DeFiChainVaultAlarm

How to get the configuration values

WLAN password

configuration item

DeFiChainVaultAlarm | DefiChainAlarm_Cfg.h | DefiChainAlarm_Eeprom.cpp | DefiChainAlarm_Eeprom.h

```
#ifndef DEFICHAINALARM_CFG
#define DEFICHAINALARM_CFG


#define DEFAULT_WLAN_SSID "Skynet"
#define DEFAULT_WLAN_PASSWORD "1234"

#define BOTtoken "0000000000:0000000000000-0000000000000000000"  // your Bot Token (Get from Botfather)
// Use @myidbot to find out the chat ID of an individual or a group
// Also note that you need to click "start" on a bot before it can
// message you
#define CHAT_ID "000000000"

#define DEFAULT_DEFICHAIN_ADDR "0000000000000000000000000000000000000000"
#define DEFAULT_DEFICHAIN_VAULT "000000000000000000000000000000000000000000000000000000000000000000000000"

#define BUZZERPIN  15 /*GPIO pin number (remark: GPIO pin number may be different to uc port number)*/


#endif
```

WLAN password

details

The WLAN (2.4GHz) connection is needed to get all the DeFiChain information it needs.
Also the Telegram bot will use this internet connection.

Please use the password „NULL“ in case you want to configure a guest-network without password.
Unfortunately this is an workaround as you cannot send an empty message with telegram.

Note: In case you change your WLAN credentials by using the Telegram bot, this hard coded ones will be used as fallback credentials in case you did update them with invalid ones.

Telegram bot

configuration item

DeFiChainVaultAlarm | DefiChainAlarm_Cfg.h | DefiChainAlarm_Eeprom.cpp | DefiChainAlarm_Eeprom.h

```
#ifndef DEFICHAINALARM_CFG
#define DEFICHAINALARM_CFG


#define DEFAULT_WLAN_SSID "Skynet"
#define DEFAULT_WLAN_PASSWORD "1234"

#define BOTtoken "0000000000:00000000000000-000000000000000000"  // your Bot Token (Get from Botfather)
// Use @myidbot to find out the chat ID of an individual or a group
// Also note that you need to click "start" on a bot before it can
// message you
#define CHAT_ID "000000000"

#define DEFAULT_DEFICHAIN_ADDR "0000000000000000000000000000000000000000"
#define DEFAULT_DEFICHAIN_VAULT "00000000000000000000000000000000000000000000000000000000000000000000"

#define BUZZERPIN  15 /*GPIO pin number (remark: GPIO pin number may be different to uc port number)*/


#endif
```

Telegram bot

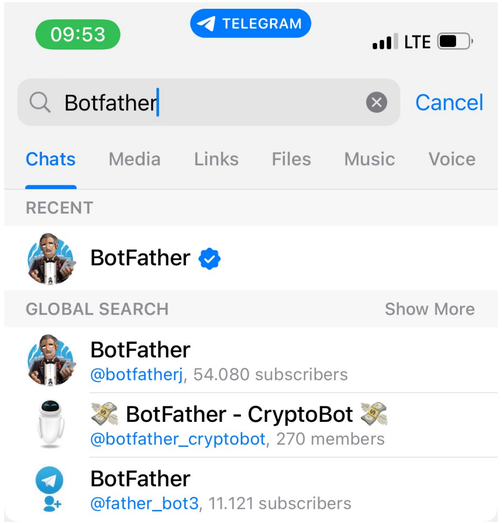
details

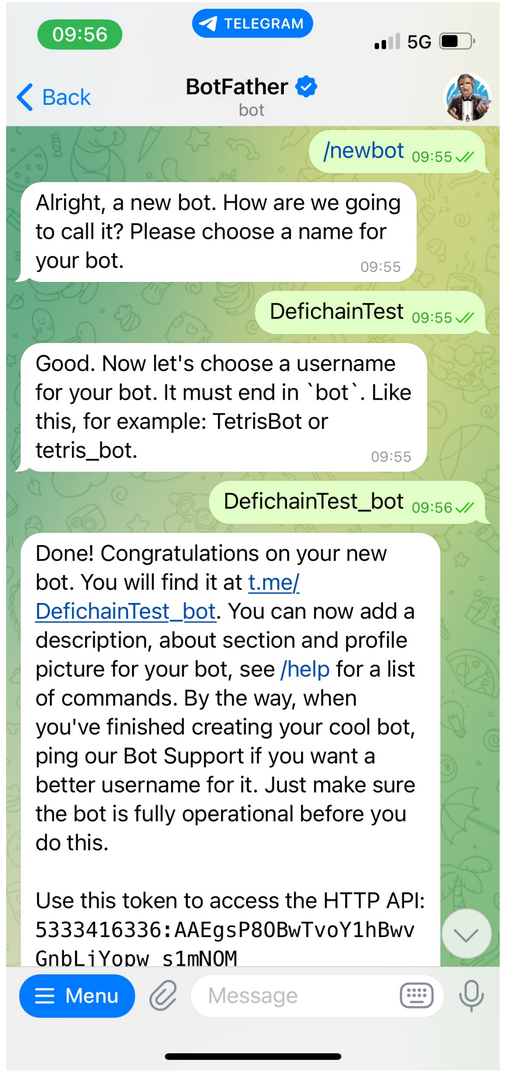
The Telegram bot will not „run“ at the device directly. The Telegram „botfather“ has to create this bot and the device will connect to it later on and control it.
So you need to create this bot using your smartphone and enter the ID, which you get from the botfather, in the code.

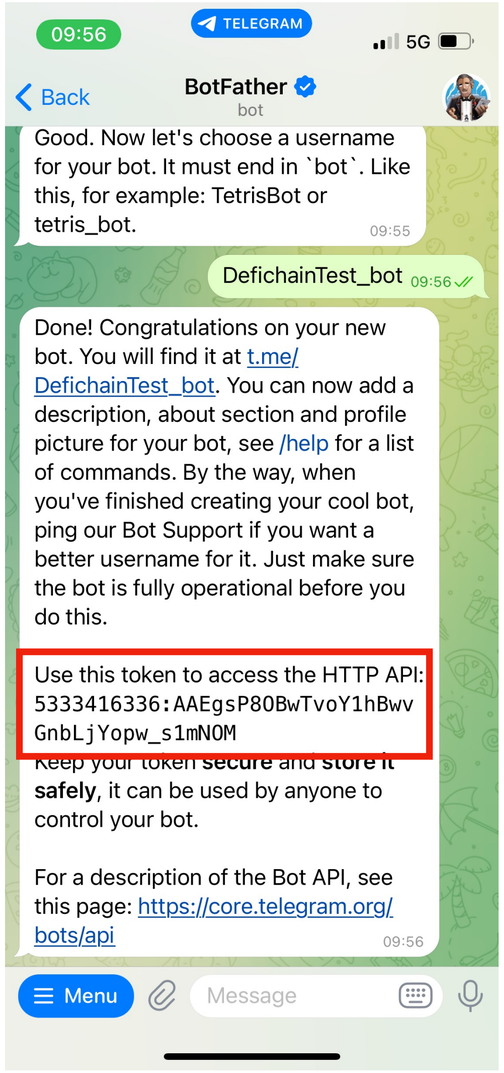
For screenshots see next page…

Telegram bot

screenshots







Telegram own chat/user ID

configuration item

DeFiChainVaultAlarm | DefiChainAlarm_Cfg.h | DefiChainAlarm_Eeprom.cpp | DefiChainAlarm_Eeprom.h

```
#ifndef DEFICHAINALARM_CFG
#define DEFICHAINALARM_CFG


#define DEFAULT_WLAN_SSID "Skynet"
#define DEFAULT_WLAN_PASSWORD "1234"

#define BOTtoken "0000000000:0000000000000-0000000000000000000"  // your Bot Token (Get from Botfather)
// Use @myidbot to find out the chat ID of an individual or a group
// Also note that you need to click "start" on a bot before it can
// message you
#define CHAT_ID "000000000"

#define DEFAULT_DEFICHAIN_ADDR "0000000000000000000000000000000000000000"
#define DEFAULT_DEFICHAIN_VAULT "00000000000000000000000000000000000000000000000000000000000000000000"

#define BUZZERPIN  15 /*GPIO pin number (remark: GPIO pin number may be different to uc port number)*/


#endif
```

Telegram own chat/user ID

details

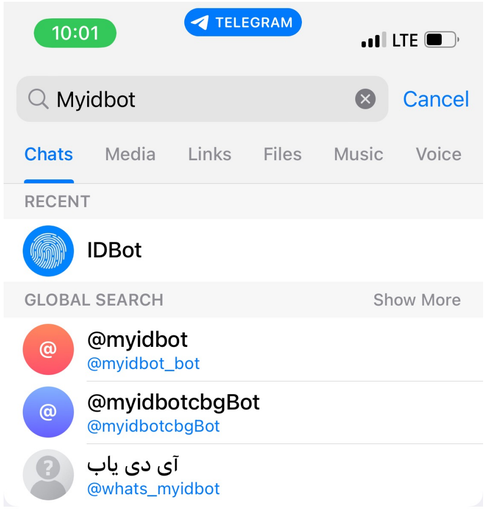
The device will check all incoming messages from Telegram if the sender was you. This will prevent others to changes settings on your device or read your vault information.

In order to get your chat/user ID you can ask the „myidbot“ on Telegram.

For screenshots see next page…

Telegram own chat/user ID

screenshots



DeFiChain address and vault ID

configuration item

DeFiChainVaultAlarm | DefiChainAlarm_Cfg.h | DefiChainAlarm_Eeprom.cpp | DefiChainAlarm_Eeprom.h

```
#ifndef DEFICHAINALARM_CFG
#define DEFICHAINALARM_CFG


#define DEFAULT_WLAN_SSID "Skynet"
#define DEFAULT_WLAN_PASSWORD "1234"

#define BOTtoken "0000000000:0000000000000-0000000000000000000"  // your Bot Token (Get from Botfather)
// Use @myidbot to find out the chat ID of an individual or a group
// Also note that you need to click "start" on a bot before it can
// message you
#define CHAT_ID "000000000"

#define DEFAULT_DEFICHAIN_ADDR "00000000000000000000000000000000000000"
#define DEFAULT_DEFICHAIN_VAULT "0000000000000000000000000000000000000000000000000000000000000000"

#define BUZZERPIN  15 /*GPIO pin number (remark: GPIO pin number may be different to uc port number)*/


#endif
```

DeFiChain address and vault ID

details

Accessing the vault status using the ocean API needs the vault ID and in addition to that the owner’s address.
In the beginning defiscan.live was used as data source, which would only need the vault ID, but using the ocean API turned out to be more stable.

Buzzer Pin connection

configuration item

DeFiChainVaultAlarm | DefiChainAlarm_Cfg.h | DefiChainAlarm_Eeprom.cpp | DefiChainAlarm_Eeprom.h

```
#ifndef DEFICHAINALARM_CFG
#define DEFICHAINALARM_CFG


#define DEFAULT_WLAN_SSID "Skynet"
#define DEFAULT_WLAN_PASSWORD "1234"

#define BOTtoken "0000000000:00000000000000-000000000000000000"  // your Bot Token (Get from Botfather)
// Use @myidbot to find out the chat ID of an individual or a group
// Also note that you need to click "start" on a bot before it can
// message you
#define CHAT_ID "000000000"

#define DEFAULT_DEFICHAIN_ADDR "0000000000000000000000000000000000000000"
#define DEFAULT_DEFICHAIN_VAULT "00000000000000000000000000000000000000000000000000000000000000000000"

#define BUZZERPIN  15 /*GPIO pin number (remark: GPIO pin number may be different to uc port number)*/


#endif
```

Buzzer Pin connection

details

The software can controll buzzers which can be activated by a simple 5V signal. The connection to this buzzer pin shall be configured. Here is a screenshot of the possible pins: