Ansible

Contains everything on Ansible IT automation tool, playbooks and tricks

- Playbook - Clearing Users' Data Files in a Group of Windows Machines
- Playbook - Update Windows Machine (Windows Update Disabled)
- Playbook - Initiate Clamscan on Machines with ClamWin Installed
- Playbook - Disable Windows Updates

Playbook - Clearing Users' Data Files in a Group of Windows Machines

The playbook below will remove all users' data in a computer that belongs in an inventory group. Below is a list of steps that this playbook will do:

1. Disable and remove the target user
2. Reboot to remove any file locks from the logged in user
3. Remove any files in the user's directory, skipping symbolic links
4. Re-create a public user with the same username and empty password that cannot be changed
5. Enable auto login for the user so that new machine will be configured for auto login as well
6. Reboot computer to enable the configuration

The playbook is as follows, please change the variables encapsulated in `< >` with the desired values:

```
---
- hosts: <inventory group / host>
  tasks:
  - name: remove user account
    win_user:
      name: <username>
      account_disabled: yes
      state: absent
  - name: reboot
    win_reboot:
      msg: "Scheduled reset started, windows will reboot in 90 seconds"
      pre_boot_delay: 90
  - name: remove any files in the folder tree
    ignore_errors: yes
    win_shell: |
      $Path = "C:\Users\<username>"
```

```
      Remove-Item "$Path" -Force -Recurse -ErrorAction SilentlyContinue
      if (Test-Path "$Path" -ErrorAction SilentlyContinue)
      {
          $folders = Get-ChildItem -Path $Path -Directory -Force -ErrorAction SilentlyContinue
          ForEach ($folder in $folders)
          {
              Remove-Tree $folder.FullName -Force -ErrorAction SilentlyContinue
          }

          $files = Get-ChildItem -Path $Path -File -Force
          ForEach ($file in $files)
          {
              Remove-Item $file.FullName -Force -ErrorAction SilentlyContinue
          }

          if (Test-Path "$Path" -ErrorAction SilentlyContinue)
          {
              Remove-Item $Path -Force -ErrorAction SilentlyContinue
          }
      }
  - name: re-add user account
    win_user:
      name: <username>
      state: present
      groups: Users
      user_cannot_change_password: yes
      password_expired: no
      password_never_expire: yes
  - name: enable auto logon
    win_shell: |
      Set-ItemProperty 'HKLM:\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon' -Name
'AutoAdminLogon' -Value '1'
      Set-ItemProperty 'HKLM:\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon' -Name
'DefaultUsername' -Value '<default username>'
      Set-ItemProperty 'HKLM:\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon' -Name
'DefaultPassword' -Value ''
  - name: reboot to apply new settings
    win_reboot:
      msg: "Scheduled reset completed, windows will reboot in 5 seconds"
      pre_boot_delay: 5
```

References:

- https://luke.geek.nz/win/using-powershell-setup-automatic-logon-windows-servers/
- https://stackoverflow.com/a/31450526
- https://docs.ansible.com/ansible/2.5/modules/list_of_windows_modules.html

Playbook - Update Windows Machine (Windows Update Disabled)

This playbook will:

1. Modify windows update service to manual in case the machine is set to disabled
2. Start the windows update service
3. Download and install the updates, reboot if required

The playbook is as follows, please change the encapsulated `< >` values to the desired values:

```
---
- hosts: <inventory group / hosts>
  tasks:
  - name: change windows update service to manual
    win_shell: Set-Service wuauserv -StartupType Manual
  - name: start windows update service
    win_shell: Start-Service wuauserv
  - name: download and install updates
    win_updates:
      reboot: yes
```

Playbook - Initiate Clamscan on Machines with ClamWin Installed

This playbook will initiate a full scan on all computers using `Clamscan` that is installed through `ClamWin`:

```
---
- hosts: <inventory group / hosts>
  tasks:
  - name: full computer scan
    win_command: '"C:\Program Files (x86)\ClamWin\bin\clamscan.exe" -rv --
move=C:\ProgramData\.clamwin\quarantine\ --database=C:\ProgramData\.clamwin\db\ --
log=C:\ProgramData\.clamwin\log\ClamScanLog.txt --enable-stats C:\'
```

Playbook - Disable Windows Updates

This playbook will download disable_windows_update.ps1 from a server, reachable by all clients and execute the script to disable windows updates on a group of windows machines. Though it is written to specifically disable windows update, it can be modified to execute other scripts as well. The playbook configuration file is ass follows, replace enclosed `< >` tags with the desired values:

```
---
- hosts: <inventory group / hosts>
  tasks:
  - name: download script to disable windows update
    win_get_url:
      url: http://<url>/disable_windows_update.ps1
      dest: C:\
  - name: execute disable windows update script
    win_shell: C:\disable_windows_update.ps1
  - name: remove script
    win_file:
      path: C:\disable_windows_update.ps1
      state: absent
```