

GitHub - XZB-1248/Spark: ✨ Spark is a web-based, cross-platform and full-featured Remote Administration Tool (RAT) written in Go that allows you control all your devices anywhere. Spark是一个Go编写的，网页UI、跨平台以及多功能的远程控制和监控工具，你可以随时随地监控和控制所有设备。

By XZB-1248

Archived: 2026-04-05 22:32:36 UTC

[English] [中文] [API Document] [API文档]

Spark is a free, safe, open-source, web-based, cross-platform, and full-featured RAT (Remote Administration Tool) that allows you to control all your devices via browser anywhere.

- ✔ **No data collection:** Spark does not collect any user information.
- ✔ **No auto-updates:** The server will not update itself.
- ✔ **Direct communication:** Clients communicate exclusively with your server.

repo size 6.1 MiB	issues 22 open	issues 40 closed
downloads 31k	downloads@latest 23k	

⚠ Disclaimer

THIS PROJECT, ITS SOURCE CODE, AND RELEASES SHOULD ONLY BE USED FOR EDUCATIONAL PURPOSES.

- ✗ **Illegal usage is strictly prohibited.**
- ✗ **Authors and developers are not responsible for any misuse.**
- ✔ **Use it at your own risk.**

If you find security vulnerabilities, **do not open an issue**. Contact me immediately via [email](#).

🚀 Quick Start

Binary Execution

1. Download the executable from the [releases](#) page.
2. Follow the [Configuration](#) instructions.
3. Run the executable and access the web interface at `http://IP:Port` .
4. Generate a client and run it on the target device.
5. Start managing your devices!

Configuration

The configuration file `config.json` should be in the same directory as the executable.

Example:

```
{
  "listen": ":8000",
  "salt": "123456abcdef123456",
  "auth": {
    "username": "password"
  },
  "log": {
    "level": "info",
    "path": "./logs",
    "days": 7
  }
}
```

Main Parameters:

- `listen` (required): Format `IP:Port` .
- `salt` (required): Max length 24 characters. After modification, all clients need to be regenerated.
- `auth` (optional): Authentication credentials (`username:password`).
 - Hashed passwords are recommended (`$algorithm$hashed-password`).
 - Supported algorithms: `sha256` , `sha512` , `bcrypt` .
- `log` (optional): Logging configuration.
 - `level` : `disable` , `fatal` , `error` , `warn` , `info` , `debug` .
 - `path` : Log directory (default: `./logs`).
 - `days` : Log retention days (default: `7`).

Features

Feature/OS	Windows	Linux	MacOS
Process Manager	✓	✓	✓

Feature/OS	Windows	Linux	MacOS
Kill Process	✓	✓	✓
Network Traffic	✓	✓	✓
File Explorer	✓	✓	✓
File Transfer	✓	✓	✓
File Editor	✓	✓	✓
Delete File	✓	✓	✓
Code Highlighting	✓	✓	✓
Desktop Monitor	✓	✓	✓
Screenshot	✓	✓	✓
OS Info	✓	✓	✓
Remote Terminal	✓	✓	✓
* Shutdown	✓	✓	✓
* Reboot	✓	✓	✓
* Log Off	✓	✗	✓
* Sleep	✓	✗	✓
* Hibernate	✓	✗	✗
* Lock Screen	✓	✗	✗

 Functions marked with * may require administrator/root privileges.

Screenshots

Spark

Generate client C I @

Hostname	Username	Ping	CPU Usage	RAM Usage	Disk Usage	OS	Arch	RAM	MAC	LAN	WAN	Uptime	Network	Operations
LOCALHOST	1248	0ms	<div style="width: 10%;"></div>	<div style="width: 10%;"></div>	<div style="width: 10%;"></div>	windows	amd64	7.85 GB	00:00:00:00:00:00	0.0.0.0	0.0.0.0	22h 53m	2.2 Kbps 1 / 4.4 Kbps 1	Terminal Process Explorer ...

- Run
- Desktop
- Screenshot
- Lock
- Logoff
- Hibernate
- Suspend
- Restart
- Shutdown
- Offline

Spark

Generate client C I @

Hostname	Username	Ping	CPU Usage	RAM Usage
LOCALHOST	1248	0ms	<div style="width: 10%;"></div>	<div style="width: 10%;"></div>

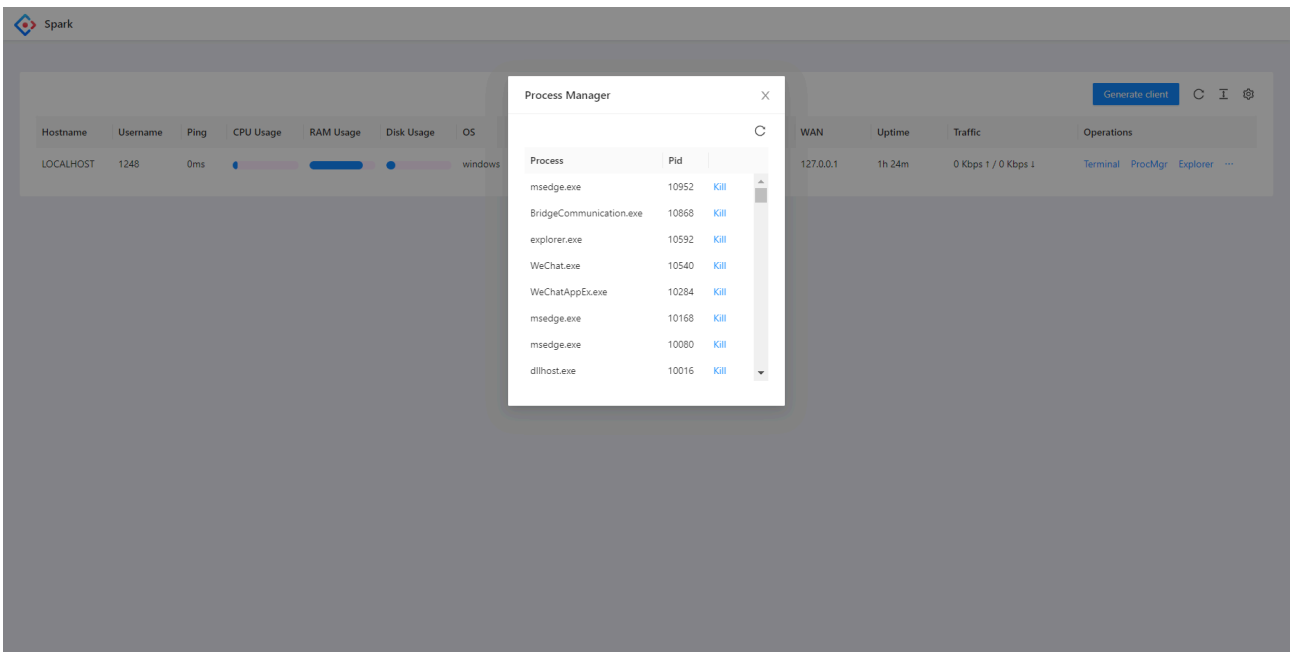
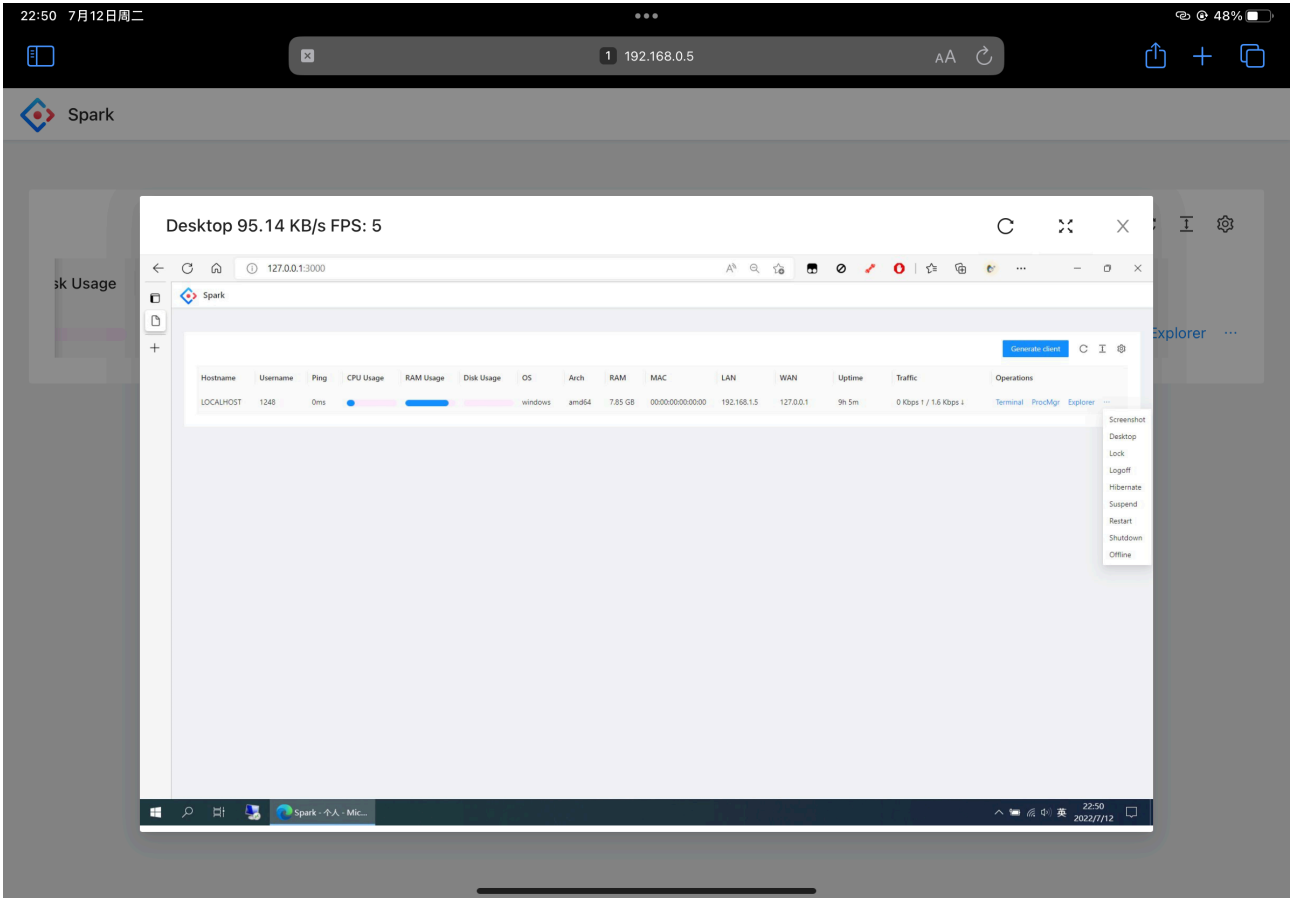
Terminal

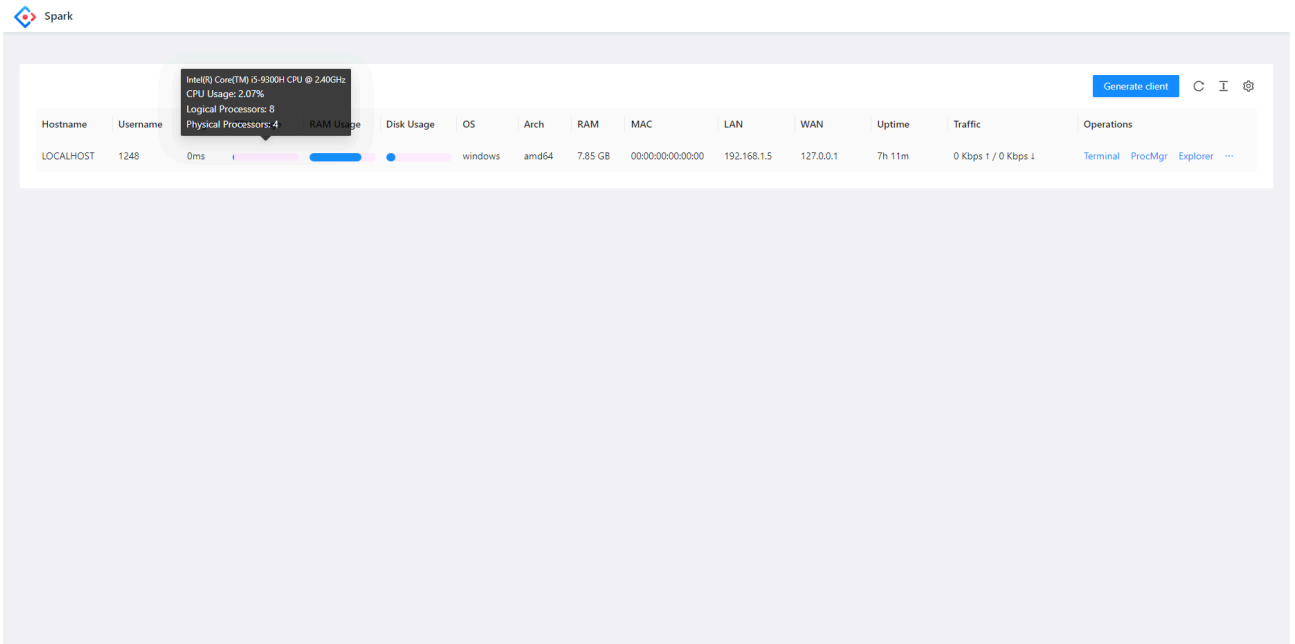
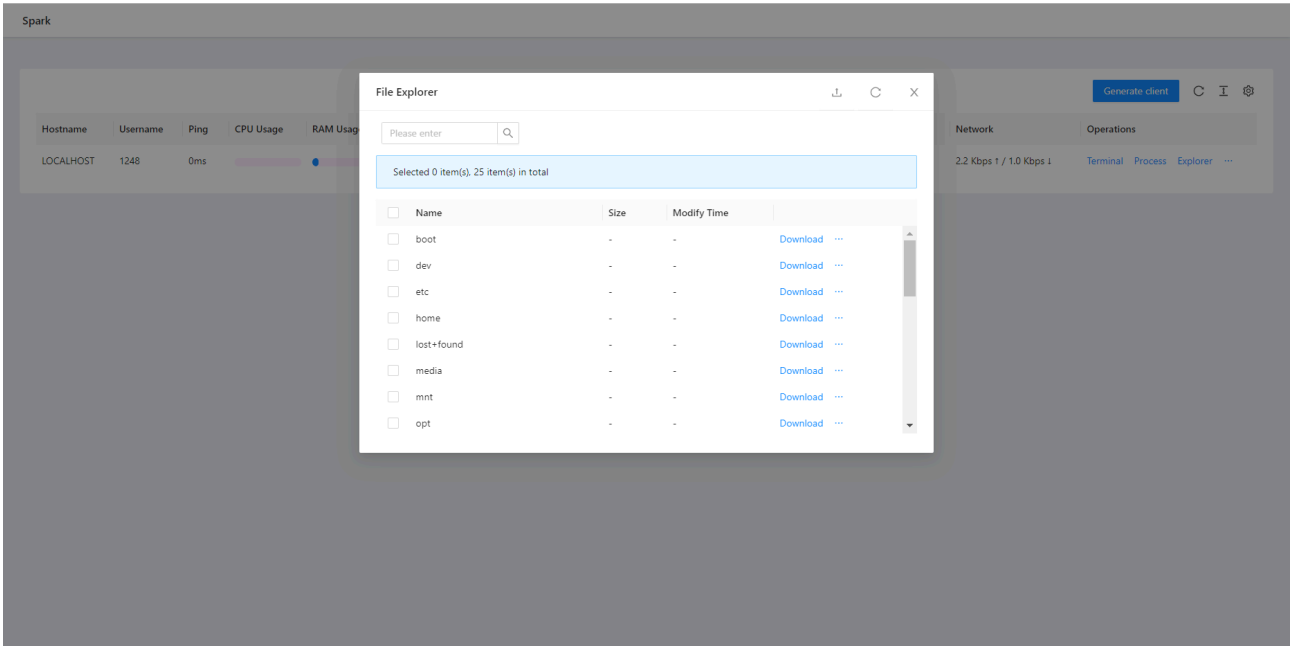
CTRL ESC TAB F1 F2 F3 F4 Function Keys ...

```
root@localhost:/#
```

Network Operations

2.2 Kbps 1 / 4.4 Kbps 1 Terminal Process Explorer ...





smb.conf

X

Save Search Replace Font Theme

```

1 #
2 # Sample configuration file for the Samba suite for Debian GNU/Linux.
3 #
4 #
5 # This is the main Samba configuration file. You should read the
6 # smb.conf(5) manual page in order to understand the options listed
7 # here. Samba has a huge number of configurable options most of which
8 # are not shown in this example
9 #
10 # Some options that are often worth tuning have been included as
11 # commented-out examples in this file.
12 # - When such options are commented with ";", the proposed setting
13 # differs from the default Samba behaviour
14 # - When commented with "#", the proposed setting is the default
15 # behaviour of Samba but the option is considered important
16 # enough to be mentioned here
17 #
18 # NOTE: Whenever you modify this file you should run the command
19 # "testparm" to check that you have not made any basic syntactic
20 # errors.
21
22 #===== Global Settings =====
23
24 [global]
25
26 ## Browsing/Identification ##
27
28 # Change this to the workgroup/NT-domain name your Samba server will part of
29 workgroup = WORKGROUP
30
31 ### Networking ###
32
33 # The specific set of interfaces / networks to bind to
34 # This can be either the interface name or an IP address/netmask;
35 # interface names are normally preferred
36 ; interfaces = 127.0.0.0/8 eth0
37
38 # Only bind to the named interfaces and/or networks; you must use the
39 # 'interfaces' option above to use this.
40 # It is recommended that you enable this feature if your Samba machine is
41 # not protected by a firewall or is a firewall itself. However, this

```



Development

Components

This project consists of three main components:

- **Client**
- **Server**
- **Front-end**

For OS support beyond Linux and Windows, additional C compilers may be required. For example, to support Android, install [Android NDK](#).

Build Guide

```

# Clone the repository
git clone https://github.com/XZB-1248/Spark
cd ./Spark

# Build the front-end
cd ./web
npm install
npm run build-prod

# Embed static resources
cd ..
go install github.com/rakyll/statik
statik -m -src="./web/dist" -f -dest="./server/embed" -p web -ns web

```

```
# Build the client
mkdir ./built
go mod tidy
go mod download
./scripts/build.client.sh

# Build the server
mkdir ./releases
./scripts/build.server.sh
```

Custom Features

If you need to customize some features, please contact me via i@1248.ink.

Dependencies

Spark contains many third-party open-source projects.

Lists of dependencies can be found at `go.mod` and `package.json`.

Some major dependencies are listed below.

Back-end

- [Go](#) ([License](#))
- [gin-gonic/gin](#) (MIT License)
- [imroc/req](#) (MIT License)
- [kbinani/screenshot](#) (MIT License)
- [shirou/gopsutil](#) ([License](#))
- [gorilla/websocket](#) (BSD-2-Clause License)
- [orcaman/concurrent-map](#) (MIT License)

Front-end

- [React](#) (MIT License)
- [Ant-Design](#) (MIT License)
- [axios](#) (MIT License)
- [xterm.js](#) (MIT License)

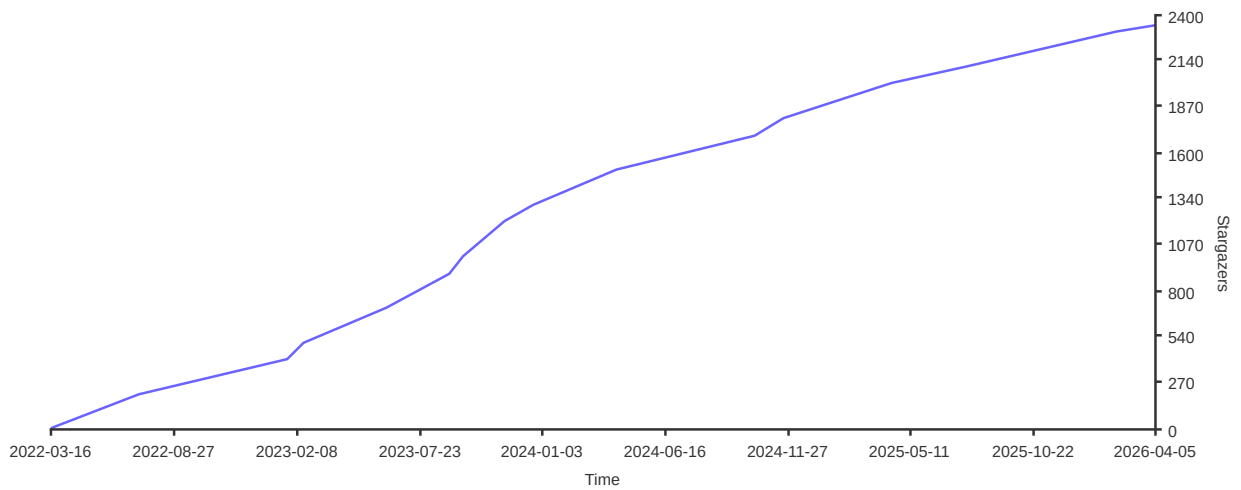
- [crypto-js](#) (MIT License)

Acknowledgements

- [natpass](#) (MIT License)



Stargazers over time



License

Distributed under the [BSD-2 License](#).

Source: <https://github.com/XZB-1248/Spark>