- **Stages of Social Engineering:**
  - Prepare.
  - Establish trust.
  - Use persuasion tactics.
  - Disconnect from the target.
- **Preventing Social Engineering:**
  - Implementing managerial controls.
  - Staying informed of trends.
  - Sharing your knowledge with others.
- **Common Social Engineering Attacks:**
  - Baiting: tempts people into compromising their security.
  - Quid pro quo: trick someone into believing that they'll be rewarded in return for sharing access, information, or money
  - Phishing: the use of digital communications to trick people into revealing sensitive data or deploying malicious software.
  - Tailgating (piggybacking): unauthorized people follow an authorized person into a restricted area.
  - Watering hole: a threat actor compromises a website frequently visited by a specific group of users.
- Potentially Unwanted Application (PUA): type of unwanted software that is bundled in with legitimate programs which might display ads, cause device slowdown, or install other software.

- Phishing kit: A collection of software tools needed to launch a phishing campaign.
- **Phishing kit tools:**
  - Malicious attachments.
  - Fake data-collection forms.
  - Fraudulent web links.
- **Phishing security measures:**
  - Anti-phishing policies.
  - Employee training resources.
  - Email filters.
  - Intrusion prevention systems.

- Malware: software designed to harm devices or networks.
- **Types of malware:**
  - Virus: malicious code written to interfere with computer operations and cause damage to data and software.
  - Worm: can duplicate and spread itself across systems on its own.
  - Trojan: looks like a legitimate file or program.
  - Adware: used to display digital advertisements in applications.
  - Spyware: used to gather and sell information without consent
  - Scareware: employs tactics to frighten users into infecting their own device.
  - Fileless Malware: uses legitimate programs that are already installed to infect a computer.
  - Rootkit: provides remote, administrative access to a computer. Spread by a combination of two components: a dropper and a loader:
    - Dropper: a type of malware that comes packed with malicious code which is delivered and installed onto a target system.
    - Loader: a type of malware that downloads strains of malicious code from an external source and installs them onto a target system.
  - Botnet (Robot Network): a collection of computers infected by malware that are under the control of a single threat actor, known as the "bot-herder."
  - Ransomware: threat actors encrypt an organization's data and demand payment to restore access.
  - Crypto-jacking: installs software to illegally mine cryptocurrencies.

- **Signs of Crypto-jacking:**
  - Slowdown.
  - Increased CPU usage.
  - Sudden system crashes.
  - Fast draining battery.
  - Unusually high electricity costs.
- Web-based Exploits: malicious code or behavior that's used to take advantage of coding flaws in a web application.
- Cross-Site Scripting (XSS): an injection attack that insets code into a vulnerable website or web application.
- **Types of XSS attacks:**
  - Reflected: when malicious script is sent to a server and activated during the server's response.
  - Stored: when malicious script is injected directly on the server.
  - DOM-based: when a malicious script exists in the webpage a browser loads.
- SQL Injection: an attack that executes unexpected queries on a database.
- **SQL Injection categories:**
  - In-band: uses the same communication channel to launch the attack and gather the results.
  - Out-of-band: uses a different communication channel to launch the attack and gather the results.
  - Inferential: when an attacker is unable to directly see the results of their attack. Instead, they can interpret the results by analyzing the behavior of the system.

- **Injection prevention ways:**
  - Prepared Statement: a coding technique that executes SQL statements before passing them onto the database.
  - Input sanitization: programming that removes user input which could be interpreted as code.
  - Input validation: programming that ensures user input meets a system's expectations.
- Threat Modeling: the process of identifying assets, their vulnerabilities, and how each is exposed to threats.
- **Threat model steps:**
  - Define the scope.
  - Identify threats.
  - Characterize the environment.
  - Analyze threats.
  - Mitigate risks.
  - Evaluate findings.
- PASTA (Process for Attack Simulation and Threat Analysis): a popular threat modeling framework that's used across many industries.
- **PASTA Stages:**
  - Define business and security objectives.
  - Define the technical scope.
  - Decompose the application.
  - Perform a threat analysis.
  - Perform a vulnerability analysis.
  - Conduct attack modeling.
  - Analyze risk and impact.