Sécurité en ingénierie du Logiciel

Le cadre des Web Services
Partie 1 : Introduction

Alexandre Dulaunoy

adulau@foo.be

Agenda (partie 1) 1/2

- Introduction
- Services Web
  - Introduction
  - Historique de Sun RPC à CORBA
  - L'avenir ?
- Sûreté du logiciel
  - Simple ? non !
  - Architecture et Design
  - Réalisation

Agenda (partie 2) 2/2

- Sûreté des Services Web
  - Risques
  - Introduction aux éléments sécurités
  - Introduction aux Bonnes pratiques
- Conclusion
- Bibliographief/réf/link
- Acronymes
- Q&R

Introduction 1/2

Pourquoi les Web Services ? L'historique...

- Protocoles de communication liés aux SEs, applications,
- API accessible à un type uniquement de SE,
- Complexité (structure base de données, langage utilisé,...) ,
- Interfaçage difficile,
- Services distribués,

Introduction 2/2

Pourquoi les Web Services ? Internet

- Tim Berner Lee créa le premier client Web en 1990,
- Conçu pour une interface Homme-Machine,
- Evolution vers machine-machine,
- Parsing automatique difficile,
- Interface instable et propre à chaque technologie/site,

Une longue route...

- Plusieurs tentatives,
  - -> SOAP, XML-RPC, WSDL, ...
- Evolution constante (!) des standards,

Services Web - Origines

Les origines du concept...

- SUN Remote Procedure Call
  - Première approche de standardiser les interactions entre un client et un serveur via un modèle de Remote Procedure Call,
  - Le serveur offre des procédures (identifiées par un nom) et le client demande au serveur l'accès à une procédure donnée avec les valeurs (paramètres),
  - XDR (eXternal Data Representation) solutionne le (une partie) problème de la représentation binaire des données,

Services Web - Origines

Microsoft DCOM (Distributed Component Object Model)
(COM -> DCOM -> COM+)

- Pas uniquement des procédures (comme SUN RPC) mais aussi un interfaçage avec des objets ou appels de méthodes,
- DCOM se veut neutre par rapport à la plateforme, au langage et même au Transport,
- DCOM est resté dans l'environnement Microsoft pour des questions de standardisation,
- DCOM est propriétaire.
- (-> .NET normalisé ECMA)

Services Web - Origines

CORBA (Common Object Request Broker Architecture)

- Fonctionne avec un ’dispatcher’ (ORB) pour les requêtes et les transferts vers un serveur donné,
- Utilisé sur des très larges projets,
- IDL (Interface Design Language) est un langage de description des services offerts par l’application,
- La complexité de CORBA est souvent une source de problèmes (IDL complexes, transport IIOP & securité,...),

Services Web - Avenir

- Insatisfaction avec DCOM, SUN RPC, DCOM, RMI,...
  - Sortir des problèmes de compatibilités systèmes, langage et format,
  - Sortir du problème propriétaire et créer des standards,
  - Utiliser l'infrastructure Internet existante,
  - Simplifier et limiter le temps de développement,
- -> XML-RPC, SOAP, ...
- Un rêve ? Les questions de performances, de gestion des erreurs, de sécurité des infrastructures distribuées utilisant les Services Web existent toujours...

Sûreté du logiciel

If our software is buggy, what does that say about its security ?

Robert H. Morris

Pourquoi est-il si difficile de réaliser des logiciels sûrs ?

- Questions techniques,
- Facteurs humains,
- Facteurs économiques,

Sûreté du logiciel

Architecture et Design : quelques bonnes questions...

- Contre qui se protéger ?
- Que voulez-vous protéger dans votre logiciel ?
- Quel est le point faible de votre logiciel ?
- Que peut-il arriver de pire à votre logiciel ?
- Quelle est votre architecture de sécurité ?
- Quelles sont vos standards ?
- Avez-vous utilisé les bonnes pratiques ?
- Avez-vous testé la sécurité du logiciel ?

quelques bonnes pratiques...

- Pensez à la sécurité de votre logiciel au début du design,
- Pensez à vos ennemis lors du design,
- Utilisez le minimum de privilèges pour votre logiciel,
- Utilisez une gestion solide et simple des erreurs,
- La simplicité sera votre maître mot,
- Utilisez des actions minimales par défaut,
- N’utilisez pas l’obscurcissement,
- ...

Sûreté du logiciel

Postfix MTA (Wietse Venema) un exemple de design sécurité pour un serveur de mail :

- Privilèges minimums (p.ex. chroot) surtout pour les composants en contact avec l'extérieur,
- Isolation des processus (aucun accès direct),
- Environnement contrôlé (p.ex. master qui contrôle les autre processus),
- Confiance minimale entre les composants,
- Contrôle des entrées importantes et tronquées,
- Contrôle du nombres du processus,
- Clarté du design et du code source (cf. le Logiciel),

Sûreté des services web

Les services web sont des logiciels donc vous avez tous les problèmes du logiciel plus :

- Une architecture décentralisée,
- Une administration décentralisée,
- Environnement hétérogène,
- (souvent) Ouvert à Internet,
- Des connections entre plusieurs points,

Ainsi que tous les problèmes relatifs au différentes couches de transport utilisées (TCP/IP) par les services web...

Sûreté des services web

Les points critiques :

- Authentification,
- Autorisation / Access Control,
- Single Sign On,
- Chiffrage (Encryption),
- Non-Repudiation (messages),
- Mesures de protection,

Conclusion

Les standards autour des services web évoluent encore et ils existent encore un nombre important de risques sécurités (comme tous les logiciels, c'est inevitable).

Une bonne compréhension de l'architecture des services web permet de mieux prévenir les risques et de créer des logiciels utilisant ces services. Le but majeur du cours est de créer un esprit de sécurité lors de la mise en pratique.

Bibliographie

- The Practice of Programming, Brian W. Kernighan, Rob Pike - Addison Wesley. 1999. ISBN 0-201-61586-X.
- Building Secure Software, John Viega, Gary McGraw - Addison Wesley. Summer, 2001. ISBN 020172152X.
- Web Services Essentials, Ethan Cerami - O'Reilly. ISBN 0-596-00224-6.
- Programming Web Services with XML-RPC, Simon St. Laurent, Joe Johnston, Edd Dumbill - O'Reilly. June 2001. ISBN 0-596-00119-3.

Liens

- http://www.w3.org/ (W3C) (de XML à SOAP en passant par HTTP)
- http://www.xmlrpc.com/ (de XML-RPC à XML-RPC)
- http://www.soapware.org/ (de SOAP à SOAP (librairies, serveur,...)
- http://www.w3.org/2002/ws/ (Services Web)
- http://www.webservices.org/

Acronymes

- XML : Extensible Markup Language
- XML-RPC : XML-Remote Procedure Call
- SOAP : Simple Object Access Protocol
- WSDL : Web Service Definition Language
- UDDI : Universal Description, Discovery, and Integration
- SSL/TLS : Secure Socket Layer / Transport Layer Security

Q&R

- adulau@foo.be
- http://www.foo.be/cours/securite-webservices/
- 3B12 DCC2 82FA 2931 2F5B 709A 09E2 CD49 44E6 CBCD