Understanding the Security Development Lifecycle - SDL

Dr. Jared DeMott
SECURITY RESEARCHER AND ENGINEER

@jareddemott

Overview

Raising security IQ

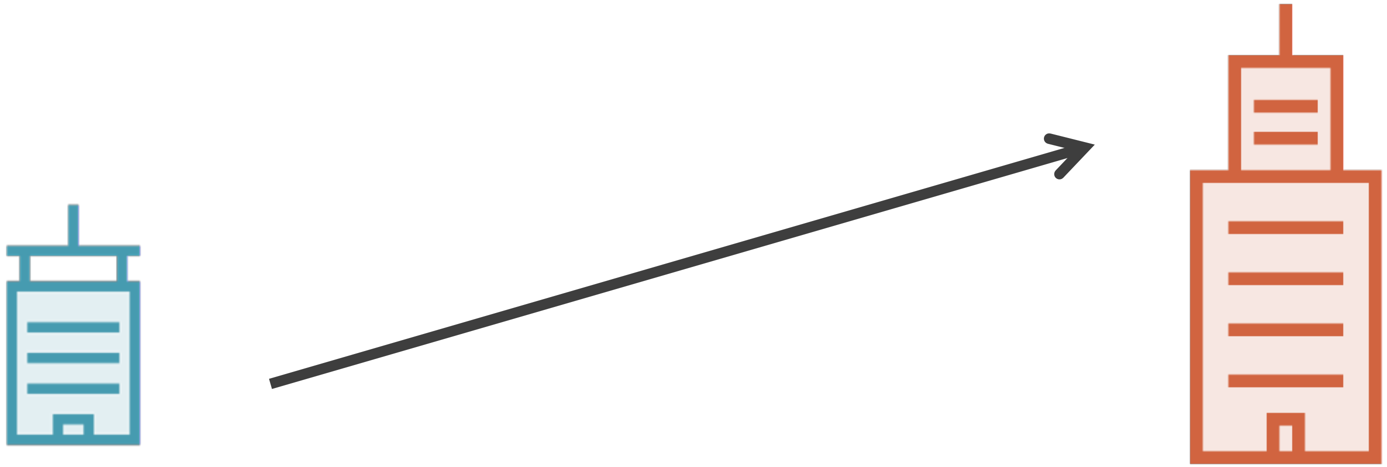
SDL

Push to the left

Attack surface reduction

Threat modeling

Executive Buy-In:
Security Is Everyone's Job



Security Engineers

**Embedded with
the development teams**

**A Central Security Team
Handles Training**

Training, IR, consulting, tools, etc.

Increase Security IQ

- Building Security Into the Culture
  - Development, test, and QA organization

Smart companies are doing:

- CBTs
- In person trainings
  - Internal and External trainers
    - Getting Devs, QA, and Testers together is key
    - Note: Devs prefer to see examples from their code

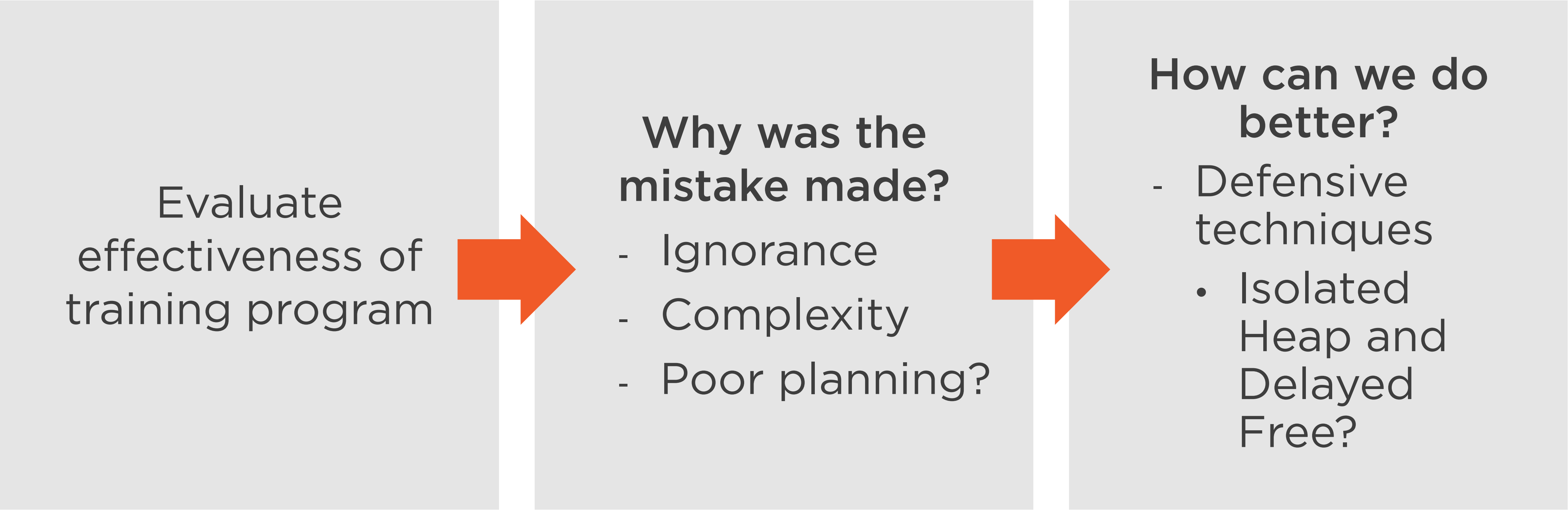
Project based opportunities

CTF

Belt Level

- Make these a part of promotion criteria
- Personnel File “Thank You”

Accountability



Evaluate effectiveness of training program

Why was the mistake made?

- Ignorance
- Complexity
- Poor planning?

How can we do better?

- Defensive techniques
  - Isolated Heap and Delayed Free?

Continuous Improvement

Microsoft’s SDL

BSIMM.com

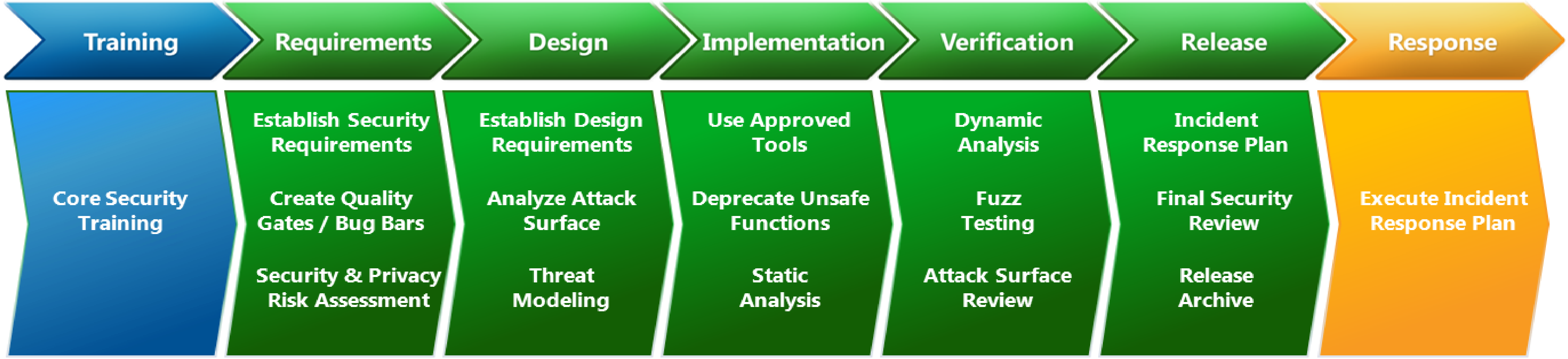
- 12 practices
  - 112 activities

SAFECode.org

Secure Development Lifecycle

New VS better than old –
E.g. CFG in 2015

Begin Fuzzing at
Beta time



Training	Requirements	Design	Implementation	Verification	Release	Response
Core Security Training	Establish Security Requirements Create Quality Gates / Bug Bars Security & Privacy Risk Assessment	Establish Design Requirements Analyze Attack Surface Threat Modeling	Use Approved Tools Deprecate Unsafe Functions Static Analysis	Dynamic Analysis Fuzz Testing Attack Surface Review	Incident Response Plan Final Security Review Release Archive	Execute Incident Response Plan

Src code checking
in build

Manual Review
and Pentest

Building Security in Maturity Model

Governance

Practices that help you organize, manage and measure your software security initiative including staff development.

- Strategy & Metrics
- Compliance & Policy
- Training

Intelligence

Practices that result in collections of knowledge to use to carry out software security activities throughout your organization.

- Attack Models
- Security Features & Design
- Standards & Requirements

SSDL Touchpoints

Common practices associated with analysis and assurance of particular software development artifacts and processes.

- Architecture Analysis
- Code Review
- Security Testing

Deployment

Practices that interface with traditional network security and software maintenance organizations.

- Penetration Testing
- Software Environment
- Config Management & Vulnerability Management

Push Security to the Left

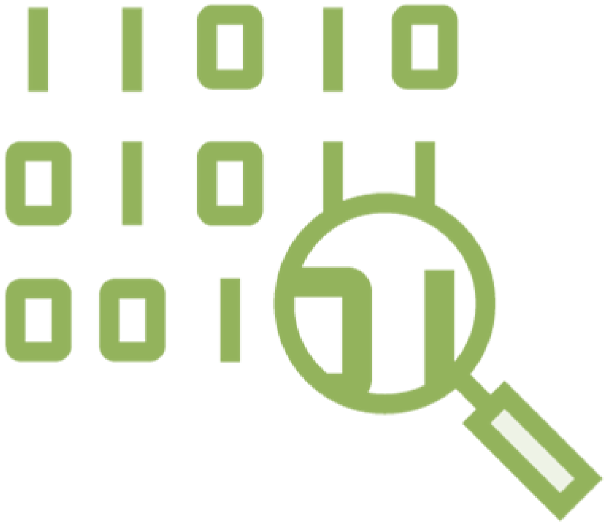
Before you code!

- How can we reduce the attack surface of this section of code?
- Safer design
- Safer development process

Attack Surface Reduction

Story about Google Chrome

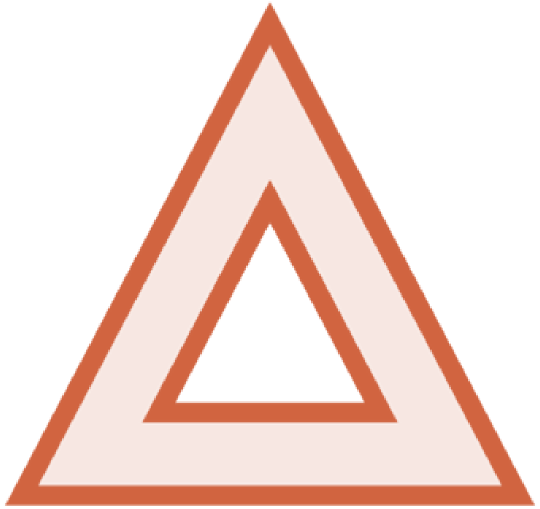
- Arbitrary clipboard formats are allowed
- In 2014, an obscure OLE format was used to escape a renderer (sandboxed tab)
    - Upon next right click, attacker code ran on the host



Attack Surface Reduction

Story about Google Chrome

- I wonder…
  - Did a software engineer ask his PM
    - “Should we limit lesser used formats? Users shouldn’t complain too much, and it would really reduce our attack surface.”
    - IF not, he should have
      - PM still might have overruled him, but at least he tried



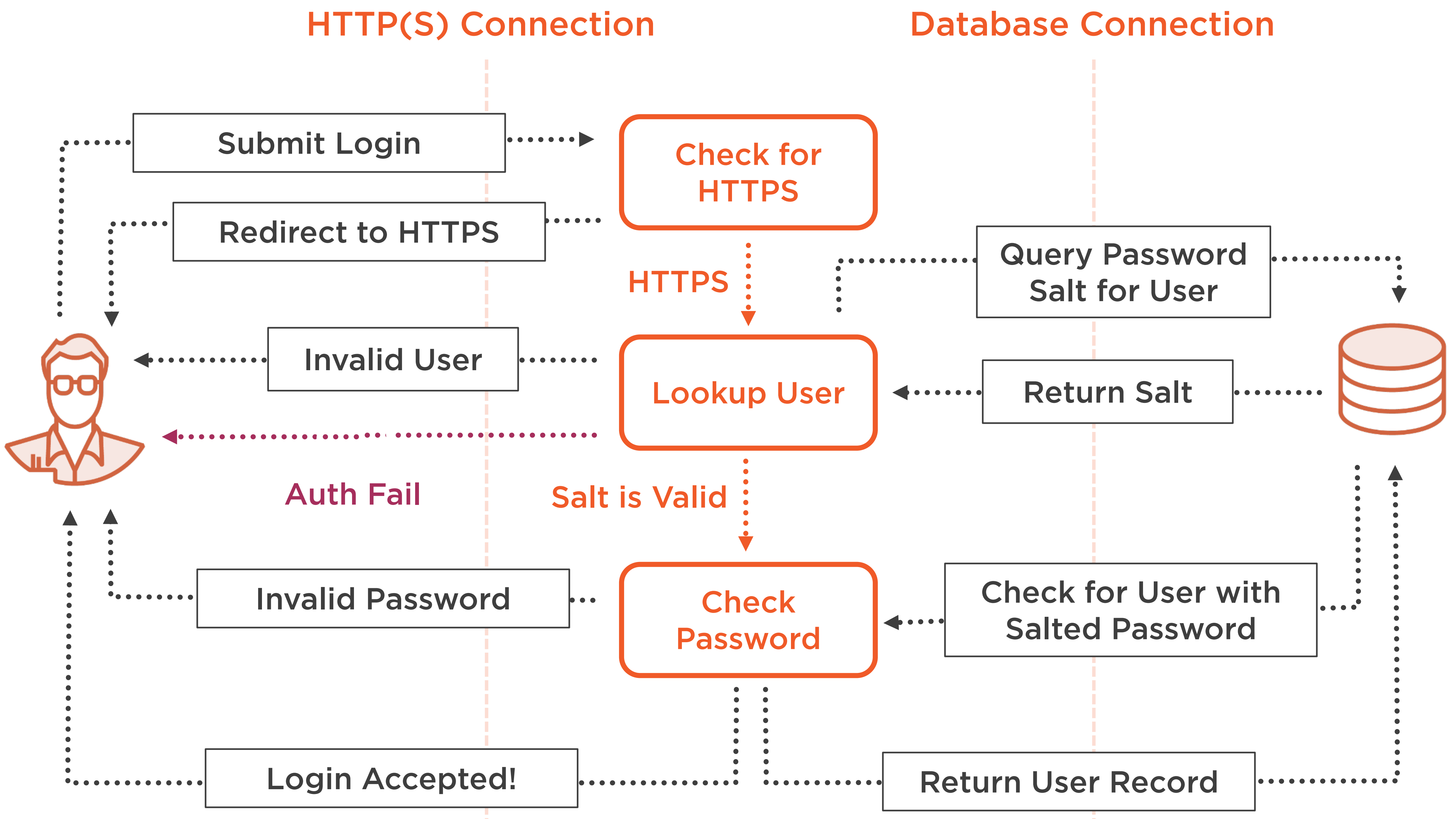
Do not *over* focus on Testing

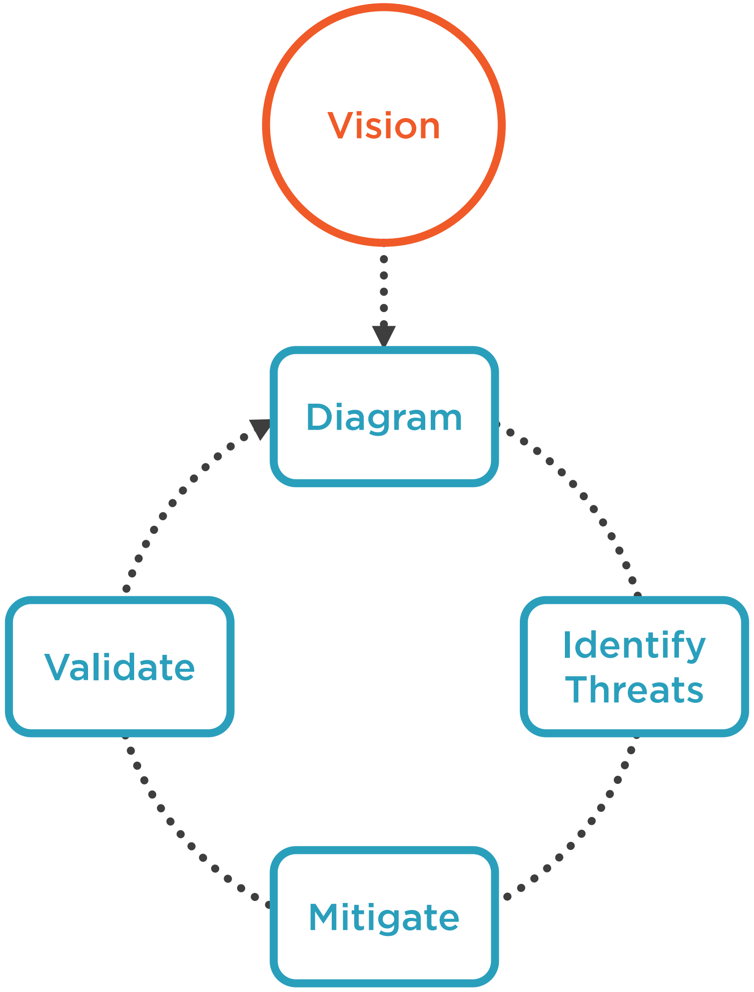
- Do threat modeling
  - Get Devs, Testers, and Operational folks together
    - Especially for today's cloud applications

Demo

Use a Data Flow Diagram

Threat Modeling Tool


HTTP(S) Connection
Database Connection
Submit Login
Check for HTTPS
Redirect to HTTPS
HTTPS
Query Password Salt for User
Invalid User
Lookup User
Return Salt
Auth Fail
Salt is Valid
Invalid Password
Check Password
Check for User with Salted Password
Login Accepted!
Return User Record



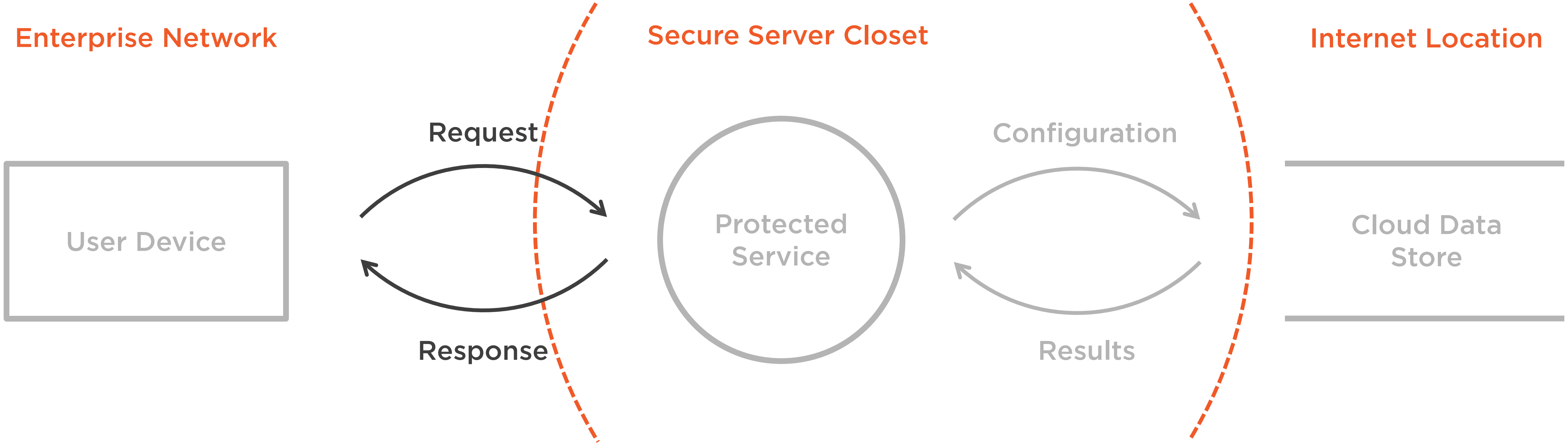
SDL Threat Modeling Process

Threat modeling

Informal or Formal?

- http://www.microsoft.com/security/sdl/adopt/threatmodeling.aspx
  - Definitions
  - Documentation
  - Free download

Threat Modeling with STRIDE



Risk Analysis

LOW

- Cafeteria menu
  - The use of static analysis may be enough

MED

- Perhaps B2B web apps
  - Static and dynamic analysis

HIGH

- Consumer desktop products
  - All, plus a more expensive pentest and manual analysis

Summary

Raising Security IQ

Push to the left

Attack surface reduction

Threat modeling