
Slide 1: Title

(Opening) Hello everyone, my name is Krishna Verma, and today I'll be presenting my capstone project on using PSADT for application deployments.

Slide 2: Introduction & Objectives

(Introduction) My project focuses on the **PowerShell App Deployment Toolkit, or PSADT**[1]. This is a powerful tool for creating, testing, and modifying application deployment packages[2].

(Objectives) The main goal of this project was to automate application installations with specific conditions, logging, and registry edits[3]. To achieve this, I had several key objectives:

- Automate application deployment using PSADT[4].
- Create a standardized installation and uninstallation process[5].
- Demonstrate the entire deployment process within a virtual machine[6].
- Generate logs and registry keys to verify the deployment's success[7].
- And, of course, gain valuable hands-on experience with PowerShell scripting[8].

Slide 3: Project Requirements & Architecture

(Requirements) To complete this project, I needed a few key components:

- A Windows 10 or 11 Virtual Machine[9].
- PowerShell version 5.1 or later[10].
- The PSADT Toolkit itself[11].
- An application installer, specifically the 7-Zip MSI[12].
- And a text editor like VS Code or PowerShell ISE[13].
- Crucially, all of this required administrator privileges for testing[14].

(Architecture Diagram) The process flows like this: We use the **PSADT Toolkit** and the **application installer** within a **virtual machine**. Once the script runs, it generates **logs and registry entries**, which we can then use for verification.

Slide 4: Project Steps

(Step 1: Initial Setup) The first step was to get everything ready. I downloaded the latest PSADT from GitHub and extracted the files into my project directory[15]. I then verified that the two most important files,

Deploy-Application.ps1 and AppDeployToolkitMain.ps1, were present[16].

(Step 2: Project Folder Setup) Next, I placed the 7-Zip MSI installer inside the "Files" folder within the PSADT directory to create the proper folder structure for my deployment package[17].

(Step 3: Virtual Machine Preparation) I prepared a clean Windows 10/11 VM, updated PowerShell to version 5.1 or later, and ensured I had the necessary admin privileges[18].

(Step 4: Script Modification) This was the core of the project. I edited the main

Deploy-Application.ps1 script to add both the installation and uninstallation logic[19]. For installation, I used the

Execute-MSI function [20], and for uninstallation, I used the same function with the

-Action Uninstall parameter[21].

(Step 5: Run Deployment) Finally, I was ready to test. I executed the script to install the application and verified that 7-Zip was successfully installed[22]. I then ran the script again, this time with the

-DeploymentType 'Uninstall' parameter, to confirm that the application could also be uninstalled correctly[23].

Slide 5: Post-Installation Actions and Verification

(Script Modification) I also customized the script to perform a post-installation action: moving the installer file to another folder[24]. This was done using the

Move-Item command[25]. I also added logic to validate the registry after the installation was complete[26].

(Verification) The verification process was crucial.

- **Logs:** I confirmed that success or failure entries were generated by checking the log files located at C:\Windows\Logs\Software or AppDeployToolkit\Logs[27].
- **Registry:** I also verified that a new entry for 7-Zip was created in the registry at HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall[28].

Slide 6: Results & Conclusion

(Results) The project was a success!

- 7-Zip was successfully installed on the virtual machine[29].
- The customized script successfully moved the installer file after the installation[30].
- All the actions were validated through both the logs and the registry entries[31].

(Conclusion) In conclusion, I successfully demonstrated how to deploy an application using PSADT in a VM environment[32]. This project proved the reliability and standardization of this approach by automating the install, uninstall, and post-installation actions and verifying the results through logs and registry entries[33].

Slide 7: Thank You

(Closing) Thank you for your time. Are there any questions?