

XIZNITBOT

CUSTOMER SETUP GUIDE

Version 1.0 | support@xiznit-bot.com | xiznit-bot.com

Welcome to Xiznit Bot

Congratulations on your purchase! Xiznit Bot connects your TradingView Pine Script alerts directly to your broker and executes trades automatically — 24/7, on your own private server, with no monthly fees. This guide will walk you through the complete setup from start to finish.

What You Need Before Starting

- Your Xiznit Bot license key (in your purchase confirmation email)
- A DigitalOcean account — digitalocean.com (free to create)
- A broker account: **Alpaca** (alpaca.markets) or **Tradier** (tradier.com)
- A TradingView account with a Pine Script strategy
- About 30–45 minutes

Setup Overview

Step 1	Create Your VPS Server on DigitalOcean
Step 2	Install Docker on Your Server
Step 3	Set Up Your Broker API Keys
Step 4	Configure and Start the Bot
Step 5	Set Up TradingView Alerts

Step 6

Verify Everything is Working

Step 1 — Create Your VPS Server

Your bot needs a server that runs 24/7. DigitalOcean offers a \$6/month server (called a Droplet) that is perfect for running Xiznit Bot.

1.1 Create a DigitalOcean Account

Go to digitalocean.com and sign up for a free account.

1.2 Create a Droplet

- Click **Create** in the top right, then select **Droplets**
- Choose region: pick the one closest to you
- Under **Choose an image**, select **Ubuntu 24.04 LTS**
- Under **Choose Size**, select **Basic** then pick the **\$6/month** plan (1GB RAM)
- Under **Authentication**, choose **Password** and set a strong password
- Click **Create Droplet** and wait about 60 seconds
- Copy your Droplet's **IP address** — you'll need it in the next step

1.3 Connect to Your Server

Once your Droplet is created, click the **Console** button in DigitalOcean to open a terminal. Log in with username **root** and the password you set.

Tip: Save your IP address and password somewhere safe. You'll need them every time you log in.

Step 2 — Install Docker on Your Server

Docker is the software that runs Xiznit Bot. Copy and paste these commands into your server terminal one at a time. Press Enter after each one and wait for it to finish.

Update your server:

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

Install Docker:

```
apt install docker.io docker-compose -y
```

Verify Docker is installed:

```
docker --version
```

*If Docker is installed correctly, the last command will show something like: **Docker version 29.x.x***

Step 3 — Set Up Your Broker API Keys

Xiznit Bot supports two brokers. Follow the instructions for **your** broker below. You only need to set up one broker.

Option A — Alpaca (Stocks + Crypto)

- Go to **alpaca.markets** and create a free account
- Once logged in, click **Paper Trading** in the left sidebar (for testing)
- Click **Your API Keys** on the right side of the screen
- Click **Generate New Key**
- Copy your **API Key ID** and **Secret Key** — save them somewhere safe
- When ready for live trading, switch to **Live Trading** and generate live keys

***Note:** Alpaca paper trading is completely free with no deposit required. Perfect for testing your strategy before going live.*

Option B — Tradier (Stocks, Options + Futures)

- Go to **tradier.com** and create a free account
- Complete identity verification (takes 1 business day)
- Once approved, log in and go to **Settings**
- Click **API Access** in the left menu
- Copy your **Access Token** — this is your TRADIER_TOKEN
- Also copy your **Account ID** — shown at the top of your account page
- For paper trading, use the **Sandbox Token** instead

***Note:** Tradier API access is completely free for brokerage account holders. Futures trading requires a separate Tradier Futures account.*

Step 4 — Configure and Start the Bot

4.1 Upload the Bot Package

You received a file called **xiznit-bot-package.zip** with your purchase. Upload it to your server by running this command in your server terminal:

- In your DigitalOcean console, create a folder for the bot:

```
mkdir -p /opt/xiznit && cd /opt/xiznit
```

- Copy the contents of **docker-compose.yml** and **config.env** from the zip file onto your server using the instructions in Section 4.2 below.

4.2 Edit Your Configuration File

Run this command to create your config file:

```
nano /opt/xiznit/config.env
```

Paste in the following, filling in your own values:

```
# Your license key LICENSE_KEY=YOUR_LICENSE_KEY_HERE # Choose your broker: alpaca  
or tradier BROKER=alpaca # paper = testing, live = real money TRADING_MODE=paper #  
Alpaca keys (if using Alpaca) ALPACA_API_KEY=YOUR_ALPACA_API_KEY  
ALPACA_SECRET_KEY=YOUR_ALPACA_SECRET_KEY # Tradier keys (if using Tradier)  
TRADIER_TOKEN=YOUR_TRADIER_TOKEN TRADIER_ACCOUNT_ID=YOUR_ACCOUNT_ID
```

When done, save with **Ctrl+X**, then **Y**, then **Enter**.

4.3 Create the Docker Compose File

Run this command:

```
nano /opt/xiznit/docker-compose.yml
```

Paste in the following exactly as shown:

```
version: '3.8' services: xiznit-bot: image: xiznit/xiznit-bot:latest  
container_name: xiznit-bot restart: always env_file: - config.env ports: -  
"8000:8000"
```

Save with **Ctrl+X**, then **Y**, then **Enter**.

4.4 Start the Bot

Run this command from the /opt/xiznit folder:

```
cd /opt/xiznit && docker-compose up -d
```

Docker will download the bot image and start it automatically. This may take 1-2 minutes the first time. When complete you will see: **xiznit-bot ... done**

4.5 Verify the Bot is Running

```
docker ps
```

You should see **xiznit-bot** listed with status **Up**. Your bot is now running 24/7!

Step 5 — Set Up TradingView Alerts

This is where the magic happens. You configure TradingView to send your strategy signals directly to your bot's webhook URL.

5.1 Your Webhook URL

Your bot's webhook URL is:

```
http://YOUR_SERVER_IP:8000/webhook
```

Replace **YOUR_SERVER_IP** with your DigitalOcean Droplet's IP address.

5.2 Create a TradingView Alert

- Open TradingView and load your strategy on a chart
- Click the **Alert** button (clock icon) in the top toolbar
- Set the **Condition** to your strategy signal
- Under **Actions**, check **Webhook URL**
- Paste your webhook URL into the Webhook URL field
- Under **Message**, paste the JSON payload (see Section 5.3 below)
- Set **Alert name** to something descriptive like "BTCUSD Buy Signal"
- Click **Create**

5.3 JSON Field Reference

Xiznit Bot accepts a flexible JSON payload. Only "ticker" and "action" are required — all other fields are optional depending on your strategy. The table below explains every supported field:

Field	Type	Required	Description
"ticker"	string	YES	The trading symbol. Accepts TradingView format (e.g. BTCUSDT, AAPL, MNQ1!)
"action"	string	YES	The trade action. See Section 5.4 for all supported values.
"price"	number	No	Current price. Use {{close}} in TradingView. Used for logging only.
"qty"	number	No	Number of shares/contracts/units. Defaults to bot preset if omitted.
"lots"	number	No	Alternative to qty. Some strategies use "lots" instead of "qty".
"sl"	number	No	Stop loss price level. Used with update_sl action or for reference.
"tp"	string	No	Take profit level. Use "TP1", "TP2", or "TP3" with exit action.
"reason"	string	No	Reason for closing. Examples: eod_flatten, sl, max_duration, weekend_gap, blackout.
"side"	string	No	Explicit side override: "buy" or "sell". Normally inferred from action.
"comment"	string	No	Optional comment or label from your strategy. Logged but not acted on.

"strategy"	string	No	Strategy name label for logging. Useful if running multiple strategies.
"exchange"	string	No	Exchange name from TradingView. Logged for reference.
"interval"	string	No	Timeframe label (e.g. "15m", "1h"). Logged for reference.

5.4 Supported Actions

Action Value	What it Does	Extra Fields Needed
"buy"	Enter a long position (buy)	qty (optional)
"sell"	Enter a short position (sell)	qty (optional)
"close"	Close the current position for this symbol	reason (optional)
"close_all"	Close ALL open positions immediately	reason (optional)
"exit"	Exit at a specific TP level	tp: "TP1", "TP2", or "TP3"
"update_sl"	Update the stop loss reference price	sl: new price level
"flat"	Alias for close — flattens the position	reason (optional)
"long"	Alias for buy — enter long	qty (optional)
"short"	Alias for sell — enter short	qty (optional)

5.5 Supported Symbols

Xiznit Bot automatically maps TradingView symbol formats to broker formats. You can use any of the formats below and the bot will handle the conversion:

TradingView Symbol	Broker Symbol	Asset Class
BTCUSDT or BTCUSD	BTCUSD	Crypto
ETHUSDT or ETHUSD	ETHUSD	Crypto
SOLUSDT or SOLUSD	SOLUSD	Crypto
AAPL, TSLA, SPY, etc.	Same symbol	Stocks / ETFs
MNQ1!, MES1!, etc.	Same symbol	Futures (Tradier)
MGC1!, SI1!, etc.	Same symbol	Futures (Tradier)
Any symbol not listed	Passed through as-is	All others

5.6 JSON Payload Examples

Basic buy order:

```
{ "ticker": "AAPL", "action": "buy", "qty": 10 }
```

Full bracket order with TP and SL:

```
{ "ticker": "{{ticker}}", "action": "buy", "price": {{close}}, "qty":  
{{strategy.order.contracts}}, "sl": 0, "tp": "TP1", "strategy": "ORB" }
```

EOD flatten (close all positions):

```
{ "ticker": "{{ticker}}", "action": "close_all", "reason": "eod_flatten" }
```

Exit at TP2 (take partial profit):

```
{ "ticker": "{{ticker}}", "action": "exit", "tp": "TP2", "qty": 1 }
```

Update stop loss to breakeven:

```
{ "ticker": "{{ticker}}", "action": "update_sl", "sl": {{plot("Entry Price"}} } }
```

Step 6 — Verify Everything is Working

6.1 Check Bot Health

Open a browser and go to:

```
http://YOUR_SERVER_IP:8000/health
```

You should see: `{"status":"online","bot":"Xiznit-Trading-Bot"}`. This confirms your bot is running and reachable.

6.2 Send a Test Alert

You can send a test order to make sure everything is connected. Run this from your server terminal (replace YOUR_SERVER_IP):

```
curl -X POST http://YOUR_SERVER_IP:8000/webhook \ -H 'Content-Type: application/json' \ -d '{"ticker":"AAPL","action":"buy","price":150,"qty":1}'
```

You should get back: `{"status":"received","broker_response":{...}}`

6.3 View the Trade Log

To see all trades and activity, run:

```
docker logs xiznit-bot
```

Useful Commands Reference

Command	What it Does
<code>docker ps</code>	Check if bot is running
<code>docker logs xiznit-bot</code>	View bot activity and trade log
<code>docker logs xiznit-bot -f</code>	Watch live bot activity
<code>docker-compose restart</code>	Restart the bot
<code>docker-compose down</code>	Stop the bot
<code>docker-compose up -d</code>	Start the bot
<code>docker-compose pull</code>	Update to latest version

Troubleshooting

Problem: Bot won't start

Solution: Check your license key is correct in config.env. Run `docker logs xiznit-bot` to see the error.

Problem: Orders not executing

Solution: Check your API keys are correct. Make sure `TRADING_MODE` matches your account type (paper vs live).

Problem: TradingView alerts not firing

Solution: Make sure your webhook URL is correct and port 8000 is open. Check that your JSON payload is valid.

Problem: Bot stopped after server restart

Solution: This shouldn't happen as `restart: always` is set. Run `docker-compose up -d` to restart manually.

Problem: Invalid license key error

Solution: Contact support@xiznit-bot.com with your license key and we'll verify it.

Legal Disclaimer & User Agreement

PLEASE READ THIS AGREEMENT CAREFULLY BEFORE USING XIZNIT BOT. BY PURCHASING AND/OR USING THIS SOFTWARE, YOU AGREE TO BE BOUND BY THE TERMS AND CONDITIONS SET FORTH BELOW. IF YOU DO NOT AGREE TO THESE TERMS, DO NOT USE THIS SOFTWARE.

1. SOFTWARE LICENSE

Xiznit Bot ("the Software") is licensed, not sold. Upon purchase, you are granted a single-user, non-transferable, non-exclusive license to install and use the Software on one server for your personal trading activities. You may not resell, redistribute, sublicense, copy, reverse engineer, decompile, or attempt to extract the source code of the Software in whole or in part. Unauthorized distribution or sharing of your license key will result in immediate revocation without refund.

2. NO FINANCIAL ADVICE

Xiznit Bot is an automated trade execution tool only. The Software does not provide financial advice, investment recommendations, or trading signals of any kind. All trading decisions, strategies, and parameters are solely determined by you, the user. Past performance of any trading strategy does not guarantee future results. Trading in financial markets, including but not limited to stocks, futures, options, forex, and cryptocurrency, involves substantial risk of loss and is not appropriate for all investors. You should only trade with money you can afford to lose.

3. LIMITATION OF LIABILITY

XIZNIT BOT, ITS CREATORS, OWNERS, EMPLOYEES, AND AFFILIATES SHALL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, CONSEQUENTIAL, OR PUNITIVE DAMAGES OF ANY KIND, INCLUDING BUT NOT LIMITED TO: trading losses, loss of profits, loss of data, missed opportunities, or any other financial losses arising from your use of or inability to use the Software. This limitation applies regardless of whether such damages were foreseeable or whether Xiznit Bot was advised of the possibility of such damages. By using the Software, you expressly agree that Xiznit Bot bears no responsibility whatsoever for any financial outcomes resulting from trades executed by the Software.

4. USER RESPONSIBILITY

You are solely responsible for: (a) ensuring the Software is properly configured with correct broker API keys and trading parameters; (b) monitoring your trading account and positions; (c) ensuring compliance with your broker terms of service and all applicable laws and regulations in your jurisdiction; (d) maintaining the security of your server, license key, and API credentials; (e) any and all trading activity executed through the Software on your account. Xiznit Bot is not responsible for errors resulting from incorrect configuration, connectivity issues, broker outages, or any circumstances beyond its control.

5. PROP FIRM & FUNDED ACCOUNT DISCLAIMER

If you are using Xiznit Bot with a proprietary trading firm ("prop firm") or funded account, you are solely responsible for ensuring that your use of automated trading software complies with that firm rules and terms of service. Xiznit Bot makes no guarantees regarding compatibility with any specific prop firm requirements. Violation of a prop firm rules resulting in account suspension or termination is not the responsibility of Xiznit Bot.

6. ALL SALES ARE FINAL — NO REFUND POLICY

ALL SALES ARE FINAL. Due to the digital nature of this product, Xiznit Bot does not offer refunds, exchanges, or credits under any circumstances once the license key has been issued and/or the software package has been delivered. By completing your purchase, you acknowledge and agree that you have read this agreement and that no refund will be issued for any reason including but not limited to: change of mind, trading losses, technical difficulties, inability to configure the software, broker incompatibility, or dissatisfaction with results.

7. CHARGEBACK POLICY

CHARGEBACKS WILL BE DISPUTED AND DENIED. If you initiate a chargeback or payment dispute with your bank, credit card provider, or payment processor after purchasing Xiznit Bot, we reserve the right to: (a) immediately and permanently revoke your license key with no possibility of reinstatement; (b) provide your payment processor and bank with a full copy of this agreement, your purchase record, and proof of software delivery as evidence that the chargeback is fraudulent; (c) pursue any and all legal remedies available. By completing your purchase you acknowledge that this constitutes a legally binding agreement and that initiating a chargeback after receiving the software constitutes fraud.

8. INTELLECTUAL PROPERTY

All code, algorithms, logic, and components of Xiznit Bot are the exclusive intellectual property of Xiznit Bot and its creators. The Software is protected by copyright law and international treaties. Any unauthorized copying, modification, distribution, or reverse engineering of the Software is strictly prohibited and will be prosecuted to the fullest extent of the law.

9. LICENSE REVOCATION

Xiznit Bot reserves the right to revoke your license key at any time, without refund, if you: (a) violate any term of this agreement; (b) attempt to reverse engineer or redistribute the Software; (c) share your license key with unauthorized users; (d) initiate a chargeback or payment dispute; (e) engage in any activity that Xiznit Bot determines, in its sole discretion, to be harmful to its business or other customers.

10. GOVERNING LAW & ENTIRE AGREEMENT

This agreement constitutes the entire agreement between you and Xiznit Bot regarding the Software and supersedes all prior agreements and understandings. This agreement shall be governed by applicable law. If any provision of this agreement is found to be unenforceable, the remaining provisions shall remain in full force and effect. Xiznit Bot reserves the right to update these terms at any time. Continued use of the Software constitutes acceptance of any updated terms.

USER ACKNOWLEDGEMENT

By purchasing and using Xiznit Bot, I confirm that I have read, understood, and agree to all terms and conditions set forth in this agreement. I understand that all sales are final, no refunds will be issued, and chargebacks will be disputed. I accept full responsibility for all trading decisions and financial outcomes resulting from my use of this software.

Purchase Date: _____ License Key: _____

Need Help?

Email us at support@xiznit-bot.com and we'll walk you through any issues. Setup assistance sessions are also available for \$100 — we'll screen share and get everything running for you.

xiznit-bot.com