HERVÉ SCHAUER CONSULTANTS
Cabinet de Consultants en Sécurité Informatique depuis 1989
Spécialisé sur Unix, Windows, TCP/IP et Internet

Netfocus

Sécurité des applications
Retour d'expérience

Nicolas Collignon <Nicolas.Collignon@hsc.fr>

Plan

- Introduction
- Sensibilisation
- Modélisation et conception
- Tests
- Conclusion

La sécurité d'aujourd'hui ...

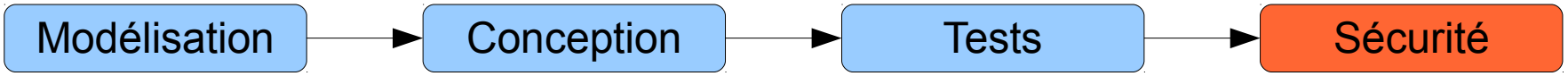
- Vulnérabilités réseau à la baisse pour les attaques externes
- Les applications sont de plus en plus exposées sur le réseau
- Tendance à la « web-isation » des applications
- Diffusion des bulletins de sécurité et des codes d'exploitation de plus en plus rapide

Les applications

- Contraintes métier
  - Évolution constante « Il faut ajouter les fonctionnalités X, Y et Z »
  - Le temps « L'application doit être prête pour demain»
  - Ressources limitées « Il faut développer, débugger, tester et faire le support technique »
- Principaux problèmes
  - Les développeurs manquent de temps
  - Les développeurs sont généralement mal sensibilisés à la sécurité
  - Absence de tests externes

Comment gérer la sécurité

- Intervention après (trop tard..)



  - Coûts potentiellement élevés

- Intervention avant

  - Sécurité intégrée au SDLC

  - 2 heures de perdues en phase de conception
    ► 1 semaine de gagnée en phase de correction des bugs

Développeurs vs. Pirates

Développeurs

- Objectif

  « Comment faire marcher la fonctionnalité XYZ pour demain ? »

- Caractéristique

  Potentiellement pas ou **mal sensibilisés à la sécurité**

Pirates / Auditeurs

- Objectifs

  « Comment casser l'application ? »

  « Comment passer administrateur sur le serveur ? »

- Caractéristique

  Bien sensibilisés à la sécurité

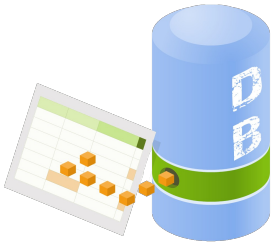
Sensibilisation

Sensibilisation à la sécurité

- Sensibiliser les personnes impliquées dans le projet
  - Développeurs, chefs de projet, RSSI, etc..
- Former les développeurs aux problèmes de sécurité liés au langage de programmation choisi
  - Les bonnes pratiques de programmation sont disponibles sur Internet
  - Le panel du type de vulnérabilités évolue avec le temps

Cas « MSSQL va mal » 1/2

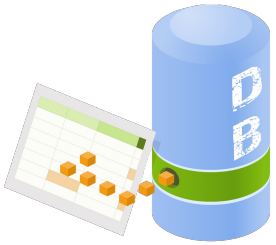
- Site Web d'un service client automobile
  - Application ASP
  - Test d'intrusion
  - Compromission de la base de données et du système



- Mauvaise validation des données utilisateurs
  - Une requête SQL intègre une donnée potentiellement malveillante
    - Injection SQL
  - Utilisation du compte MSSQL « sa »
    - Élévation de privilèges en tant que SYSTEM sur le serveur Windows

Cas « MSSQL va mal » 2/2

- Correction de la majorité des interactions avec la base de données



Manque de sensibilisation

↓

Sécurité «  »

↓

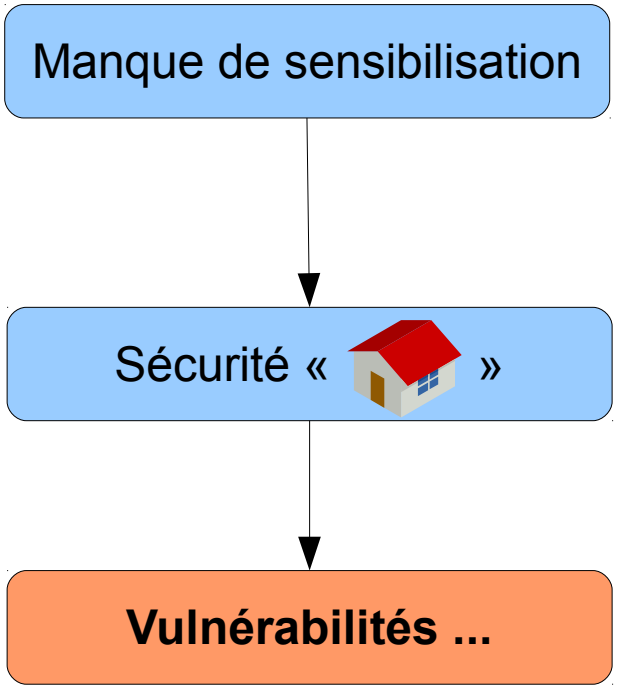
Vulnérabilités ...

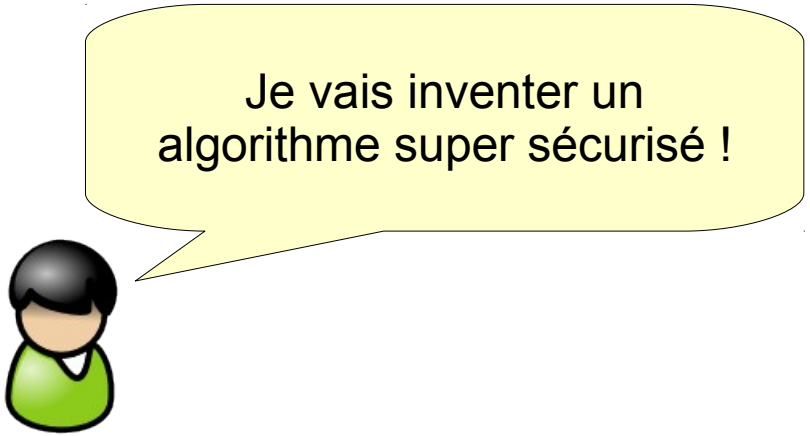
Cas « crypto maison » 1/2

- Partage d'informations vidéo P2P
  - Application propriétaire (C++)
  - Analyse boîte-noire (reverse-engineering)
  - Contournement de l'authentification
- Mauvaise implémentation cryptographique lors de l'authentification
  - La transformation du mot de passe génère de nombreuses collisions
  - Le mot de passe n'est pas suffisamment utilisé dans le calcul clé RC4
  - Attaque « plain-text » sur RC4 en 256 itérations
  - Authentification sans connaître le mot de passe

Cas « crypto maison » 2/2

- Obligation de gérer deux versions du protocole car tous les clients ne peuvent pas être mis à jour





Modélisation et conception

Modélisation et conception

- La sécurité doit être prise en compte dès la modélisation

  ► Évite les lourdes corrections après découverte de failles

- Schématiser les interactions dans l'application entre les blocs de confiance et les blocs sensibles

  ► Permet de mieux comprendre le code qui est sensible

- Suivre les bonnes pratiques de sécurité

Cas « je vois ton compte client » 1/2

- Application Web Java / Affacturage
  - Application de gestion de comptes clients
  - Test d'intrusion externe
  - Usurpation d'identité

- Absence de contrôle sur l'identité qui accède à un compte
  - Le client indique au serveur quelle identité utiliser
  - Le serveur ne vérifie pas l'information
  - Injection de l'identité dans les cookies

Cas « je vois ton compte client » 2/2

- Conséquences
  - Ré-implémentation de la gestion des sessions utilisateurs
  - Impact sur le code serveur et le code client

Contraintes de temps
+
Manque de sensibilisation

↓

Sécurité «  »

↓

Vulnérabilités ...

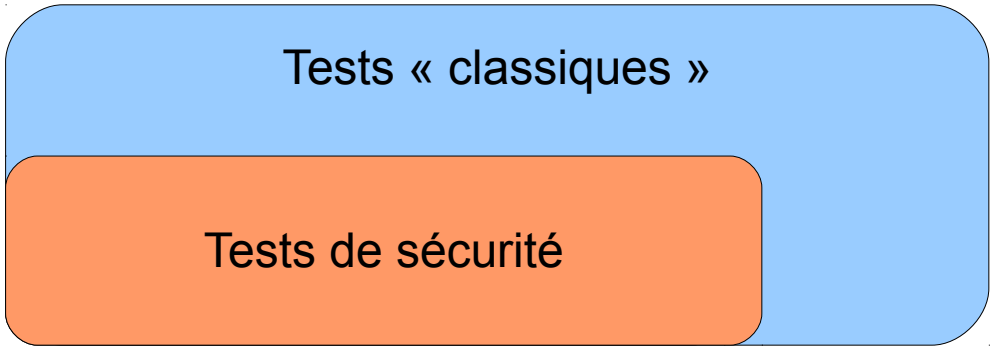
C'est beaucoup plus simple quand je stocke l'identifiant dans les cookies.

J'afficherai uniquement les comptes de l'utilisateur pour pas qu'il accède aux autres comptes

Tests

La phase de tests

- 2 types de tests



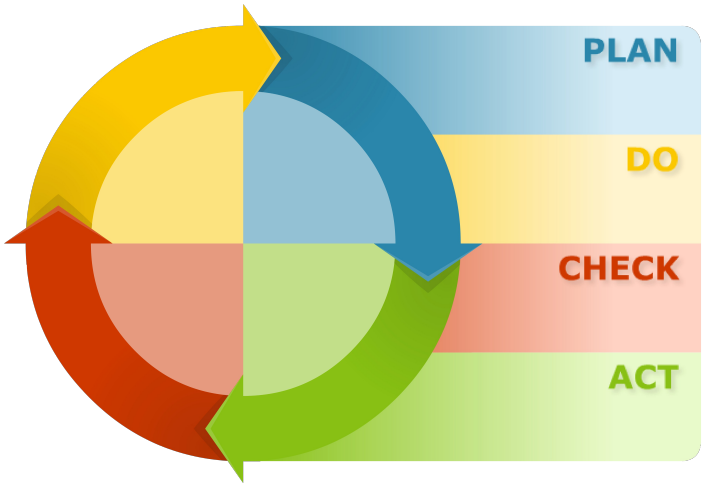
- Procéder aux tests de sécurité pendant les tests unitaires
- La phase de test / debug est plus longue que la conception
- Tester est souvent aussi compliqué que développer

Qui teste ?

- Relecture en interne
  - Nécessite des ressources et du temps
  - Relecture par des développeurs autres que ceux qui ont développé
- Relecture assistée par un logiciel
  - De gratuit à très cher, d'inutile à très rentable
  - Ne permet pas de détecter les problèmes de logique métier
- Tests externes
  - Audits de code
  - Tests d'intrusion

Et après ?

- L'application vit …
  - Modification du code source
  - Modification de la configuration
  - Modification de la configuration système
  - Modification de la configuration réseau



- Intégrer la sécurité dans le cycle de vie de l'application

- Prier ...

Cas « je peux changer les cotes » 1/2

- Portail de paris sportifs
  - Serveur d'application Web + application Web
  - Audit de code source (~280 000 lignes de C et TCL)
  - Développeurs fortement sensibilisés à la sécurité
- Compromission du serveur, possibilité de changer les cotes des paris
- Modification récente du code non revue en interne
  - Fonctionnalité non utilisée mais présente dans le code
  - Manque de vérification des données utilisateurs
  - Exécution de code TCL arbitraire

Cas « je peux changer les cotes » 2/2

- Fonctionnalité inutile supprimée
  - Correction rapide

Forte expertise en sécurité
+
Mauvais processus de relecture du code

↓

Sécurité « presque très bien »

↓

Vulnérabilités ...

Conclusion

En bref

- Impliquer la sécurité en amont des projets
- Sensibiliser les acteurs des projets à la sécurité
- Intégrer la sécurité au SDLC
- La sécurité ne se résume pas au code source
  - Configuration de l'application
  - Configuration système
  - Configuration réseau

Questions ?