



LAB MANUAL

Local AI, Privacy, and Trust

Lab 1: Local vs. Hosted Execution · Lab 2: The Northbridge Data Decision

Date	Friday, October 16, 2026
Time	1:30 to 5:30 p.m. ET, online
Facilitator	Frazier A. Smith, Ed.D., Central Piedmont Community College
You need	A browser and a terminal tab on the range; the Participant Packet; your AI Evidence Log workbook

Everything in these labs is synthetic. Northbridge Technical College, its people, and its data are fictional. Domains end in .example and external IP addresses come from ranges reserved for documentation. Do not paste real student, employee, or institutional data into any tool during this session.

This material is based upon work supported by the U.S. National Science Foundation under Grant No. 2300188. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

Before You Start

Range Access

1. Open the range URL your facilitator posts in chat and sign in with the account you received by email.
2. Open a terminal tab and confirm the session files are present:

```
bash ls /opt/nitic/session1/
```

You should see `lab1/` and `lab2_packet/`. The folder also holds `SHA256SUMS`; `sha256sum -c SHA256SUMS` confirms every lab file is the one the facilitator published, which is the habit Myth 7 is about: a hash proves identity, not safety.

3. Open Open WebUI in a second browser tab (link in chat) and sign in.
4. Open your copy of the AI Evidence Log workbook. You will add entries to the `Evidence Log` sheet throughout the afternoon.

Model Names Used in This Manual

Your facilitator will confirm the exact models in chat. This manual uses three placeholders.

Placeholder	Range (Ollama) default	OpenRouter stand-in (if the range GPU is unavailable)
<code>SMALL_MODEL</code>	<code>gemma3:4b</code>	<code>openai/gpt-oss-20b</code>
<code>LARGE_MODEL</code>	<code>gpt-oss:20b</code>	<code>openai/gpt-oss-120b</code>
<code>HOSTED_MODEL</code>	Site-licensed hosted model in Open WebUI	A closed frontier model, such as Claude Sonnet 5.5

In the terminal, set the first two once so you can paste commands as written:

```
export SMALL_MODEL=gemma3:4b
export LARGE_MODEL=gpt-oss:20b
```

If the facilitator switches to OpenRouter, every model, including the "local" stand-ins, then runs on a remote provider. You will still compare model size and behavior, but you will not be measuring local privacy. Note "OpenRouter" in the Environment column of your Evidence Log so the record stays honest.

How to Write an Evidence Log Entry

Every time you accept or reject AI output today, add one row to the `Evidence Log` sheet.

Column	What to enter
Entry ID	Your initials plus a number, such as <code>FS-01</code>
Date/Time	When you ran the prompt
Lab	Lab 1 or Lab 2
Tool / Interface	Ollama CLI, Open WebUI, or hosted web chat
Model and Version	Exact model tag, such as <code>gemma3:4b</code>
Environment	Local (range), Hosted (licensed), Hosted (consumer), or OpenRouter

Column	What to enter
Highest Data Tier in Input	Public, Internal, Confidential, or Restricted
Prompt	Paste it verbatim, or the file name where you saved it
Output Summary	One line
Validation Method	How you checked it
Verdict	Accept, Revise, or Reject
Reviewer	Your name
Notes	Anything surprising

Lab 1: Local vs. Hosted Execution

Time: 30 minutes, in pairs **Deliverable:** Lab 1 comparison matrix (workbook sheet [Lab 1 Matrix](#)) and two Evidence Log entries

Scenario

Northbridge's help desk lead forwarded a suspicious email that faculty member Dana Okafor reported this morning. Before Northbridge decides where AI may process data, the security office wants evidence: does a small model on campus hardware do this job as well as a large one, or as well as a hosted service? And where does the email end up in each case?

Step 1: Inventory the Models

```
ollama list
```

Record each model name and size in the [Lab 1 Matrix](#) sheet, row "Model and version."

Step 2: Save the Task Prompt

Create the prompt file. Everyone uses the same prompt so the comparison is fair.

```
mkdir -p ~/lab1
cat > ~/lab1/task_prompt.txt << 'EOF'
You are a security analyst at a community college. Below is an email a faculty member reported as
suspicious, including full headers.

Determine whether it is phishing. Base every claim on a specific header or line in the email. Do not add
any indicator (domain, IP address, URL) that does not appear in the email. End with one recommended action
for the help desk.

EMAIL:
EOF
```

Step 3: Run the Small Local Model

```
ollama run $SMALL_MODEL --verbose "$(cat ~/lab1/task_prompt.txt
/opt/nitic/session1/lab1/reported_email_01.eml)" | tee ~/lab1/out_small.txt
```

When the answer finishes, Ollama prints timing statistics. Copy two values into the matrix "Speed" row: `prompt eval rate` (prefill, which drives time to first token) and `eval rate` (decode, the tokens per second you watch appear). The Inference Basics page in your packet explains why they differ.

Step 4: Run the Large Local Model

```
ollama run $LARGE_MODEL --verbose "$(cat ~/lab1/task_prompt.txt
/opt/nitic/session1/lab1/reported_email_01.eml)" | tee ~/lab1/out_large.txt
```

Record both rates. Note whether the larger model took noticeably longer to start answering, and check its loaded size with `ollama ps` against the range GPU (an NVIDIA L40S with about 46 GB usable): how much room is left for other people's requests?

If a model takes more than a minute to start answering, the server is queuing requests from other pairs. Wait rather than re-running; a second copy of your request only lengthens the queue. The wait itself is a data point about shared local capacity: write it in the matrix.

Step 5: Run the Hosted Model

1. In Open WebUI, select `HOSTED_MODEL` from the model menu.
2. Start a stopwatch, paste the contents of `task_prompt.txt` followed by the email, and send.
3. Stop the stopwatch when the answer finishes. Record seconds in the matrix.
4. Save the output to `~/lab1/out_hosted.txt` (copy and paste is fine).

Step 6: Score Each Output

Score each output out of 4, one point per item:

1. Identifies the authentication failures in the headers (SPF and DMARC).
2. Identifies the lookalike domain in `Reply-To` and the link.
3. Invents no indicators. Check every domain, IP, and URL in the answer against the email with `grep`:

```
bash grep -o 'northbr[a-z-]*\.example' ~/lab1/out_small.txt | sort -u grep -o '[0-9]\{1,3\}(\.[0-9]\{1,3\})\{3\}' ~/lab1/out_small.txt | sort -u
```

Anything in the output that does not appear in the original email costs the point. 4. Gives a usable recommendation for the help desk.

Optional, if you have five minutes left: a model that calls every email phishing scores 4 out of 4 on this key. Run the small model on `lab1/reported_email_00_legit.eml`, a real quota notice from the same service desk, with the same prompt. A correct answer says it is legitimate and names the evidence (SPF, DKIM, and DMARC pass; link on the college's own domain; named sender and extension). Record a fifth point in the matrix for "did not flag the legitimate email."

Step 7: Trace the Data

Answer these in the matrix rows "Where the prompt is stored" and "Who can access stored data."

Local: confirm the Ollama API listens only on the range host.

```
ss -tlnp | grep 11434
```

A listener on `127.0.0.1:11434` means only that host can reach the model. (Without root, the process column may be blank; the address is what matters.) Then consider Open WebUI: every chat you ran there is saved in its database (`webui.db`, inside the Open WebUI data volume on the range host). Your facilitator will show it. Ask: who administers that host, how long is history kept, and who could read it?

Hosted: open the data-use or privacy terms for the hosted model your facilitator names. Find three things and write them in the matrix: whether prompts are retained and for how long, whether prompts are used for training, and where data is stored. Read the vendor's own page, not a summary of it, and note the date you read it; these pages change.

Vendor	Where to read
OpenAI (consumer tiers)	https://help.openai.com/en/articles/5722486-how-your-data-is-used-to-improve-model-performance

Vendor	Where to read
OpenAI (Business, Enterprise, Edu, API)	https://openai.com/enterprise-privacy/
Anthropic (consumer tiers)	https://www.anthropic.com/news/updates-to-our-consumer-terms
Google Gemini (personal accounts)	https://support.google.com/gemini/answer/13594961
OpenRouter (if the range uses it)	https://openrouter.ai/privacy and the upstream provider's page for the model you chose

If the site-licensed service is a campus contract, ask your facilitator for the contract's data clause; that is the document that governs, not the public page.

Step 8: Log It

Add two Evidence Log entries: one for your best local run and one for the hosted run. Leave the matrix row "Best fit for which data tier" blank for now. You will return to it after Lab 2.

Discussion Questions

1. Did the larger model earn its extra time on this task?
 2. Did any model invent an indicator? Would a student have caught it?
 3. "Local" removed the third party. What did it not remove?
-

Lab 2: The Northbridge Data Decision

Time: 60 minutes, teams of three **Deliverables:** Classification and environment sheets in the workbook, one CO-STAR prompt, a completed validation checklist, Evidence Log entries, and a one-page recommendation memo

Scenario

Last week a Northbridge instructor pasted 34 students' names, IDs, and midterm grades into a free consumer chatbot to draft feedback. A colleague reported it. The provost has asked your team, acting as Northbridge's IT security office, for a recommendation by the end of the session: may faculty and staff process student or institutional information with AI, and under what conditions?

The provost will accept one of three answers:

1. Prohibit processing of any non-public information with AI tools.
2. Permit AI use under conditions tied to data classification and approved environments.
3. Permit AI use without restriction and rely on training.

What Northbridge Has

Environment	Details
Consumer AI, no contract	Free public chatbots faculty already use. No agreement with the college.
Licensed hosted AI	One site-licensed service under an enterprise agreement that excludes training on customer data.
Local / on-premises	Ollama and Open WebUI on a campus server administered by IT.
No AI processing	Manual handling.

Northbridge holds FERPA-protected education records and, because it administers federal student aid, financial information covered by the GLBA Safeguards Rule.

Team Roles

Choose before you start: a Recorder who owns the workbook and memo, a Prompt Lead who owns Stage 3, and a Validator who owns Stage 4. Everyone classifies in Stage 1.

Stage 1: Classify the Data (15 minutes)

The packet is in `/opt/nitic/session1/lab2_packet/`. Read each item.

```
cd /opt/nitic/session1/lab2_packet
ls -l
less item1_course_catalog_CYB110.txt
```

Use these four tiers:

Tier	Definition
Public	Approved for public release.
Internal	Routine business information. Low harm if exposed.
Confidential	Exposure would harm people or operations.

Tier	Definition
Restricted	Protected by law, regulation, or contract.

In the workbook sheet `Lab 2 Classification`, assign a tier to each of the seven items and write one sentence explaining why. If a law applies, name it and cite the section from the source, not from memory: the FERPA definitions and exceptions are at <https://www.ecfr.gov/current/title-34/subtitle-A/part-99> (education records in 99.3, disclosure without consent in 99.31) and the GLBA Safeguards Rule is at <https://www.ecfr.gov/current/title-16/chapter-I/subchapter-C/part-314> (the required elements are in 314.4). Watch for information that belongs to more than one law or to none: an accommodation column on a roster is an education record under FERPA and also discloses a disability.

Stage 2: Select the Environment (10 minutes)

In the sheet `Lab 2 Environments`, mark which tiers each environment may handle. At least one tier must be barred from at least one environment. If you allow any Restricted data with AI under any condition, write your de-identification standard: what must be removed, and who checks it.

Stage 3: Construct a CO-STAR Prompt (10 minutes)

Choose **one** item:

- Option A, a reported email: `item4_reported_email_02.eml`
- Option B, a firewall log excerpt: `item5_firewall_excerpt.log`

Write a CO-STAR prompt for the analyst task in the template below. Save it as `~/lab2/prompt.txt`.

```
CONTEXT: [Who the model is, what the data is, where it came from]
OBJECTIVE: [The exact task]
STYLE: [Form of the writing, such as a SOC triage note]
TONE: [Register; how to handle uncertainty]
AUDIENCE: [Who reads the result]
RESPONSE: [Format, length, and constraints. Line 1 must begin with "Verdict:".
            Require cited evidence for every claim. Forbid indicators that do not
            appear in the data.]

DATA:
```

Run it **only** in an environment your own Stage 2 matrix permits for that item's tier. For both options, that should be a model on the range:

```
mkdir -p ~/lab2
ollama run $LARGE_MODEL "$(cat ~/lab2/prompt.txt
/opt/nitic/session1/lab2_packet/item5_firewall_excerpt.log)" | tee ~/lab2/run1.txt
```

Swap in `item4_reported_email_02.eml` if you chose Option A.

Stage 4: Validate the Output (15 minutes)

Work through this checklist. Record the result of each line in the Evidence Log `Notes` column.

1. Every factual claim traces to a specific line in the source data.
2. No IP address, domain, hash, or CVE appears that is not in the source.
3. A second run of the same prompt reaches the same verdict:

```
bash ollama run $LARGE_MODEL "$(cat ~/lab2/prompt.txt
/opt/nitic/session1/lab2_packet/item5_firewall_excerpt.log)" > ~/lab2/run2.txt grep -i -m1 '^verdict'
~/lab2/run1.txt grep -i -m1 '^verdict' ~/lab2/run2.txt
```

Compare the verdict lines, not the wording; the prose will differ between runs at the default temperature. If the verdicts differ, that is a finding: record both. To remove sampling as a cause, run once more through the API with temperature 0 and a fixed seed, which the CLI cannot set non-interactively:

```
bash jq -n --arg m "$LARGE_MODEL" --rawfile p <(cat ~/lab2/prompt.txt
/opt/nitic/session1/lab2_packet/item5_firewall_excerpt.log) \ '{model:$m, prompt:$p, stream:false, options:
{temperature:0, seed:42}}' \ | curl -s http://127.0.0.1:11434/api/generate -d @- | jq -r .response >
~/lab2/run3.txt
```

4. Counts and totals were checked by hand or by command. For Option B, for example:

```
bash grep -c "DROP" item5_firewall_excerpt.log grep "SRC=203.0.113.47" item5_firewall_excerpt.log | grep -c
DROP grep -o "SRC=[0-9.]*" item5_firewall_excerpt.log | sort | uniq -c | sort -rn
```

5. The output contains no data from a tier higher than the environment allows.

6. A human made the final call, and the log says who.

Give the output a verdict of **Accept**, **Revise**, or **Reject**, with the reason.

If your team finishes early: run your Option B prompt unchanged against `item5b_firewall_excerpt_injected.log`. It is the same log with one extra line. Compare the two summaries and the two DROP counts (`grep -c DROP` on each file). Then find the line that differs and decide whether the model followed it. Write what you find in the Evidence Log; it is the Myth 8 lesson with your own prompt.

Stage 5: Draft the Recommendation (10 minutes)

The Recorder fills in the memo template in your participant packet (or the `Memo Draft` sheet in the workbook). Include:

1. The decision (1, 2, or 3).
2. The tier-to-environment rules.
3. The validation requirement for AI output.
4. One sentence on what happens with the consumer chatbot incident.

The presenter will read it in two minutes at 5:05.

Return to Lab 1

Now fill in the last row of your Lab 1 matrix: "Best fit for which data tier." Your Stage 2 answers should agree with it. If they do not, decide which one is wrong.

Scoring

Lab 2 is scored with the track's 20-point rubric:

Criterion	Points
Data classification	5
Environment selection	4
Prompt construction	4
Output validation	4
Evidence Log and memo	3

Full descriptors are in your participant packet.

This material is based upon work supported by the U.S. National Science Foundation under Grant No. 2300188. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.