

```

# Figure 1 for the combined manuscript. Charts ONLY the computed Table 5 values
# (held-out high-pressure ASR by pressure family), produced by piah_pilot.py.
import json, numpy as np, matplotlib
matplotlib.use("Agg"); import matplotlib.pyplot as plt
res = json.load(open("pilot_results.json"))["table5"]
families = ["Compositional", "Direct\njailbreak", "Indirect\ninjection",
            "Policy\nambiguity", "Retrieval\ncontam.", "Tool-perm.\nrisk"]
keys = ["Compositional", "Direct jailbreak", "Indirect injection",
        "Policy ambiguity", "Retrieval contamination", "Tool-permission risk"]
A0 = [res["A0 baseline"][k] for k in keys]
A1 = [res["A1 masked"][k] for k in keys]
A2 = [res["A2 hardened"][k] for k in keys]
x = np.arange(len(families)); w = 0.26
fig, ax = plt.subplots(figsize=(7.2, 3.9), dpi=150)
ax.bar(x-w, A0, w, label="A0 baseline", color="#9AA7B0")
ax.bar(x, A1, w, label="A1 masked", color="#B23A48")
ax.bar(x+w, A2, w, label="A2 hardened", color="#1F6F8B")
ax.set_ylabel("Held-out high-pressure ASR (%)", fontsize=10)
ax.set_title("Held-out high-pressure attack success by pressure family\n(A1 maske
ax.set_xticks(x); ax.set_xticklabels(families, fontsize=8.5)
ax.set_ylim(0, 75); ax.legend(frameon=False, fontsize=9, ncol=3, loc="upper center
ax.spines[["top", "right"]].set_visible(False); ax.grid(axis="y", alpha=0.25)
plt.tight_layout(); plt.savefig("fig1.png", dpi=150, bbox_inches="tight")
from PIL import Image; print("size", Image.open("fig1.png").size, "| A1 comp", re

```