

# Answers

Reference answers for Lesson 03 lab and homework.

## Part 1 — Lab Answers

### Exercise 1 — tcpdump Triage

---

#### Expected findings

- A protocol breakdown dominated by TCP + DNS with small amounts of ICMP.
- At least one non-standard high port connection to an external IP — the candidate C2.
- An ICMP volume anomaly or ARP pattern if the capture includes tunneling.
- One flow flagged as "most suspicious" because of its combination of destination, port, and periodicity.

### Exercise 2 — DNS Exfiltration Hunting

---

#### Expected findings

- A single second-level domain receiving many long-subdomain queries at high rate.
- Subdomain labels that decode (base32 or base64) into readable text — often shell output or file chunks.
- TXT or NULL records being requested, a strong indicator of tunneling.

- One internal host responsible for most of the queries — that host is compromised.

## Exercise 3 — HTTP Session Reconstruction

---

### Expected findings

- At least one cleartext credential in Basic auth, form POST, or query string.
- A beacon: same client POSTing the same URL on a regular interval with payload of similar shape.
- A User-Agent that signals tooling (curl, python, a C2 framework string).
- A large outbound POST indicating staged data exfil.

## Exercise 4 — NetFlow Beacon Detection

---

### Expected findings

- One (src, dst, port) tuple with <30% jitter and >10 flows — the beacon.
- Asymmetric data-ratio flows suggesting exfil (small request, large response or vice versa).
- A long-lived TCP session hinting at an interactive reverse shell.

## Final Exercise — Attack Story

---

### Expected chain

Initial access via HTTP (credential from Exercise 3) → DNS tunnel for C2 (Exercise 2) → NetFlow beacon confirms periodicity (Exercise 4) → tcpdump confirms

protocol mix (Exercise 1). A good synthesis connects the same compromised internal IP across three or four sources.

---

## Part 2 — Homework Answers

### Task 1 — Beacon Hunting Script

---

#### Expected shape

A script of ~60 lines using `collections.defaultdict(list)` to group, `statistics` for stddev, and jitter computed as `stddev/mean * 100`. The main output should flag a single tuple that matches the lab's C2 finding.

### Task 2 — DNS Exfil Decoder

---

#### Expected shape

Extract the leftmost label(s) per query, concatenate in arrival order, attempt `base64.b32decode` then `base64.b64decode` with padding repair, print whichever yields >80% printable ASCII. A good solution handles case-sensitivity and missing padding gracefully.

### Task 3 — Egress Filter Recommendation

---

#### Expected rules

- Deny outbound DNS to any resolver except the corporate resolver.
- Deny outbound to destinations newer than 30 days (newly registered domains).

- Deny outbound on non-standard high ports (4444, 8080 on non-proxy hosts, 31337).
- Rate-limit DNS queries per client (caps high-rate tunneling).
- Deny outbound TLS with mismatched SNI / cert CN.

Each rule should list realistic false-positive scenarios (e.g., "breaks captive portal login for visitor wifi").