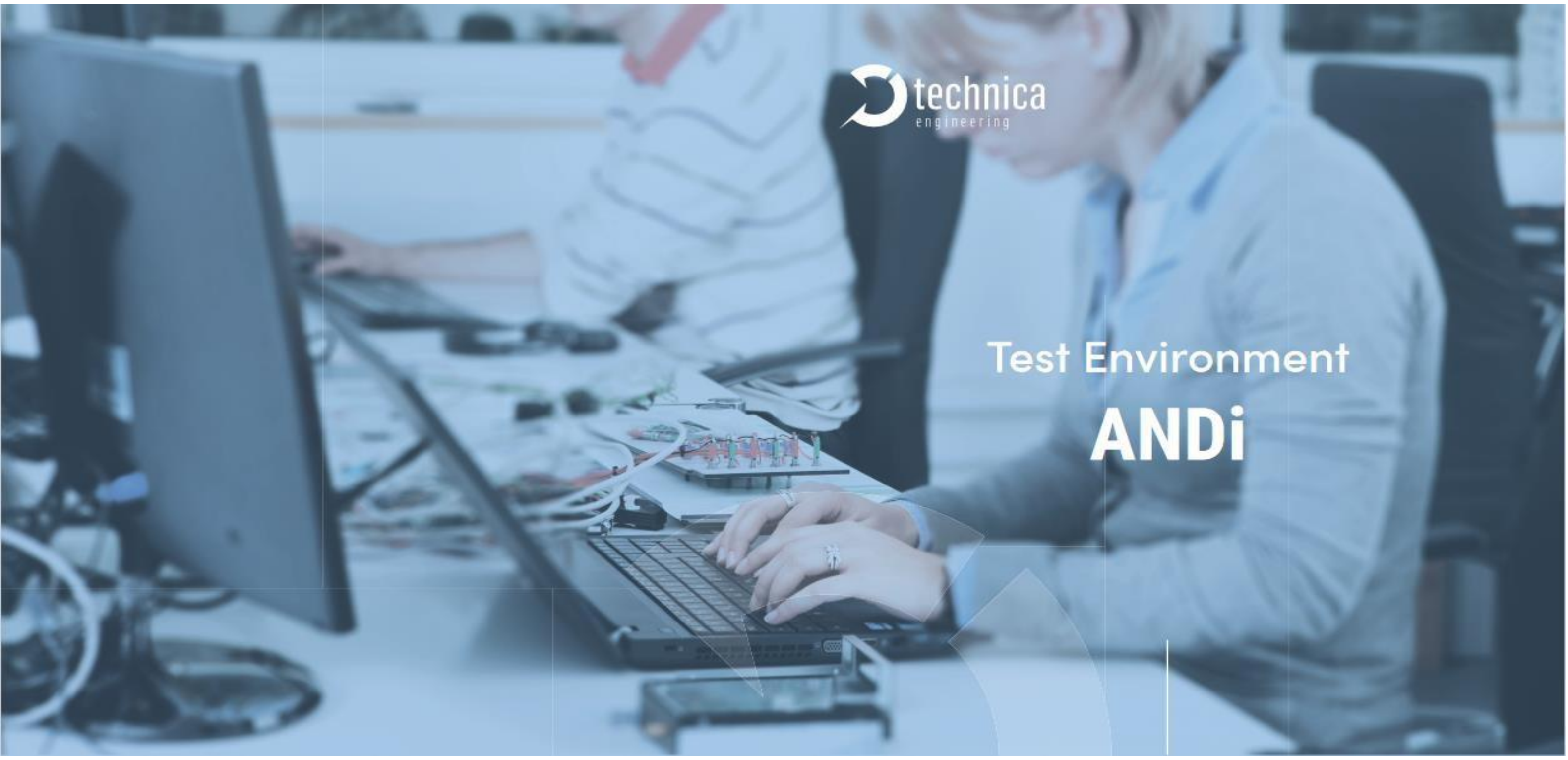


ANDi SDK Manual

As Of June 2025

Manual Version : 3.0

Technica Engineering GmbH, Leopoldstr. 236, D - 80807 München, Phone: +49 89-200-072410.
info@technica-engineering.de . www.technica-engineering.com

Contents

Introduction

This document represents a manual on how to install and use ANDi SDK.

ANDi SDK is the portable lightweight ANDi software development kit. This kit comes shipped with python library that allows the use of ANDi scripting API directly from python 3.

Requirements

ANDi SDK is portable, it can be used on both Windows and Linux machines. Before running

ANDi SDK, the following requirements need to be met:

- .NET 8 runtime: responsible for running ANDi library files (DLLs)
- CodeMeter: responsible for the handling of the licensing for ANDi SDK
- Npcap or WinPcap (Windows specific): responsible for hardware and packet handling

ANDi SDK also uses Python.NET library which enables running .NET code from python runtime. This library is installed automatically when installing ANDi SDK, so an internet connection is needed during packager installation.

Note: ANDi SDK is not functional on virtual machines.

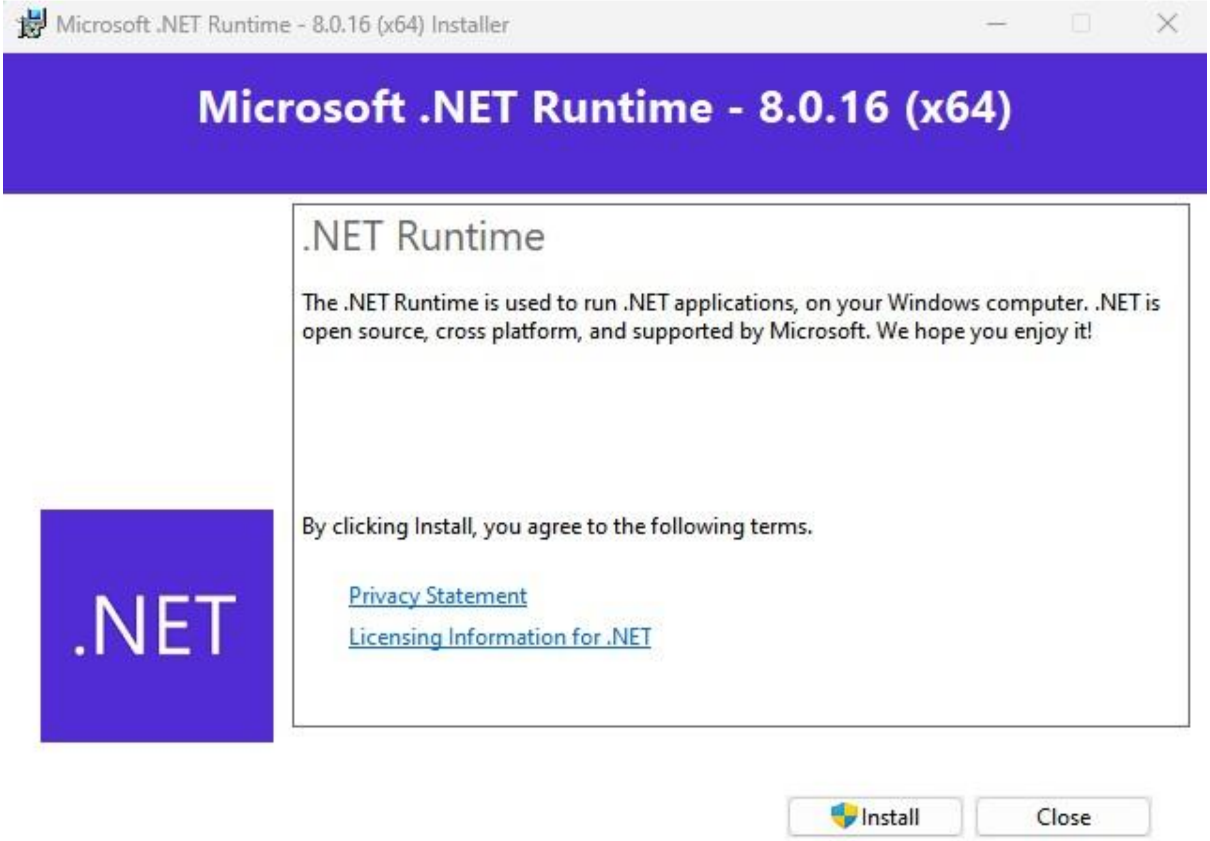
Redistributables for Windows system

Installing .NET

1- Download .NET runtime for windows from https://builds.dotnet.microsoft.com/dotnet/Runtime/8.0.16/dotnet-runtime-8.0.16-win-x64.exe

2- Run the downloaded executable

3- Follow the instructions in the installer, click on install



Microsoft .NET Runtime - 8.0.16 (x64) Installer
Microsoft .NET Runtime - 8.0.16 (x64)
.NET
.NET Runtime

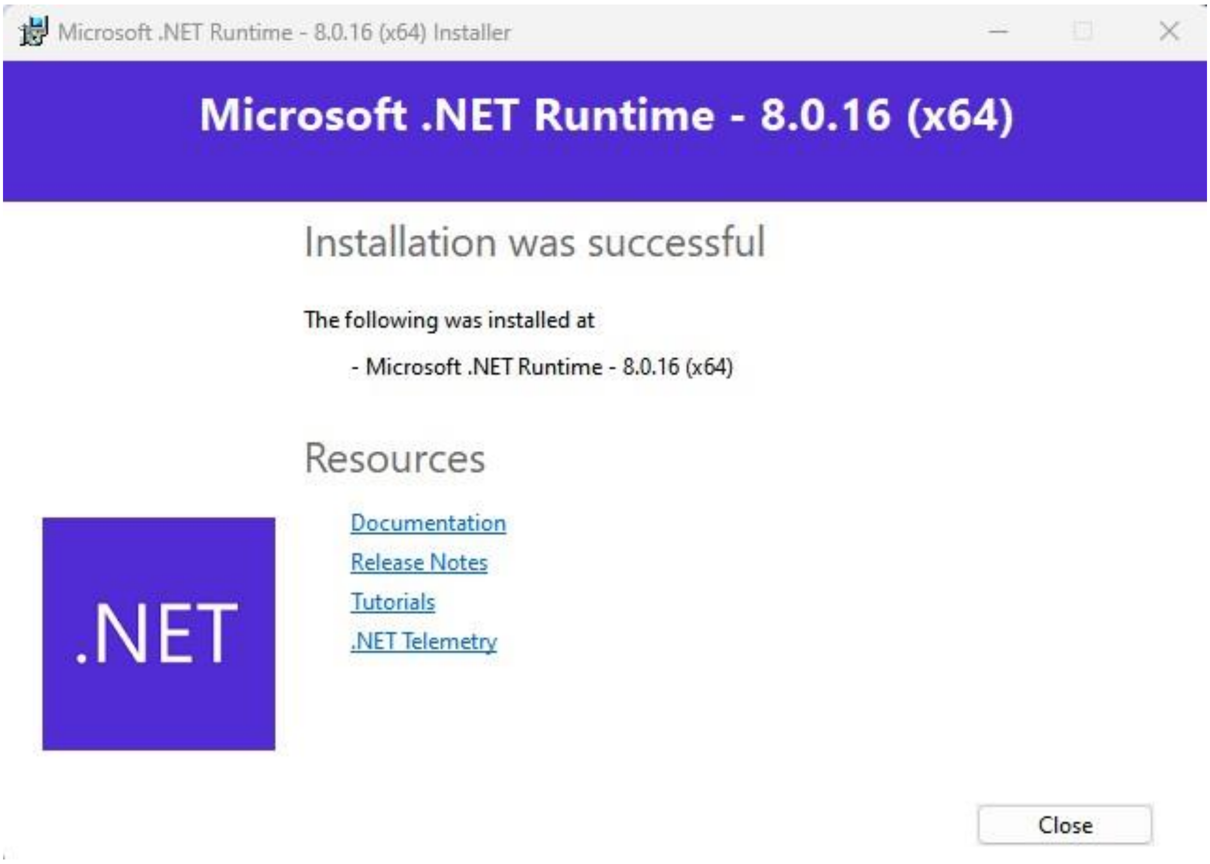
The .NET Runtime is used to run .NET applications, on your Windows computer. .NET is open source, cross platform, and supported by Microsoft. We hope you enjoy it!

By clicking Install, you agree to the following terms.

Privacy Statement

Licensing Information for .NET

4- Wait for the installation to be concluded and close the installer



Microsoft .NET Runtime - 8.0.16 (x64) Installer

Microsoft .NET Runtime - 8.0.16 (x64)

Installation was successful

The following was installed at

- Microsoft .NET Runtime - 8.0.16 (x64)

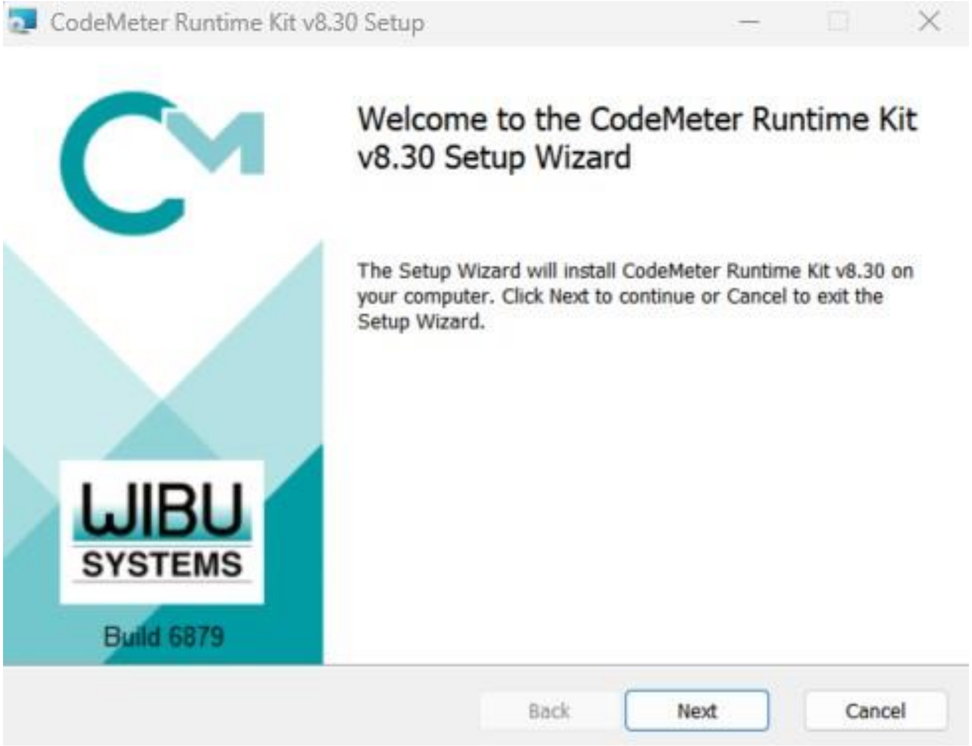
Resources

Installing CodeMeter

1. Download CodeMeter User Runtime for Windows from https://files.technica-engineering.de/ANDi/Redistributables/CodeMeterRuntime_8_30.msi

2. Run the downloaded executable

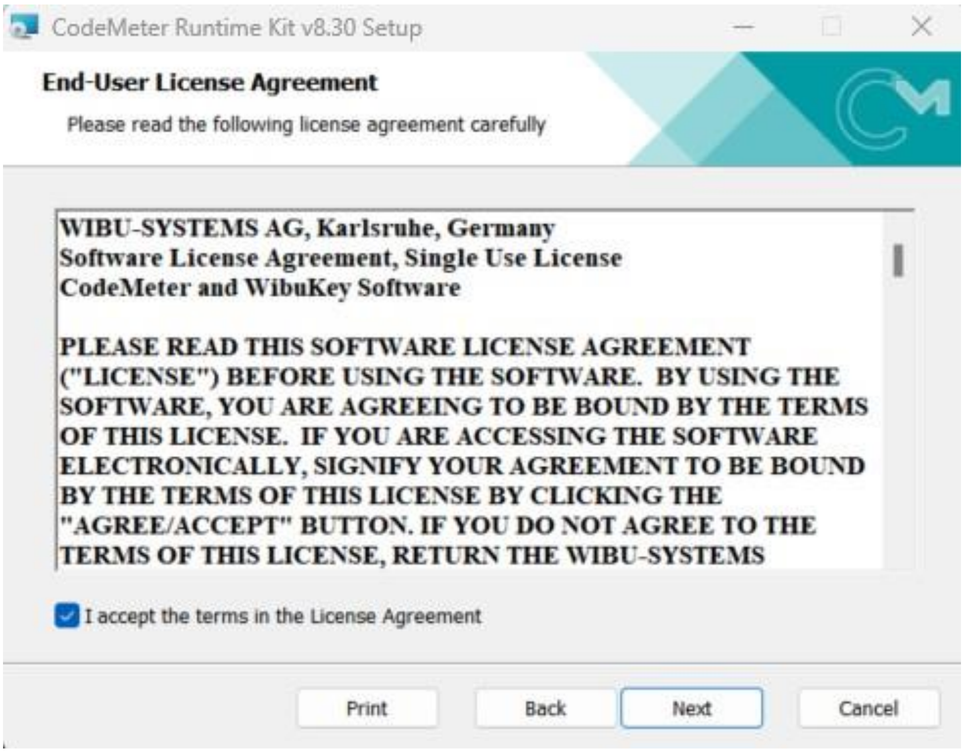
3. Follow the instructions in the installer



Welcome to the CodeMeter Runtime Kit v8.30 Setup Wizard

The Setup Wizard will install CodeMeter Runtime Kit v8.30 on your computer. Click Next to continue or Cancel to exit the Setup Wizard.

4. Accept the license agreement



End-User License Agreement

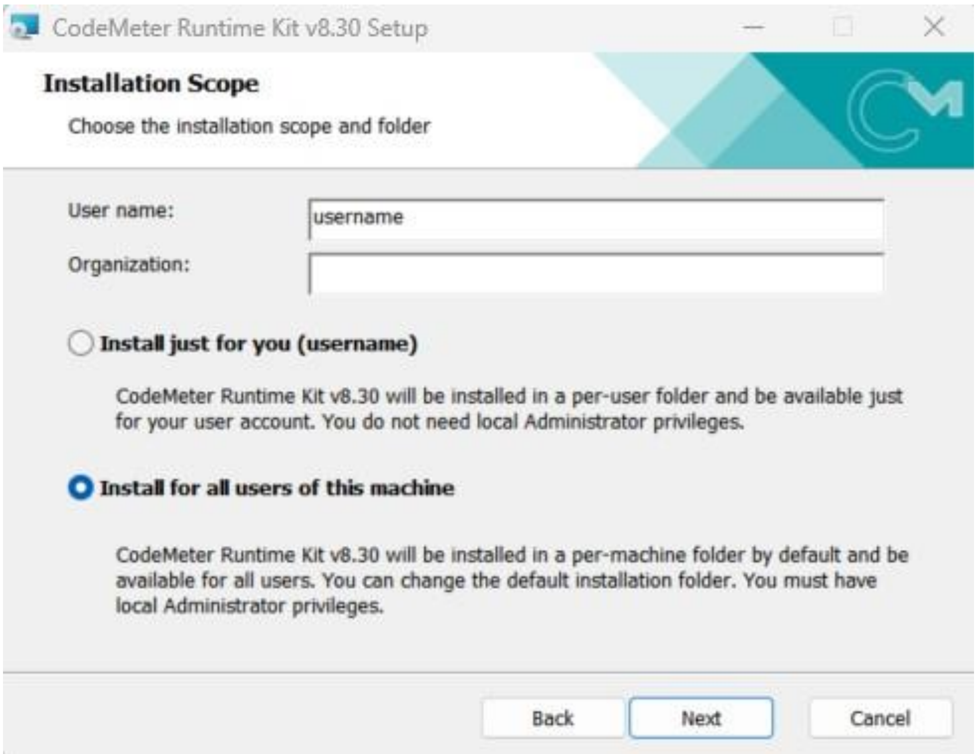
Please read the following license agreement carefully

WIBU-SYSTEMS AG, Karlsruhe, Germany
Software License Agreement, Single Use License
CodeMeter and WibuKey Software

PLEASE READ THIS SOFTWARE LICENSE AGREEMENT ("LICENSE") BEFORE USING THE SOFTWARE. BY USING THE SOFTWARE, YOU ARE AGREEING TO BE BOUND BY THE TERMS OF THIS LICENSE. IF YOU ARE ACCESSING THE SOFTWARE ELECTRONICALLY, SIGNIFY YOUR AGREEMENT TO BE BOUND BY THE TERMS OF THIS LICENSE BY CLICKING THE "AGREE/ACCEPT" BUTTON. IF YOU DO NOT AGREE TO THE TERMS OF THIS LICENSE, RETURN THE WIBU-SYSTEMS

☑ I accept the terms in the License Agreement

5. Choose a username and install for all users (Recommended)



CodeMeter Runtime Kit v8.30 Setup

Installation Scope

Choose the installation scope and folder

User name: username

Organization:

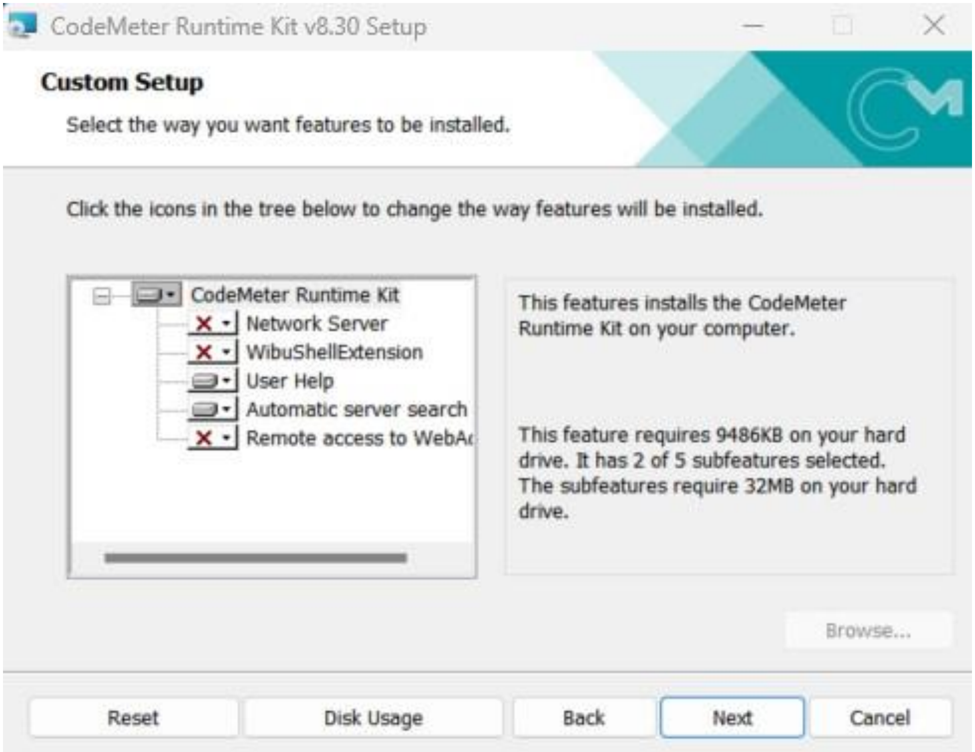
○ **Install just for you (username)**

CodeMeter Runtime Kit v8.30 will be installed in a per-user folder and be available just for your user account. You do not need local Administrator privileges.

● **Install for all users of this machine**

CodeMeter Runtime Kit v8.30 will be installed in a per-machine folder by default and be available for all users. You can change the default installation folder. You must have local Administrator privileges.

6. We could choose to install remote access to web admin and network server for remote licensing, but it is not recommended, so it’s better to keep the default values



CodeMeter Runtime Kit v8.30 Setup

Custom Setup

Select the way you want features to be installed.

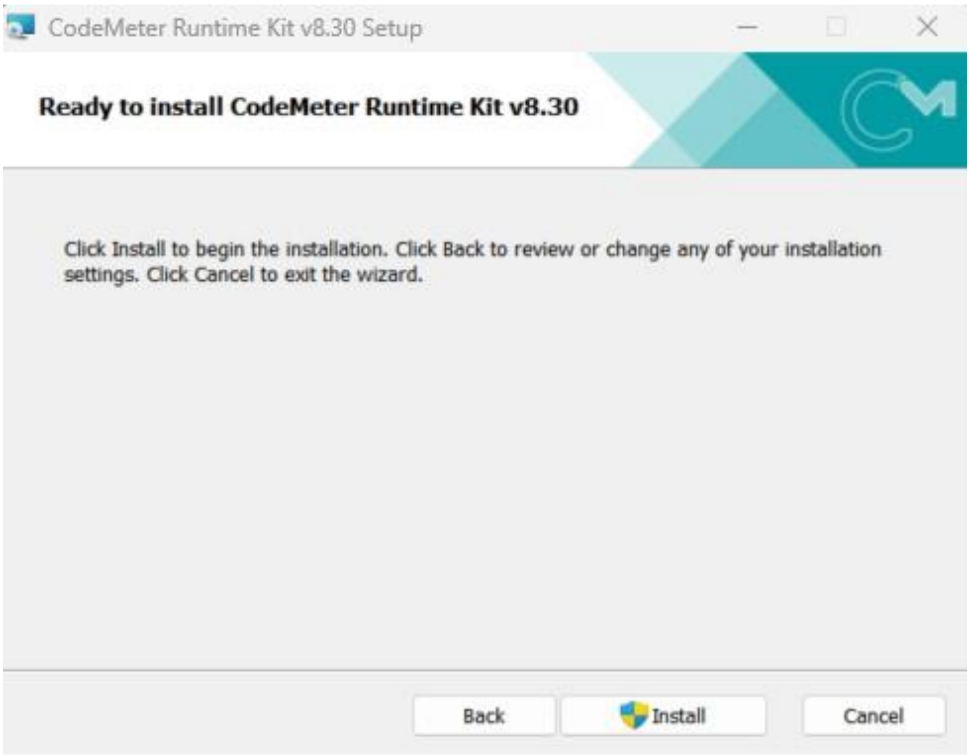
Click the icons in the tree below to change the way features will be installed.

- CodeMeter Runtime Kit
  - ☒ Network Server
  - ☒ WibuShellExtension
  - User Help
  - Automatic server search
  - ☒ Remote access to WebA

This features installs the CodeMeter Runtime Kit on your computer.

This feature requires 9486KB on your hard drive. It has 2 of 5 subfeatures selected. The subfeatures require 32MB on your hard drive.

7. Click install and wait for the installation to finish

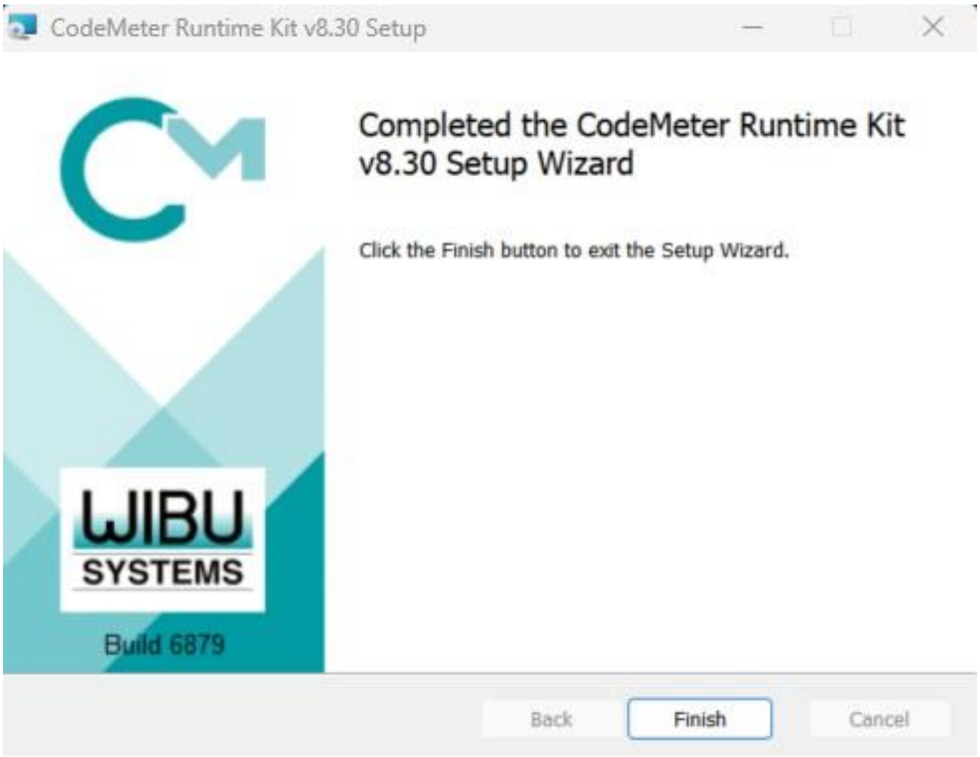


CodeMeter Runtime Kit v8.30 Setup

Ready to install CodeMeter Runtime Kit v8.30

Click Install to begin the installation. Click Back to review or change any of your installation settings. Click Cancel to exit the wizard.

8. Click on finish when the installation is completed



CodeMeter Runtime Kit v8.30 Setup

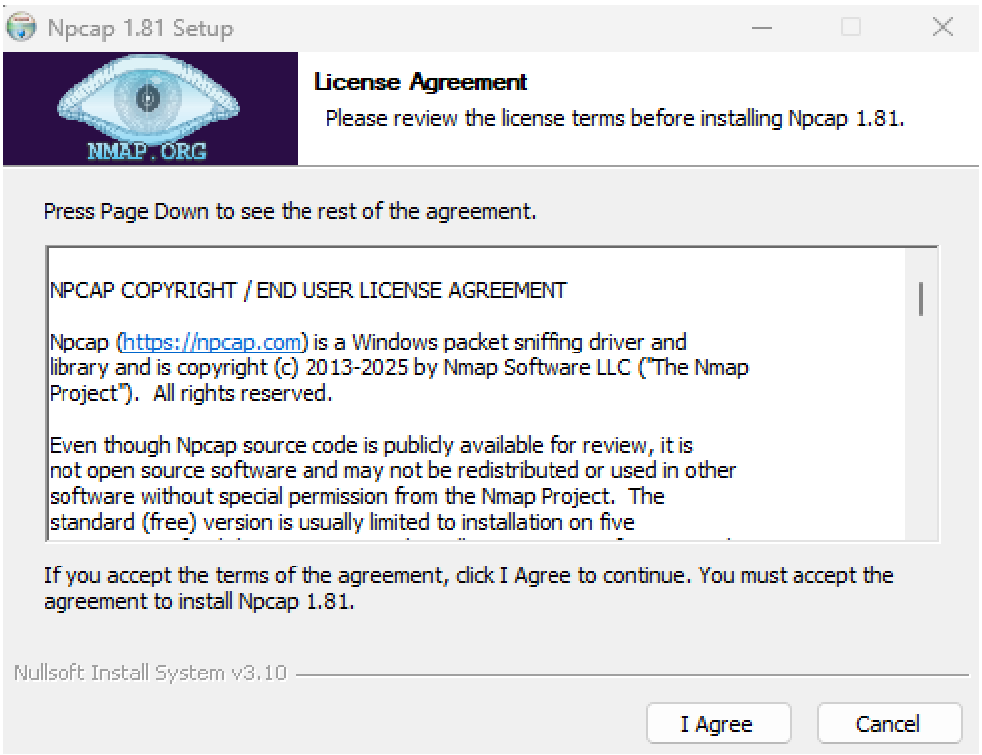
Completed the CodeMeter Runtime Kit v8.30 Setup Wizard

Click the Finish button to exit the Setup Wizard.

Installing Npcap/WinPcap

Officially, ANDi SDK needs Npcap at least version 1.79 to run properly, installing WinPcap works as well.

1- Download the Npcap installer from the website https://npcap.com/#download
2- Run the downloaded executable
3- Follow the instructions in the installer
4- Agree to the license agreement



License Agreement

Please review the license terms before installing Npcap 1.81.

Press Page Down to see the rest of the agreement.

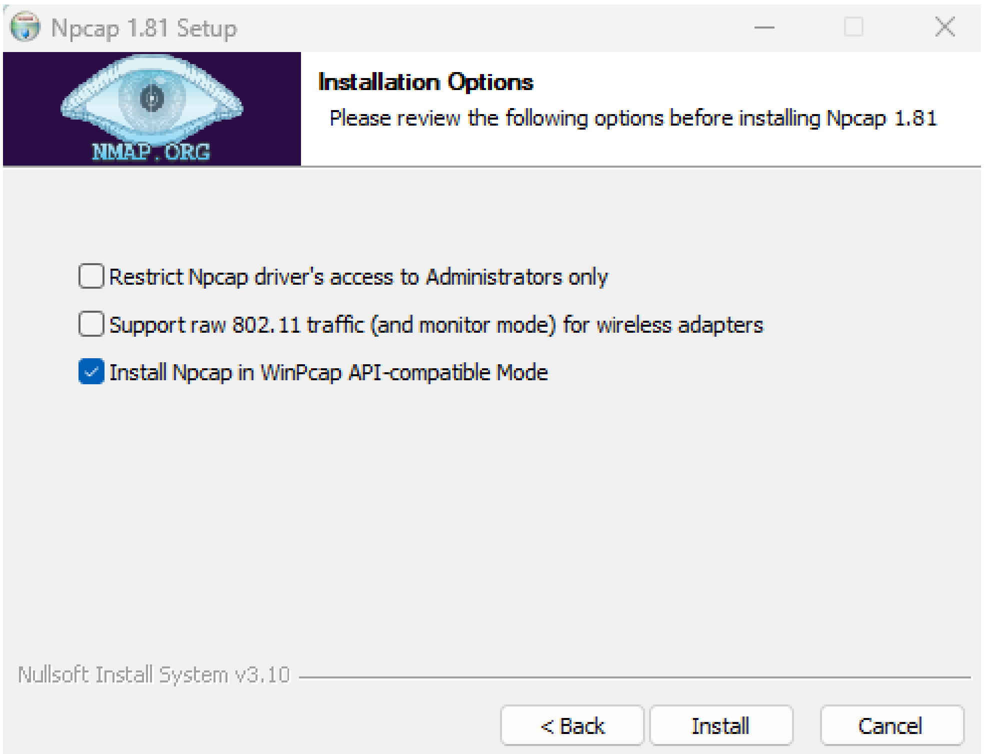
NPCAP COPYRIGHT / END USER LICENSE AGREEMENT

Npcap (https://npcap.com) is a Windows packet sniffing driver and library and is copyright (c) 2013-2025 by Nmap Software LLC ("The Nmap Project"). All rights reserved.

Even though Npcap source code is publicly available for review, it is not open source software and may not be redistributed or used in other software without special permission from the Nmap Project. The standard (free) version is usually limited to installation on five

If you accept the terms of the agreement, click I Agree to continue. You must accept the agreement to install Npcap 1.81.

5- Keep the option “Install Npcap in WinPcap API-compatible mode” selected and click install



Npcap 1.81 Setup

Installation Options

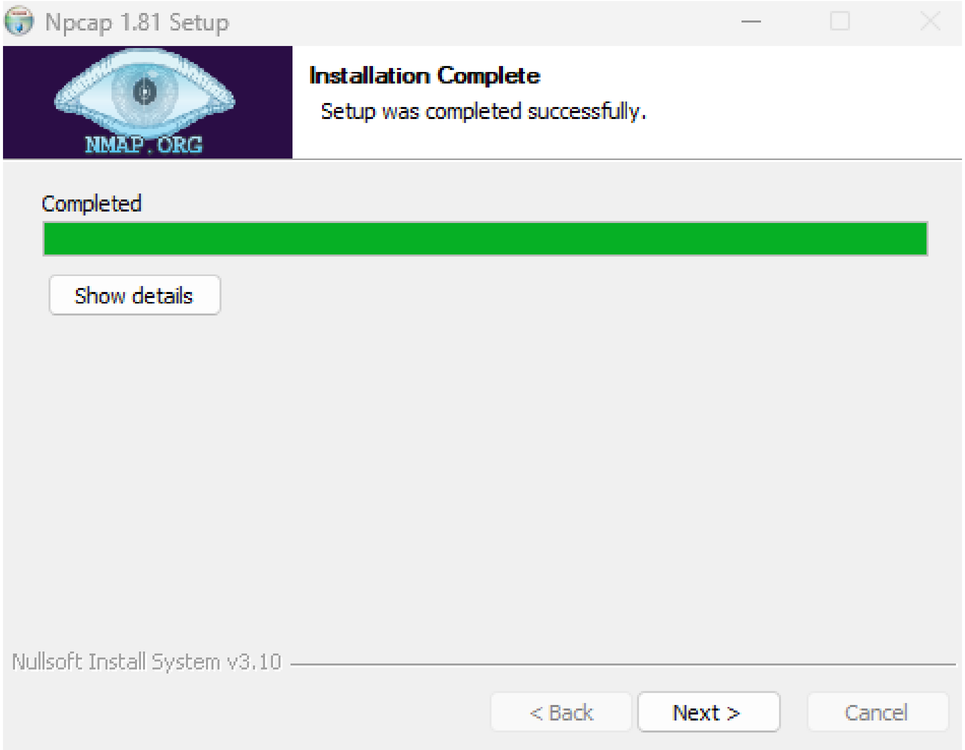
Please review the following options before installing Npcap 1.81

□ Restrict Npcap driver's access to Administrators only

□ Support raw 802.11 traffic (and monitor mode) for wireless adapters

☑ Install Npcap in WinPcap API-compatible Mode

6- Wait for the installation to finish and click next



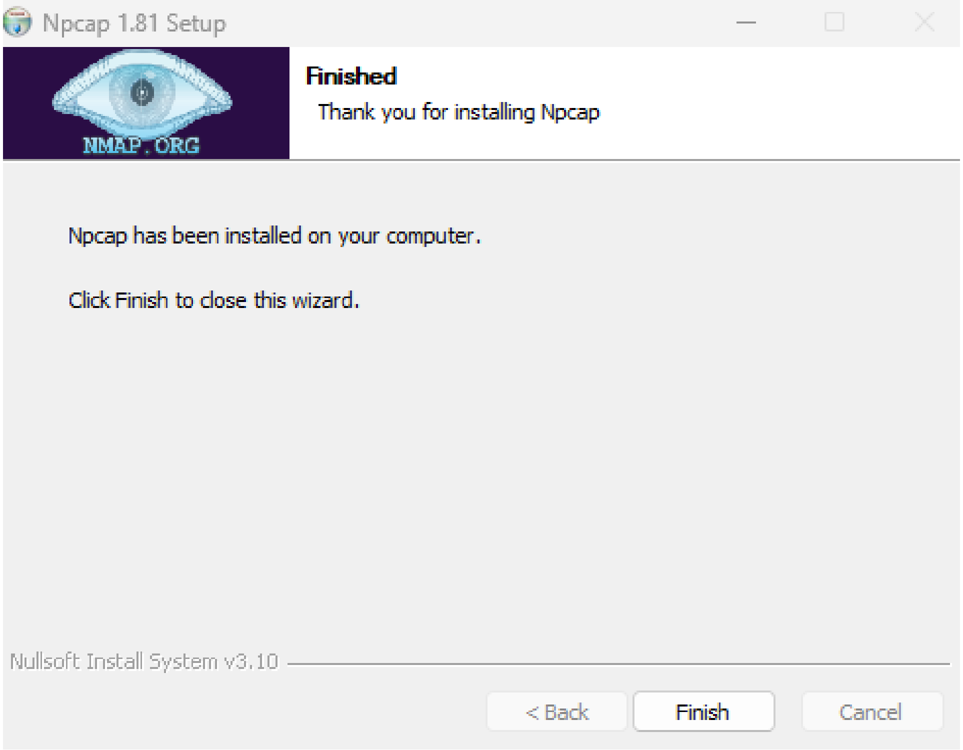
Npcap 1.81 Setup

Installation Complete

Setup was completed successfully.

Completed

7- Click Finish to close the installer



Finished

Thank you for installing Npcap

Npcap has been installed on your computer.

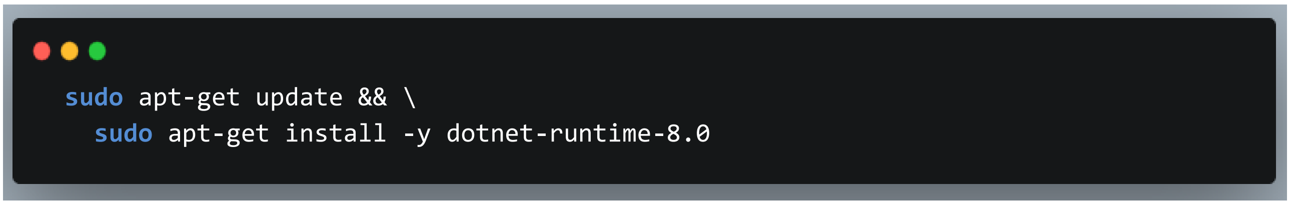
Click Finish to close this wizard.

⇨ Wireshark comes bundled with the latest version of Npcap, so If Wireshark is installed on the machine, this step can be skipped.

Redistributables for Linux system

Installing .NET

- Update package list and install .NET 8 Runtime

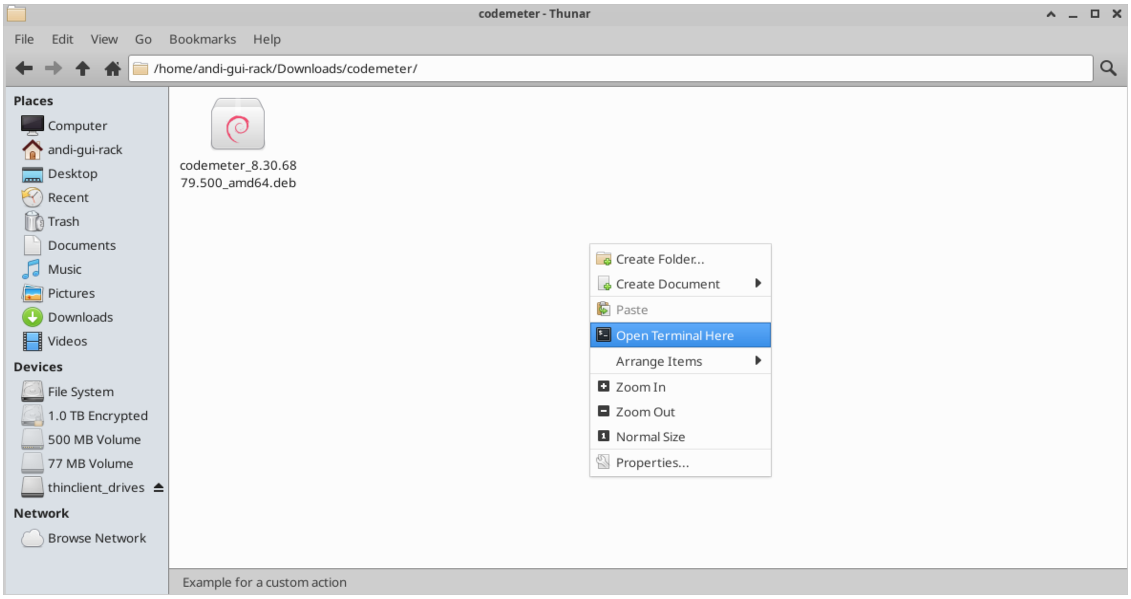


```
sudo apt-get update && \
  sudo apt-get install -y dotnet-runtime-8.0
```

Note:
If this doesn’t work on your machine, you can check this page for your machine installation:
Install .NET on Linux distributions - .NET | Microsoft Learn

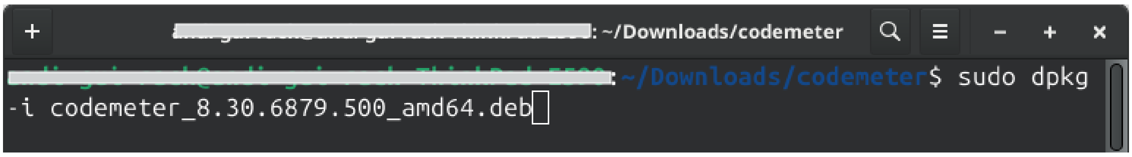
Installing CodeMeter

1- Go to the official Wibu-Systems page.
2- Navigate to **Support & Downloads** → **User Software**.
3- Choose the appropriate package (**.deb** for Debian/Ubuntu, **.rpm** for RHEL/CentOS).
4- Open a terminal in the folder where the runtime file is



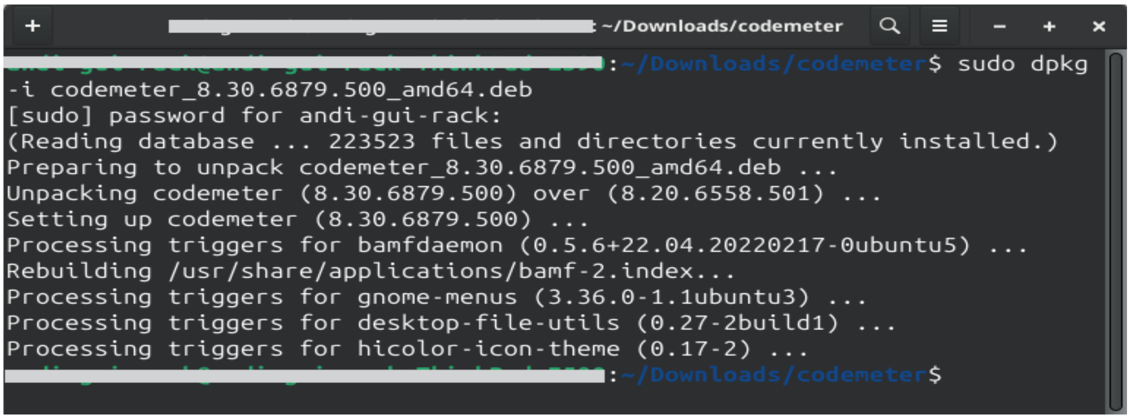
5- Run the following command

```
sudo dpkg -i runtime_file_name.deb
```



```
[redacted]:~/Downloads/codemeter$ sudo dpkg
-i codemeter_8.30.6879.500_amd64.deb
```

6- Wait for the installation to be over, the result should look like this



```
[redacted]:~/Downloads/codemeter$ sudo dpkg
-i codemeter_8.30.6879.500_amd64.deb
[sudo] password for andi-gui-rack:
(Reading database ... 223523 files and directories currently installed.)
Preparing to unpack codemeter_8.30.6879.500_amd64.deb ...
Unpacking codemeter (8.30.6879.500) over (8.20.6558.501) ...
Setting up codemeter (8.30.6879.500) ...
Processing triggers for bamfdaemon (0.5.6+22.04.20220217-0ubuntu5) ...
Rebuilding /usr/share/applications/bamf-2.index...
Processing triggers for gnome-menus (3.36.0-1.1ubuntu3) ...
Processing triggers for desktop-file-utils (0.27-2build1) ...
Processing triggers for hicolor-icon-theme (0.17-2) ...
[redacted]:~/Downloads/codemeter$
```

After installing CodeMeter, an ANDi-tool-license and a USB-dongle are needed. Both are provided by Technica Engineering. The license is fixed to the dongle. The dongle needs to be plugged into the machine before working with ANDi SDK.

For license related inquiries, this email: support@technica-engineering.de is available from Technica Engineering.

Installing Libpcap

Officially, ANDi SDK needs Libpcap on Linux to run properly.

The Libpcap can be installed using the following command line:

```
sudo apt-get install libpcap-dev
```

ANDi SDK installation

ANDi SDK can be installed using the following command line:

```
pip install andisdk
```

ANDi SDK update

From PyPi Server

If ANDi SDK is already installed, an upgrade is possible using the following command

```
pip install --upgrade andisdk
```

User Manual

ANDi SDK or the needed packages from it can be imported directly in python scripts.

```
from andisdk import load_project
# or
import andisdk
# or, starting from version 0.0.11 message_builder and andi objects are usable without the need to load a project
from andisdk import message_builder, andi
```

ANDi SDK API is available after the appropriate imports.

- ⇨ When using IDEs that are compatible with python autocomplete, ANDi SDK provides autocomplete and type hinting capabilities.
- ⇨ In Linux, several ANDi SDK features (for example, adapters) require admin privileges.
- ⇨ Script execution will stop if a valid ANDi license is not detected.

ANDi SDK uses python logging module to log the messages happening in ANDi functionalities. Logging configured to log to a file using the function basicConfig

```
import logging
logging.basicConfig(filename="log_output_file.txt",  filemode="w")
```

For a better and richer logging experience, rich logging handler can be used as a logging handler

This is done by installing rich python module using the command

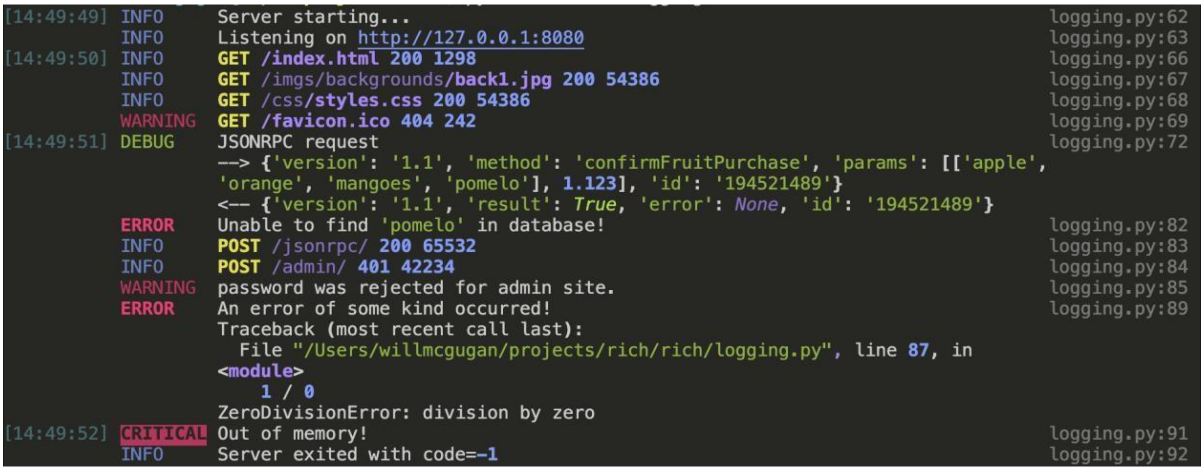
```
pip install rich
```

And then logging is configured to use the rich handler

```
import logging
from rich.logging import RichHandler

FORMAT = "%(message)s"
logging.basicConfig(
    level="NOTSET", format=FORMAT, datefmt="[%X]", handlers=[RichHandler()]
)
```

This will give logging a look similar to this:



```
[14:49:49] INFO     Server starting...                                                   logging.py:62
           INFO     Listening on http://127.0.0.1:8080                                   logging.py:63
[14:49:50] INFO     GET /index.html 200 1298                                             logging.py:66
           INFO     GET /imgs/backgrounds/back1.jpg 200 54386                            logging.py:67
           INFO     GET /css/styles.css 200 54386                                        logging.py:68
           WARNING  GET /favicon.ico 404 242                                             logging.py:69
[14:49:51] DEBUG    JSONRPC request                                                      logging.py:72
                    --> {'version': '1.1', 'method': 'confirmFruitPurchase', 'params': [['apple',
                    'orange', 'mangoes', 'pomelo'], 1.123], 'id': '194521489'}
                    <-- {'version': '1.1', 'result': True, 'error': None, 'id': '194521489'}
           ERROR    Unable to find 'pomelo' in database!                                 logging.py:82
           INFO     POST /jsonrpc/ 200 65532                                             logging.py:83
           INFO     POST /admin/ 401 42234                                               logging.py:84
           WARNING  password was rejected for admin site.                                logging.py:85
           ERROR    An error of some kind occurred!                                      logging.py:89
                    Traceback (most recent call last):
                      File "/Users/willmcgugan/projects/rich/rich/logging.py", line 87, in
                    <module>
                        1 / 0
                    ZeroDivisionError: division by zero
[14:49:52] CRITICAL Out of memory!                                                       logging.py:91
           INFO     Server exited with code=-1                                           logging.py:92
```

Scripts samples

Creating and sending a UDP message

```python
from andisdk import load_project
import sys
api = load_project("empty_project.atp")

adapters = api.andi.get_adapters()

if (len(adapters) <= 0):
    print("No adapters found, stopping script")
    sys.exit()

adapter = adapters[0]
print("using adapter " + adapter.id + " to send udp message")
channel = api.andi.create_channel("Ethernet")
message = api.message_builder.create_udp_message(channel, channel)

message.payload = tuple([0x01, 0x02, 0x03, 0x04])
message.udp_header.port_source = 1234

print("sending udp message with payload " + str([x for x in message.payload]))
message.send()
```

Script version without a project (starting from version 0.0.11)

```python
from andisdk import message_builder, andi
import sys

adapters = andi.get_adapters()

if (len(adapters) <= 0):
    print("No adapters found, stopping script")
    sys.exit()

adapter = adapters[0]
print("using adapter " + adapter.id + " to send udp message")
channel = andi.create_channel("Ethernet")
message = message_builder.create_udp_message(channel, channel)

message.payload = tuple([0x01, 0x02, 0x03, 0x04])
message.udp_header.port_source = 1234

print("sending udp message with payload " + str([x for x in message.payload]))
message.send()
```

Creating and sending a SOME/IP message with a database

```
from andisdk import load_project
import sys
api = load_project("empty_project.atp")

adapters = api.andi.get_adapters()
database = api.andi.load_database("someip_database.xml")

if (len(adapters) <= 0):
    print("No adapters found, stopping script")
    sys.exit()

adapter = adapters[0]
print("using adapter " + adapter.id + " to send someip message")
channel = api.andi.create_channel("Ethernet")
message = api.message_builder.create_someip_message(channel, channel)
message.data_base = database
message.someip_header.service_identifier = 0xb011
message.someip_header.method_identifier = 0x000e
message.set_input_param("statusReceiptRequestInstrument", 2)
message.send()
```

Script version without a project (starting from version 0.0.11)

```
from andisdk import message_builder, andi
import sys

adapters = andi.get_adapters()
database = andi.load_database("someip_database.xml")

if (len(adapters) <= 0):
    print("No adapters found, stopping script")
    sys.exit()

adapter = adapters[0]
print("using adapter " + adapter.id + " to send someip message")
channel = andi.create_channel("Ethernet")
message = message_builder.create_someip_message(channel, channel)
message.data_base = database
message.someip_header.service_identifier = 0xb011
message.someip_header.method_identifier = 0x000e
message.set_input_param("statusReceiptRequestInstrument", 2)
message.send()
```