So, my topic is Intune App Deployment – Errors and Packaging Complexity.

First, what is Intune App Deployment?

It’s a cloud service that allows us to deploy, update, and manage applications across different devices — like Windows, macOS, iOS, and Android.

It supports different types of apps — such as Win32 applications, Microsoft Store apps, MSI packages, and even web-based apps.

Now, let’s see some common reasons why deployment errors happen.

Incorrect Packaging – If an app isn’t wrapped properly into the .intunewin format or is in the wrong format, it can fail to deploy.

Outdated Tools – Sometimes, we may be using older packaging or deployment utilities, which can cause compatibility issues.

Configuration Mistakes – Things like wrong install or uninstall commands, incorrect detection rules, or missing required files can cause problems.

Network or Policy Issues – Even if packaging is correct, poor internet connectivity, firewall blocks, or the device not syncing can cause the deployment to fail.

Lastly, let’s look at packaging complexity.

File Size Limits – Very large packages may lead to compatibility issues or upload failures.

Multiple Dependencies – If an app requires other apps or files to be installed first, we need to ensure the correct install order.

Accurate Detection Rules – These rules must exactly match the installed app’s footprint; otherwise, Intune might think the app is missing or reinstall it unnecessarily.

Today, I want to talk about Process Explorer. It’s a powerful tool for monitoring and troubleshooting developed by Microsoft and initially it is part of the Sysinternals. You can think of it is like as a better version of the Windows Task Manager.

While Task Manager will shows a simple overview of running programs and system resources, Process Explorer lets you explore deeper.It also gives detailed information about each process, like the files or libraries it’s using.

Key features:

1st is real-time process monitoring

One of the main things Process Explorer does is real-time monitoring. This means you can see, moment by moment, how much CPU, memory is using. This is especially useful when your system slows down, or you suspect that some application is consuming too many resources.

Tree view of processes : Another feature is the process tree view. This shows the relationship between processes, like parent and child processes. For example, if a program opens several helper processes, you can see exactly which process started them. This makes troubleshooting much easier, especially if you’re trying to track down like malware programs

Detailed process properties

Process Explorer also provides detailed properties for each process. You can check the threads it’s running, even environment variables. This is helpful for system administrators and developers who need precise information about how software behaves.

DLL & handle viewing: Process Explorer lets you view DLLs and handles. DLLs are libraries that programs use, and handles are references to system resources like files and registry keys. With this feature, you can find exactly which resources a process is using, which is very useful for debugging or investigating system issues.

Search capability: Process Explorer has a powerful search capability. If you’re trying to figure out which process is using a particular file or resource, you can search for it, and the tool will quickly show you the answer.