



HERVÉ SCHAUER CONSULTANTS

Cabinet de Consultants en Sécurité Informatique depuis 1989

Spécialisé sur Unix, Windows, TCP/IP et Internet

Web Services et sécurité



Espace RSSI du Clusif

10 Septembre 2003

Hervé Schauer

<Herve.Schauer@hsc.fr>

- x Qu'est-ce que les Web Services ?
- x Ce que permettent les Web Services
- x Comparaison IP / HTTP
- x Objectif des Web Services en sécurité
- x Expérience HSC avec les Web Services
- x Quels sont les vulnérabilités ?
- x Exemples de failles applicatives
- x Causes des difficultés actuelles
- x Recommandations
- x Conclusion
- x Ressources – Acronymes

- × Une infrastructure d'échange de messages
 - × Sur le protocole HTTP/HTTPS
 - × Donc au travers de clients et de serveurs Web
 - × Entre applications
 - × Entre clients et applications
- × Permet d'offrir une application comme tout autre ressource adressable par une URL
 - × Notamment sur Internet
- × Les clients et les applications n'ont pas à savoir comment un service est implémenté.
- × Promu par l'ensemble du marché : J2EE, .Net, etc

- × L'échange de données structurées (XML)
- × La réutilisation d'applications
- × La distribution d'applications
 - × Comme les RPC (*Remote Procedure Call*)
 - × A l'échelle globale, sur Internet
 - × Comme le ver Blaster avec les RPC DCOM
- × Un système de dénomination global unique (URI)
- × Des projets de systèmes répartis de gestion des authentifications et des permissions (SAML)
- × Des projets d'annuaires de Web Services (UDDI)

x	Services sur IP	Services sur HTTP
x	Normalisation	OASIS & W3C
x	Identification d'un service	URI
x	Authentification/chiffrement	SSL/TLS
x	Protocole de transport	SOAP
x	Présentation des données	XML

- × Les *firewalls* et les systèmes de contrôle d'accès en place
 - × Sont figés
 - × Ne permettent pas les RPC
- × Peu d'entreprises ont accepté d'ouvrir les flux pour des protocoles comme FIX ou IIOP pour une application
- × ⇒ Contournement des *firewalls* par réencapsulation sur la couche supérieure
- × ⇒ Besoin de *firewalls* HTTP/SOAP/XML pour appliquer sa politique de sécurité
- × Déport du problème de sécurité et de la réactivité

- × Croisés lors de diverses prestations
 - × Audits de sécurité
 - × Etudes d'architecture de sécurité
 - × Tests d'intrusion
- × Projets
 - × Migration de systèmes EDI X400 vers Web Services XML
 - × Création de nouveaux services d'échanges
 - × Plateformes de marché
 - × Coté fournisseur de service
 - × Coté entreprise utilisatrice
- × Tout secteurs d'activités
 - × Industriels, opérateurs, banques, secteur public, opérateurs de services de confiance

- x Les principales vulnérabilités rencontrées :
 - x 1) Programmes entrent et sortent du système d'information
 - x Sans contrôle d'accès
 - x Sans analyse de contenu
 - x ⇒ Pénétration du réseau privé par envoi de code malveillant
 - x 2) Logiciels mis en frontal sur internet sans avoir été prévu pour
 - x S'appellent "connecteur XML" mais constituent un serveur web
 - x ⇒ Dénis de services
 - x ⇒ Intrusion par les failles applicatives

- × Failles rencontrées par hasard
- × Envoi de code XML mal formé \Rightarrow plantage du connecteur XML
 - × Exemples : Webmethods, Baltimore, Entrust, etc
- × Inclusion dans les messages XML d'attachements MIME, et inclusion dans les attachements MIME de binaires Windows
 - × Exemples : tous les profils XML rencontrés le permettent
- × Injection de XML dans les données XML \Rightarrow changement complet du sens du message XML
- × Exécution de commandes au niveau système d'exploitation

- × Absence de sécurité à priori
 - × Installation systématique sans sécurité
 - × Mots de passe par défaut
 - × Exemple : les installations SAP utilisent toujours les mots de passe Oracle par défaut
 - × Fichiers des applications ou des bases de données accessibles en lecture ou lecture/écriture pour tous au niveau du système d'exploitation
 - × Mots de passe stockés en clair sur le disque
- × Discretion
 - × Les systèmes utilisant les Web Services entrent partout dans l'entreprise sans être vus : EDI sur ADSL dans le bureau, extranets, développements métiers, applications transparentes, etc

- × Chocs culturels et erreurs conceptuelles :
 - × Confusion entre transport et application
 - × Utilisation du même certificat pour la session SSL et la signature XML
 - × Exemple : profil rosettanel
 - × "C'est un partenaire en qui j'ai confiance", il ne m'enverra jamais de données malveillantes
- × Désintérêt
 - × Pas de budget alloué aux aspects sécurité des Web Services
 - × Peu d'études ou de travaux demandés sur le sujet
 - × Pas de moyens pour travailler à la recherche de solutions
 - × Pas d'affaires pour les éditeurs de logiciels de sécurité traitant du problème
 - × Exemple : Vordel

- × Usage de SSL/TLS + certificats
- × Usage de signatures XML
- × Minimisent considérablement les risques tels que :
 - × Vol d'identité
 - × Interception
 - × Atteinte à la confidentialité ou l'intégrité des données
- × N'apportent **rien** pour minimiser les risques d'intrusions dans les applications
 - × Une application dont le contrôle a été acquis acceptera n'importe quelle signature, enverra n'importe quel code malveillant, etc
- × Journalisation généralement possible à tous les niveaux
- × A gérer : non-répudiation

- × Imposer la prise en compte de la sécurité dans ces projets
- × Intégrer dans l'analyse du risque, les fonctions présentes mais pas visibles, et les menaces potentielles
 - × Ne pas attendre qu'il y ait un ver XML ... exploitant sur le coeur du système d'information le déni de service de Cisco IOS de Juillet 2003
 - × Considérer la menace à sa juste valeur dès l'origine du projet
- × Ne pas faire confiance aux données XML reçues
 - × Validation syntaxique et lexicale avant tout transfert à un connecteur XML
 - × Décodage
 - × Analyse du contenu
 - × Envoi à des analyseurs tiers, à l'anti-virus, récursivité
- × Utiliser un relais HTTP/HTTPS/SOAP/XML de sécurité

- × Ne pas faire confiance aux données XML émises
 - × Pas de maîtrise sur les applications dans l'entreprise qui envoient ces données
 - × Que maîtrisez-vous de SAP ?
 - × Même recommandation que sur le HTML vis-à-vis de l'exécution croisée de code
- × Utiliser un contrôle d'accès par filtrage IP quand les partenaires sont listables
 - × ⇒ Jusqu'à 40 partenaires est généralement gérable
- × Utiliser les fonctions de contrôle d'accès applicatives des connecteurs XML
 - × Permissions par *Client Groups* ↔ Liste de *Distinguished Names* sur les applications / les documents

- × Ne pas donner accès aux services de supervision des applications sur le réseau de production
- × Imposer aux fournisseurs de service des règles de sécurité draconiennes
 - × Infogérance
 - × Plate-formes de marché

- × L'intégration des processus d'affaires entre entreprises ne peuvent que se développer
- × La prise en compte de la sécurité est un élément clé de la réussite de cette évolution
- × Les Web Services peuvent permettre la mise en oeuvre d'attaques variées et conséquentes
- × Il faut sensibiliser pour prendre dès le début les bonnes décisions et intégrer pleinement la sécurité applicative

×

× Questions ?

- × Bases de données et sécurité
 - × Alain Thivillon et Nicolas Jombart, Hervé Schauer Consultants
 - × Journées CFSSI de la DCSSI, Avril 2002
 - × <http://www.hsc.fr/ressources/presentations/secsgbd/index.html.fr>
- × Architectures de sécurité pour les EAI
 - × Emmanuel Attali, FT Euralba
 - × Forum XML et Web Services, Novembre 2002, non publié
- × WebMethods B2B 4.x Security Best Practices
 - × Jeremy Epstein, WebMethods, Mai 2001, confidentiel, soumis à NDA
- × SAML Basics
 - × Eve Maler, Sun, Décembre 2001
 - × <http://xml.coverpages.org/Maler-saml-basics.ppt>

- x FIX : Financial Information eXchange protocol
 - x www.fixprotocol.org
- x IIOP : Internet Inter-ORB Protocol
 - x www.omg.org
- x SAML : Security Assertion Markup Language
 - x www.oasis-open.org
- x SOAP : Simple Object Access Protocol
 - x www.w3c.org
- x URI : Uniform Resource Identifier
- x UDDI : Universal Description, Discovery & Integration
 - x www.uddi.org