

NWinfo

Ask Zread
Ask DeepWiki

NWinfo is a Win32 program that allows you to obtain system and hardware information.

Features

- Retrieves detailed information about SMBIOS, CPUID, S.M.A.R.T., PCI, EDID, and more.
- Supports exporting in JSON, YAML, and HTML formats.
- Gathers information directly without relying on WMI.

Download

- Source code: https://github.com/a1ive/nwinfo
- Latest Release: GitHub Releases
- Nightly Build: Github Actions | Direct Link
- Libraries: Direct Link

CLI (nwinfo)

Usage

```
.\nwinfo.exe OPTIONS
```

Example Commands

```
.\nwinfo.exe --format=json --output=report.json --cp=UTF8 --sys --disk
```

Exports system and disk information to `report.json` in JSON format.

```
.\nwinfo.exe --pci=03
```

Prints information about all PCI devices in the class code `03` (Display Controllers).

```
.\nwinfo.exe --format=html --output=report.html --net=active,phys,ipv4
```

Exports active physical network interfaces with IPv4 addresses to `report.html` in HTML format.

General Options

- --format=`FORMAT`
  Specify output format.
  `FORMAT` can be `YAML` (default), `JSON`, `LUA`, `TREE`, or `HTML`.
- --output=`FILE`
  Write to `FILE` instead of printing to screen.
- --cp=`CODEPAGE`
  Set the code page of output text.
  `CODEPAGE` can be `ANSI` or `UTF8`.
- --human
  Display numbers in human readable format.
- --bin=`FORMAT`
  Specify binary format.
  `FORMAT` can be `NONE` (default), `BASE64` or `HEX`.
- --debug
  Print debug info to stdout.
- --hide-sensitive
  Hide sensitive data (MAC & S/N).

Hardware Details

- --cpu[=`FILE`]
  Print CPUID info.
  Driver is required to access sensors (e.g. temperature).
  Intel, AMD and VIA/Zhaoxin CPUs are supported.
  `FILE` specifies the file name of the CPUID dump.
- --net[=`FLAG,...`]
  Print network info.
  - `GUID`
    Specify the GUID of the network interface, e.g. `{B16B00B5-CAFE-BEEF-DEAD-001453AD0529}`
  - `FLAGS`
    `ACTIVE` Exclude inactive network interfaces.
    `PHYS` Exclude virtual network interfaces.
    `ETH` Include Ethernet network interfaces.
    `WLAN` Include IEEE 802.11 wireless addresses.
    `IPV4` Show IPv4 addresses only.
    `IPV6` Show IPv6 addresses only.
- --acpi[=`SGN`]
  Print ACPI info.
  Driver is required to access some ACPI tables.
  `SGN` specifies the signature of the ACPI table, e.g. `FACP` (Fixed ACPI Description Table).
- --smbios[=`TYPE`]
  Print SMBIOS info.

`TYPE` specifies the type of the SMBIOS table, e.g. `2` (Base Board Information).

- --disk[=`FLAG,..`]
  Print disk info.
  - `PATH`
    Specify the path of the disk, e.g. `\\.\PhysicalDrive0`, `\\.\CdRom0`.
  - `FLAGS`
    `NO-SMART` Don't print disk S.M.A.R.T. info.
    `PHYS` Exclude virtual drives.
    `CD` Include CD-ROM devices.
    `HD` Include hard drives.
    `NVME` Include NVMe devices.
    `SATA` Include SATA devices.
    `SCSI` Include SCSI devices.
    `SAS` Include SAS devices.
    `USB` Include USB devices.
- --smart=`FLAG,...`
  Specify S.M.A.R.T. features.
  Features enabled by default:
  `WMI`, `ATA`, `NVIDIA`, `MARVELL`, `SAT`, `SUNPLUS`, `IODATA`, `LOGITEC`, `PROLIFIC`, `USBJMICRON`, `CYPRESS`, `MEMORY`, `JMICRON`, `ASMEDIA`, `REALTEK`, `MEGARAID`, `VROC`, `ASM1352R` and `HIDERAID`.
  Use `DEFAULT` to specify the above features.
  Other features are `ADVANCED`, `HD204UI`, `ADATA`, `NOWAKEUP` and `RTK9220DP`.
- --display[=`FILE`]
  Print EDID info.
  `FILE` specifies the file name of the SPD dump.
- --pci[=`CLASS`]
  Print PCI info.
  `CLASS` specifies the class code of PCI devices, e.g. `0C05` (SMBus).
- --spd[=`FILE`]
  Print DIMM SPD info.
  Driver is required to access SPD data.
  ⚠ This option may damage the hardware.
  `FILE` specifies the file name of the SPD dump.
- --usb
  Print USB info.
- --battery
  Print battery info.
- --uefi[=`FLAG,..`]
  Print UEFI info.
  - `FLAGS`
    `MENU` Print UEFI boot menus.
    `VARS` List all UEFI variables.
- --audio
  Print audio devices.

- --gpu
  Print GPU utilization and sensors (e.g. temperature).
  GPU drivers are required to access this information.
  NVIDIA (NVAPI), AMD (ADL2) and Intel (IGCL) are supported.
- --device[=`TYPE`]
  Print device tree.
  `TYPE` specifies the type of the devices, e.g. `ACPI`, `SWD`, `PCI`, or `USB`.
- --sensors[=SRC,..]
  Print sensors.
  `SRC` specifies the provider of sensors.
  Available providers are:
  `LHM`, `HWINFO`, `GPU-Z`, `CPU`, `DIMM`, `GPU`, `SMART`, `NET` and `IMC`.

System Information

- --sys
  Print system info.
- --shares
  Print network mapped drives and shared folders.
- --public-ip
  Print public IP address.
- --product-policy[=`NAME`]
  Print ProductPolicy.
  `NAME` specifies the name of the product policy.
- --font
  Print installed fonts.

PowerShell Script for System Diagnostics

`hw_report.ps1` is a PowerShell script designed to generate and display a detailed hardware and system report using `nwinfo`.

You might need to temporarily allow script execution by running the following command:

```
Set-ExecutionPolicy -Scope Process -ExecutionPolicy Bypass
```

Supported Hardware

CPU

Vendor	CPUID	Temperature	Voltage	Power	Clock
Intel	✔	✔	✔	✔	✔
AMD	✔	✔	✔	✔	✔
VIA / Zhaoxin	✔	✔	✖	✖	✖

GPU

Vendor	API	GPU Usage	VRAM	Temperature	Power	Frequency	Voltage	Fan Speed
NVIDIA	NVAPI	✔	✔	✔	✔	✔	✔	✔
AMD	ADL2	✔	✔	✔	✔	✔	✔	✔
Intel	IGCL	✔	✔	✔	✔	✔	✔	✔
Generic	D3D	✔	✔	✔	✔	✖	✖	⚠

Notes:

- `VRAM` refers to the dedicated video memory only.
- `Frequency` refers to the GPU core frequency.
- `Power` refers to the board power draw.

Memory Module SPD

- SMBus: Intel PCH, PIIX4 / AMD SB / Hygon
- Memory Module: SDR, DDR, DDR2, DDR3, DDR4, DDR5
- Thermal Sensor: DDR4, DDR5

HDD / SSD S.M.A.R.T.

NWinfo uses libcdi to access S.M.A.R.T. data.

`libcdi` is a dynamic library based on CrystalDiskInfo.

Note: NVMe requires Windows 10 or later.

Supported Drivers

The program searches for and loads drivers in the following order: **CPUZ161 -> EVGA -> HwRwDrv -> WinRing0 -> PawnIO**.

Driver	Filename	Security Status	Bundled	CPU Sensor	SPD	ACPI
PawnIO	PawnIO.sys	✔ Safe to use	✔	⚠	✔	✖
HwRwDrv	HwRwDrvx64.sys	✖ Vulnerable	✖	✔	✔	✔
WinRing0	WinRing0x64.sys	✖ Blocked by AV	✖	✔	✔	⚠
CPUZ161	cpuidx64.sys	✔ Safe to use	✖	✔	✔	✔
EVGA	HwIox64.sys	✔ Safe to use	✖	✔	✔	✖

Note: The program can still run normally even without drivers, but some hardware information may not be accessible.

PawnIO Driver Installation

Install the PawnIO driver silently using the following command:

```
.\PawnIOSetup.exe -install -silent
```

Uninstall the PawnIO driver silently using the following command:

```
.\PawnIOSetup.exe -uninstall -silent
```

File List

This section describes all files included in the final release package.

File Name	Category	Description
`nwinfo.exe`	Executable	Main executable (x64)
`nwinfox86.exe`	Executable	Main executable (x86)
`gnwinfo.exe`	Executable	GUI executable (x64)
`gnwinfox86.exe`	Executable	GUI executable (x86)
`libcdi.dll`	Library	S.M.A.R.T. data access library (x64)
`libcdix86.dll`	Library	S.M.A.R.T. data access library (x86)
`hw_report.ps1`	Script	Example PowerShell script
`gnwinfo.ini`	Configuration	Configuration file for the GUI
`pci.ids`	Database	PCI database
`usb.ids`	Database	USB database
`pnp.ids`	Database	PnP (monitor) vendor database
`jep106.ids`	Database	JEDEC memory module vendor database
`PawnIOSetup.exe`	Driver	PawnIO driver installer (x64)
`IntelMSR.bin`	PawnIO Module	Intel MSR module for PawnIO driver
`AMDFamily0F.bin`	PawnIO Module	AMD K8 MSR module for PawnIO driver
`AMDFamily10.bin`	PawnIO Module	AMD K10 MSR module for PawnIO driver
`AMDFamily17.bin`	PawnIO Module	AMD Zen MSR module for PawnIO driver
`RyzenSMU.bin`	PawnIO Module	AMD Ryzen SMU module for PawnIO driver
`SmbusPIIX4.bin`	PawnIO Module	PIIX4 SMBus module for PawnIO driver
`SmbusI801.bin`	PawnIO Module	I801 SMBus module for PawnIO driver

Licenses & Credits

This project is licensed under the Unlicense license.

- libcpuid
- CrystalDiskInfo
- Nuklear
- stb
- optparse
- hwdata
- PawnIO