Intern Name: Ankita Kolapte
Intern ID: 270
Date:06-08-2025
Task Title: MITRE ATTACK PoC — Persistence Tactic

🔐 Persistence (MITRE ATT&CK Tactic)

Persistence is a tactic used by adversaries (like malware or threat actors) to maintain long-term access to a system, even after restarts, user logouts, or system crashes.

Purpose: Attackers want to remain on a compromised system so they can continue their malicious activities without needing to reinfect the system.

🔍 Techniques under Persistence:

1. **T1547.001:** Registry Run Keys / Startup Folder

   Malware modifies registry keys to auto-run at boot.

2. **T1053.005**: Scheduled Task/Job - Scheduled Task

   Malware creates scheduled tasks to run on reboot.

3. **T1543.003**: Create or Modify System Process - Windows Service

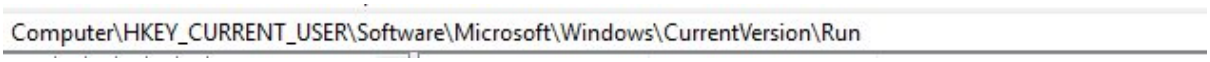
   Malware installs itself as a service.

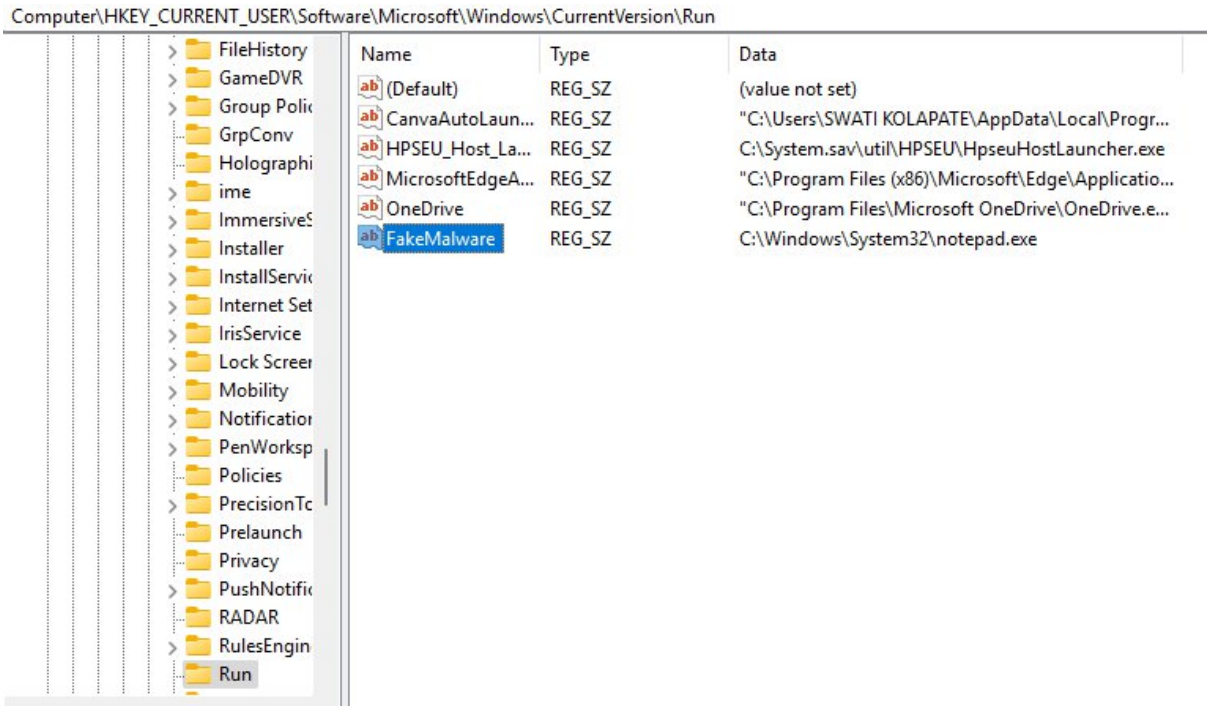
🛠 Procedures:

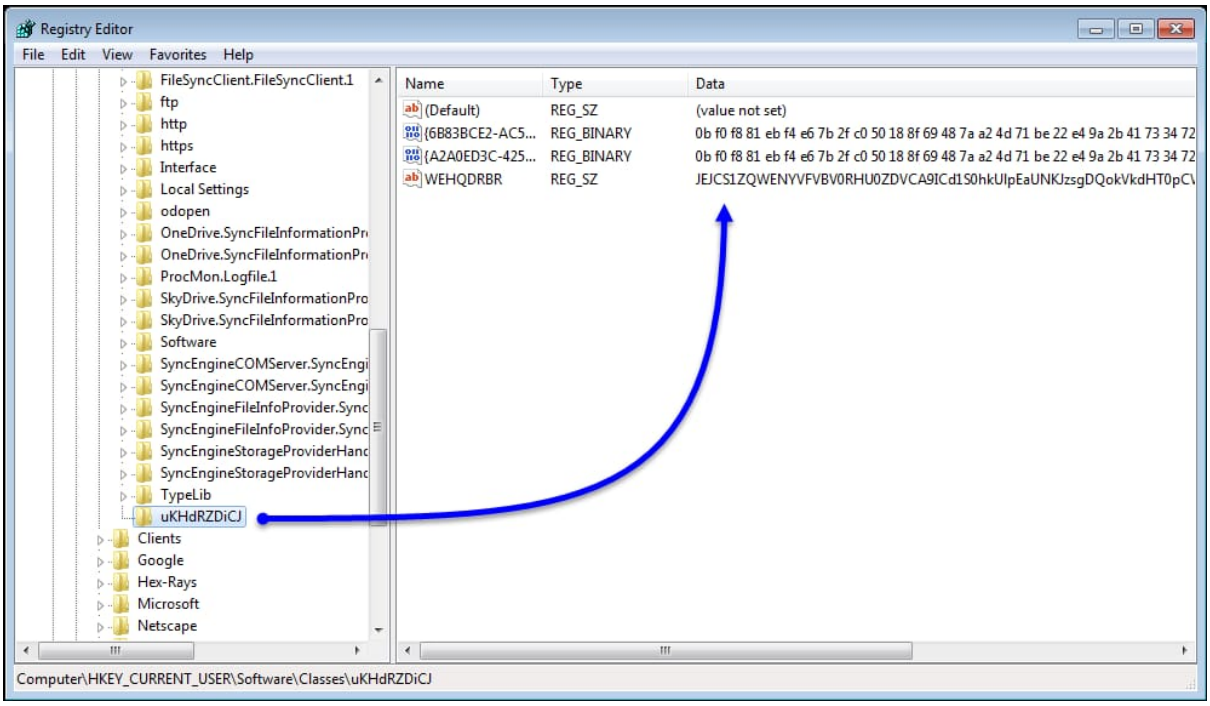
Procedure 1: Registry Key Injection

- The Trojan modifies the Windows registry key:
  HKCU\Software\Microsoft\Windows\CurrentVersion\Run
- This ensures it runs every time the system reboots.
- Mapped to:
  - Tactic: Persistence
  - Technique: T1547.001 - Registry Run Keys

- Registry Persistence Entry via Run Key:



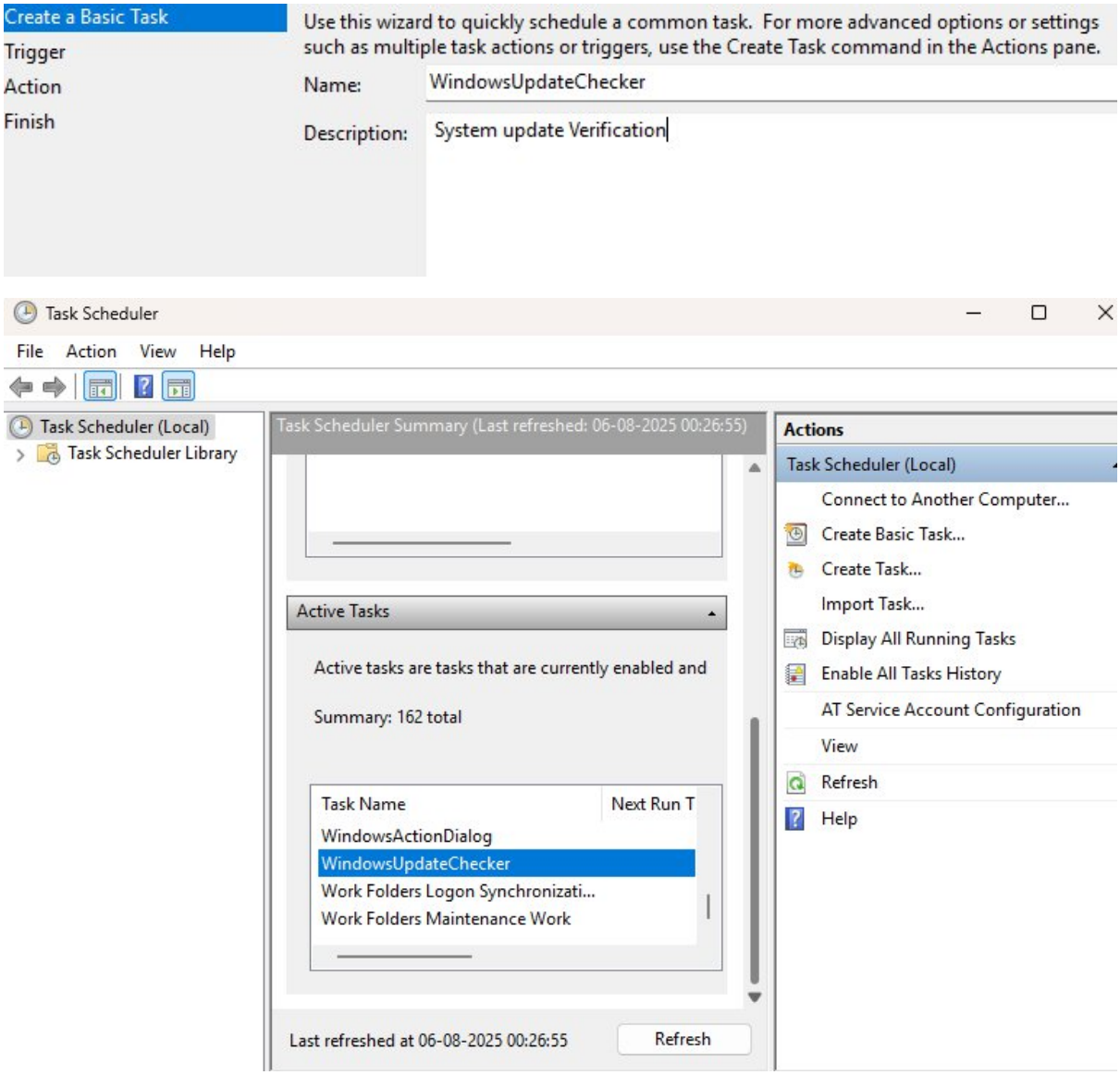




This key is commonly used by malware to maintain persistence by configuring the system to automatically execute a program upon user login. In this case, a sample entry (e.g., FakeMalware) can be added to simulate how a Trojan would store its executable path to ensure it runs every time the user logs into the system. This behavior is mapped to MITRE ATT&CK technique T1547.001 - Registry Run Keys / Startup Folder.

Procedure 2: Scheduled Task Creation

- Malware creates a scheduled task using schtasks.exe to execute the payload every hour or at startup.
- Mapped to:
  - Tactic: Persistence
  - Technique: T1053.005 - Scheduled Task/Job
- Simulated Malicious Task via Task Scheduler:



As seen in the screenshot, a scheduled task is created to execute a command at system startup, mimicking how malware gains persistence.

Detection & Mitigations

1. **Registry & Task Monitoring**

   Detect unauthorized persistence by monitoring registry keys (Run, RunOnce) and scheduled task creation logs (Event ID 4698).

2. **File Integrity & Script Logging**

   Use file integrity monitoring and enable logging for PowerShell, scripts, and startup file changes.

3. **Least Privilege Enforcement**

   Restrict user rights to prevent unapproved creation of persistent mechanisms like tasks or autorun entries.

4. **Use of Security Tools**

   Implement antivirus/EDR solutions and tools like Sysinternals Autoruns to detect and block persistence behavior.

📝 **Summary:**

In this Proof of Concept (PoC), we analyzed a Trojan malware sample using both static and dynamic analysis techniques. The malware exhibited persistence behavior by modifying registry keys and creating scheduled tasks. It also showed suspicious network activity and attempted to connect to external domains. Key findings include system modifications, file system changes, and observable Indicators of Compromise (IOCs). Proper detection mechanisms and mitigations were suggested to help prevent such threats in real-world environments.

Submitted To: Digisuraksha Parhari Foundation