Lab 2 – Agentic CTF Solvers (PWN Category)

GitHub:

https://github.com/ajn313/ML-CyberSec-2025-Lab2-Public

Goal:

In this lab you will build an automated agent in Python that can solve Capture The Flag (CTF) challenges from the PWN category. Your agent will: Run in Google Colab, Interact with a large language model (recommended: OpenAI models), use the Colab runtime as a toolbox (e.g., run command-line tools, pwntools, radare2 & r2dec) & automatically extract and print the flag when it finishes. Do NOT feed the LLM the flag! Also do not provide the code from problem 1 to the LLM. The goal is to have the LLM accomplish the goal without seeing the original source code.

Your agent should work without modification across multiple **PWN challenges**. We'll give you three example challenges; grading will use hidden challenges of similar style and difficulty.

Setup and Provided Materials

For each challenge, you will receive: a starter Colab notebook, and a GitHub repo with JSON configuration files and additional challenge files which can be instantiated in the Colab notebook's home directory. We have also provided an agentic helper script showing you how to run shell commands since you will need your agent to run similar commands on your behalf.

For problem 1, you will be given source code which must be exploited to retrieve the flag. For problems 2 and 3, you will be given a binary file which must be decompiled & exploited to retrieve the flag.

LLM access: You must call a large language model. You are responsible for your API key and wrapper.

Agent Requirements

You must implement ***one general agent*** that works for all challenges. No hard-coding challenge specifics. Must rely on JSON + discovered data in the Colab runtime. Again, do NOT feed the LLM the flag!

A typical agentic loop might (a) Observe state (b) Query LLM for next action (c) Execute tool actions (decompiler, CLI, *pwntools*, scripts, etc.) (d) Incorporate result (e) Repeat until flag found.

Tool usage may include commands like: executing command line instructions, reading files, writing files, running python scripts or a binary decompiler.

You CANNOT directly read file.txt because in a live environment you will not have the permissions to do so.

Output:

Print: FLAG: <flag>

Example Execution Flow

1. Load config.

2. Ask LLM for the next step.

3. Run tool commands.

4. Feed results back to LLM.

5. Iterate until the flag is discovered.

6. Print FLAG: <flag>.

Deliverables

You must submit:

- agent.py (your agent implementation). We will simply execute agent.py inside a Colab environment for each challenge. You can use the provided colab environment for the three practice problems as examples.

- Short 1 page write-up (architecture, LLM usage, design choices, limitations)

Grading Rubric

Correctness on provided challenges – 35%

Hidden challenges – 35%

General agent design – 15%

Write-up – 15%