Experiment 3: Challenge–Response Authentication

Experiment 3: Challenge–Response Authentication and Replay Attack Prevention

This document summarizes the authentication demonstrations implemented in:

1) auth-demo-phases.py

- Multi-phase authentication demo
- Shows challenge–response, replay attacks, timestamps, nonce tracking, and mutual authentication

2) Expt3.py

- Simple challenge–response script
- Includes nonce tracking to prevent replay

PHASE SUMMARY (auth-demo-phases.py)

Phase 1: Basic challenge–response

- Server generates a nonce
- Client hashes nonce + shared secret
- Server verifies hash

Phase 2: Replay attack demonstration

- Shows how fixed messages lead to authentication bypass

Phase 3: Timestamp-based protection

- Client hashes nonce + timestamp + secret
- Server accepts only within time window
- Replay after timestamp expiry fails

Phase 4: Nonce-based freshness

- Server tracks used nonces
- Reusing a nonce triggers replay detection

Phase 5: Mutual authentication

- Server authenticates client
- Client authenticates server
- Uses role-tagged hashing to prevent reflection attacks

Expt3.py Summary

- Server generates challenge + timestamp
- Client computes SHA256(challenge + secret)

- Server verifies and checks replay using used_nonces set
- Demonstrates replay attack after delay

How To Run

```
python auth-demo-phases.py
python Expt3.py
```

Project Structure:

```
Experiment-3/
■■■ auth-demo-phases.py
■■■ Expt3.py
■■■ README_Exp3.pdf
```