HAILO

Empowering Intelligence

Hailo Log Collector User Guide

Release 1.0.3

26 December 2022

Table of Contents

Disclaimer and Proprietary Information Notice

Copyright

This version of the document supersedes all previous versions.

General Notice

Hailo, to the fullest extent permitted by law, provides this document "as-is" and disclaims all warranties, either express or implied, statutory or otherwise, including but not limited to the implied warranties of merchantability, non-infringement of third parties' rights, and fitness for particular purpose.

Although Hailo used reasonable efforts to ensure the accuracy of the content of this document, it is possible that this document may contain technical inaccuracies or other errors. Hailo assumes no liability for any error in this document, and for damages, whether direct, indirect, incidental, consequential or otherwise, that may result from such error, including, but not limited to loss of data or profits.

The content in this document is subject to change without prior notice and Hailo reserves the right to make changes to content of this document without providing a notification to its users.

1. Overview

Hailo Log Collector tool is used to create an informative .zip and .csv files that contains information about Hailo packages, latest Hailo log files and system information.

You are encouraged to use this tool whenever Hailo is contacted - whether for a feature request or for a bug report - to help us address the issue in the most accurate and fast way.

1.1. Details

Hailo releases four main software components:

1. Dataflow Compiler (Model conversion and compilation to Hailo binary format)
2. HailoRT (Runtime environment and driver for running networks and interacting with Hailo devices)
3. Model Zoo (Pre-trained models to run and evaluate on Hailo devices)
4. TAPPAS (Deployment framework, examples and multi-network pipelines)

These components can be download as standalone packages from hailo developer zone, or as part of Hailo Software Suite (whether as a self-extracted package or a Docker file).

When Hailo is contacted, instead of figuring out which files will be of help, this tool automatically collects logs from the above packages, as well as system information, Hailo installation logs and driver issues:

- dmesg log
- Hailo software components information
- System information (CPU, RAM, OS, Kernel version, Python version, PIP version, PCI information)
- .json, .hef, .har, .alls files from a Dataflow Compiler working directory (if specified)
- Collects Hailo Dataflow Compiler logs (DFC v3.20.0 and on)
- Collects HailoRT package & HailoRT PCI Driver installation logs
- Collects Hailo Model Zoo logs (MZ v2.6.0 and on)
- Collects HailoRT Driver logs (HRT v4.12.0 and on)
- Collects pyHailoRT logs (HRT v4.12.0 and on)
- Collects HailoRT Multi-Process Service logs (HRT v4.12.0 and on)

2. Changelog

Hailo Log Collector v1.0.3

- Collect HailoRT Multi-Process Service log

Hailo Log Collector v1.0.2

- Fixed collection of HailoRT logs

Hailo Log Collector v1.0.1

- Output log collected directory will be inside the current directory instead of the given working directory

Hailo Log Collector v1.0.0

- Initial release:
  - Collects dmesg log
  - Collects Hailo software components information
  - Collects System information (CPU, RAM, OS, Kernel version, Python version, PIP version, PCI information)
  - Collects .json, .hef, .har, .alls files from a Dataflow Compiler working directory (if specified)
  - Collects Hailo Dataflow Compiler logs (DFC v3.20.0 and on)
  - Collects HailoRT package & HailoRT PCI Driver installation logs
  - Collects Hailo Model Zoo logs (MZ v2.6.0 and on)
  - Collects HailoRT Driver logs (HRT v4.12.0 and on)
  - Collects pyHailoRT logs (HRT v4.12.0 and on)

3. Installation

The tool is made of two parts:

1. Collect all files, logs and system information into a folder
2. Create a .zip file out of that folder's contents, and also a summary .csv file

The first part is based on Bash script files (.sh), so they could run on almost any Linux-based machine. The second part requires Python because it includes some logic for the .csv file creation.

Because not all machines have Python installed, we have decided to release the tool in two forms:

- **Python package (.whl)**
  - Contains the .sh files and also the report generation logic (.zip, .csv)
  - Use to collect the logs, and also to create the .zip and .csv file
  - Can be used on the output folder of the Bash script, for cases where the folder is created on another machine
- **A .zip file containing only the Bash scripts (.sh)**
  - Use this form whenever you don't have Python installed on your machine (you'll most probably have at least HailoRT installed)
  - Use to create the folder with all of the collected files and information
  - You might copy this folder to another machine with the Python package to create the .zip and .csv files

3.1. Installing the .zip archive

Supported OS

- Linux

Supported Architectures

- x86_64
- aarch64

Installation

- Download the *hailo_log_collector-VERSION.zip* file from hailo developer zone
- Extract the .zip file to create the log collector folder with .sh files
- Change dir into that folder, and view the tool options with: *./log_collector.sh -h*

3.2. Installing the Python package

Pre-requisites

- Python3.6 and above, pip

Supported OS

- Linux

Supported Architectures

- x86_64
- aarch64

Installation

- Download the *hailo_log_collector-VERSION-py3-none_any.whl* file from hailo developer zone
- Run *pip install WHEEL_FILE* (from within any virtual environment)
  - If the *pip* command isn't recognized, try using *python -m pip*
- View the tool options with: *hailo_log_collector -h*

4. Usage

As mentioned on the Installation page, the full tool is released as a Python .whl package, and the collection-only tool is released as a .zip file containing Bash .sh scripts.

For the Python package: Run *hailo_log_collector -h* to see options.

For the Bash tool: Run *./log_collector.sh -h* to see options.

Run *hailo_log_collector collect-and-report* (or *./log_collector.sh* on the Bash tool) when inside the Dataflow Compiler virtualenv, to automatically collect the following logs:

- dmesg log
- Hailo software components information
- System information (CPU, RAM, OS, Kernel version, Python version, PIP version, PCI information)
- Hailo Dataflow Compiler logs (DFC v3.20.0 and on)
- HailoRT package & HailoRT PCI Driver installation logs
- Hailo Model Zoo logs (MZ v2.6.0 and on)
- HailoRT Driver logs (HRT v4.12.0 and on)
- pyHailoRT logs (HRT v4.12.0 and on)
- HailoRT Multi-Process Service logs (HRT v4.12.0 and on)

When outside of the Dataflow Compiler virtualenv, only the following logs will be collected:

- dmesg log
- Hailo software components information
- System information (CPU, RAM, OS, Kernel version, Python version, PIP version, PCI information)
- HailoRT package & HailoRT PCI Driver installation logs
- HailoRT Driver logs (HRT v4.12.0 and on)
- HailoRT Multi-Process Service logs (HRT v4.12.0 and on)

When running inside the Dataflow Compiler virtualenv, you might also use the following flags:

- `--verbosity-level DETAILED` to also collect hailo_sdk.core.log that contains the model's compilation and allocation data. It will also look for .json, .hef, .har, .alls files inside the current working directory, but if you want to point the tool to another Dataflow Compiler working directory, use the next flag.
- `--working-dir PATH_TO_DFC_WORKING_DIR` to change the default working directory from which we collect the compilation files.

If you only wish to run the Python package's report generation step, on a folder that was created with the collection-only Bash tool:

- Run the following command: *hailo_log_collector report-only PATH_TO_log_collector__DD_MM_YYYY__HH_MM_SS_FOLDER*
- The .zip and .csv files will be created inside that folder.

Outputs:

- The folder `log_collector__DD_MM_YYYY__HH_MM_SS` will be created on the current directory. Contents:
  - The folder `collected_logs`. Contents:
    - All of the collected files described above.
  - hailo_log_collector_archive__HH_MM_SS.zip: Contains the files that are inside collected_logs. **Only on Python package.**
  - hailo_log_collector_report__HH_MM_SS.csv: Containing the summarized information. **Only on Python package.**

