequence?**

A **task sequence** is a **list of steps** that a computer will follow in order, such as:  
Install App A  
Restart computer  
Install App B  
Install App C  
Restart computer  
Install App D

You **set it up once**, and the computer **does everything automatically**.

* **Why use a task sequence?**

Saves **time and effort** (no need to manually install each app).

Ensures **apps are installed in the correct order**.

Handles **reboots automatically** when needed.

**Perfect for large organizations** that need to set up many computers.

**How Does It Work?**

Let’s say we use **Microsoft Endpoint Configuration Manager (MECM)** or **Microsoft Deployment Toolkit (MDT)**, both popular for creating task sequences.

Here’s what happens:  
1️. You create a **task sequence** with steps to install each app.  
2️. You tell it when to **restart the computer**.  
3️. The system runs the steps:

**Installs an app**

**Reboots if needed**

**Continues with the next app**  
When finished, you have a fully set-up computer.

* **Simple Example:**

Imagine installing:

Chrome

Microsoft Office

Zoom

SAP

And you know that **Office needs a reboot** before installing the next app.

**Task Sequence Steps:**  
Install Chrome  
Install Office  
Restart computer  
Install Zoom  
Install SAP  
Restart computer  
Finish

**Why Multiple Reboots?**

* Some apps need a **reboot to finish installing correctly** (like Office or system updates).
* Without the reboot, the next app might **fail to install**.

**Summary (in Simple Words):**

* **Task Sequence**: A list of steps to install apps in order.
* **Why Use It**: Saves time, handles reboots, and automates installation.
* **Multiple Reboots**: Some apps need a restart before the next one installs.
* **Log Files and Event Viewer for Troubleshooting MSI Errors**

**What are we talking about?**

When you try to install a program on Windows using an **MSI file** (Microsoft Installer), sometimes the installation **fails**.

* **MSI files** are packages used to install software.
* **MSI errors** are messages that something went wrong.

To figure out **why it failed**, we use:  
**Log files** – Records of what happened during the install.  
**Event Viewer** – A tool that shows system messages and errors.

**1️. Log Files**

* When an MSI installer runs, it creates a **log file**.
* This file keeps a **step-by-step record** of what the installer did:
  + What files did it copy
  + What settings did it change
  + What errors did it run into

**Why use log files?**

* If something fails, the log will say **where and why it happened**.
* Look for lines with **"error"**, **"failed"**, or **error codes** (like 1603, 1618).

**Where to find the log file?**

* You can tell the MSI installer to **create a log file** using a command like:

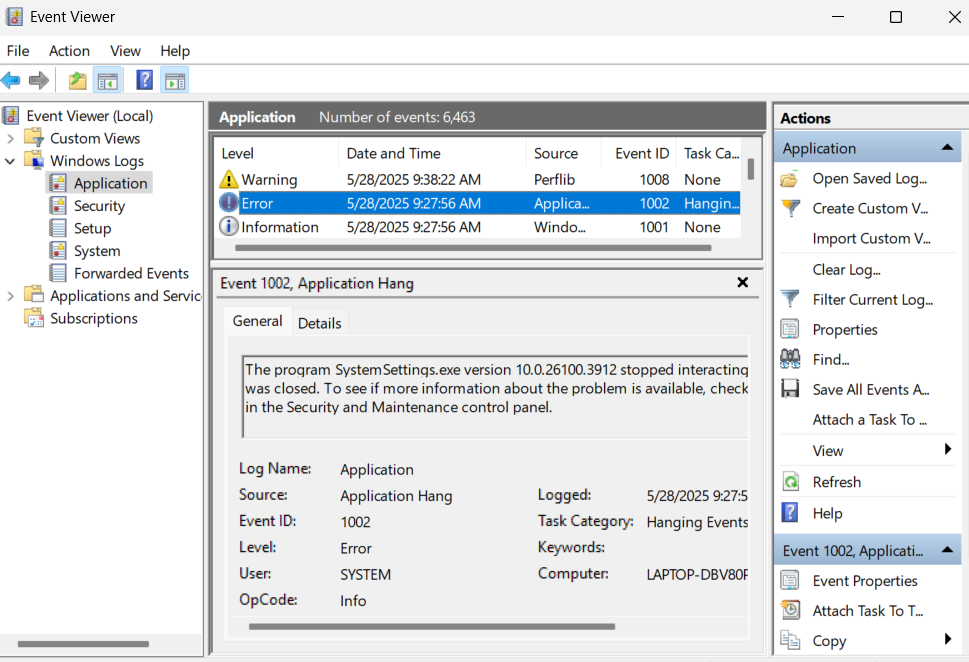
msiexec /i appname.msi /l\*v C:\InstallLogs\appname.log

* The /l\*v tells it to log everything, in detail.

**2️. Event Viewer**

* **Event Viewer** is a Windows tool that shows all **system events**.
* When an MSI installation fails, Windows logs an **event** about it.

**How to open Event Viewer?**  
1️. Press **Windows key + R**, type eventvwr, and press Enter.  
2️. Go to **Windows Logs → Application**.  
3️. Look for **MSI Installer errors** with a **red X** or warning sign.



**Why use Event Viewer?**

* It shows you **error messages**, including the **Event ID** and a **description** of the problem.
* You can see **when and where** the error happened.

**Example:**

Let’s say you tried installing Microsoft Office (with an MSI), but it failed.

1️. You check the **log file** and find:

Error 1603: Fatal error during installation

2️. You open **Event Viewer**, go to Application logs, and find:

* Event ID: **11706**
* Message: "Product: Office -- Installation failed."

This tells you there was a **specific error** during the Office install.

**Why is this important?**

* **MSI errors don’t always tell you exactly what went wrong**.
* **Log files** and **Event Viewer** give you **detailed clues** to solve the problem.
* You can Google the **error code** (like 1603) to find a solution.

**Simple Summary:**

* **Log Files**: Detailed records of the installation steps and errors.
* **Event Viewer**: Shows system error messages related to MSI installs.
* Both help you **figure out why an installation failed**.
* **What is Intune?**

Intune is a **cloud service from Microsoft** that helps you **manage devices** (like laptops and phones) and **applications** in your organization.

* You can control how devices are used.
* You can install apps remotely.
* You can secure devices if they’re lost or stolen.

**How to Enroll in a Free Trial of Intune**

**Step-by-Step:**

**1️. Go to the Microsoft Endpoint Manager Website**

* Open your browser and go to:  
   <https://www.microsoft.com/en-us/microsoft-intune>

**2️. Click on "Free Trial" or "Start Free Trial."**

* Look for a button that says **"Free trial"** or **"Try now"**.
* Click on it.

**3️ Sign in or Create an Account**

* If you already have a **Microsoft account** (like a work email), sign in.
* If not, you’ll need to **create a new Microsoft Work or school account**.
* You might need to provide a **business email address** (not a personal one like Gmail).

**4️. Fill in Business Details**

* Microsoft will ask for:
  + Company name
  + Number of employees
  + Phone number
  + Country
* This is for verification and to set up the trial.

**5️ Set Up Microsoft 365 Tenant**

* Intune is part of Microsoft 365, so it will **set up a tenant** (a space for your business in Microsoft’s cloud).
* You’ll choose a **domain name** (e.g., mycompany.onmicrosoft.com).

**6️ Confirm and Activate**

* You’ll receive a **verification email** or phone code.
* Enter the code and finish setting up.

**7️ Log in to Microsoft Endpoint Manager Admin Center**

* Once set up, go to:  
   <https://endpoint.microsoft.com>
* This is where you **manage Intune**.

**Now you can:**

* Add devices.
* Enroll users.
* Create policies.
* Test app deployment and management.

**How Long is the Free Trial?**

* Microsoft offers a **30-day free trial** of Intune (sometimes longer depending on the offer).

**Bonus Tips:**

Use the **Microsoft 365 Admin Center** to manage billing and subscriptions.  
Try setting up **test devices** to explore features without affecting real devices.  
Explore **Microsoft Learn** for tutorials.

**Simple Summary:**

1️. Go to Microsoft’s Intune page.  
2️. Click **Free Trial** and sign in or create an account.  
3️. Fill in your business details.  
4️. Verify your account and set up a tenant.  
5️. Log in to **Endpoint Manager** and start testing Intune!

* **Walkthrough: Intune Applications for Windows**

**1️. What are Intune Applications?**

* **Intune** lets you manage and deploy applications (apps) to **Windows PCs** (and other devices).
* You can:
  + **Install software remotely** (like Office, Zoom, Chrome).
  + **Set up rules** for which apps can be installed.
  + **Update apps automatically**.
  + **Uninstall apps remotely**.

**2️. Types of Applications in Intune**

When adding an app to Intune, you choose the **type**. Common types for Windows are:  
**Win32 app** – For classic Windows installers (like .exe or .msi files).  
**Microsoft Store app** – Apps from the Microsoft Store.  
**Line-of-business (LOB) app** – Custom business apps (usually .msi files).  
**Microsoft 365 apps** – Office apps like Word, Excel, and Outlook.

**3️. Adding a Windows App in Intune**

**Step 1: Go to Intune Admin Center**

* Open your browser and go to:  
  <https://endpoint.microsoft.com>
* Sign in with your **Intune admin account**.

**Step 2: Open the "Apps" Section**

* In the left menu, click **"Apps"**.
* Then click **"All apps"**.

**Step 3: Click Add (to add a new app)**

* Choose the **type of app** you want to add.  
  For example, to add a classic Windows installer, choose **Windows app (Win32)**.

**Step 4: Configure App Information**

* **Upload the app file** (e.g., setup.msi or .intunewin file for Win32 apps).
* Enter:
  + **Name of the app** (like "Zoom").
  + **Description**.
  + **Publisher** (who made the app).

**Step 5: Assign the App**

* Choose **who should get this app**:
  + **All devices** – every Windows device enrolled in Intune.
  + **Specific groups** – for example, only the sales team’s devices.
* Choose **install intent**:
  + **Required** – installs automatically.
  + **Available** – users can install from the Company Portal.
  + **Uninstall** – removes the app.

**Step 6: Configure Requirements (Optional)**

* You can set:
  + **Minimum OS version** (like Windows 10 21H2).
  + **Disk space needed**.
  + **RAM required**.
  + If the computer doesn’t meet these requirements, the app won’t install.

**Step 7: Configure Detection Rules (For Win32 Apps)**

* This helps Intune **check if the app is already installed**.
* You can set a **file path or registry key** to detect the app.

**Step 8: Review and Create**

* Review your settings.
* Click **Create**.

**Step 9: Sync and Test**

* Make sure devices **sync with Intune**.
* Check if the app installs properly on Windows devices.

**4️. Managing Apps Later**

In the **Intune Admin Center**:

* Go to **Apps → All apps**.
* You can:
  + **Update** an app.
  + **Uninstall** an app.
  + **Monitor installation status** (success, failed, pending).

**🎥 Simple Summary**

✅ **Intune apps** let you manage software on Windows computers from the cloud.  
✅ You choose the app type (Win32, Store, Office, LOB).  
✅ Upload the app, configure settings, assign it to devices, and Intune takes care of the rest.

Top of Form

Bottom of Form