

Theory

Reading time: ~20 minutes · Prerequisites: Lesson 01 (Introduction to Hashcat), basic command-line familiarity

1. System Requirements Overview

Hashcat is an advanced password recovery tool designed to leverage massively parallel computing hardware. While it can run on modest hardware, its true potential is unlocked with modern GPUs. Understanding the hardware requirements helps you plan your setup and set realistic expectations for cracking performance.

CPU Requirements

Hashcat primarily uses the GPU for hash computation, but the CPU handles important tasks: managing the attack workflow, applying rules, distributing work across devices, and serving as a fallback compute device when no GPU is available.

- **Architecture:** x86_64 (64-bit) is required. 32-bit systems are not supported.
- **Instruction sets:** AVX and AVX2 support significantly accelerates CPU-based operations. AVX-512 provides additional gains on supported processors. Without at least SSE2, hashcat will not run.
- **Recommended CPUs:** Intel Core i7/i9 (10th generation or later) or AMD Ryzen 7/9. These provide sufficient cores and modern instruction set extensions.
- **Core count:** At least 4 cores recommended. More cores help when running multi-device attacks or when the CPU itself is used as a compute device.

GPU Requirements

The GPU is the primary compute engine in hashcat. Modern GPUs contain thousands of shader cores that can compute hashes in parallel, making them orders of magnitude faster than CPUs for most hash types.

NVIDIA GPUs

NVIDIA GPUs are supported through both CUDA and OpenCL. CUDA generally provides better performance with hashcat on NVIDIA hardware.

- **Minimum:** GTX 1060 6GB (Pascal architecture) with driver version 440.0+
- **Recommended:** RTX 3080 10GB or RTX 3090 24GB (Ampere architecture)
- **Professional:** RTX 4090 24GB or A100 80GB for maximum throughput
- **VRAM:** 6 GB minimum; 10 GB+ recommended. Some hash types and large wordlist-based attacks require significant VRAM.

AMD GPUs

AMD GPUs are supported through OpenCL and ROCm (on Linux). Performance has improved significantly with recent driver updates.

- **Minimum:** RX 580 8GB (Polaris architecture)
- **Recommended:** RX 6800 XT 16GB or RX 6900 XT 16GB (RDNA 2)
- **Professional:** RX 7900 XTX 24GB or Instinct MI250
- **Drivers:** Adrenalin 21.5+ on Windows, ROCm 4.0+ on Linux

Apple Silicon (M1/M2/M3)

Hashcat has experimental Metal support on Apple Silicon, but performance is significantly lower than dedicated NVIDIA or AMD GPUs. Apple's unified memory architecture is not optimised for the massively parallel workloads hashcat requires. For production cracking, a dedicated GPU rig running Linux is strongly recommended.

RAM and Storage

- **RAM:** 4 GB minimum, 16 GB+ recommended. Large wordlists and rule files are loaded into memory. A 10 GB wordlist (like the full RockYou2024) will require proportional RAM.
- **Storage:** 20 GB free space minimum. SSD is strongly recommended — wordlist loading from an HDD can take minutes instead of seconds for large files. NVMe is ideal for multi-gigabyte wordlists.

Operating System

Hashcat requires a 64-bit operating system:

- **Linux:** Ubuntu 20.04+, Kali Linux, Parrot OS, or any distro with modern GPU drivers. Linux generally provides the best performance and driver support.
- **Windows:** Windows 10 or 11 (64-bit). Ensure GPU drivers are current.
- **macOS:** macOS 11 (Big Sur) or later. Limited GPU acceleration due to Apple's deprecation of OpenCL.

2. Installation on macOS

macOS offers two installation methods: Homebrew (recommended for most users) and building from source (for access to the latest development features).

Method 1: Homebrew

Homebrew is the simplest approach. It handles dependencies automatically and makes updates straightforward.

```
# Install Homebrew if you don't have it
/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/

# Install hashcat
brew install hashcat

# Verify the installation
hashcat --version
# Expected output: v6.2.6 (or later)

# List available compute devices
hashcat -I
```

Method 2: Build from Source

Building from source gives you access to the latest commits and development features before they are packaged into a release.

```
# Ensure build tools are available
xcode-select --install
```

```
# Clone the repository
git clone https://github.com/hashcat/hashcat.git
cd hashcat

# Checkout a specific version (optional, for stability)
git checkout v6.2.6

# Build using all available cores
make -j$(sysctl -n hw.ncpu)

# Verify
./hashcat --version

# Optionally install system-wide
sudo make install
```

macOS OpenCL Deprecation

Apple deprecated OpenCL in macOS 10.14 and removed it from the default SDK in macOS 12. Hashcat uses OpenCL as its primary compute API, which means GPU acceleration on macOS is increasingly limited. The hashcat team has been working on Metal backend support, but it is not yet feature-complete. If you need full GPU performance, consider dual-booting Linux or using a dedicated Linux machine.

3. Installation on Linux

Linux is the recommended platform for serious hashcat use. It provides the best GPU driver support and performance.

Method 1: Package Manager

```
# Debian / Ubuntu
sudo apt update
sudo apt install hashcat

# Fedora / RHEL
sudo dnf install hashcat
```

```
# Arch Linux
sudo pacman -S hashcat

# Verify
hashcat --version
```

Kali Linux

Hashcat comes pre-installed on Kali Linux. Simply update to the latest version with `sudo apt update && sudo apt upgrade hashcat`.

Method 2: Build from Source

```
# Install build dependencies
sudo apt install build-essential git

# Clone and build
git clone https://github.com/hashcat/hashcat.git
cd hashcat
make -j$(nproc)
sudo make install

# Verify
hashcat --version
```

GPU Driver Setup (Linux)

GPU drivers must be installed before hashcat can use GPU acceleration. Without drivers, hashcat will only see the CPU as a compute device.

NVIDIA Drivers

```
# Ubuntu: Use the recommended driver
sudo ubuntu-drivers autoinstall
sudo reboot

# Verify NVIDIA driver
nvidia-smi
```

```
# Install CUDA toolkit (optional, for CUDA backend)
sudo apt install nvidia-cuda-toolkit
```

AMD Drivers (ROCm)

```
# Add ROCm repository (Ubuntu 22.04 example)
wget -q -O - https://repo.radeon.com/rocm/rocm.gpg.key | sudo apt-key add -
echo 'deb [arch=amd64] https://repo.radeon.com/rocm/apt/latest/ ubuntu main' |
sudo apt update
sudo apt install rocm-ocl-runtime

# Verify
rocm-smi
hashcat -I
```

4. Installation on Windows

Windows installation uses pre-compiled binaries downloaded from the official hashcat website. No compilation is required.

```
# 1. Download the latest release from:
#   https://hashcat.net/hashcat/
#   (Choose the .7z or .zip archive)

# 2. Extract the archive to a permanent location
#   Example: C:\Tools\hashcat

# 3. Open Command Prompt or PowerShell
cd C:\Tools\hashcat

# 4. Verify installation
.\hashcat.exe --version

# 5. List compute devices
.\hashcat.exe -I
```

Adding to PATH (Windows)

To use hashcat from any directory, add the hashcat folder to your system PATH:

1. Open **System Properties** → **Advanced** → **Environment Variables**
2. Under **System variables**, select **Path** and click **Edit**
3. Add `C:\Tools\hashcat` (or wherever you extracted it)
4. Click **OK** and restart your terminal

GPU Drivers on Windows

Ensure you have the latest GPU drivers installed:

- **NVIDIA:** Download from [nvidia.com/drivers](https://www.nvidia.com/drivers). Use the Game Ready or Studio driver. Verify with `nvidia-smi` in Command Prompt.
- **AMD:** Download Adrenalin drivers from [amd.com/en/support](https://www.amd.com/en/support). Ensure OpenCL runtime is included in the installation.

5. Verifying Your Installation

After installation, run these commands to confirm everything is working correctly:

```
# Check hashcat version
hashcat --version

# List all detected compute devices
hashcat -I

# Expected output (example with NVIDIA GPU):
# OpenCL Platform ID #1
#   Vendor...: NVIDIA Corporation
#   Name.....: NVIDIA CUDA
#   Version..: OpenCL 3.0 CUDA 12.2.138
#
# Backend Device ID #1
#   Type.....: GPU
#   Vendor.ID.....: 32
#   Name.....: NVIDIA GeForce RTX 3080
#   Memory.Total....: 10240 MB
#   Driver.Version.: 535.129.03

# Run a quick benchmark on MD5
hashcat -b -m 0
```

```
# Run a quick test crack
echo -n "password" | md5sum
# Output: 5f4dcc3b5aa765d61d8327deb882cf99
echo "5f4dcc3b5aa765d61d8327deb882cf99" > /tmp/test.hash
```

6. Environment Configuration

Environment variables allow you to set default behaviours without passing flags on every command. This is especially useful for automation and scripting.

Variable	Purpose	Example Value
<code>HASHCAT_HOME</code>	Base directory where hashcat looks for modules, rules, and charsets	<code>/opt/hashcat</code>
<code>HASHCAT_LOG</code>	Directory for hashcat log files	<code>\$HOME/.hashcat/logs</code>
<code>HASHCAT_GPU</code>	Default GPU device ID(s) to use	<code>1</code> or <code>1,2</code>
<code>HASHCAT_CPU</code>	Enable CPU-only mode (set to 1)	<code>0</code> (default)
<code>HASHCAT_FORCE</code>	Bypass device type and capability warnings	<code>1</code> (use with caution)
<code>HASHCAT_WORKLOAD_PROFILE</code>	Default workload tuning profile (1=low, 2=default, 3=high, 4=nightmare)	<code>3</code>

Setting Up on Linux / macOS

```
# Add to ~/.bashrc, ~/.zshrc, or ~/.profile
export HASHCAT_HOME="/opt/hashcat"
export HASHCAT_LOG="$HOME/.hashcat/logs"
export HASHCAT_WORKLOAD_PROFILE=3

# Create the log directory
mkdir -p "$HOME/.hashcat/logs"
```

```
# Reload your shell configuration
source ~/.bashrc # or source ~/.zshrc
```

Setting Up on Windows

```
# PowerShell (persistent, user-level)
[Environment]::SetEnvironmentVariable("HASHCAT_HOME", "C:\Tools\hashcat", "User")
[Environment]::SetEnvironmentVariable("HASHCAT_LOG", "$env:USERPROFILE\.hashcat", "User")
[Environment]::SetEnvironmentVariable("HASHCAT_WORKLOAD_PROFILE", "3", "User")

# Create the log directory
New-Item -ItemType Directory -Force -Path "$env:USERPROFILE\.hashcat\logs"
```

Workload Profiles Explained

Profile	Name	GPU Impact	Use Case
1	Low	Minimal — desktop remains responsive	Background cracking while using the computer
2	Default	Moderate	General use on a workstation
3	High	Heavy — desktop may stutter	Dedicated cracking machine
4	Nightmare	Maximum — system nearly unusable	Headless server / dedicated rig

The --force Flag

Setting `HASHCAT_FORCE=1` or using `--force` on the command line bypasses all device warnings and capability checks. This can lead to **incorrect results**, **hardware damage** (overheating), or **crashes**. Only use it when you understand exactly why hashcat is warning you and have confirmed the warning does not apply to your situation.

7. Directory Structure

A well-organised hashcat working directory will save you time as your projects grow:

```
~/hashcat-workspace/
hashes/           # Hash files to crack
wordlists/        # Dictionary files (rockyou.txt, etc.)
rules/            # Rule files (best64.rule, etc.)
masks/            # Custom mask files
potfiles/         # Cracked hash results
sessions/         # Session restore files
logs/             # Benchmark and run logs
```

```
# Create the workspace
mkdir -p ~/hashcat-workspace/{hashes,wordlists,rules,masks,potfiles,sessions,logs}
```

8. Troubleshooting Common Issues

hashcat: command not found

The hashcat binary is not in your PATH. Either add the installation directory to PATH, use the full path to the binary, or create a symlink:

```
sudo ln -s /opt/hashcat/hashcat /usr/local/bin/hashcat
```

No devices found

GPU drivers are not installed or not loading correctly. Run `nvidia-smi` (NVIDIA) or `rocm-smi` (AMD) to check driver status. On macOS, ensure you have not removed the OpenCL framework.

CL_OUT_OF_RESOURCES or similar GPU errors

The GPU ran out of memory or hit a compute timeout. Try reducing the workload profile (`-w 1`), using a smaller wordlist, or increasing the GPU's TDR (timeout detection recovery) limit on Windows.

Benchmark speeds are lower than expected

- Ensure no other GPU-intensive applications are running (games, other compute workloads)

- Check GPU temperatures — thermal throttling reduces performance. Target below 80 degrees C.
- Verify you are using the correct workload profile (`-w 3` or `-w 4` for maximum speed)
- Update GPU drivers to the latest version

Further Reading

- [Hashcat FAQ](#) — Official FAQ covering common installation and runtime issues.
- [Hashcat Wiki](#) — Comprehensive documentation for all features and options.
- [Hashcat on GitHub](#) — Source code, issue tracker, and build instructions.
- [Benchmark Reference](#) — Community-submitted benchmark results for various hardware configurations.

Next lesson: **Lesson 03 — Command Structure & Parameters** — understanding hashcat's command-line syntax and core options.