



uProf Release Notes

Publication Number: 63856 Revision: 5.1

Date: June 2025

Contents

Chapter 1: Release Highlights	3
Chapter 2: Supported Processors	5
Chapter 3: Supported Operating Systems	7
Chapter 4: System Requirements	8
4.1 CPU Profiling	8
4.2 GPU Profiling and Tracing	8
4.3 OS Tracing	8
Chapter 5: Known Issues	9
5.1 Installer	9
5.2 CPU Profiling	9
5.3 HPC - MPI Tracing	10
5.4 Power Profiling	10
5.5 AMDuProfPcm	10
5.6 AMDuProfSys	11
5.7 GPU Profiling and Tracing	11
5.8 Others	11
Chapter 6: Limitations	12
6.1 CPU Profiling	12
Chapter 7: Support	13
Appendix A: Additional Resources and Legal Notices	14
A.1 Revision History	14
Appendix B: Notices	15
B.1 Trademarks	15

Chapter 1: Release Highlights

- AMDuProfPCM
 - `profile` command to generate timeseries, cumulative and roofline CSV and HTML reports from single run
 - `hreport` command to generate HTML reports from timeseries CSV report
 - New CLI options `--collect-host` and `--collect-guest` for monitoring host and guest metrics from the host system
 - L3 Access/Misses (per thousand instructions {pti}) metrics
 - Set required capabilities for MSR mode using `AMDPcmSetCapability.sh`
 - Monitor NUMA nodes with `-c numa=<node#>`
 - New Python script has been implemented to divide metrics in csv file based on IP or aggregation type
 - Support for comparing roofline sessions has been added
- New Platform Support
 - Rapheal X3D
 - Granite Ridge X3D
 - Ryzen EPYC 4004
 - Ryzen EPYC Server X3D 4004
 - Ryzen EPYC 4005
- Hotspots Analysis
 - `TOTAL_CPU_TIME` of functions if callstack is enabled
 - Hotspots support is enabled on Windows (GUI and CLI)
- Threading Analysis
 - Java support
 - Profile control API support
 - Pause and Resume from GUI
 - CPU Profile Samples swimlane will not display in Timeline views when the session is collected with OS trace data
- OS Tracing
 - Removing `--trace os` and `--trace user` options
 - New CLI options for function tracing (`--trace func`) and memory tracing (`--trace memory`)
 - Run the script `AMDuProfSetup.sh` with root user to setup the OS tracing.

`AMDuProfSetBpfCapability.sh` is deprecated

- Memory tracing, diskio tracing, kernel function tracing without root access
- Profile Control APIs and Instrumentation APIs
 - Supports strict Pause/Resume using `AMDProfileStrictPause()` and `AMDProfileStrictResume()` APIs
 - Supports `Attach to Process` with Instrumentation APIs
 - Supports different configurations like `ibs`, `assess`, etc. with Instrumentation APIs
 - Supports Pause/Resume control APIs from Python script
 - Added Task Hotspot Summary table in GUI
- CpuProfiler
 - Support for new OS - Windows Server® 2025, Windows® 11 - 24H2
 - Faster sample data processing and report generation
- HPC
 - MPI Fortran-2008 support for MPICH implementation
 - Support for additional Open MPI APIs in Fortran-90 and Fortran-2008
 - Display of OpenMP parallel region instance execution count
- GPU Tracing
 - Latest ROCm version support (until ROCm 6.3.3)
- GPU Profiling
 - Latest ROCm version support (until ROCm 6.3.3)
- UProfSys
 - `--wait-for-signal` for start/stop control using SWIFT signal
 - By default, Summary report is enabled for all Timeseries reports
- Other updates
 - GUI shows profiling overhead for selected profile type
 - Added option in GUI to re-direct application `stdout/stderr` output to a file
 - Numerous timeline improvements in GUI
 - Specify MPI processes by rank, and OpenMP threads by OpenMP thread number in GUI

Chapter 2: Supported Processors

AMD uProf supports the following processors.

Table 2.1: Supported Processors

Segment	Processor	CPU Profiling	Live Power Profiling	uProfPc m	uProfSy s	OS Runtime Tracing	HPC
Server	AMD EPYC™ 9005 and 97x5	Yes	Yes	Yes	Yes	Yes (Linux)	Yes (Linux)
	AMD EPYC™ 9004 and 97x4	Yes	Yes	Yes	Yes	Yes (Linux)	Yes (Linux)
	AMD EPYC™ 7003	Yes	Yes	Yes	Yes	Yes (Linux)	Yes (Linux)
	AMD EPYC™ 7002	Yes	Yes	Yes	Yes	Yes (Linux)	Yes (Linux)
	AMD EPYC™ 7001	Yes	Yes	Yes	Yes	Yes (Linux)	Yes (Linux)
Workstation	AMD Ryzen™ Threadripper™	Yes	Yes	Yes	Yes	Yes (Linux)	Yes (Linux)
	AMD Ryzen™ PRO	Yes	Yes	Yes	Yes	Yes (Linux)	Yes (Linux)
Desktop / Mobile	AMD Ryzen™ 9000 series	Yes	Yes	Yes	Yes	Yes (Linux)	Not applicable
	AMD Ryzen™ 7000 series	Yes	Yes	No	No	Yes (Linux)	Not applicable
	AMD Ryzen™ 5000 series	Yes	Yes	No	No	Yes (Linux)	Not applicable
	AMD Ryzen™ Server EPYC 4004 Series (EPYC 4384PX)	Yes	Yes	Yes	Yes	Yes (Linux)	Not applicable

Table 2.1: Supported Processors (continued)

Segment	Processor	CPU Profiling	Live Power Profiling	uProfPc m	uProfSy s	OS Runtime Tracing	HPC
	Grado EPYC-4005 (EPYC 4565P)	Yes	Yes	Yes	Yes	Yes (Linux)	Not applicab le
	AMD Ryzen™ 3000 series	Yes	Yes	No	No	Yes (Linux)	Not applicab le

 **Note:** GPU Profiling and Tracing support AMD Instinct™ MI200 and MI300A accelerators.

Chapter 3: Supported Operating Systems

AMD uProf supports the 64-bit version of the following **operating systems**.

- Microsoft®
 - Windows® 10 (up to 22H2)
 - Windows® 11 (up to 24H2)
 - Windows Server® 2019, Windows Server® 2022, and Windows Server® 2025
- Linux
 - Ubuntu® 22.04 and later
 - RHEL® 8.6 and later
 - SLES & openSUSE® Leap 15.5*, Debian 12
 - RHEL based distros - Rocky Linux 9.3*, Alma Linux 9.4
- FreeBSD® 13, FreeBSD® 14

Virtualization

- Linux KVM
- Windows Hyper-V
- VMware ESXi
- Citrix Xen

Cloud Environments

- AWS
- Azure

Containers

- Docker (on Linux)

For OS support on AMD EPYC™ processors, see this [webpage](#).

- * Sanity tested. Support subject to commitment to compatibility with Red Hat Enterprise Linux.
- Version 5.1 is the last version of uProf that supports RHEL 7.x. Support for RHEL 7.x will be deprecated in future releases of uProf.

Chapter 4: System Requirements

AMD uProf contains a host of development features with different system requirements.

4.1 CPU Profiling

- Time-Based Profiling can be performed on AMD and other x86-64-compatible processors.
- The Event-Based Profiling (EBP) and Instruction-Based Sampling (IBS) session types require an AMD CPU or APU.
- CPU Profiling on Linux platforms - Limitations of Linux perf:
 - For AMD “Zen” microarchitecture processors, EBP and IBS profiling on Linux requires Linux kernel 4.9 or later.
 - Call chain analysis on Linux depends on the call chain information provided by Linux perf. This requires the profiled binaries to have stack frame pointer (that is, compiled with `-fno-omit-frame-pointer`).
 - Timer and EBP profiling works with the default `/proc/sys/kernel/perf_event_paranoid (2)` to profile any user-space application.
 - Timer and EBP profiling require `perf_event_paranoid (<= 1)` to profile the kernel space and `perf_event_paranoid (<= 0)` to profile the entire system.
 - IBS profile requires `perf_event_paranoid(<= 0)` due to HW limitations; IBS cannot be configured to profile any specific process.
- IBS may not be enabled by default. Check BIOS settings to enable it.

4.2 GPU Profiling and Tracing

GPU profiling and tracing uses AMD ROCm 6.3.3 or lower versions. For steps to install AMD ROCm, refer to the [AMD uProf User Guide](#).

4.3 OS Tracing

- OS tracing (on Linux) requires BPF Compiler Collection (BCC) and eBPF. Refer to the AMD uProf 5.0 user guide for the steps to install these dependencies.
- OS tracing requires Linux kernel 4.7 or later. We recommended you use kernel 4.15 or later.

Chapter 5: Known Issues

Here is a list of known issues.

5.1 Installer

1. On a Windows system with Microsoft Visual Studio versions 2019 or 2022 installed, and on which Microsoft Visual C++ redistributable package was uninstalled for any reason:
 - a. AMD uProf installation may not detect that the redistributable package is missing and thus fails to install the requisite redistributable package.
 - b. We recommend you verify or initiate installation of the Microsoft Visual C++ redistributable package before installing AMD uProf on a Windows system.
2. On Linux, while upgrading AMD uProf using RPM package installer, the upgrade process might leave some folders of the previous version in the `/opt/AMDuProf_<old version>` directory. These files or folders from a previous version will have no impact on the upgraded version of AMD uProf.

5.2 CPU Profiling

- Some samples may be missed when performing a combined PMC and Timer-based sampling profile run.
- If AMD uProf is installed in a path that includes non-ASCII Unicode characters, profiling does not work.
- The CPU Profiler does not support profiling of a Windows Store app.
- The CPU Profiler report command invocation with `--symbol-server` and `--symbol-cache-dir` options may crash if **Ctrl-C** is pressed.
- Unsupported instructions (for example, VDPBF16PS) are shown as BAD DASM in IMIX and Source view.
- Occasionally, core PMC samples may not be collected while profiling on KVM guest environment.
- On an ESXi guest environment, a high number of samples is collected in comparison to the same application running on a bare-metal system.
- Custom profiling with IBS and TBP or EBP events might show zero or fewer samples in the report. To overcome this issue, do not combine IBS with the other profiling events (TBP or EBP) during the custom profiling.
- `libclang-cpp.so` library bundled with AOCC 3.2 is not CXX11 ABI compliant. AMD uProf OS tracing functionality depends on libclang library (with abi cxx11 support). While running applications compiled with AOCC 3.2, the `LD_LIBRARY_PATH` points to AOCC 3.2 `bin/lib` paths and hence, AMD uProf fails giving an undefined symbols error. This issue is not observed if `LD_LIBRARY_PATH` is not set to AOCC 3.2 libs.
- For Java applications running for a long time (more than an hour), the sample data attributed to Java

methods might not be accurate.

- IBS and MPI profiling are not recommended to be run concurrently as a very large dataset will accumulate.
- GUI may crash when the thread timeline option is selected for importing a session using the option `--category cpu`.
- CPU Profiling is not supported on FreeBSD-14.
- CPU Profiler does not support more than 256 cores on FreeBSD platforms.
- On KVM-Windows, uProf may fail to generate report.
- uProf may fail to profile custom config events if the `perf_event_mlock_kb` value is > 516. As a workaround, you can increase the mlock limit through root user.
- IMIX Summary report fails to build report in csv format for Java application.

5.3 HPC - MPI Tracing

The data volume shown in an MPI report is not evenly distributed among all the ranks involved for the MPI APIs performing more than one communication operation. For example, `MPI_SendRecv`, `MPI_Waitall`, and so on, due to which, the MPI report might not match with other MPI tracing tools evenly distributing the volume.

5.4 Power Profiling

- If the profiled system goes into Sleep/Hibernate state during a Live Power Profiling session only data collected before hibernation is displayed, and the navigation slider does not respond.
- Energy analysis is not supported.
- Power Profiling does not work as intended if trying to collect data from offline cores.
- On a Windows Hyper-V enabled server system, the Power Profiling equivalent CLI command is missing in the Live Power Profiling UI.
- On Genoa AWS virtual machine and Windows Hyper-V systems, uProf GUI crashes on profiling displayed PP counters.
- On Windows, Power Profile API is broken.

5.5 AMDuProfPcm

- Collecting multiple groups of Core PMC based metrics across all the cores in root mode may result in reduced accuracy of the collected data.
- All virtual machine (VM) configurations have not been tested.
- On Windows, the Roofline model may display incorrect bandwidth data for certain applications.
- uProfPcm reports incorrect topology on Turin Dense with Hyper-V enabled system.
- On Microsoft Azure, uProfPcm reports data on package 0 only for DF metrics.

5.6 AMDuProfSys

- Use the `--metrics` option to collect custom perfCtl based metrics provided in the command line. MSR based events (aperf, mperf, irperf, and tsc) are not allowed to collect without at least one valid PMC event in the given metrics. In such a case, behavior will be undefined.
- While using the `--use-linux-perf` flag, you cannot use the options `-C` and `-p` together in Linux.
- All VM configurations have not been tested.
- uProfSys throws an "invalid token" error when generating a report for default metric using `--use-linux-perf` + `--group-by` options.
- Data inconsistency may be observed for core metrics when data is collected for core + I3 (multiple metrics) using `--use-linux-perf`.

5.7 GPU Profiling and Tracing

- Core dump may be generated sporadically when GPU data is collected using the `profile` option.
- GUI may crash sporadically during translation when the GPU Profile session generated using CLI is imported.

5.8 Others

- Importing only GPU traces is not supported.
- AMDuProf crashes when a timer event is selected through the filter option in the Custom configuration page.

Chapter 6: Limitations

Here is a list of limitations.

6.1 CPU Profiling

- CPU Profiling (EBP and IBS) is disabled on Windows platforms when Hyper-V is enabled.
- Profiling of Java apps running on 32-bit JVM on the Linux platform is not supported.
- On Linux, IBS Fetch profiling shows extremely low number of samples on AMD “Zen1 and Zen2” generation processors.
- OpenMP Tracing is not supported on RHEL 7.x.
- OpenMP tracing does not support tracing through attach to process.
- OpenMP parallel region name resolution is supported only when OpenMP tracing and CPU Profiling are performed together.
- On Windows, AMD uProf does not support profiling through attach to a process running under a different account (for example, SYSTEM, NETWORK, and another user). To profile such a process, launch AMD uProf under the same account as the target process.
- On RHEL8.2, IBS and cache analysis profiles throws NMI errors.
- In the Cache analysis report on the AMD “Zen3” generation processor, the remote cache hit (M) metric may not be reported.
- Importing GPU profile DB is not supported.

Chapter 7: Support

For support options, latest documentation, and downloads, see the AMD uProf [webpage](#).

Appendix A: Additional Resources and Legal Notices

A.1 Revision History

Date	Revision	Description
June 2025	5.1	Updated this doc for the 5.1 release.
October 2024	5.0	Created this doc for the 5.0 release.

Appendix B: Notices

© Copyright 2022-2025 Advanced Micro Devices, Inc.

The information presented in this document is for informational purposes only and may contain technical inaccuracies, omissions, and typographical errors. The information contained herein is subject to change and may be rendered inaccurate for many reasons, including but not limited to product and roadmap changes, component and motherboard version changes, new model and/or product releases, product differences between differing manufacturers, software changes, BIOS flashes, firmware upgrades, or the like. Any computer system has risks of security vulnerabilities that cannot be completely prevented or mitigated. AMD assumes no obligation to update or otherwise correct or revise this information. However, AMD reserves the right to revise this information and to make changes from time to time to the content hereof without obligation of AMD to notify any person of such revisions or changes.

THIS INFORMATION IS PROVIDED "AS IS." AMD MAKES NO REPRESENTATIONS OR WARRANTIES WITH RESPECT TO THE CONTENTS HEREOF AND ASSUMES NO RESPONSIBILITY FOR ANY INACCURACIES, ERRORS, OR OMISSIONS THAT MAY APPEAR IN THIS INFORMATION. AMD SPECIFICALLY DISCLAIMS ANY IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, OR FITNESS FOR ANY PARTICULAR PURPOSE. IN NO EVENT WILL AMD BE LIABLE TO ANY PERSON FOR ANY RELIANCE, DIRECT, INDIRECT, SPECIAL, OR OTHER CONSEQUENTIAL DAMAGES ARISING FROM THE USE OF ANY INFORMATION CONTAINED HEREIN, EVEN IF AMD IS EXPRESSLY ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

B.1 Trademarks

AMD, the AMD Arrow logo, and combinations thereof are trademarks of Advanced Micro Devices, Inc.

Other product names used in this publication are for identification purposes only and may be trademarks of their respective companies.