Threat Modeling

This material is licensed under CC BY-NC 4.0. See final slide for details.

What is Threat Modeling?

A process for identifying security vulnerabilities and deciding how to address them.

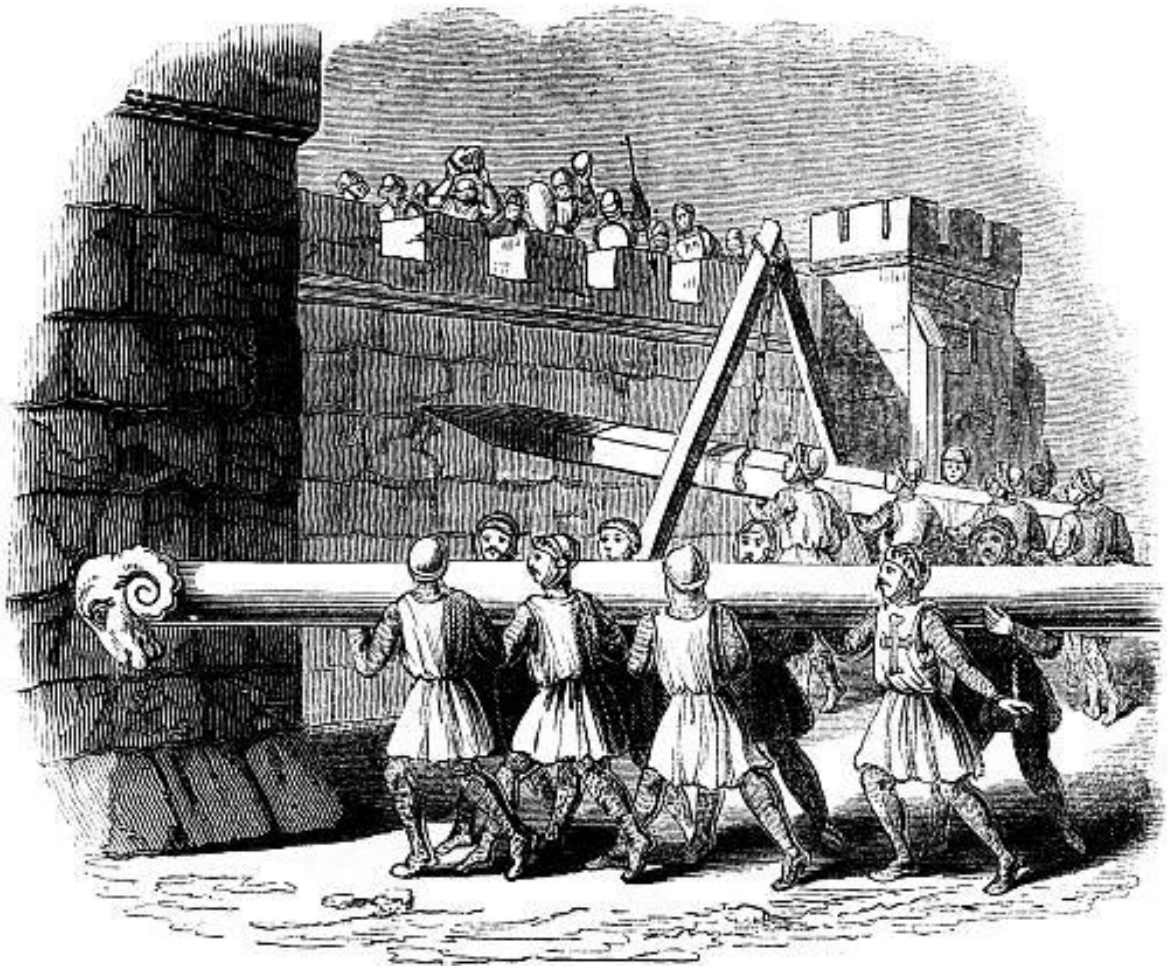
Why Do Threat Modeling?

- Find and address threats to your product
- Approach security in a structured, systematic way
- Create a reliable, repeatable process

What is the Process?

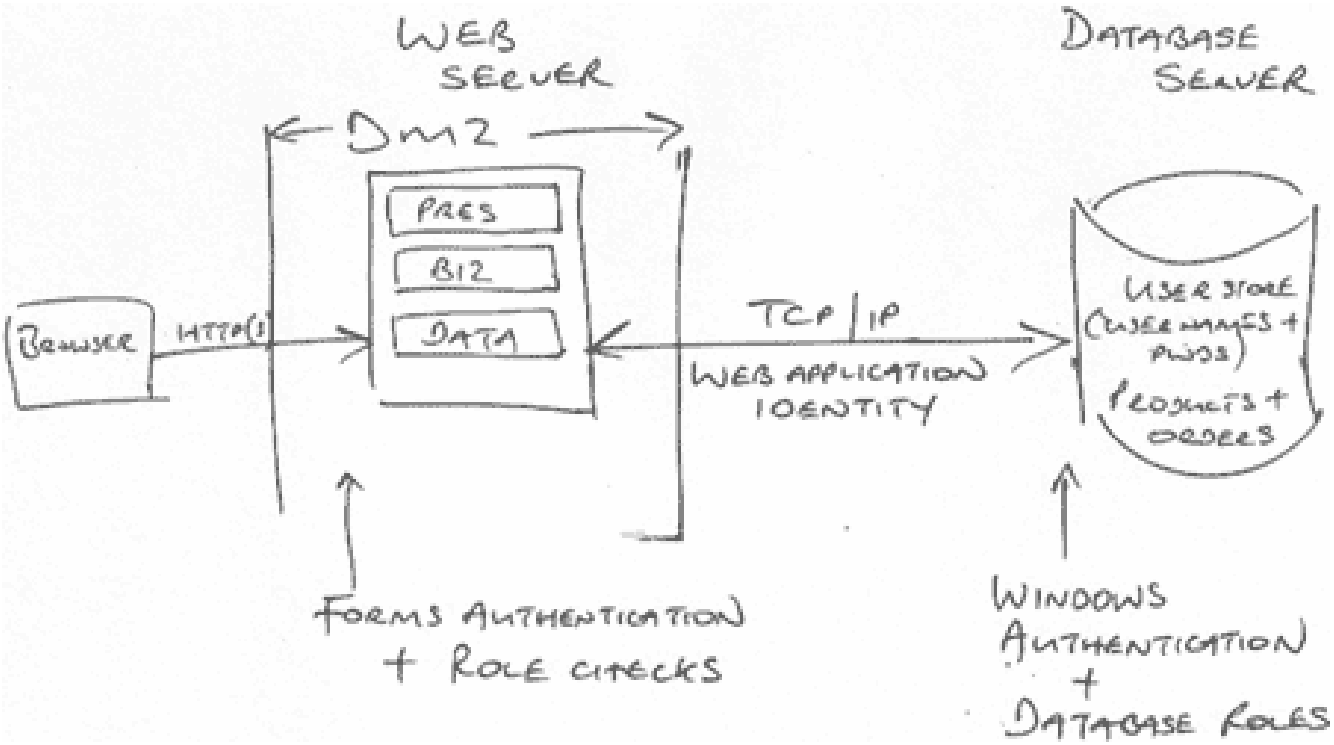
1. Describe the attack surface.
2. Identify threats.
3. Identify countermeasures.
4. Assess risk.
5. Decide how to handle each threat.

1. Describing the Attack Surface

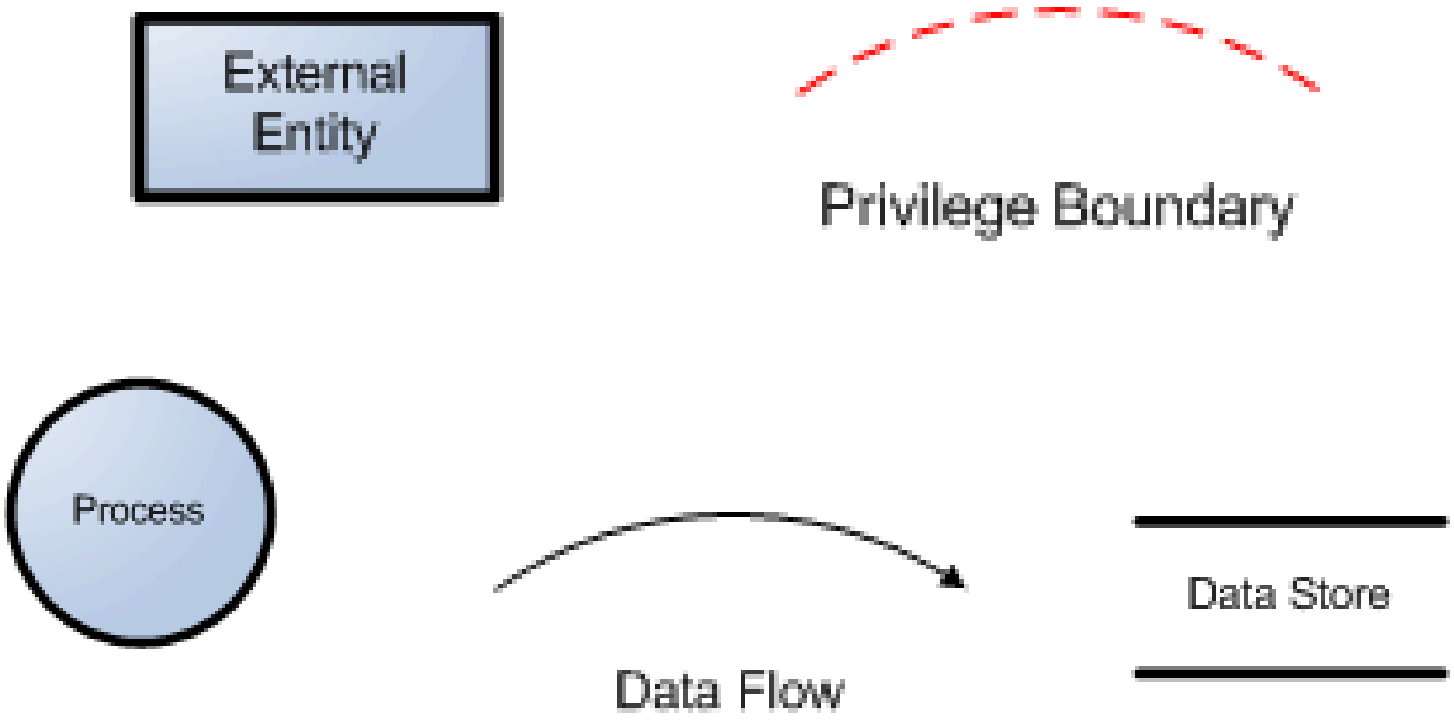


1. Describing the Attack Surface

- Start on a white board. Create a diagram.
- Show agents, data flow, and trust boundaries



Common Objects in DFDs



1. Describing the Attack Surface

- Data doesn't magically appear.

  Show who creates it.

- Data isn't stored for no reason.

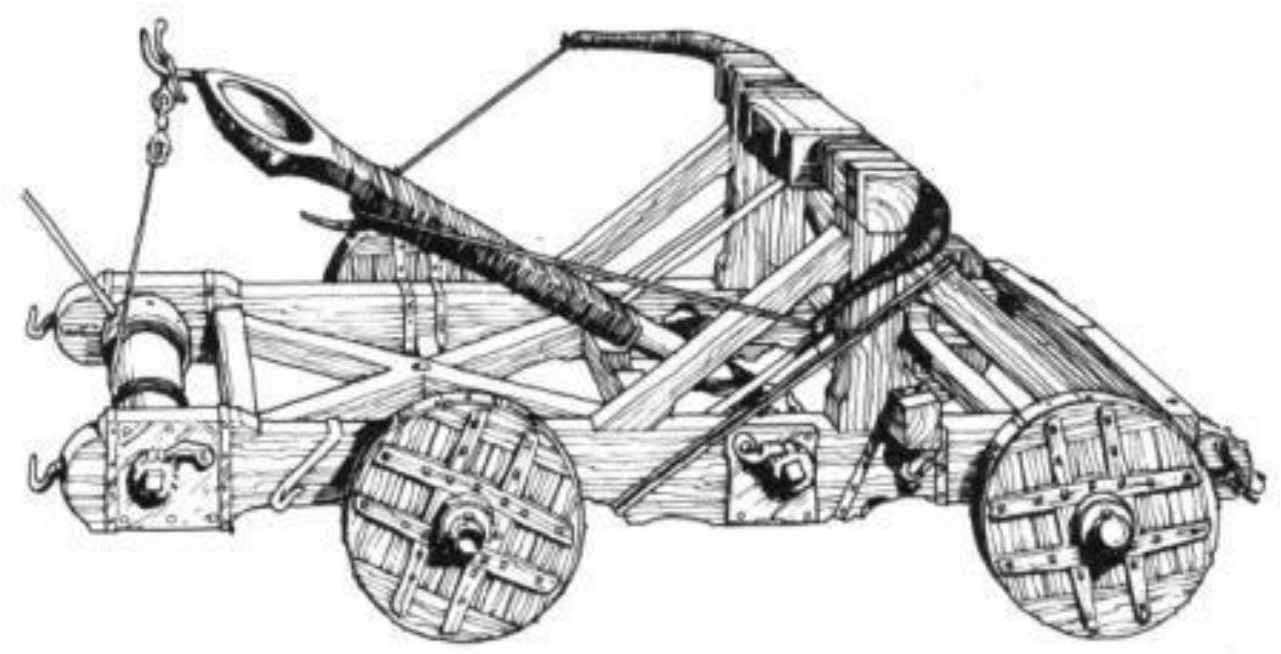
  Show who reads it.

- Data doesn't simply flow from one place to another.

  Show what moves it.

- DFDs are not flow charts, class diagrams, or call graphs.

2. Identifying Threats



2. Identifying Threats: STRIDE

S poofing identity

T ampering with data

R epudiation

I nformation disclosure

D enial of service

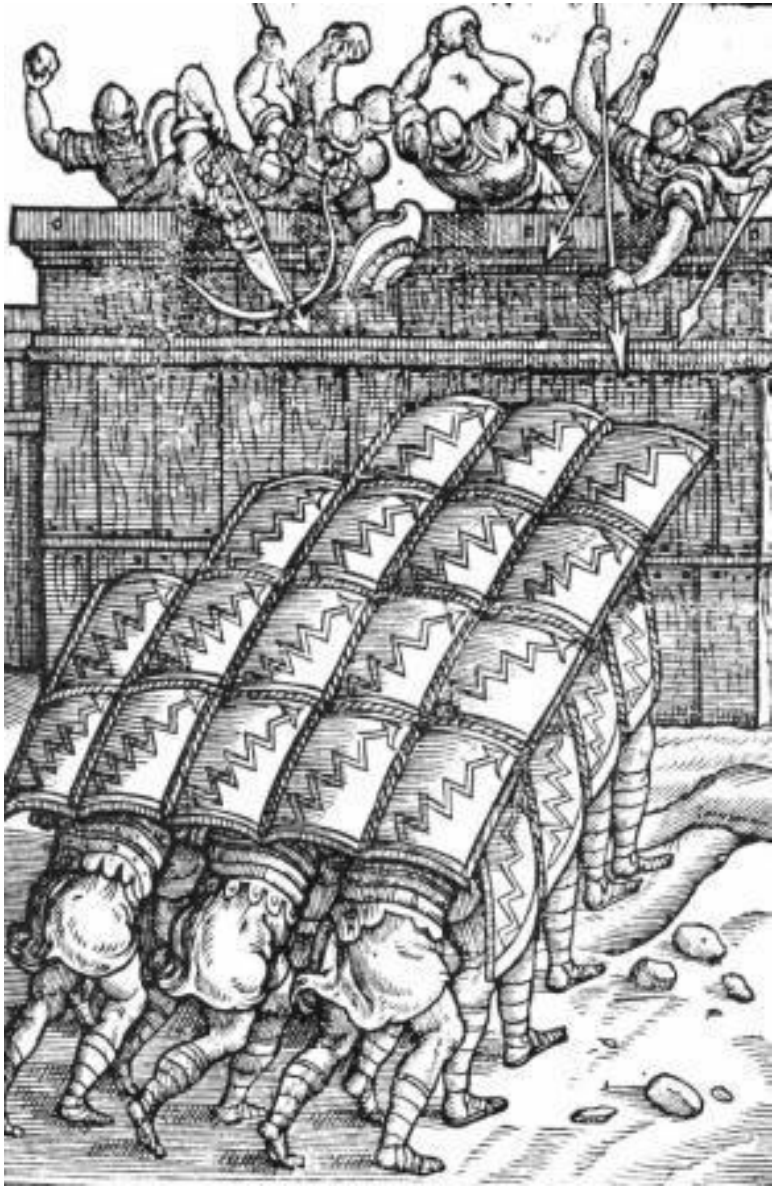
E levation of privilege

2. Identifying Threats: Examples

- Attacker may be able to read other user's messages
- User may not have logged off on a shared PC
- Data validation may allow SQL injection
- Authorization may fail, allowing unauthorized access
- Browser cache may contain contents of message
- Attacker may sniff packets and read network traffic

For more see OWASP Top 10.

3. Identifying Countermeasures



3. Identifying Countermeasures

Spoofing	Use strong authentication. Do not store secrets in plaintext. Do not pass credentials in plaintext over the wire. Protect authentication cookies with SSL.
Tampering	Data hashing and signing. Digital signatures. Strong authorization. Tamper-resistant protocols.
Repudiation	Secure audit trails. Digital signatures.
Information Disclosure	Strong authorization. Strong encryption. Protocols that provide message confidentiality. Do not store secrets in plaintext.
Denial of Service	Use resource and bandwidth throttling techniques. Validate and filter input.
Elevation of Privilege	Follow the principle of least privilege and use least privileged service accounts to run processes and access resources.

3. Identifying Countermeasures

For authentication threats:

- Credentials and authentication tokens are protected with encryption in storage and transit
- Protocols are resistant to brute force, dictionary, and replay attacks
- Strong password policies are enforced
- Trusted server authentication is used instead of SQL authentication
- Passwords are stored with salted hashes
- Password resets do not reveal password hints and valid usernames
- Account lockouts do not result in a denial of service attack

4. Assessing Risk

4. Assessing Risk: DREAD

D iscoverability

R eproducibility

E xploitability

A ffected users

D amage potential

5. Deciding What to Do

5. Deciding What to Do

5. Deciding What to Do

Choose a strategy for each threat:

- **Do nothing** (hope for the best)
- **Accept the risk** (evaluate business impact)
- **Warn of the risk** (give notices or training)
- **Mitigate the risk (**put countermeasures in place)
- **Terminate the risk** (turn off, omit, shut down)

Recommendations

- Start early.

  Threat modeling belongs in the planning phase.
- Identify owners.

  Architect? Dev Manager? Product Manager?
- Include:
  - product managers
  - testers
- Document risks discovered and counter-measures adopted.

License and Attribution

This material is licensed under the Creative Commons Attribution-NonCommercial 4.0 International (CC BY-NC 4.0) License.

Attribution: Please credit Brian Myers.

NonCommercial Use Only: Internal use permitted. Commercial use prohibited.

For full license terms, visit:
https://creativecommons.org/licenses/by-nc/4.0