Обзор практики разработки защищенных приложений

Андрей Иванов, RTO
Microsoft

Цепочка поставок ПО

Шаги, предпринимаемые правительствами

США – Defense Procurement Regulations; Draft Legislation

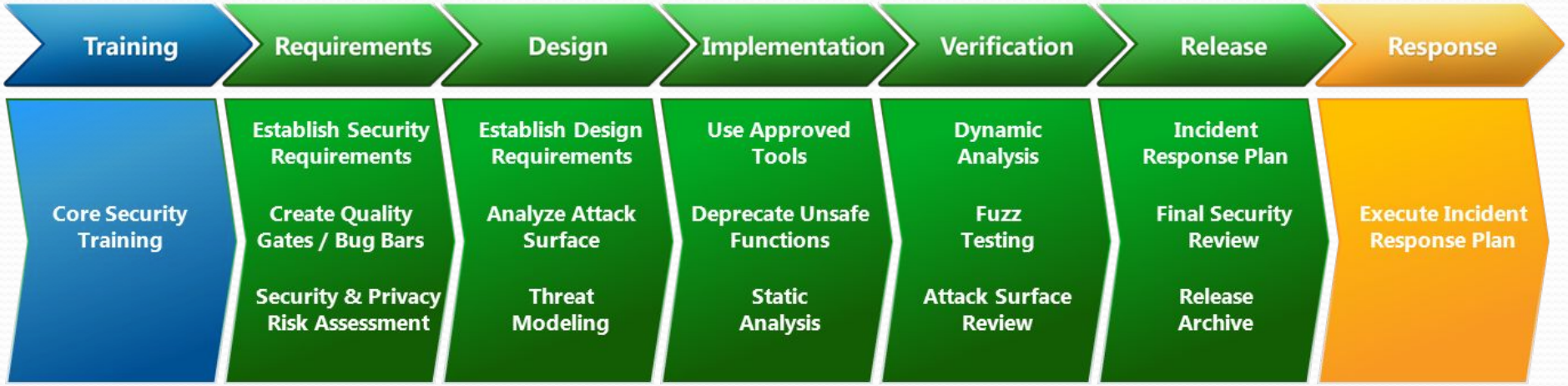
Россия – Сертификация на НДВ, НСД; Национальная программная платформа

Индия – Indigenous Telecom Innovation Proposals

Китай – 11th Five-Year Plan; Multi-level Protection Scheme (MLPS)

Что такое защищенное ПО

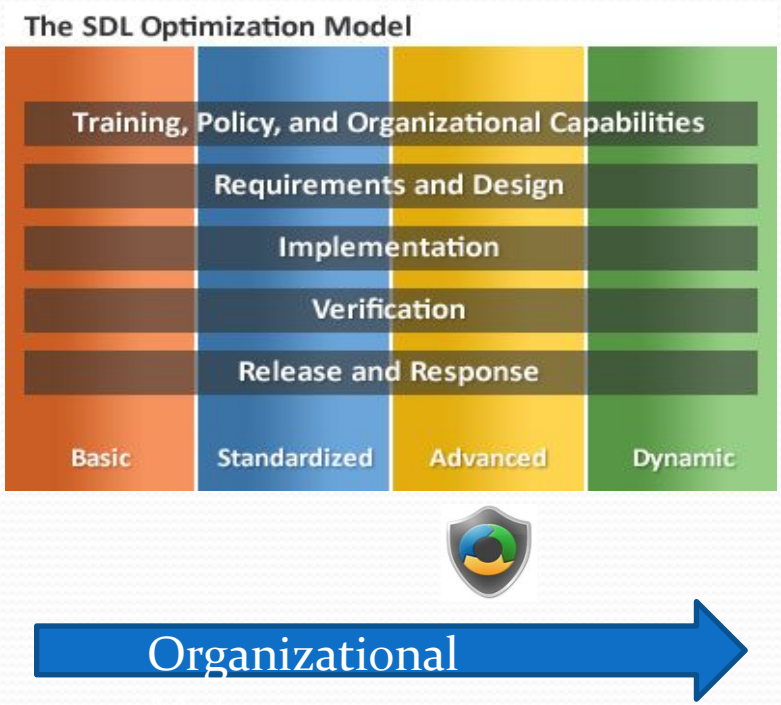
Security Development Lifecycle (SDL)



- 16 основных процессов. От тренинга до выпуска продукта
  - Максимальное использование автоматизации
  - Подразумевает распределение ролей и включает активности для архитекторов, Program/Project Managers, разработчиков и тестеров
  - Замечание: Последний этап “Response” формально не часть SDL, но тесно с ним связан
- Пост-релизные процессы
  - Исследование причин появления найденных уязвимостей. (из-за чего она возникла – человеческая ошибка?, несовершенство процесса?, ошибка автоматизации?)
  - Анализ уязвимостей схожих приложений
  - Тесты на проникновение

SDL Optimization Model

- Модель помогает определить текущий уровень зрелости компании и разработать план действий по внедрению соответствующих процессов для реализации полноценного цикла безопасной разработки



Что такой Simplified (упрощенный) SDL?

- Минимальный набор условий, соответствие которым оценивает Advanced уровень модели оптимизации SDL
- Включает в себя
  - Формализацию ролей и обязанностей всех вовлеченных в процесс разработки сотрудников
  - Обязательные меры обеспечения безопасности
  - Дополнительные меры обеспечения безопасности
  - Процесс проверки соответствия требованиям безопасности

Pre-SDL Requirements: Security Training

Assess organizational knowledge on security and privacy – establish training program as necessary

- Establish training criteria
  - Content covering secure design, development, test and privacy
- Establish minimum training frequency
  - Employees must attend n classes per year
- Establish minimum acceptable group training thresholds
  - Organizational training targets (e.g. 80% of all technical personnel trained prior to product RTM)

Phase One: Requirements

Opportunity to consider security at the outset of a project

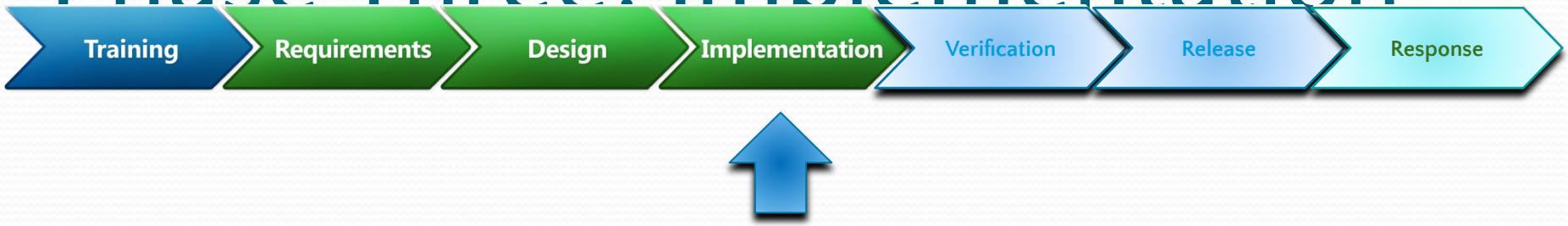
- Development team identifies security and privacy requirements
- Development team identifies lead security and privacy contacts
- Security Advisor assigned
- Security Advisor reviews product plan, makes recommendations,
  may set additional requirements
- Mandate the use of a bug tracking/job assignment system
- Define and document security and privacy bug bars

Phase Two: Design

Define and document security architecture, identify security critical components

- Identify design techniques (layering, managed code, least privilege, attack surface minimization)
- Document attack surface and limit through default settings
- Define supplemental security ship criteria due to unique product issues
  - Cross-site scripting tests
  - Deprecation of weak crypto
- Threat Modeling
  - Systematic review of features and product architecture from a security point of view
  - Identify threats and mitigations
- Online services specific requirements

Phase Three: Implementation



Full spectrum review – used to determine processes, documentation and tools necessary to ensure secure deployment and operation

- Specification of approved build tools and options
- Static analysis (/analyze (PREfast), FXCop)
- Banned APIs
- Use of operating system "defense in depth" protections (NX, ASLR and HeapTermination)
- Online services specific requirements (e.g., Cross-site scripting , SQL Injection etc)
- Consider other recommendations (e.g., Standard Annotation Language (SAL))

Phase Four: Verification

Started as early as possible – conducted after "code complete" stage

- Start security response planning – including response plans for vulnerability reports
- Re-evaluate attack surface
- Fuzz testing – files, installable controls and network facing code
- Conduct "security push" (as necessary, increasingly rare)
  - Not a substitute for security work done during development
  - Code review
  - Penetration testing and other security testing
  - Review design and architecture in light of new threats
- Online services specific requirements

Phase Five:
Release – Response Plan

Creation of a clearly defined support policy – consistent with MS corporate policies

- Provide Software Security Incident Response Plan (SSIRP)
  - Identify contacts for MSRC and resources to respond to events
  - 24x7x365 contact information for 3-5 engineering, 3-5 marketing, and 1-2 management (PUM and higher) individuals
- Ensure ability to service all code including “out of band” releases and all licensed 3rd party code.

Phase Five: Release – Final Security Review

Verify SDL requirements are met and there are no known security vulnerabilities

- The FSR provides an independent view into "security ship readiness"
- The FSR is NOT:
  - A penetration test – no "penetrate and patch" allowed
  - The first time security is reviewed
  - A signoff process
  - Key Concept: The tasks for this phase are used as a determining factor on whether or not to ship – not used as a "catchall" phase for missed work in earlier phases

Post-SDL Requirement: Response

"Plan the work, work the plan…"

- Execution on response tasks outlined during Security Response Planning and Release Phases

Что такое Software Integrity (SI)?

- Все чаще задается вопрос о том, а можем ли мы доверять тому или иному программному обеспечению
- Основная угроза обычно позиционируется как «внутренний нарушитель, обладающий определенными полномочиями»
- SI предназначен для снижения риска умышленной подмены функциональности продукта или сервиса.
- Применяемые методы хорошо известны и используются при защите интеллектуальной собственности от кражи.
  - Управление доступом к исходному коду, защита систем, обрабатывающих критические данные и т.д.
  - Цифровая подпись кода
- Дополняет SDL

Software Integrity

- Оценка угроз целостности в рамках существующей модели разработки и идентификация наиболее высоких рисков
- Обычно предпринимаются следующие категории мер:
  - Доказательство подлинности
  - Управление доступом
  - Разработка процессов контроля и мониторинга (аудита)
  - Сканирование ПО на вирусы
  - Цифровая подпись кода

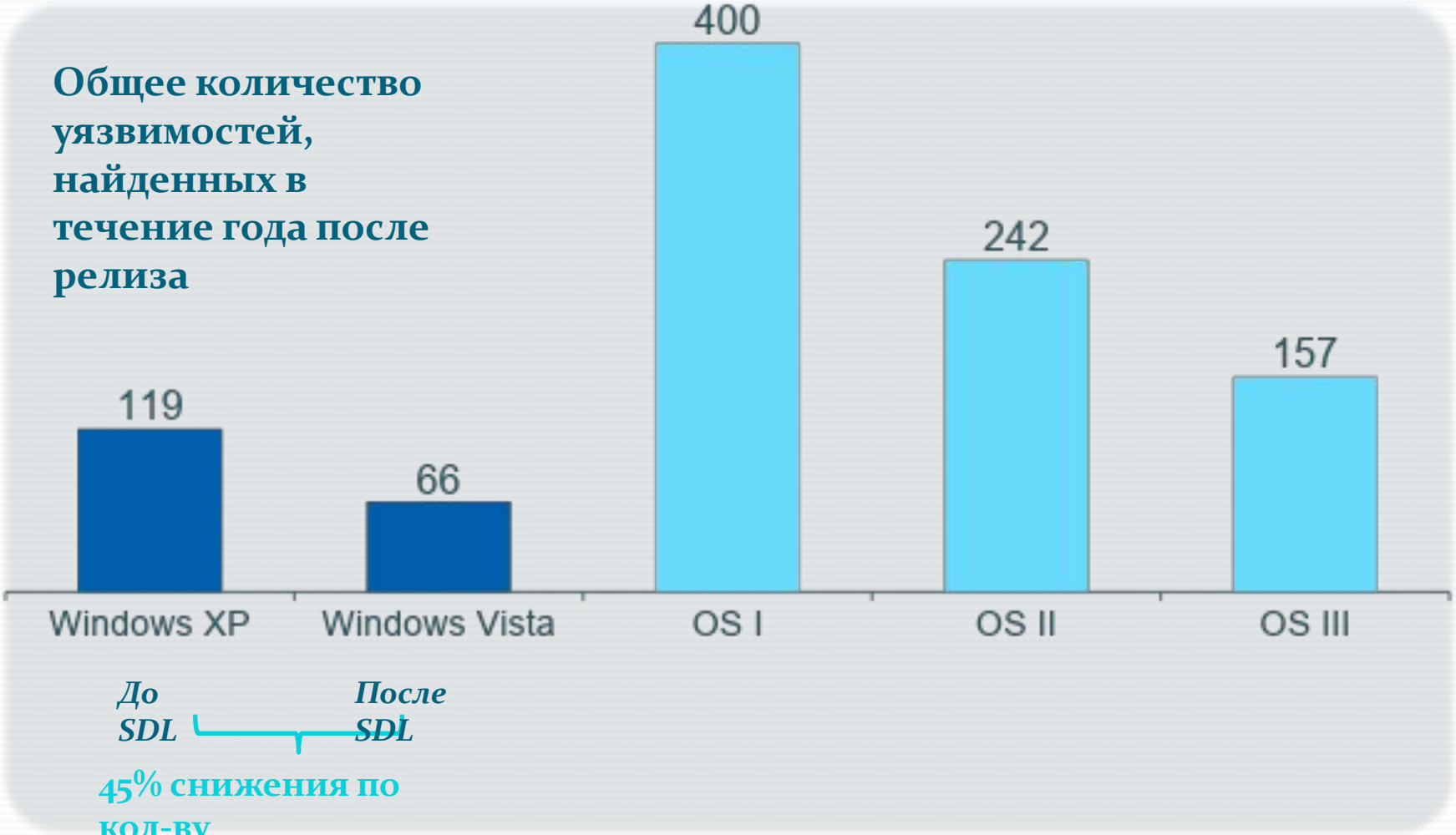
Выгоды

Недавние исследования выявили взаимосвязь между стремлением повысить безопасность разработки и эффективностью бизнеса

– Aberdeen Study Findings:

- • Prevention of a single security issue nearly offsets the total
- annual cost of average application security
- • 4X return on investments in applications security
- – Forrester Study –Key findings:
- • Application security is not a mature practice for many
- • Coordinated approaches experienced a stronger ROI
- • Those using SDL specifically reported visibly better ROI
- results than the overall population

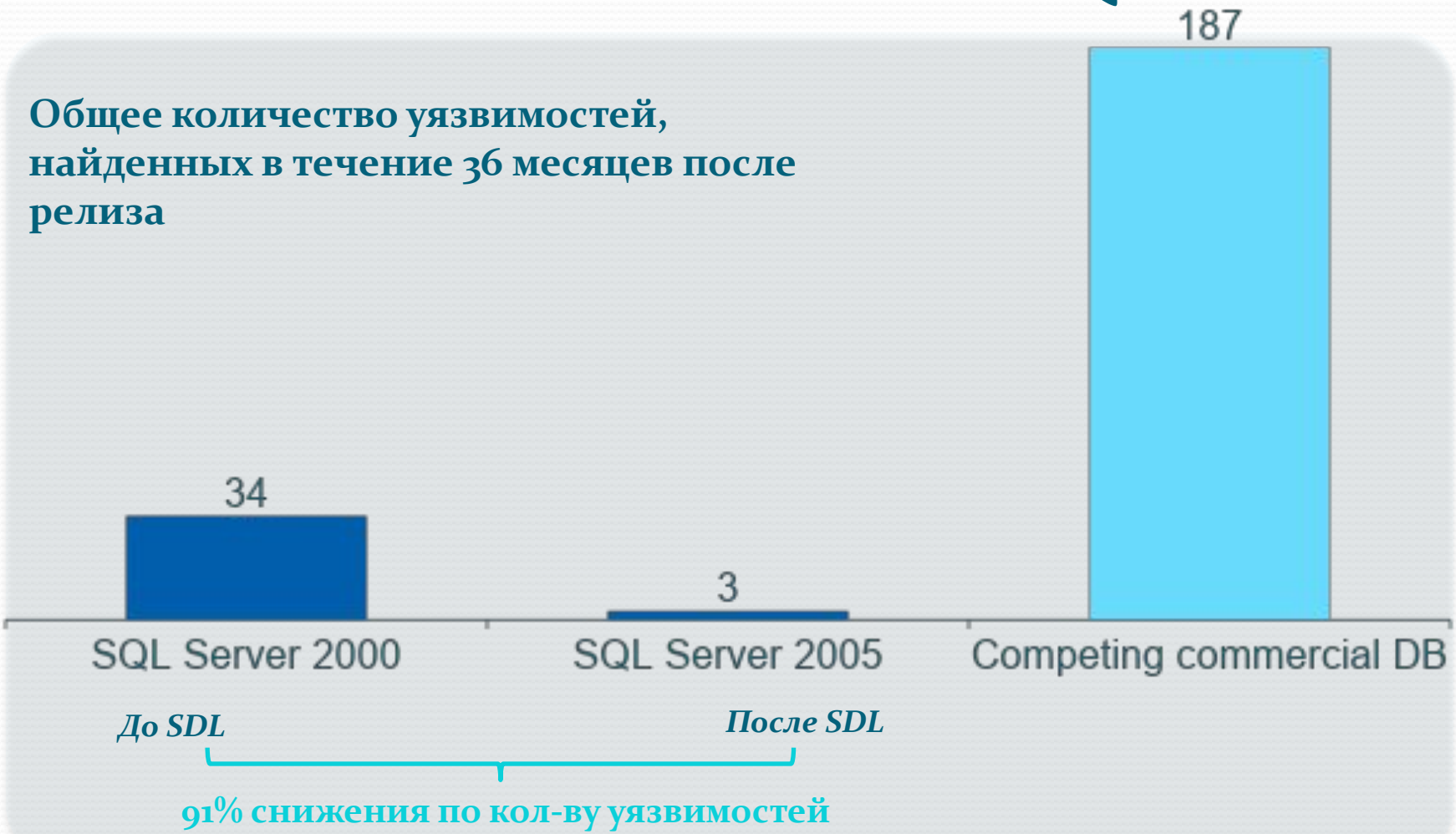
Microsoft SDL and Windows



Source: Windows Vista One Year Vulnerability Report, Microsoft Security Blog 23 Jan 2008

Microsoft SDL and SQL Server

Общее количество уязвимостей, найденных в течение 36 месяцев после релиза



Sources: Analysis by Jeff Jones (Microsoft technet security blog)

Resources

Trustworthy Computing

http://www.microsoft.com/twc

TwC Blogs

http://www.microsoft.com/mscorp/twc/blogs/default.mspx

SDL Portal

http://www.microsoft.com/sdl

SDL Process on MSDN (Web)

http://msdn.microsoft.com/en-us/library/cc307748.aspx

SI Whitepapers

Cyber Supply Chain Risk Management: Toward a Global Vision of Transparency and Trust

http://www.microsoft.com/download/en/details.aspx?id=26826

Toward a Trusted Supply Chain: A Risk Based Approach to Managing Software Integrity

http://www.microsoft.com/download/en/details.aspx?id=26828

Спасибо!

Андрей Иванов
andreyi@microsoft.com