Introducing: LangSmith Sandboxes (Now in Private Preview)

3 MIN · YOUTUBE · HTTPS://WWW.YOUTUBE.COM/WATCH?V=HBCIDNOFU6A
https://www.youtube.com/watch?v=HBcIDnOFU6A

SUMMARY

LangChain has introduced sandboxes, secure environments for agents to execute code safely without risking infrastructure. These sandboxes allow agents to run applications, analyze data, and interact with APIs while maintaining security against untrusted code.

- *LangChain sandboxes provide secure, ephemeral environments for code execution by agents.*
- *Agents can run entirely within sandboxes, defined by templates that set configurations and proxy rules.*
- *Proxy rules prevent prompt injection attacks by allowing API communication without exposing secrets.*
- *Agents have full root access, enabling them to run Docker and other applications as if on a personal computer.*
- *Sandboxes can be used to execute code, generate HTML, take screenshots, and handle various tasks like analyzing PDFs and processing videos.*
- *LangSmith deployments can automatically manage sandboxes, enhancing integration and functionality.*
- *Users interested in these capabilities are encouraged to sign up for more information.*

Hey everyone, I'm Will Kukulcan from LangChain. Today I'm excited to show you LangChain sandboxes. Over the last couple of months, we've seen code execution emerge as one of the most powerful primitives for agents. Agents can run code, they can analyze data, call APIs, and validate their own output to build applications from scratch. The challenge is that agent-generated code is untrusted and unpredictable. You can't just let it run on your infrastructure. That's why we built LangChain sandboxes, secure ephemeral environments where your agents can execute code. Let me show you how it works. We've seen two main ways that people use sandboxes. The first one which is running the agent completely inside of the sandbox. To start, every sandbox is defined by a template. A template defines a base set of configurations that you'll use to provision the rest of your sandboxes.

Let's create one that has some requirements for running the Deep Agent CLI. I'll list the Deep Agent CLI demo. I'll use a base image that has some of the Deep Agent CLI base configurations. I'll bump up our default requests. And also add a few proxy rules. Proxy rules allow you to talk to APIs and things like that without actually mounting secrets on your containers. This is important to avoid nasty prompt injection attacks. So I'll start by adding a few proxy rules to talk to OpenAI and the LangChain SDK API.

Now that we've turned template, we can go ahead and create a box. So take a second or two to spin up. And you can see that we have a full box with full root access. Now I'll actually try running the Deep Agent CLI. You'll see that this fails. the reason for this is that we don't actually have an OpenAI API key on the box. We're relying on our proxy. I'll set a foo API key.

And then we're off. Because we have full root access, D agents can even access things like Docker, pretty much anything that you would want to run on a VM. And so we'll try it and start running Docker. You'll see that this also works because we're able to inject the open AI header into all requests. And well, as every D agent CLI could spin up a Docker container, I can even curl Oh, we have Engine X running and you can pretty do pretty much anything else you'd want to do on your personal laptop.

Your agents have access to their own personal computer. The other main way we see agents use sandboxes by using the sandboxes as a tool to execute code. Here, I've set up a sample web app using LangSmith deployments. All LangSmith deployments are automatically able to spin up, interact with, and delete sandboxes using their innate LangSmith access. I'm going to hop over to Studio so I can actually interact with this deployment. I'm going to task my application with generating some HTML, taking a screenshot using a headless browser, and then sending it back to us here.

Here's a fancy prompt that I have set up already. We can see that it's asking it to spin up a bespoke image. And if we actually go to the sandboxes tab, you can see that we'll try to spin up a sandbox. And you can see that it's actually already spun up. If I go back, it's going to be executing. This might take a few seconds. And there you have it. Our application was successfully able to spin up a sandbox, render HTML, and then send us a screenshot using the prompt that we gave it.

You can do many other things like run user inputted scripts, analyze PDFs, process videos, and much, much more. Again, do anything that your agent could do with its own personal computer. If any of these use cases interest you, please sign up using the sign up form attached to this blog.