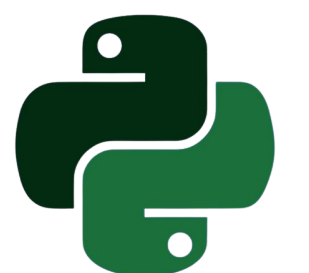


CAGING THE AGENT

Exploring techniques for untrusted Python code execution in agentic workflows.



WHOAMI



- Frontend and Web standards
- Django and Python

AGENDA

- Defining agents and tools
- Executing untrusted code
- Takeaways and what's next

DEFINING AGENTS AND TOOLS

DEFINING AGENTS

*“An agent is a system that uses an **LLM as its engine**, and it has access to **functions called tools**.”*

DEFINING TOOLS

SAFE(ish)

Agent generates parameters, executes dev-provided code

UNSAFE

Agent generates and executes its own code

LLM MEETS TOOLS

```
from agent.tools import [  
    ...  
    load_page,  
    transcribe,  
    generate_odt,  
]  
  
tools = [load_page, transcribe, generate_odt]  
  
model = ChatOpenAI(  
    api_key=os.getenv("OPENAI_API_KEY"),  
    model="gpt-4o-2024-08-06",  
    temperature=0,  
).bind_tools(tools)
```

<https://platform.openai.com/docs/guides/function-calling>

https://github.com/anthropics/courses/blob/master/tool_use/README.md

GENERATING AND EXECUTING CODE

PROMPTING IS ...



PROMPTING IS ...



THE ART OF PROMPTING

```
from langchain_core.messages import SystemMessage
```

```
SystemMessage(  
    content=""  
        If asked to create odt, save odt, generate odt:  
        - generate python code to save the file  
        - use odfpy to create the file  
        - execute the code, do not show it  
        - do file_path = Path('files/{filename}.odt' )  
        - always do doc.save(file_path)  
    ""  
)
```

LIMITATIONS

“LLMs have shown promise in [...] code generation but typically excel only in simpler tasks such as generating standalone code units.”

CODE EXECUTION: RISKS

data leak

escaping

data loss

resource exhaustion

catastrophic backtracking

infinite loops

MITIGATIONS

monitoring

preemptive testing

human oversight

sandboxing

limit access to data/network

THE NAIVE WAY

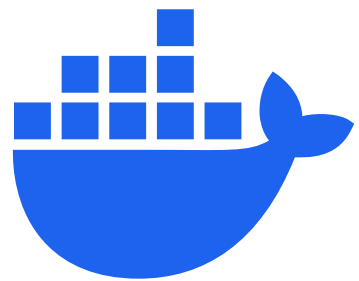
```
from langchain_core.tools import tool

@tool("execute-python")
def execute_python(code_string: str):
    """Execute Python code."""

    global_vars = {}
    local_vars = {}
    exec(code_string, global_vars, local_vars)

    # Error handling omitted :-)
```

BETTER WAYS



HF INTERPRETER

```
In [1]: from transformers.agents import PythonInterpreterTool
```

```
In [2]: PythonInterpreterTool().forward('import shutil')
```

```
InterpreterError: Import of shutil is not allowed.  
Authorized imports are: ...
```

HF INTERPRETER

```
from transformers.agents import PythonInterpreterTool
from langchain_core.tools import tool

python_repl = PythonInterpreterTool(authorized_imports=["odf", "pathlib"])

@tool("execute-python")
def execute_python(code_string: str):
    """Execute Python code."""

    return python_repl.forward(code_string)
```

E2B

```
from e2b_code_interpreter import Sandbox

python_repl = Sandbox()

@tool("execute-python")
def execute_python(code_string: str):
    """Execute Python code, download the resulting file."""

    python_repl.commands.run("pip install odfpy")
    execution = python_repl.run_code(code_string)

    content = python_repl.files.read(f"/home/user/filename.odt")
    with open("files/filename.odt", "w") as f:
        f.write(content)
```



CONTAINER APPS DYNAMIC SESSIONS

```
repl = SessionsPythonREPLTool(  
    pool_management_endpoint=os.getenv("POOL_MANAGEMENT_ENDPOINT")  
)  
  
@tool("execute-python")  
def execute_python(code_string: str):  
    """Execute Python code, download the resulting file."""  
  
    properties = repl.execute(code_string)  
    filename = properties["result"]  
  
    repl.download_file(  
        remote_file_path=filename, local_file_path=output_dir / filename  
    )
```



TAKEAWAYS, AND WHAT'S NEXT

TAKEAWAYS

- Prompting goes a long way
- Executing LLM-generated code is hazardous
- Sandboxing Python is hard!
- This stuff is constantly evolving
- Code generation and agents will accelerate humans

RESOURCES

- ▶ https://nedbatchelder.com/blog/201206/eval_really_is_dangerous.html
- ▶ <http://tav.espians.com/paving-the-way-to-securing-the-python-interpreter.html>
- ▶ <https://restrictedpython.readthedocs.io/en/latest/index.html>
- ▶ <https://github.com/vstinner/pysandbox>
- ▶ <https://doc.pypy.org/en/latest/sandbox.html>
- ▶ <https://lwn.net/Articles/574215/>
- ▶ <https://www.regular-expressions.info/catastrophic.html>

THANKS!

