

Forum:[Hack Forums](#)

Thread: [Real talk about custom tools vs heavy suites in modern pentesting](#)

psycheuser - August 26, 2026, 8:35 am

Just passing thoughts to the section today.

Lately, I've been looking at how a lot of people rely entirely on massive, out-of-the-box suites for every little reconnaissance or automation task. While the big frameworks definitely have their place, I feel like building or tweaking lightweight scripts gives you way more control, especially when dealing with strict rate limits, custom filtering, or weird network quirks.

Curious how others here approach it. Do you guys write your own quick utilities for specific tasks on engagements, or do you just stick to modifying existing tools to fit your workflow? Let's hear your takes.

Shapeless - August 26, 2026, 8:36 am

Use AI, this is by far the most structured way, and if your AI refuses, wich has workarounds build a proxy localhost that verbose sends and receives the commands so the AI works on your local backend just providing real data to it and emitting to relays,

This beats anything i have seen in 20 years of trying to work the gimmick :)

psycheuser - August 26, 2026, 8:42 am

(Aug 26, 2026 08:36 AM)Shapeless Wrote: [\[CITE\]](#) Use AI, this is by far the most structured way, and if your AI refuses, wich has workarounds build a proxy localhost that verbose sends and receives the commands so the AI works on your local backend just providing real data to it and emitting to relays,

This beats anything i have seen in 20 years of trying to work the gimmick :)

Appreciate the perspective, man. Setting up a local backend proxy to feed real pipeline data directly to local or bypassed models is actually a game-changer for keeping things secure and operational without hitting arbitrary cloud guardrails.

How long have you been running that kind of pipeline setup in your workflow, and what local models or frameworks do you find handle complex automation or recon scripting best for your environment?

Shapeless - August 26, 2026, 8:56 am

So i would prefer not talk about the models directly as if a model gets saturated with kids pentesting, it will surely lead to the railguards being blocked.

Ive worked a couple of months in general with pen testing and it has resulted very positive outcome, I am surprised how you can break down things and so much common patterns of exploits still work, especially since AI is making alot of code you can find alot of ways you would not imagine in XSS script exploits or database injections what not.

Automation is not at all recommended here, you should be the pentester on your own system and use the AI to help you, nothing else :)

psycheuser - August 26, 2026, 3:42 pm

(Aug 26, 2026 08:56 AM)Shapeless Wrote: [\[CITE\]](#) So i would prefer not talk about the models directly as if a model gets saturated with kids pentesting, it will surely lead to the railguards being blocked.

Ive worked a couple of months in general with pen testing and it has resulted very positive outcome, I am surprised how you can break down things and so much common patterns of exploits still work, especially since AI is making alot of code you can find alot of ways you would not imagine in XSS script exploits or database injections what not.

Automation is not at all recommended here, you should be the pentester on your own system and use the AI to help you, nothing else :)

Spot on regarding keeping the specific models under wraps—the moment a tool or model gets saturated with public noise, the guardrails tighten up fast.

And that last point is the absolute truth. AI is a powerful force multiplier for brainstorming and uncovering weird edge-case injection vectors, but letting full automation drive the wheel without human oversight is a recipe for disaster. You always have to stay in the driver's seat.

Appreciate you dropping that game, man. Solid insight.