

# Adrenaline Rx (Anti-Ransomware / File-Monitor)

**Document Version: 0.0.1.3**

**Software Version: 3.5.0100.1**

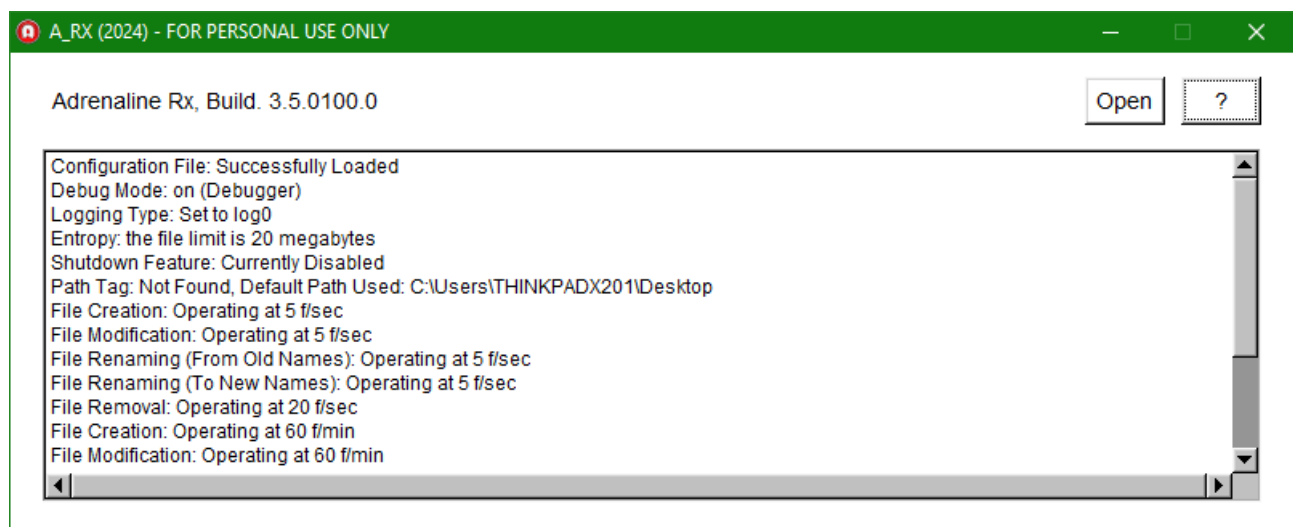
**Author: Roberto Mazzoni, Federico Bombardi**

## Introduction

Adrenaline RX is an anti-ransomware written in C++ designed to protect your sensitive data and prevent unauthorized access to your files. With a combination of advanced monitoring and detection techniques, Adrenaline offers robust defense against computer threats such as ransomware. Thanks to its entropy calculation features, sound alarm, and automatic shutdown, Adrenaline is an indispensable tool for keeping your data safe from ransomware attacks when normal defenses have fallen or even do not exist or are too weak to block the ongoing attack.

## Initial Log During startup

Adrenaline displays an initial log to provide information about the status of the program and the current configuration. This log includes details such as the type of log, the status of the shutdown function, the monitoring path, and alarm settings for creation, modification, renaming, and removal of files.



## Main Features

Adrenaline RX's filters generate a score based on the calculation of Entropy, Headers, Extensions, and I/O streams for the evaluation of generated and compromised files. This allows for the quick identification of files that have been encrypted or modified in an unusual way.

$$H(X) = - \sum_{i=1}^n P(x_i) \log_b P(x_i)$$

## Sound Alarm

When suspicious or potentially harmful activity is detected, Adrenaline activates a sound alarm to alert the user of the ongoing suspicious activity. You can find the "ALARM.wav" file in the root folder of Adrenaline.

## Automatic Shutdown

In case of detection of malware or harmful activities, Adrenaline can automatically activate the system shutdown function to prevent further damage and protect data integrity.

## Magic Bytes Configuration File (magic.cfg)

Adrenaline's magic bytes configuration file (magic.cfg) contains file extensions and their corresponding "Magic Bytes" to identify the file type.

.zip 50 4B

.rar 52 61 72 21

.....

## Installation To install Adrenaline Rx, follow these steps:

- Download the installation file from the official project page on GitHub.
- Run the installation file and follow the instructions displayed on the screen.
- Configure the config.cfg file according to your needs.
- Once the installation is complete, start Adrenaline Rx.

## Configuration File (config.cfg)

The Adrenaline configuration file (config.cfg) contains the main settings of the program, such as the type of log, the status of the shutdown function, the monitoring path, and the thresholds for the file alarm.

**Attention:** note that the command characters are case-sensitive, so make sure to correctly respect the syntax.

List of commands that you can configure inside the config.cfg file:

**Command: LOG=<log0 | log1 | log2>**

The log file, essential for tracking activities, is generated daily and can be found in the main directory of Adrenaline, specifically in the “.\log” folder.

**log0:** This recording mode is inactive, not recording any events.

**log1:** In this mode, only suspicious files, those that might have been encrypted, are recorded. The log follows a standardized format, showing the date, time, and path of the file.

**log2:** “Monitor mode”, This mode records all file creation and manipulation events, without exceptions. Every activity is noted in the log file.

**Command: DEBUG\_MODE=<on|off>**

Activates the debug mode. !IMPORTANT!: use DEBUG\_MODE=off in production.

**Command: POWER=<on|off>**

This command activates or deactivates the computer shutdown function in response to a critical alarm detected by the anti-ransomware.

**Command: PATH=<path>**

This command specifies the path where the monitor starts scanning in recursive mode, checking all subdirectories within the specified path.

**Command: FILE\_LIMITER=<KiloBytes>**

This command determines the size of the first data segment, or “frame”, that the entropic engine analyzes.

**Command: CREATE=<file per second>**

Sets the trigger for file creation alarms, expressed in seconds. If the file creation time exceeds the specified value, the alarm is triggered.

**Command: CREATE\_M=<file per minute>**

Sets the trigger for file creation alarms, expressed in minutes. If the file creation time exceeds the specified value, the alarm is triggered.

**Command: MODIFY=<file per second>**

Sets the trigger for file modification alarms, expressed in seconds. If the time elapsed since the last modification of a file exceeds the specified value, the alarm is triggered.

**Command: MODIFY\_M=<file per minute>**

Sets the trigger for file modification alarms, expressed in minutes. If the time elapsed since the last modification of a file exceeds the specified value, the alarm is triggered.

**Command: RENAME\_OLD=<file per second>**

Sets the trigger for rename alarms (old name), expressed in seconds. If the time elapsed since the last modification of the original name of a file exceeds the specified value, the alarm is triggered.

**Command: RENAME\_OLD\_M=<file per minute>**

Sets the trigger for rename alarms (old name), expressed in minutes. If the time elapsed since the last modification of the original name of a file exceeds the specified value, the alarm is triggered.

**Command: RENAME\_NEW=<file per second>**

Sets the trigger for rename alarms (new name), expressed in seconds.

**Command: RENAME\_NEW\_M=<file per minute>**

Sets the trigger for rename alarms (new name), expressed in minutes.

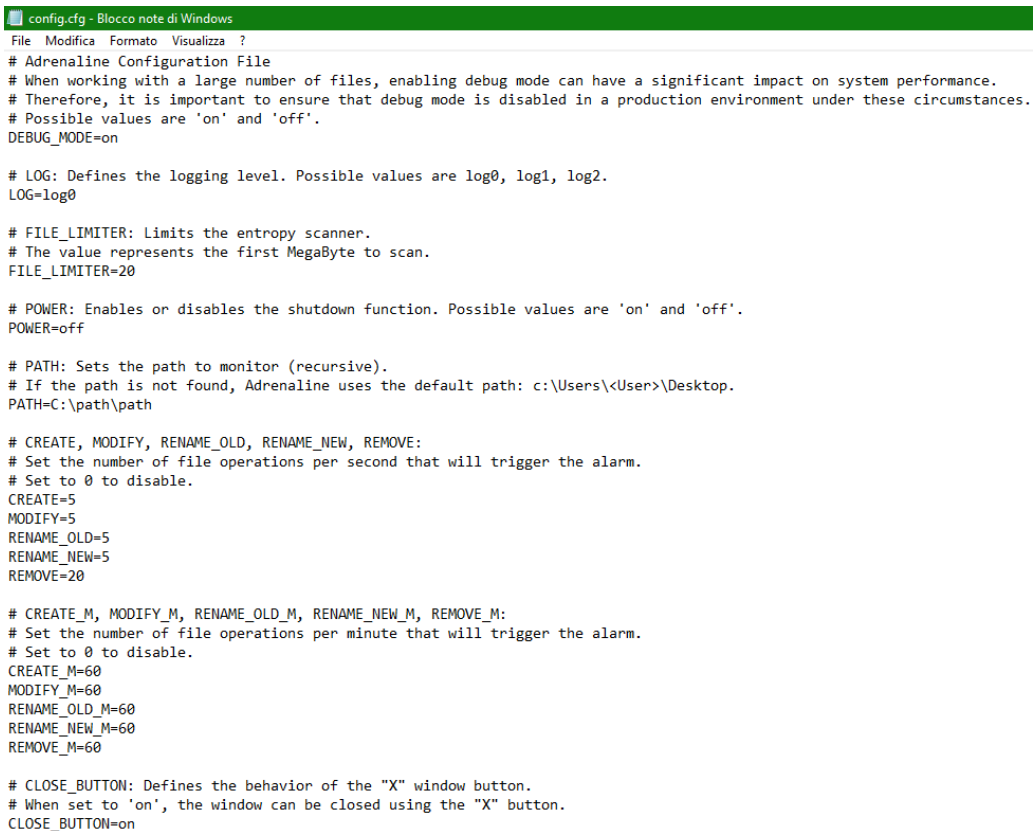
**Command: REMOVE=<file per second>**

Sets the trigger for file removal alarms, expressed in seconds. If the time elapsed since a file was removed exceeds the specified value, the alarm is triggered.

**Command: REMOVE\_M=<file per minute>**

Sets the trigger for file removal alarms, expressed in minutes. If the time elapsed since a file was removed exceeds the specified value, the alarm is triggered.

## config.cfg



```
File Modifica Formato Visualizza ?
# Adrenaline Configuration File
# When working with a large number of files, enabling debug mode can have a significant impact on system performance.
# Therefore, it is important to ensure that debug mode is disabled in a production environment under these circumstances.
# Possible values are 'on' and 'off'.
DEBUG_MODE=on

# LOG: Defines the logging level. Possible values are log0, log1, log2.
LOG=log0

# FILE_LIMITER: Limits the entropy scanner.
# The value represents the first MegaByte to scan.
FILE_LIMITER=20

# POWER: Enables or disables the shutdown function. Possible values are 'on' and 'off'.
POWER=off

# PATH: Sets the path to monitor (recursive).
# If the path is not found, Adrenaline uses the default path: c:\Users\<User>\Desktop.
PATH=C:\path\path

# CREATE, MODIFY, RENAME_OLD, RENAME_NEW, REMOVE:
# Set the number of file operations per second that will trigger the alarm.
# Set to 0 to disable.
CREATE=5
MODIFY=5
RENAME_OLD=5
RENAME_NEW=5
REMOVE=20

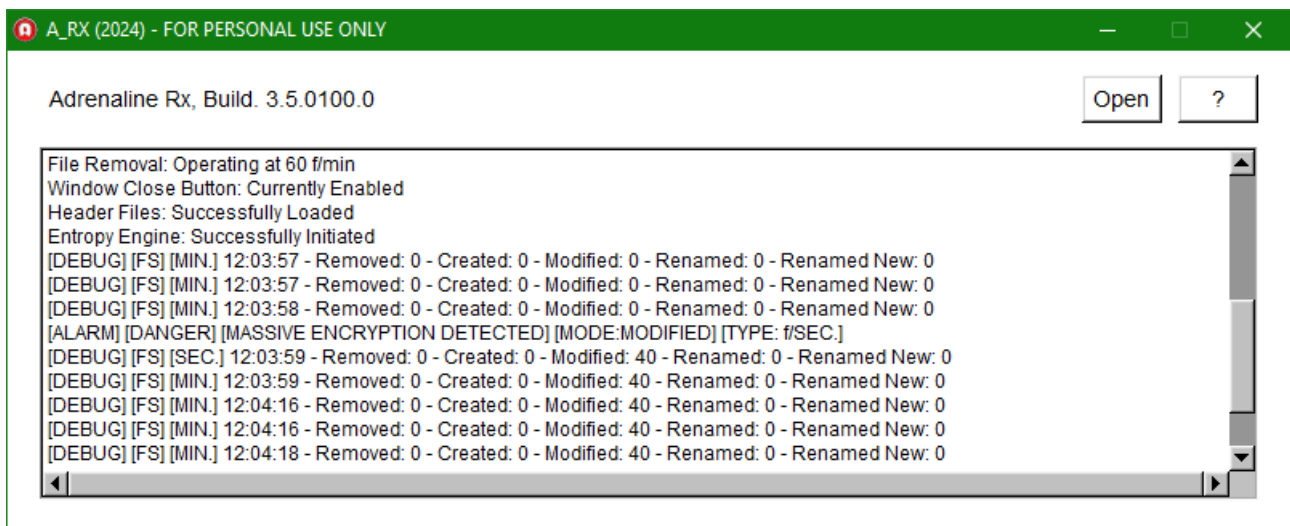
# CREATE_M, MODIFY_M, RENAME_OLD_M, RENAME_NEW_M, REMOVE_M:
# Set the number of file operations per minute that will trigger the alarm.
# Set to 0 to disable.
CREATE_M=60
MODIFY_M=60
RENAME_OLD_M=60
RENAME_NEW_M=60
REMOVE_M=60

# CLOSE_BUTTON: Defines the behavior of the "X" window button.
# When set to 'on', the window can be closed using the "X" button.
CLOSE_BUTTON=on
```

## Debug Mode:

In `config.cfg` use `DEBUG_MODE=off`

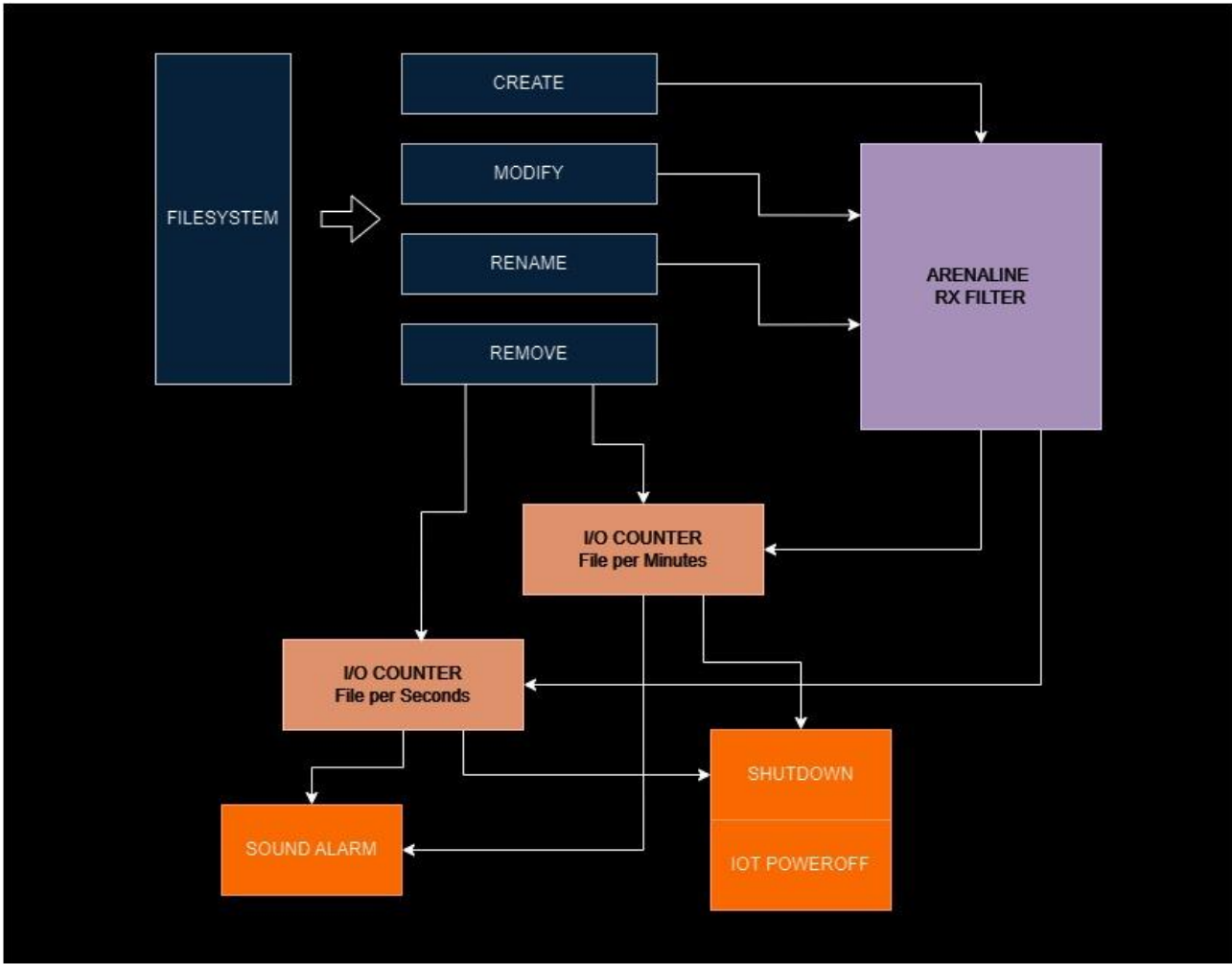
WARNING!, do not use Debug mode in production. All values are expressed in minute/second(\*).



\*The "Removed" counter keeps track of all the deleted files, while the other parameters are filtered through Adrenaline's algorithms.

\*Files that Adrenaline has deemed clean will not appear in the counters.

**Adrenaline RX AntiRansomware Filter:**





## Versioning Scheme

The versioning system of Adrenaline Rx follows a standard scheme composed of four numbers:

- Main Version: Indicates a significant version of the software with major changes or additions of features.
- Major Revision: Represents a smaller revision compared to the main version, generally characterized by significant improvements or important updates.
- Minor Revision: Indicates a smaller revision compared to the major revision, which includes minor adjustments, bug fixes or performance improvements.
- Bug Number: Represents the number of bugs fixed from the version.

For example, consider the version “3.5.0100.0”:

- “3” is the main version.
- “5” is the major revision.
- “0100” is the minor revision.
- “0” indicates the number of bugs fixed.