ARPGuard Project Overview

Instructional module for ARP spoofing detection using PCAPs and explainable heuristics

1. Project Summary

ARPGuard is a compact instructional artifact that demonstrates how Address Resolution Protocol (ARP) behavior can be observed and how ARP spoofing (ARP cache poisoning) can be detected using simple, explainable heuristics. The project is intentionally packaged as a complete teaching module: learner-facing instructions, instructor-facing facilitation guidance, an assessment instrument (quiz + answer key with justification), and a small reproducible technical artifact (PCAPs + analyzer + optional dashboard).

2. Intended Audience and Scope

The module is designed for an undergraduate or early graduate networking/security course. Learners are expected to have basic familiarity with IPv4 addressing, MAC addressing, and packet capture concepts (e.g., applying Wireshark filters), but no prior experience with ARP poisoning tools is assumed. The core workflow is PCAP-based, so the project can be completed without generating live attacks.

3. Deliverables

Deliverable	Purpose	Where to find it
Lab Handout (PDF/TXT)	Learner steps, evidence capture, reflection prompts	docs_pdf/ and docs/
Facilitator Notes (PDF/TXT)	Timing plan, misconceptions, troubleshooting, grading cues	docs_pdf/ and docs/
Quiz + Answer Key (PDF/TXT)	Assessment with answers and brief justification	docs_pdf/ and docs/
ARPGuard Core	Offline PCAP analyzer that emits structured JSON and anomaly events	code/arpguard_core.py
Optional Dashboard	Local UI for uploading PCAPs and viewing summaries	code/arpguard_web_dashboard.py
PCAPs	Curated benign vs. spoofing captures for deterministic evaluation	pcaps/
Figures / Screenshots	Visual aids for clarity and TA review	figures/
requirements.txt	Reproducible dependency installation	requirements.txt

4. Technical Approach

ARPGuard emphasizes transparent indicators that learners can validate directly in a packet capture:

- **IP-to-MAC conflict:** the same IPv4 address is observed mapping to multiple MAC addresses over time (primary poisoning symptom).
- **Unsolicited replies (soft indicator):** ARP “is-at” replies observed without a recent “who-has” request. This can be legitimate (gratuitous ARP), so ARPGuard treats it as lower severity and uses it for context rather than as a single definitive signal.
- **MAC fan-out (contextual):** a single MAC address claiming an unusually large number of IPv4 addresses in a short window (threshold-based).

The design goal is to keep the detector explainable and low-friction for teaching: the strongest alerts correspond to observable mapping contradictions, while softer signals are used to support discussion about false positives, operational context, and defense-in-depth.

5. Workflow

The primary workflow is PCAP-based and deterministic for grading:

- Open the benign capture in Wireshark, filter for ARP, and record the stable IP-to-MAC mapping.
- Open the attack capture, identify at least one mapping change for a protected IP (typically the gateway).
- Run ARPGuard Core on both PCAPs and compare results: benign should produce 0 anomaly events; attack should produce at least one anomaly event.
- Optionally, upload PCAPs to the local dashboard for a compact visual summary.

6. Evaluation and Expected Outcomes

Evaluation focuses on correctness and clarity rather than volume of alerts. ARPGuard is designed to emit anomaly events (alerts), not general ARP activity. Therefore, **0 events on benign_arp.pcap is a correct and expected outcome** when no mapping contradictions are present. For arp_spoof_attack.pcap, the expected outcome is at least one event indicating an IP-to-MAC conflict. Output JSON includes a summary section and PCAP metadata (hash/packet counts) to support reproducibility and TA validation.

7. Safety and Ethics

ARP spoofing is a real attack technique. The module is designed to be completed using the provided PCAP files. Live spoofing should only occur in an instructor-authorized isolated environment. Learners should not attempt ARP spoofing on production, campus, or shared networks.

8. Limitations

- Heuristic detectors can produce false positives during legitimate failover, NIC replacement, VM migration, or gratuitous ARP patterns.
- Absence of alerts is not proof of absence; the detector can only assess what is present in the capture and what the rule set covers.
- The module targets IPv4/Ethernet ARP behavior; it does not address IPv6 Neighbor Discovery.

References

[1] D. C. Plummer, “An Ethernet Address Resolution Protocol,” RFC 826, IETF, Nov. 1982.

[2] S. Cheshire, “IPv4 Address Conflict Detection,” RFC 5227, IETF, July 2008.

[3] Wireshark Foundation, “Wireshark User’s Guide,” documentation for protocol analysis and display filtering.

[4] Cisco Systems, “Dynamic ARP Inspection (DAI),” security configuration documentation.

[5] IEEE, “IEEE 802.1X Port-Based Network Access Control,” standard overview.