Driving Secure Development Using a Threat Model

LESS WORK, MORE BENEFIT

2011-10-07

STACH&LIU

What is a Threat Model?

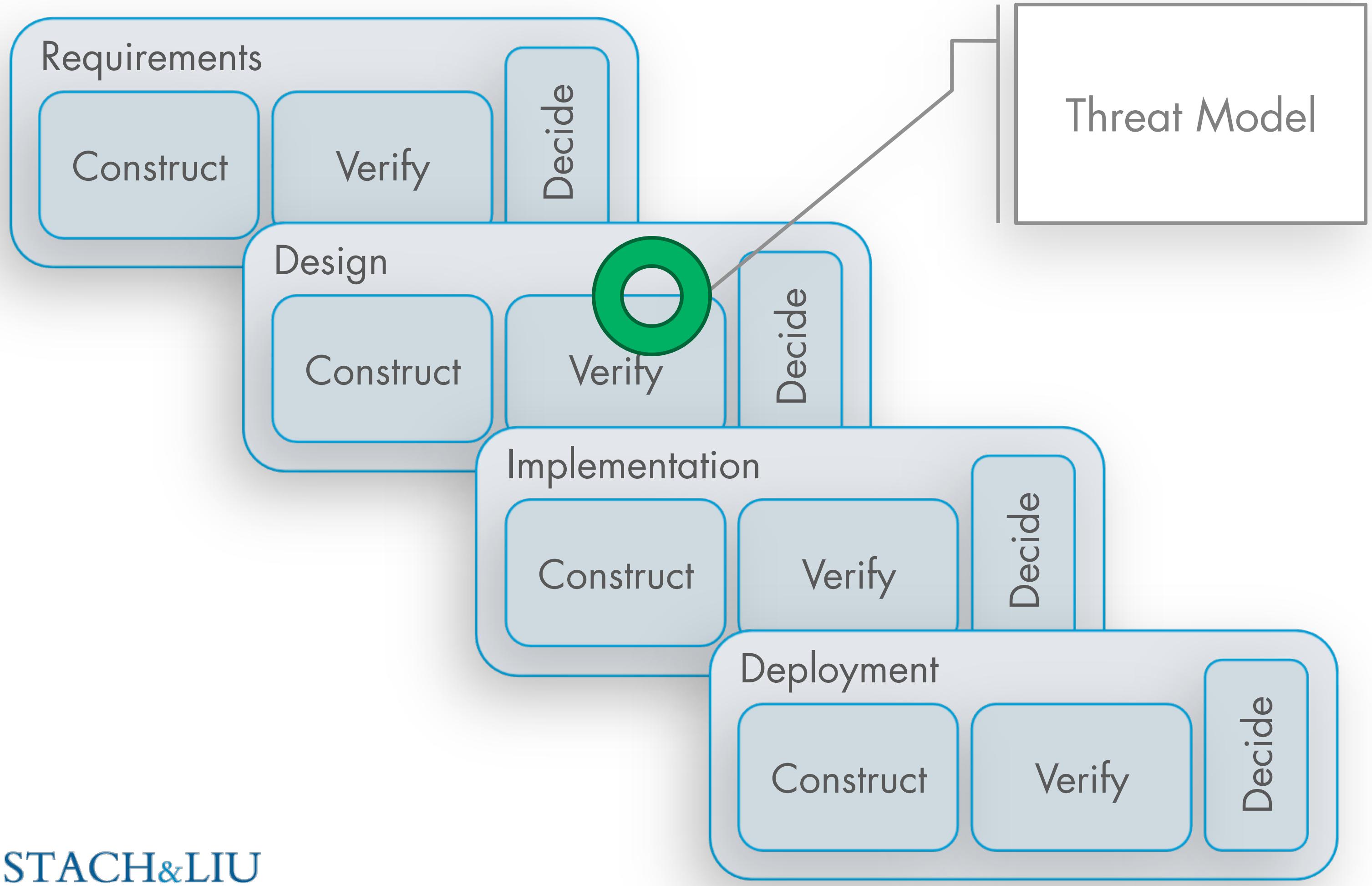
SECURITY CARTOGRAPHY

- A threat model is a model of:
  - Actions an attacker could [try to] take, using this system
  - System defenses against these threats

STACH&LIU

Threat Modeling in the SDL

HISTORICALLY



Threat Modeling in the SDL

HISTORICAL OUTCOMES

Very little changes

Worst Case

- Author has spent more time thinking about security
- Nobody reads it but the authors & reviewers
- Soon, it is out of date

Best Case

- Designers mitigate design flaws they noticed
- Someone updates it as the design changes
- Developers & QA read it

Model-Driven Development

DESIRED OUTCOMES

- Security activities pruned by what is actually needed
- System models contain exactly enough security to meet security objectives
- Development decisions based on accurate security information
- Threat model updated whenever system model is updated

Model-Driven Development

STRATEGY

- Start at requirements time
- Integrate as much as possible with existing system models
- Update continuously
- Consult the model when making decisions
  - Which design option
  - What activities to do (e.g. pen testing)
  - Specifics within activities (e.g. which test cases)

Inside a Threat Model

ANATOMY

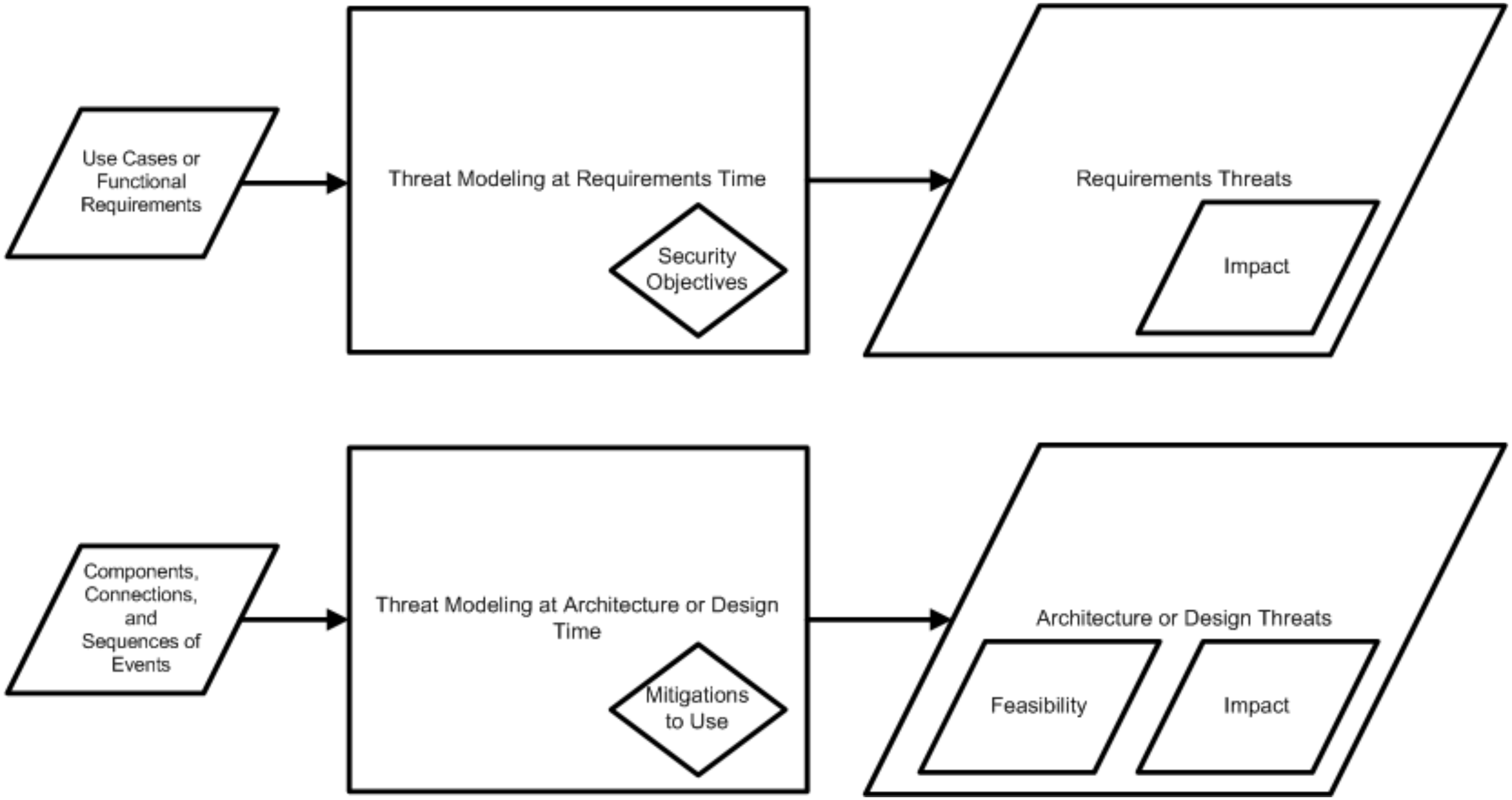
System Analysis

- Purpose of system
- High-level security goals
  - In-scope attackers
- Deployment environment
- System architecture
  - Static view
  - Dynamic view
  - Security attributes & technology

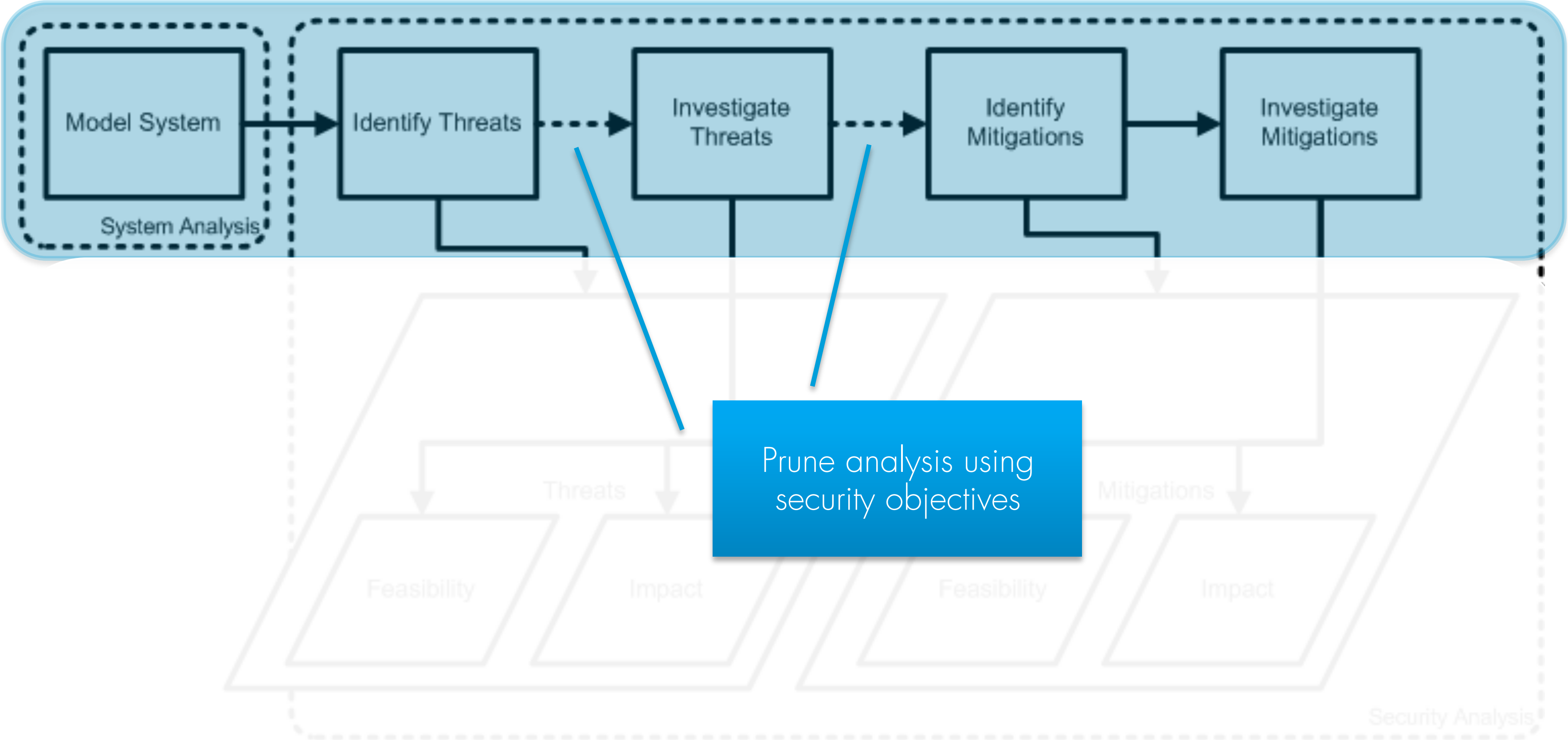
Security Analysis

- High-level threats
- Lower-level attacks
- Relationships between threats and attacks
- Impacts
- Feasibility of attacks
- Mitigations

Creation/Update Timing


Use Cases or Functional Requirements
Threat Modeling at Requirements Time
Security Objectives
Requirements Threats
Impact
Components, Connections, and Sequences of Events
Threat Modeling at Architecture or Design Time
Mitigations to Use
Architecture or Design Threats
Feasibility
Impact

Pruning Model Creation



Prune analysis using security objectives

Pruning Analysis Activities

BY SECURITY OBJECTIVES

- Many uses of risk assessment can be replaced by agreeing on security objectives up front
- In each following phase, investigate only topics noted from the preceding phase
- Would it help an attacker break the security objectives?
  - If not, it doesn't matter, don't investigate

Making Decisions

BY THREAT MODEL

- Identify a project decision that should be affected by security
    - E.g. Whether application is ready to launch
- Identify information that should inform that decision
    - E.g. Does the expense reports application meet its security objectives?
- Extract that information from the model
    - E.g. Examine threats that are still feasible for unbroken chains from attacker starting privileges to prohibited threats

Choosing Designs

BY THREAT MODEL

- Security objectives should be met
- Defenses should be protecting against threats
- Apply design patterns appropriately to respond to threats (e.g. input trust boundary, centralized input validation library)
- Best design has either fewer or easier threats to defend against

Security Tests

BY THREAT MODEL

- Confirm protections are in place
- Confirm responsibilities are met
- Try to perform all the relevant threats identified in the threat model
    - Start with those that are more beneficial to the attacker

Thanks

- Eleanor Saitta
- OWASP

- Bsides!