Microsoft SDL in practice

Alex Thissen
Principal Architect, Achmea

alex.thissen@achmea.nl  @alexthissen
Alex Thissen

- Architect with a focus on Microsoft technologies and products
  - Security
  - Competencies
- Trainer/coach in software development
- Regional Director for The Netherlands
- Most Valuable Professional for Visual C#
Agenda

• Overview of Microsoft SDL
• Phases of SDL
• Implementing SDL at Achmea
• Lessons learned
• Questions and answers
Think security

• Force yourself to pay attention to security during application development

• Security is often first victim
Security Development Lifecycle

- Embedding security into software and culture
- Platform agnostic approach
  - Proven benefits
- Microsoft internal adoption
  - Extensive experience with security
  - Trustworthy computing
SDL optimization model

Phased Approach

1. Assess
2. Identify
3. Evaluate and Plan
4. Deploy

The SDL Optimization Model

- Training, Policy, and Organizational Capabilities
- Requirements and Design
- Implementation
- Verification
- Release and Response

Basic
- Security is reactive
- Customer risk is undefined

Standardized
- Security is proactive
- Customer risk is understood

Advanced
- Security is integrated
- Customer risk is controlled

Dynamic
- Security is specialized
- Customer risk is minimized
Achmea SDL optimization

Start
- Basic: Security is undefined, Customer risk is undefined
- Standardized: Security is understood, Customer risk is understood

Goal
- Advanced: Security is controlled, Customer risk is controlled
- Dynamic: Security is specialized, Customer risk is minimized

Introduction
Self-assessment guide
Implementer's guide Basic -> Standardized
Implementer's guide Standardized -> Advanced
Implementer's guide Advanced -> Dynamic
Phases of Simplified SDL

- Training
  - Core Security Training
    - Establish Security Requirements
    - Create Quality Gates / Bug Bars
    - Security & Privacy Risk Assessment

- Requirements
  - Establish Design Requirements
  - Analyze Attack Surface
  - Threat Modeling

- Design
  - Use Approved Tools
  - Deprecate Unsafe Functions
  - Static Analysis

- Implementation
  - Dynamic Analysis
  - Fuzz Testing
  - Attack Surface Review

- Verification
  - Incident Response Plan
  - Final Security Review
  - Release Archive

- Release
  - Execute Incident Response Plan
Combining SDL and agile

- Requirements defined by frequency, not phase
  - Every-Sprint (most critical)
  - One-Time (non-repeating)
  - Bucket (all others)
Embedding SDL in process

- Guidance for process changes
- Process template for Visual Studio ALM integration
  - SDL
  - MSF Agile with SDL
IMPLEMENTING SDL AT ACHMEA
Focus at Achmea

• Emphasis on implementation at MScc
  ▪ Line-of-business apps
  ▪ Web portals

• Part of chain: bigger scope

• Embed SDL into “existing” development process
  ▪ Sync with quality gates
## Deliverables SDL for Achmea

### The Microsoft Security Development Lifecycle

#### Training
- Core training
  - Define quality gates/bug bar
  - Analyze security and privacy risk

#### Requirements
- Attack surface analysis
- Threat modeling

#### Design
- Specify tools
- Enforce banned functions
- Static analysis

#### Implementation
- Dynamic/Fuzz testing
- Verify threat models/attack surface

#### Verification
- Response plan
- Final security review
- Release archive

#### Release
- Response execution

### Best Practices
- **Tools:**
  - TFS work items
  - Security bug
  - SDL threat model
  - TFS work items
  - Security bug
  - Fx.Con
  - Watcher
  - TFS work items
  - Security bug
  - Fx.Con
  - Watcher
  - TFS work items

- **Documents (templates):**
  - PSA
  - (T)PID
  - SRS
  - Use Case Spec
  - SAD
  - Mis Use Case Spec

- **Reports:**
  - Threat model
  - Static analysis
  - Security bug
  - Test results
  - Static analysis
  - Security bug
  - Test results
  - Final security review

- **Coding Guidelines:**
  - Code review Guideline

- **Final security review:**
Training

- Online assessment and awareness course
- Security expert training
- Roadshow for all MScc employees
  - Focus on different phases in SDL for different roles
Requirements

• Business Impact Analysis (BIA)
  • Determines CIA rating
  • Weighs in on initial Architecture design and documentation
Design

- Combined Attack Surface Analysis and Threat model
- Change design to reduce surface
- Threat models as part of architecture
- Use SDL Threat Modeling Tool
- Determine risks from STRIDE
- Part of security view of SAD
Implementation

• Adopted Patterns & Practices guidance
  ▪ Best practices
  ▪ Guidelines and checklists
  ▪ Tooling

• Included CAT.NET in build

• Watcher
Verification

- BTOcc testplan adopted from OWASP
  - Testing for OWASP Top 10
  - ASVS testing
  - Dynamic, static and manual penetration testing
- Code reviews
Release

- Final Security Review (FSR)
  - Check on deliverables of previous phases
- Approval by Design Authority
- Ultimate quality gate
Response plan

• Incident response part of other departments
  ▪ IT Operations (IDS, monitoring)
  ▪ Security departments

• Close loop by applying lessons learned
LESSONS LEARNED
Taking hurdles

• Security as a hurdle
  ▪ “False positives”
• Break perception
  ▪ “Security takes time, budget and in not cool”
• Missing or sub-optimal tooling
Visibility

• Make sure you have security experts
  ▪ Advocating security
  ▪ People to ask questions
• Pick people that like it
• Find management that demands it
Achievable goals

• Small steps
• Not all at once
• Prioritize and pick from top 3
Continuous metrics

- Include security metrics in build
- Tooling is essential
- Testing only at end leads to disaster
Business and management

- Buy-in from management is essential
- Awareness at business is critical
- Don’t end in a showdown with business
Ongoing training

• Training alone is not enough
  • Offer help on-the-job
  • Not just before but during project as well

• Fast-moving field of security, attacks, vulnerabilities
Responsibility

• Define clear roles
  ▪ Who does what?
• Sharing responsibility
WRAPPING UP
Summary

• Embed security in your process
• It’s not easy
• Microsoft SDL turned out to be a good choice
• OWASP initiatives helped a lot
• You’re never done
Questions and Answers