A BUYER'S CHECKLIST

The Questions to Ask Before You Buy an Automated Strategy

Written by someone who sells them.

You are about to send money to somebody you have never met, for software you cannot read, that will trade a live account while you are at work.

That is the transaction. Strip away the website and the testimonials and that is what is actually happening — and most people walk into it having asked four or five questions, none of which tell them anything useful.

I build and sell automated futures strategies for a living, so I am on the receiving end of these conversations constantly. I know which questions people ask. I know which ones I wish they would ask. The gap between those two lists is wide, and it is why a lot of people lose money on tools that were never going to work for them.

Print this. Take it to the call.

FIRST — THE QUESTIONS THAT TELL YOU NOTHING

- ***"How many trades does it take?"*** Frequency is not quality. A strategy that skipped today's setup is not broken — that setup did not meet its conditions. Fewer, better trades mean smaller drawdowns.
- ***"What's the best strategy?"*** There is no answer that stays true. One market suits a breakout, another suits a fade. The best strategy in March is not the best in September.
- ***"What results can I expect?"*** Anybody who answers with a number is telling you something they cannot know.
- ***"Can I just turn it on and never worry about it?"*** No. Automation is a job you can do in less time than discretionary trading. It is not passive income.
- ***"Show me how this tune performed eighteen months ago."*** Your concern is right — you are worried about curve fitting, and you should be. But the artifact is wrong. A tune built for today's market, judged against a market from eighteen months ago, tells you only that nobody would have run it then.

THE QUESTIONS THAT ACTUALLY TELL YOU SOMETHING

☐ 1. How adaptable is this to changing market conditions?

If you only ask one question from this section, ask this one. What makes a strategy good is not whether it worked over years — it is whether it can be adjusted to the market you are in today.

Good answer: shows you the tuning process and what they change. **Dodge:** points back at the long-run track record.

☐ 2. When it gets bad, how bad does it get?

The matched pair to the question above. One asks how deep the hole goes; the other asks how you climb out. Ask all four parts: what is the worst drawdown, how long did it last, what do you do to recover, and what do you do to prevent it happening the same way twice.

Good answer: all four parts, unprompted. **Dodge:** a single number with no recovery plan — that is half an answer.

☐ 3. Is every parameter tunable, or are some locked?

Ten seconds to ask, and it tells you a lot. If parameters are blacked out you cannot adapt the strategy to current conditions — which is the thing that decides whether it keeps working. A strategy you cannot tune has an expiration date somebody else set for you.

Good answer: everything is open, and here is what I would touch first. **Dodge:** "the settings are proprietary."

☐ 4. When is this designed to fail?

The set that separates the real from the rest. What conditions does it thrive in? When would you recommend I *not* run it? What kind of market does it start losing money in? What do you do when that happens — sit out, or run something else? And what tells you it is working again?

Good answer: names the conditions specifically. **Dodge:** "it works in all market conditions." Every strategy fails eventually. Every single one.

☐ 5. What's the thesis behind it?

Why does this work, and why did you build it this way? They cannot give away the secret sauce and you should not ask them to — but they should be able to explain the reasoning. Then you either understand it or you do not, and that is the test. Not whether the explanation impresses you. Whether it makes sense to you. You cannot read the code; if nobody explains the logic, you are flying blind.

Good answer: plain-English reasoning you can follow. **Dodge:** jargon, or "proprietary algorithm."

6. What happens after it fires?

Almost nobody asks about execution, and it is where real money gets lost. Does it close positions properly? Do the stops work as specified? If it has trailing stops, do they actually activate and trail? What errors does it throw? Do I have the right data feed for it? A strategy that picks good trades and manages them badly will still lose you money.

Good answer: knows the failure modes and the data requirements cold. **Dodge:** has never thought about it.

7. What do you recommend, and why?

The reasoning is the test, not the recommendation. Follow it up: what is your solution to the problem you just described? What do I need to make this work properly, and why do I need that? You are testing the knowledge of the person selling, not just the product.

Good answer: a recommendation with conditions attached. **Dodge:** whatever is most expensive.

8. Can I have thirty minutes of your time?

People get busy and I understand that. But send an email and see whether you get a response. A vendor who will not answer you before the sale will not answer you after it.

Good answer: yes, here are some times. **Dodge:** silence.

9. Can you show me — live?

On that call, do not settle for being told. Ask them to pull up a strategy and tune it in front of you: this is how I adjust it for current conditions, this is what changed, this is what is different. Anybody who cannot demonstrate that live does not understand their own product well enough to sell it to you.

Good answer: shares a screen and does it. **Dodge:** sends a report instead.

10. What am I actually getting besides the code?

Look at what they are selling. If it is the strategy and a backtest and nothing else — no educational material, no explanation of how it works, no thesis, no community, no willingness to walk you through the logic — that is your answer, and you can see it before you ever speak to them. Selling a strategy raw fails every time. You cannot fly a plane unless somebody teaches you how.

Good answer: a system around the software. **Dodge:** a download link.

IF YOU ONLY GET ONE QUESTION

"When does this break, what conditions break it, and what do you do about it?"

It is the most revealing question you can ask and also the most polite. It does not sound like an accusation — it sounds like a technical question, because it is one. Nobody gets offended by it.

And there is no sales pitch available for the answer. You cannot market your way through "here is when my product stops working." You can only answer it with knowledge, or fail to answer it at all.

Every strategy breaks. If they cannot tell you when, and what they do about it, they are selling you junk.

TWO MORE THINGS, AND THEN GO MAKE THE CALL

A confident seller is not upset by any question. He has an answer for all of them, because he has already stress-tested himself. If he cannot answer, that is the problem — not the question. Defensiveness is information.

And part of this is on you. Somebody else's tunes and guidance are fine at the start — that is what you are paying for. But they are training wheels. Eventually you have to understand what the strategy does, what makes it work, and what makes it stop, because the person who wrote it cannot decide how you are going to manage the trade. That part was always going to be yours.

Now run this checklist on me.

I wrote it, which means I do not get to be exempt from it. Every question in here has an answer on my site — how my strategies are tuned, when they are designed to fail, what the data requirements are, and what you get besides the code. If something is missing, ask me directly and I will answer it.

turnkeyfuturesstrategies.com

— Crystal Curry, TurnKey Futures Strategies