Problem Class 3 (Security Engineering 26/27)

Network Security

1. “De-perimeterization is great. I don’t have to worry anymore about malicious or out-of-date devices on my network anymore.” To what extent is this statement true?
2. Which threat actors and attack techniques are most/least impacted by zero trust network access?
3. Why does spoofing an email address require the social engineering to succeed right away? How might attackers get around this limitation and conduct back-and-forth social engineering?
4. How comfortable would you be logging into your bank account on a public airport WiFi? Would a VPN offer you additional protection in this scenario?
5. What kinds of web attacks would be made more difficult by domain registars enforcing Know Your Customer whenever a new domain is registered? What kinds of web attacks would likely increase as a result?

Detection & Response

1. A detection system flags 99% of attacks and wrongly flags 1% of benign logins. Only 1 in 10,000 logins is malicious. What fraction of alerts are real attacks, and what does this mean for a SOC?
2. “Our threat intel provider flags 50,000 malicious IP addresses a day, so we automatically block them all.” Give three reasons this could be a bad policy, and say what evidence would tell you whether it is.
3. Allowlisting is the most effective blocking control and the least popular. Why? Would you recommend it for a 50-person firm, a hospital, and a bank, and how would your answer differ?
4. After a breach, a firm routes its forensic investigation through its lawyers so the findings are privileged. Who benefits and who loses? What happens to the “learn and improve” stage of incident response?
5. Automated response can isolate a laptop or revoke a session token in seconds, but a human analyst takes hours. When should a firm let the tool act without approval? Give one example where automation is clearly right and one where it is clearly wrong.
6. “Backups mean we don’t need to worry about ransomware.” Discuss.

Assurance & Compliance

1. Why do military and safety critical software buyers maintain requirements for high-assurance, whereas B2C software builders have dropped this requirement? What is lost as a result?
2. How can formal methods as a route to assurance be reconciled with a focus on user testing?
3. What is the relationship between prices in a bug bounty program and assurance over software security? What complicates this?
4. Discuss the defensibility of the four patch management strategies introduced in the lecture for a CISO facing allegations they were negligent.
5. If a vendor goes bankrupt, what are the security benefits and risks associated with "open-sourcing" their code bases?
6. Name three traditional security assumptions that are no longer true due to advances in AI systems and why.
7. How can a defender avoid being misled by forum shopping when it comes to assessing their vendors' security posture?