Install Windows

- export to https://repo.sxl.net/install_windows.html
- windows_7
- Win10 update assistant https://is.gd/OK28yz
- Win11 update assistant https://is.gd/nLSnJd

OOBE

- OOBE bypass
- Shift+F10 then OOBE/BYPASSNRO
- defaults apps `control /name Microsoft.DefaultPrograms`
- language `control /name Microsoft.Language`
- regional `intl.cpl`
- date/time `timedate.cpl`
- Network connections `ncpa.cpl`
- firewall `firewall.cpl`
- Printers `control.exe printers`
- programs add/remove `appwiz.cpl`
- windows features `OptionalFeatures.exe`
- system propoerties `sysdm.cpl`
- device manager `devmgmt.msc`
- restart computer `shutdown.exe /r /t 0`
- windows update `control.exe update`
- user management
- `control.exe userpasswords`
- `control.exe userpasswords2`
- `lusrmgr.msc`

create local admin Owner

```
net user /add Owner /y
net localgroup administrators Owner /add
net user Owner "" /passwordreq:no /expires:never #empty password
```

set language

```
Set-ExecutionPolicy Bypass -Force
$repo="http://repo.sxl.net/_h/bin/"
$drv="c:\drv"; $null=(mkdir -Force "$drv"); cd "$drv"
$f="drv-setlang.ps1"; iwr "$repo/$f" -outfile "$f"; & "./$f" help
.\drv-setlang.ps1 us
```

ESSENTIALS

Essential DRV

```
<# prepare DRV & utils #>
Set-ExecutionPolicy Bypass -Force
$repo="http://repo.sxl.net/_h/bin"
$drv="c:\drv"; $null=(mkdir -Force "$drv"); cd "$drv"
$f="curl.exe"; if(!(Get-Command $f -EA SilentlyContinue ))
  { Invoke-WebRequest "$repo/$f" -outfile "$env:windir\$f" }
$f="7za.exe"; curl.exe -LO "$repo/$f"
$f="caffeine.exe"; curl.exe -LO "$repo/$f"
& "c:\drv\caffeine.exe" -stes -replace
```

```
$f="drv-mail2sys.ps1"; curl.exe -LO "$repo/$f"
$f="drv-setlang.ps1"; curl.exe -LO "$repo/$f"
Add-Content "$env:windir\reboot.cmd" "shutdown.exe /r /t 0"
### windows settings
Enable-ComputerRestore -Drive C: -EA SilentlyContinue #System protection
### set registry mailpw
$p="HKLM:\SOFTWARE\SXLnet"; New-item -Path "$p" -EA SilentlyContinue
New-ItemProperty "$p" -Name 'pw' -Value "A13c0BRpgWFWXQmi0lB2" -Force | Out-Null
### Disable OOBE Lets Finish Setting Up
$k="HKCU\Software\Microsoft\Windows\CurrentVersion\ContentDeliveryManager"
reg.exe add $k /v SubscribedContent-310093Enabled /t REG_DWORD /d 1 /f
### Powershell PROFILE
$f="$PROFILE"; if(!(Test-Path $f)) { New-Item $f -Force }
$re='\$\S+ErrorForegroundColor\s*='
(Get-Content $f) | Select-String $re -NotMatch | Set-Content $f
'$host.PrivateData.ErrorForegroundColor="Magenta"' | Add-Content $f
(Get-Content $f) | Where-Object {$_ -ne ""} | Set-Content $f # del blank lines
. "$PROFILE" # source profile
Get-Content $PROFILE
. $PROFILE
### Windows defender
if ((Get-MpComputerStatus -EA SilentlyContinue).AntivirusEnabled) {
  Set-MpPreference -EnableControlledFolderAccess Disabled
  Set-MpPreference -DisableEmailScanning $False
  Set-MpPreference -UILockdown $False
  Set-MpPreference -PUAProtection Enabled
  #$d="c:\drv\no_avscan"; mkdir -Force $d; cd $d; Set-MpPreference -ExclusionPath $d
}
### keep this session alive
$f="caffeine.exe"; if (!(Test-Path "$f")) {curl.exe -LO "$repo/$f"}
& "./$f" -stes -replace
### Making pcinfo
#$f="drv-pcinfo.ps1"; curl.exe -LO "$repo/$f"; & "./$f"

### Refresh environment
$env:Path=[System.Environment]::GetEnvironmentVariable("Path","Machine")
$env:Path+=";"+[System.Environment]::GetEnvironmentVariable("Path","User")
<# done #>
```

install/upgrade chocolatey

- chocolatey

```
$d="c:\drv"; $null=(mkdir -Force "$d"); cd "$d"
Set-ExecutionPolicy Bypass -Scope Process -Force
[System.Net.ServicePointManager]::SecurityProtocol = `
  [System.Net.ServicePointManager]::SecurityProtocol -bor 3072
$u="https://community.chocolatey.org/install.ps1"
if (Get-Command choco -EA SilentlyContinue ) { choco upgrade chocolatey } `
else {iex ((New-Object System.Net.WebClient).DownloadString($u))}
Import-Module $env:ChocolateyInstall\helpers\chocolateyProfile.psm1
refreshenv
# fix history problems
$p="sudo"; if ( choco list --lo -r -e "$p" ) { choco uninstall "$p" }
$p="gsudo"; if ( choco list --lo -r -e "$p" ) { choco uninstall "$p" }
$p="choco-cleaner"; if ( choco list --lo -r -e "$p" ) { choco uninstall "$p" }
$p="teamviewer"; if ( choco list --lo -r -e "$p" ) { choco uninstall "$p" }
$p="teamviewer.host"; if ( choco list --lo -r -e "$p" ) { choco uninstall "$p" }
# config chocolatey
$env:ChocolateyToolsLocation="$env:ProgramData\tools"
setx.exe ChocolateyToolsLocation $env:ProgramData\tools
```

```
choco feature enable -n allowGlobalConfirmation
choco feature enable -n useRememberedArgumentsForUpgrades
refreshenv
choco upgrade chocolatey
# install essentials
choco install 7zip.commandline wget curl aria2 gsudo nircmd
"`nImport-Module 'gsudoModule'" | Add-Content $PROFILE
choco upgrade all
# done
```

choco cleaner

```
choco upgrade all
choco install choco-cleaner  --params "'/NOTASK:TRUE'"
choco-cleaner
choco uninstall choco-cleaner;
#done
```

winget install/upgrade

```
# install/fix/upgrade winget
$repo="http://repo.sxl.net/_h/bin"; $f="drv-winget.ps1"
$drv="c:\drv"; $null=(mkdir -Force "$drv"); cd "$drv"
Invoke-WebRequest "$repo/$f" -outfile "$f"; & "./$f"
./drv-winget.ps1 init
#./drv-winget.ps1 upgrade
#
```

hardware

vmware

```
choco install vmware-tools
```

PS Windows Update

```
### PS Windows Update
Install-PackageProvider -Name NuGet -Force -Confirm:$false
Install-Module PSWindowsUpdate -Force -Confirm:$false
Add-WUServiceManager -MicrosoftUpdate -Confirm:$false
Install-WindowsUpdate -MicrosoftUpdate -AcceptAll -IgnoreReboot
Get-WindowsUpdate -IgnoreReboot
#
```

Backups

- Iperius - https://www.iperiusbackup.com/
- Cobian - https://www.cobiansoft.com/ `choco install cobian-backup`
- Duplicati - https://www.duplicati.com/ duplicati

Enable File History

- windows explorer, right-click drive, configure Shadow copies
- Enable shadow copies on the drive (e.g C:\ or D:\)

- click Settings > Max Size Limit (example: 20GB=20480MB; 100GB=102400MB, 400GB=409600MB)
- https://cdn.sxl.net/sharex/2018/20180613190536.jpg
- Change schedule: Daily, 1PM and 9PM
- https://cdn.sxl.net/sharex/2018/20180613191027.jpg

Veeam agent

```
<# install veeam agent #>
choco install veeam-agent --pin
#
```

```
<# create recovery ISO #>
& "$env:ProgramFiles\Veeam\Endpoint Backup\Veeam.EndPoint.Manager.exe"
/createrecoverymediaiso
```

COMMON

choco install apps

```
<# choco apps #>
choco install peazip.install # open, set assoc > standard, set lang
choco install pasteintofile ; PasteIntoFile /filename yyyyMMdd-HHmmss
choco install file-converter
choco install vlc
choco install xnviewmp.install
#
```

rustdesk

download exe on desktop

```
<# rustdesk.exe on desktop#>
choco install dra
$repo="rustdesk/rustdesk"
$tag="rustdesk-{tag}-x86_64.exe"
$out=[Environment]::GetFolderPath("CommonDesktopDirectory")
dra download $repo --select $tag --output $out\rustdesk.exe
#
```

rustdesk install

```
<# install rustdesk #>
$rd=[Environment]::GetFolderPath("CommonDesktopDirectory")
$rd+="\rustdesk.exe"
& "$rd" --silent-install
while (Test-Path "$rd") { Start-Sleep 1; Remove-Item "$rd" -Force }
<# set rustdesk password #>
$rd="$env:ProgramFiles\RustDesk\rustdesk.exe"
while (!(Test-Path "$rd")) { Start-Sleep 1 }
& $rd --install-service
while (!(Get-Service "Rustdesk" -EA SilentlyContinue)) { Start-Sleep 1 }
& $rd --password Super@123
& $rd --option allow-remote-config-modification Y
& $rd --get-id | Out-String
#
```

uninstall winget

```
<# uninstall rustdesk #>
winget uninstall rustdesk.rustdesk
#
```

aspia install/upgrade

install/upgrade

```
## download
$drv="c:\drv"; $null=(mkdir -Force "$drv"); cd "$drv"
$repo="https://repo.sxl.net/_h/proj/"
curl.exe -LO "$repo/remote/aspia-host-2.7.0-x86_64.msi"
#$url="https://github.com/dchapyshev/aspia/releases/download"
#curl.exe -LO "$url/v2.7.0/aspia-relay-2.7.0-x86_64.msi"
curl.exe -LO "$repo/remote/aspia/aspia3.json"
Start-Process msiexec.exe "/qb /i aspia-host-2.7.0-x86_64.msi" -Wait
## install/upgrade service
$d="$env:ProgramFiles\Aspia\Host"
& "$d\aspia_host.exe" --import=c:\drv\aspia3.json --silent
& "$d\aspia_host_service.exe" --stop
& "$d\aspia_host_service.exe" --remove
& "$d\aspia_host_service.exe" --install
& "$d\aspia_host_service.exe" --start
& "$d\aspia_host_service.exe" --version
& "$d\aspia_host_service.exe" --host-id
## done
```

uninstall

```
winget uninstall "Aspia Host"
# winget uninstall "ARP\Machine\X64\{152430E2-C5E5-4086-A24B-A733C1B693C6}"
```

customize win10/11

```
# explorer
$p="HKCU:\Software\Microsoft\Windows\CurrentVersion\Explorer\Advanced"
Set-Itemproperty -path "$p" -Name "HideFileExt" -value 0 #show file exts
Set-ItemProperty -Path "$p" -Name "TaskbarMn"   -Value 0 #hide win11 chat
# win10 disable news feed (old method)
$k="HKCU:\SOFTWARE\Microsoft\Windows\CurrentVersion\Feeds"
Set-ItemProperty -Path "$k" -Name "ShellFeedsTaskbarViewMode" -Value 2 -EA SilentlyContinue
# win10 disable news feed (after 2024)
$url="https://repo.sxl.net/_h/bin/drv-disable-w10news.ps1"
& ([ScriptBlock]::Create((Invoke-RestMethod $url)))
# win11 enable classic context menu
$k="HKCU\Software\Classes\CLSID\{86ca1aa0-34aa-4e8b-a509-50c905bae2a2}"
New-Item -Path "$k\InprocServer32" -Value "" -Force
#Remove-Item -Path "$k\InprocServer32" -Force # Undo
# win11 start menu to left
$k="HKCU:\software\microsoft\windows\currentversion\explorer\advanced"
Set-ItemProperty -Path "$k" -Name "TaskbarAl" -Type "DWord" -Value 0
# win11 remove news widget (Windows web experience Pack)
winget uninstall "Windows web experience Pack"
winget uninstall --id 9MSSGKG348SP --accept-source-agreements
# restart windows explorer
Get-Process explorer | Stop-Process
#
```

adobe reader

choco install

```
<# choco install adobe reader #>
mkdir -force c:\drv ; Set-Location c:\drv
choco install adobereader --pin --params '"/DesktopIcon /NoUpdates"'
# MANUALLY set as default
```

install asian font pack

```
<# install asian font pack #>
$u="https://ardownload3.adobe.com/pub/adobe/acrobat/win/AcrobatDC"
curl.exe -o adobefont.msi "$u/2300820421/AcroRdrALSDx64_2300820421_all_DC.msi"
msiexec.exe /i adobefont.msi /qb
#
```

config adobe reader

```
<# config adobe reader #>
$subkeys = @("HKLM\SOFTWARE","HKLM\SOFTWARE\WOW6432Node")
ForEach ( $subkey in $subkeys) {
  $regkey = "$subkey\Policies\Adobe\Adobe Acrobat\DC\FeatureLockDown"
  reg add "$regkey" /v bLimitPromptsFeatureKey /t REG_DWORD /d 1 /f
  reg add "$regkey" /v bToggleDCAppCenter /t REG_DWORD /d 1 /f
  reg add "$regkey" /v bShowRhpToolSearch /t REG_DWORD /d 0 /f
  reg add "$regkey" /v bAcroSuppressUpsell /t REG_DWORD /d 1 /f
  reg add "$regkey" /v bPurchaseAcro /t REG_DWORD /d 0 /f
  reg add "$regkey" /v bReaderRetentionExperiment /t REG_DWORD /d 0 /f
  reg add "$regkey" /v bEnableTrialistLaunchCard /t REG_DWORD /d 0 /f
  reg add "$regkey" /v bToggleSophiaWebInfra /t REG_DWORD /d 0 /f
  reg add "$regkey" /v bMerchandizingEnabled /t REG_DWORD /d 0 /f
}
$regkey = "HKLM\SOFTWARE\Policies\Adobe\Adobe Acrobat\DC\FeatureLockDown\cIPM"
reg add "$regkey" /v bDontShowMsgWhenViewingDoc /t REG_DWORD /d 0 /f
reg add "$regkey" /v bShowMsgAtLaunch /t REG_DWORD /d 0 /f
reg add "$regkey" /v bAllowUserToChangeMsgPrefs /t REG_DWORD /d 0 /f
$regkey = "HKLM\SOFTWARE\Policies\Adobe\Adobe Acrobat\DC\FeatureLockDown\cCloud"
reg add "$regkey" /v bDisableADCFileStore /t REG_DWORD /d 1 /f
reg add "$regkey" /v bEnableADCFileStore /t REG_DWORD /d 0 /f
$regkey = "HKLM\SOFTWARE\Policies\Adobe\Adobe Acrobat\DC\FeatureLockDown\cServices"
reg add "$regkey" /v bUpdater /t REG_DWORD /d 0 /f
$regkey = "HKCR\Software\Adobe\Acrobat Reader\11.0\Workflows"
reg add "$regkey" /v bEnableAcrobatHS /t REG_DWORD /d 0 /f
# open it, make it DEFAULT
```

MS Office

microsoft_office

```
choco install office365business -pin
choco pin add -n office365business
```

activate

```
& ([ScriptBlock]::Create((irm https://massgrave.dev/get))) /Ohook ## office only
```

uninstall

```
$s="microsoft-office-deployment"; choco pin remove -n $s; choco uninstall $s
```

```
winget uninstall Microsoft.Office
cd c:\drv ; curl.exe -o uninst_office.exe -L https://aka.ms/diag_officeuninstall;
.\uninst_office.exe;
```

google chrome

choco install

```
choco install googlechrome --ignore-checksums
```

uninstall extension

```
<# uninstall chrome and extensions #>
winget uninstall Google.Chrome --force
$k="HKCU:\SOFTWARE\Policies\Google\Chrome\ExtensionInstallForcelist"
Remove-Item -Path "$k" -Recurse -Force -EA SilentlyContinue
#
```

Xnview

- not for commercial use
    - run xnviewmp as admin > Tools > Settings > Integration : Enable Browse with XnViewMP

```
choco install xnviewmp.install
```

Symantec SEP 64 and 32

- https://is.gd/sPpEfq
- https://repo.sxl.net/proj/antivirus/

```
# install SEP
mkdir -force c:\drv ; cd c:\drv
if ([System.Environment]::Is64BitOperatingSystem) {$bit="64"} else {$bit="32"}
$lang = switch -Wildcard ((Get-WinUserLanguageList)[0].LanguageTag)
  {"zh-Hans*" {'CS'}; "zh-Hant*" {'CH'}; default {'EN'}}
$url="https://repo.sxl.net/.hidden/proj/antivirus/SEP_14.3.0_RU4_Win$bit-bit_Client_$lang.exe"
curl.exe -L -o "SEP.exe" "$url"
.\SEP.exe /s /v"SYMREBOOT=ReallySuppress /qb"
# Restart-Computer
```

Uninstall

```
winget uninstall "Symantec Endpoint Protection"
```

Other antivirus

- NOD32 - https://www.eset.hk/
- Clamwin for Win98-XP - http://clamsentinel.sourceforge.net/ '

MS Security Essentials (for vista-7) - https://bit.ly/3iXkoA0 choco install MicrosoftSecurityEssentials

OPTIONAL

dontsleep

```
<# dontsleep #>
choco install dontsleep.install
dontsleep_x64.exe -bg block_standby=1 block_screensaver=1 block_shutdown=0 block_logoff=0
#
```

caffeine

```
<# caffeine #>
mkdir -force c:\drv; cd c:\drv
curl.exe -LO "http://repo.sxl.net/_h/bin/$f"
& "c:\drv\caffeine.exe" -stes -replace
#
```

cad

```
choco install dwgtrueview designreview
```

```
choco install autodesk-fusion360
choco install meshmixer
choco install cura
choco install autocad
```

graphics apps

```
choco install inkscape
choco install gimp
choco install drawio
```

utils

Internet Download Manager IDM

```
choco install internet-download-manager
irm https://massgrave.dev/ias | iex
```

misc

```
locale-emulator #app locale
firefox
choco install
hwinfo.install
autoruns
autologon
bleacbit.portable
sidebar-diagnostics
```

misc

```
set-alias wg c:\drv\drv-winget.ps1
<# misc #>
wg install Apple.iTunes
wg install 9NKSQGP7F2NH # whatsapp

wg installTelegram.TelegramDesktop
```

china

```
wg install tencent.wechat
wg install YY.YYAnchor
wg install tencent.qq
wg install tencent.tim
wg install Tencent.WeCom #企业微信
wg install tencent.qqmusic
wg install tencent.yingyongbao #应用宝
wg install Sogou.SogouInput #搜狗输入法
wg install NetEase.CloudMusic #网易云音乐
```

quickq

```
mkdir -force c:\drv ; sl c:\drv
curl.exe -LO https://d.asdfgh.win/quickq/download/win32-quickq.exe
```

caming

```
choco install obs-studio.install itunes
```

SERVER

create local adminsxl

```
net user /add adminsxl sxlSuper@123 ; localgroup administrators adminsxl /add
```

Connect server

create the batch file

```
$desktop = ([Environment]::GetFolderPath('Desktop'))
$null >> "$desktop\connect-server.bat"
notepad "$desktop\connect-server.bat"
```

file content

```
net use n: /delete
net use /persistent:yes
net use N: \\server\share password /user:domain\username /savecred
```

reboot shortcut

```
# create reboot shortcut
Add-Content "c:\drv\reboot.cmd" "shutdown.exe /r /t 0"
$TargetFile = "c:\drv\reboot.cmd"
$ShortcutFile = "$env:Public\Desktop\reboot.lnk"
$WScriptShell = New-Object -ComObject WScript.Shell
$Shortcut = $WScriptShell.CreateShortcut($ShortcutFile)
$Shortcut.TargetPath = $TargetFile
```

```
$Shortcut.Save()
```

conemu

```
choco install conemu
```

Setting > Keys & Macros > Paste >

- UNCHECK Multi-line paste
- UNCHECK Long text-paste

windows terminal

```
choco install microsoft-windows-terminal
```

```
# download the ico
$icon="https://raw.githubusercontent.com/microsoft/terminal/master/res/terminal.ico"
iwr -OutFile "$env:LOCALAPPDATA\\wt.ico" -uri $icon
$DesktopPath=[Environment]::GetFolderPath("Desktop")
$WshShell=New-Object -comObject WScript.Shell
$Shortcut=$WshShell.CreateShortcut("$DesktopPath\Windows Terminal.lnk")
$Shortcut.TargetPath = (Get-Command wt.exe).Path
$shortcut.IconLocation = "$env:LOCALAPPDATA\\wt.ico"
$Shortcut.Save()
#
```

notepad++

install

```
choco install notepadplusplus.install
regsvr32 "C:\Program Files\Notepad++\contextMenu\NppShell.dll"
#choco install notepadreplacer --params "/NOTEPAD:C:\Program Files\Notepad++\notepad++.exe"
```

notepad plugins

```
choco install notepadplusplus-nppluginmanager
```

* open Plugins > Plugin Admin, add `MarkdownViewer++`, `AutoCodePage`, `AutoEolFormat`

acronis drive monitor

```
choco install acronis-drive-monitor
```

* Option Alerts Email notification > Change settings > `systems@sxl.net`

virtio

```
choco install virtio-drivers
```

hwinfo

```
choco install hwinfo
```

```
nircmd shortcut "%ProgramFiles%\HWiNFO64\HWiNFO64.EXE" "~$folder.desktop$" "HWINFO"
```

Gsmartcontrol

```
choco install gsmartcontrol
nircmd shortcut "%ProgramFiles%\GSmartControl\gsmartcontrol.exe" "~$folder.desktop$"
"GSmartControl"
```

Server 2016/2019 Windows Defender

Enable Windows Defender GUI

```
Install-WindowsFeature -Name Windows-Defender
Install-WindowsFeature -Name Windows-Defender-GUI
```

Disable Windows Defender GUI

```
Uninstall-WindowsFeature -Name Windows-Defender-GUI
Uninstall-WindowsFeature -Name Windows-Defender
```

ACTIVATE

- download from Github or Bitbucket or Massgrave

exclude a dir in msdefender

```
$d="c:\drv\no_avscan"; mkdir -Force $d; cd $d
Set-MpPreference -ExclusionPath $d
```

activate

```
irm https://massgrave.dev/get | iex  ## interactive mode
& ([ScriptBlock]::Create((irm https://massgrave.dev/get))) /HWID  ## win only
& ([ScriptBlock]::Create((irm https://massgrave.dev/get))) /Ohook ## office only
```

adobe acrobat

```
winget install Adobe.Acrobat.Pro # genp patch app
```

adobe cc

- read https://www.reddit.com/r/GenP/ > genp patch cc > install apps > genp patch apps
- login user `nobodyboss@outlook.com` pass `aSuper@123` - adobe email forward to sxlderek@gmail

winget install Adobe.CreativeCloud
autodesk

```
choco install autocad --pin
```

TROUBLESHOOT

clean disk

```
cleanmgr.exe /lowdisk
cleanmgr.exe /verylowdisk
```

choco cleaner

```
choco install choco-cleaner
choco-cleaner
choco uninstall choco-cleaner
```

HARDWARE

drivers / platform specific

- Other windows_platform_specific_tools

choco

```
choco install virtio-drivers; choco pin add -n virtio-drivers
choco install qemu-guest-agent
choco install vmware-tools
choco install virtualbox-guest-additions-guest.install
choco install lenovo-thinkvantage-system-update
choco install sdio  #download index only > select all > install
```

winget

```
wg install 9WZDNCRFJ4MV  # lenovo vantage
```

DEVELOP

```
# common
choco install python              # python
choco install vscode.install      # vscode
choco install javaruntime         # java latest runtime
choco install temurin8            # java8 runtime
choco install adoptopenjdk13      # jdk 13
# java8 with Java WebStart
$f="FeatureMain,FeatureEnvironment,FeatureJarFileRunWith,"
$f+="FeatureJavaHome,FeatureIcedTeaWeb,FeatureJNLPFileRunWith"
choco install Temurin8 --params="/both /ADDLOCAL=$f /INSTALLDIR=$env:Programfiles\Eclipse
Adoptium\ /quiet"
```

Win7/8/8.1/2012R2

```
<# old windows 7/8 #>
choco install powershell
choco install conemu
```

old firefox

```
choco install firefoxesr --version=115.9.1 --params "/NoAutoUpdate"
choco pin add -n firefoxesr
```

old chrome
ver 109