



链滴

Docker 搭建 HFish0.3

作者: [someone43608](#)

原文链接: <https://ld246.com/article/1567130367472>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

HFish 是一款基于 Golang 开发的跨平台多功能主动诱导型蜜罐框架系统，为了企业安全防护测试做了精心的打造

- 多功能 不仅仅支持 HTTP(S) 蜜罐，还支持 SSH、SFTP、Redis、Mysql、FTP、Telnet、暗网 等
- 扩展性 提供 API 接口，使用者可以随意扩展蜜罐模块 (WEB、PC、APP)
- 便捷性 使用 Golang 开发，使用者可以在 Win + Mac + Linux 上快速部署一套蜜罐平台

HFish 官方

- 官网: <https://hfish.io>
- 使用文档: <https://hfish.io/docs>

Github

- Git: <https://github.com/hacklcx/HFish>
- Download: <https://github.com/hacklcx/HFish/releases>

码云(Gitee)

- Git: <https://gitee.com/lauix/HFish>
- Download: <https://gitee.com/lauix/HFish/releases>

本人是在CentOS上搭建的，不会docker，纯属娱乐，大佬勿喷

1. 安装Docker

[百度一下](#)

由于我在此之前安装了Docker，但是在安装HFish出了点问题，百度后发现是需要更新Docker。

更新参考的文章: [如何将Docker升级到最新版本 - yaobo - 博客园](#)

2. 下载镜像

```
docker pull imdevops/hfish
```

3. 部署

```
docker run -d --name hfish -p 21:21 -p 22:22 -p 23:23 -p 3306:3306 -p 6379:6379 -p 9000:9000 -p 9001:9001 -p 9002:9002 -e USERNAME=账号 PASSWORD=密码 API_IP=api_ip:9001 imdevops/hfish:latest
```

需要注意的是：某些端口可能会冲突，可以使用其他端口代替，所需要端口的数量可以参考[config.ini](#)自己所需来确定。

4. 修改配置文件

docker exec -it hfish sh

cd opt/HFish

vi config.ini

将端口冲突的设置为其他端口.

```
[rpc]
status = 1
addr = 0.0.0.0:7879
name = Server
# RPC ..... 0 ..... 1 ..... 2 .....
# RPC ..... 1 ..... or ..... 2 .....
# ..... 1 ..... ..... 2 .....

[admin]
addr = 0.0.0.0:9001
# RPC ..... 2 ..... admin .....
# .....
# .....

[api]
status = 1
web_url = /api/v1/post/report
deep_url = /api/v1/post/deep_report
plug_url = /api/v1/post/plug_report
sec_key = 9cbf8a4dcb8e30682b927f352d6559a0
# ..... API 0 ..... 1 .....
# WEB ..... API
# ..... API
# ..... API
# API .....

[plug]
status = 1
addr = 0.0.0.0:8989
# ..... 0 ..... 1 ..... API
# .....

[web]
status = 1
addr = 0.0.0.0:9002
template = wordpress/html
index = index.html
static = wordpress/static
url = /
# ..... WEB 1 ..... 0 ..... API ... WEB .....
# WEB ..... 0.0.0.0 ..... 0.0.0.0 ..... Nginx .....
# WEB .....
# WEB .....
# WEB .....
# WEB .....
# WEB ..... / ..... index.html index.asp index.php

[deep]
status = 1
addr = 0.0.0.0:9003
template = deep/html
index = index.html
static = deep/static
url = /
# ..... 1 ..... 0 ..... API ...
# ..... WEB .....
# ..... WEB .....
# ..... WEB .....
# ..... WEB .....
# ..... WEB .....
# ..... WEB ..... / ..... index.html index.asp index.php

[ssh]
status = 2
addr = 0.0.0.0:9010
# ..... SSH 0 ..... 1 ..... 2 .....
# SSH ..... openssh .....

[redis]
status = 1
addr = 0.0.0.0:6379
# ..... Redis 0 ..... 1 .....
# Redis .....

[mysql]
status = 1
addr = 0.0.0.0:9011
files = /etc/passwd,/etc/group
# ..... Mysql 0 ..... 1 .....
# Mysql .....
# Mysql .....

[telnet]
config.ini 2/61 3%
```

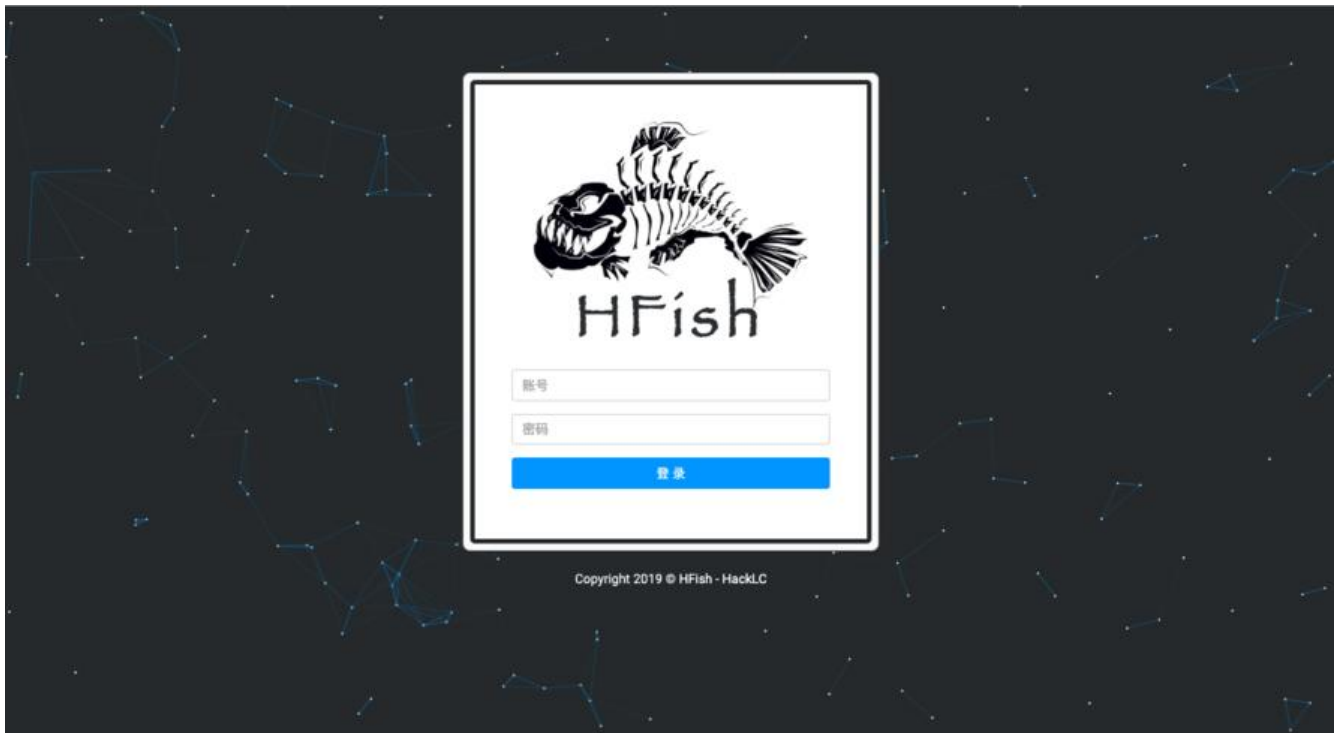
5. 重启容器

docker ps -a #查看容器ID

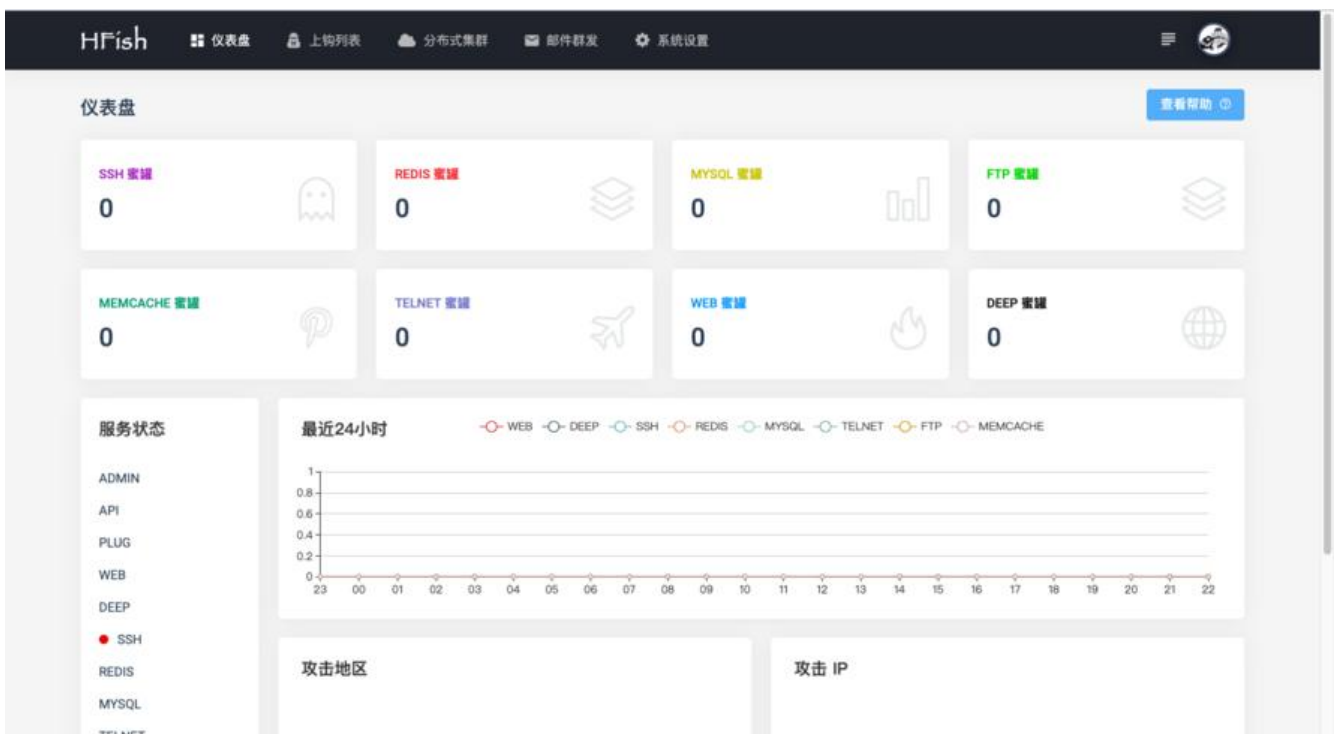
docker restart 容器ID

效果展示

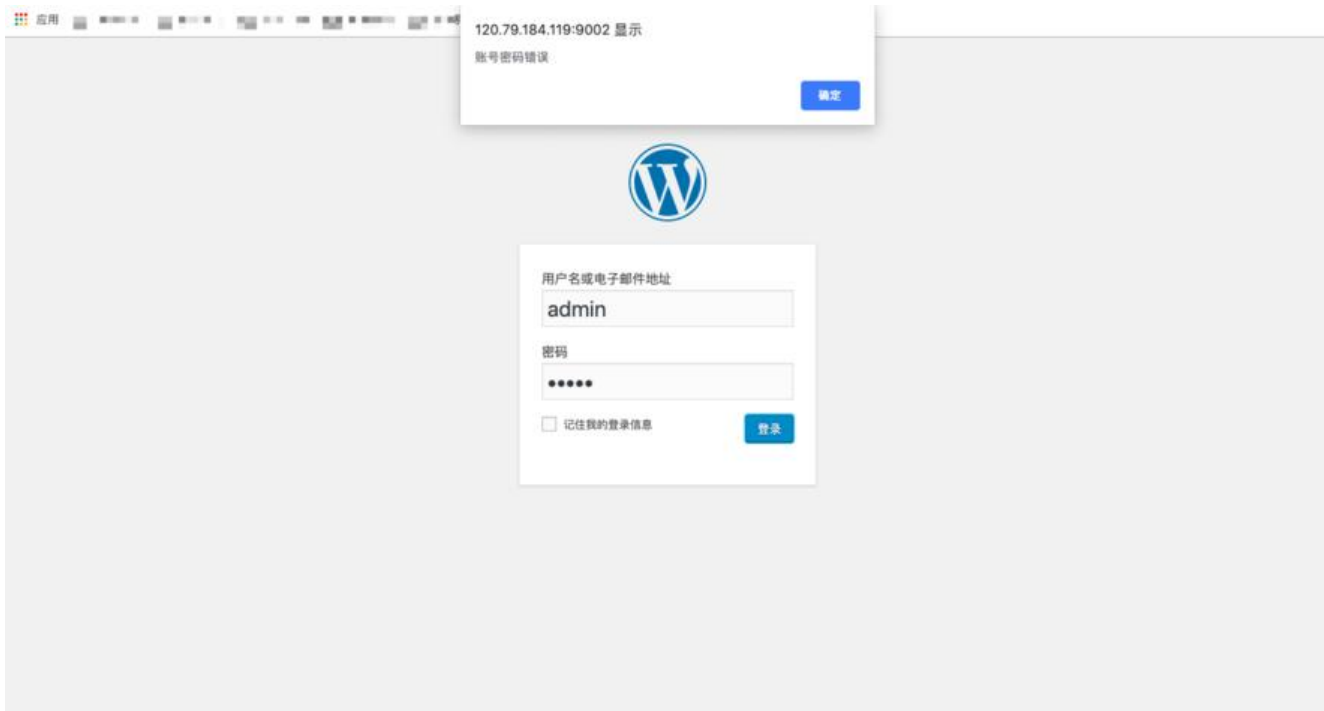
后台:



仪表盘:



尝试登录 wordpress:



后台显示:

