You cannot build a secure system until you understand your threats

1) Threat model: a security-based analysis that helps determine the (highest level) security risks posed to an application and how attacks can manifest themselves.
2) Threat models provide structure in terms of security to the design process.
3) Threat models help formalize the way you think about security as related to an application.
4) Most important skill in threat modeling: the ability to correctly question any and all assumptions about the data.
5) A secure system cannot be built without threat evaluation.
6) Benefits of threat modeling
    a) You will better understand your app
    b) You will find bugs
    c) New team members will get up to speed faster
    d) Other teams that work on your app will be better able to understand it
    e) Testers will be able to better test your app
7) Basic threat modeling process
    a) Assemble a (threat modeling) team: should include those with some security background
    b) Decompose the app: DFDs (identify processes, data stores, interactors, data flows), Activity diagrams and other UML components.
8) Determine threats
    a) Use FAIR model to categorize threats (https://www.fairinstitute.org/)
        1. Spoofing: attacker poses as another (trusted) user or rogue server poses as a trusted one (examples include DNS cache poisoning).
        2. Tampering with data: malicious modification of the data.
        3. Repudiation: user denies performing some action (like an online purchase) – non-repudiation is the ability to prove the transaction occurred.
        4. Information Disclosure: data that should be hidden/protected is viewed in the open.
        5. Denial of Service: app does not respond because it has crashed or is overloaded with requests.
        6. Elevation of privilege: allows for access to restricted data and files – if an attacker "gets root" on your machine, you are "owned." Some threats interrelate
        7. Elevation of privilege is probably the worst.