

Compte-Rendu des TechDays 2007

par [Patrick OTTAVI MVP Visual C++](#)

Date de publication : 15/02/2007

Dernière mise à jour : 15/02/2007

Le 5, 6,7 Février j'ai assisté aux TechDays 2007 en compagnie de quelques collègues (garçons et fille) de la rédaction, Notre point de ralliement était le stand de DVP dans l'emplacement des communautés. Je vais essayer de vous faire une synthèse des sessions auxquelles j'ai assistées.

- I - Session Ecrire du code sécurisé : modélisation des menaces
- II - Session Windows vista : nouveautés de la sécurité
- III - Session Longhorn Server de A à Z
- IV - C++ Mixed code : Migration VC6-VC8
- V - Introduction au HPC pour les développeurs
- VI - Tout sur le nouveau fichier de format OpenXML
- VII - Conclusions

Le message de Microsoft à ce salon c'était la sécurité, l'interopérabilité et la virtualisation.

Le tout s'exprimant à travers les produits phares de la firme à savoir Windows Vista, office 2007 et dans une certaine mesure Longhorn server (pour les principaux).

I - Session Ecrire du code sécurisé : modélisation des menaces

Commençons par la sécurité, Microsoft en a fait son cheval de bataille en appliquant la méthode de développement **SDLC** : Secure Development Life Cycle , en quelques mots :

Les étapes sont fixées à l'avance et on ne peut passer à la suivante que lorsque l'étape est validée.

On peut les résumer ainsi :

- Définitions des exigences (cahier des charges)
- Conception générale
- Implémentation
- Vérification
- Lancement
- Support et entretien

Cette méthode nous a été présentée pendant la session et les deux intervenants Eric Vernier et Eric Mittelette nous ont enjoint à un petit exercice d'étude de menace à travers un projet fictif.

Cette démarche de sécurité des produits se retrouve bien sûr à travers les différents produits présentés.

Pour la sécurité du code Visual 2005 n'est pas en reste avec par exemple la directive **/Analyse** qui lance un analyse du code sur la sécurité, les cast de données les exceptions etc ..

Pour en savoir plus lire l'article sur MSDN

<http://www.microsoft.com/france/msdn/securite/vs05security.msp>

II - Session Windows vista : nouveautés de la sécurité

Dans le prolongement de la session précédent voici quelques exemples sur les mécanismes de sécurités mis en place dans Vista.

Le principe du moindre privilège pour les comptes utilisateurs avec l'**UAC :User account Contrôle**.

Le principe étant que chaque utilisateur ou compte d'administrateur fonctionne avec un nombre restreint de privilèges, dès qu'une action requière un privilège important une fenêtre apparaîtra nous invitant à confirmer l'action dans le cas d'un compte administrateur.

Je me pose une question : est ce que les utilisateurs ne vont pas être excédés par ces messages intrusifs et être tentés de désactiver le mécanisme ?

Réduction de la surface d'attaque sur les services en leur associant un minimum d'autorisation

Limitant ainsi les dégâts aux seuls droits attribués en cas de détournement de celui-ci.

Organisation du noyau en couche avec une hiérarchisation, la couche supérieure peut dépendre de la couche inférieure mais pas l'inverse.

Contexte d'exécution limité des programmes, un programme ne peut envoyer d'informations à une application dont les privilèges seraient supérieurs.

L'exemple le plus concret est internet explorer 7 qui possède deux processus, un de bas niveau ieuser.exe avec des privilèges élevés, et le processus lié à iexplorer.exe avec des privilèges réduits qui gère l'interface graphique.

On le voit bien les modifications apportées aux systèmes Vista et Longhorn sont résolument tournées vers la sécurité.

Un lien intéressant sur le site de Microsoft France sur le développement **SDL** :

<http://www.microsoft.com/france/technet/securite/sdl.aspx>

D'une manière générale vous pouvez consulter le site sécurité pour plus d'informations :

<http://www.microsoft.com/france/technet/security/default.aspx>

Toutes ces modifications systèmes ne manqueront pas d'avoir un impact sur nos applications et futurs développements.

III - Session Longhorn Server de A à Z

Cette session a été riche en découvertes :

Le produit est encore en bêta et normalement disponible pour la fin de cette année.

Les plus résumés en quelques lignes :

Nouveau noyau 6.0 et partage un bonne partie du code de **Vista**.

La sécurité à été renforcée comme pour **Vista** .

La console de gestion du serveur à été revue

L'intégration de Windows Power Shell le nouveau langage de script très puissant de Microsoft.

Pour les aficionados de la ligne de commande **LongHorn** sera disponible en version core.

Dans ce mode pas de bureau, juste une boite de commande MSDOS et la possibilité d'appeler Notepad 2.0.

Toute la configuration du serveur ce fera avec **Windows Power Shell** .

Intégration de Nap : Network Access Protection :

Le principe étant de définir au niveau d'un pc Mobile client un bulletin de santé nécessaire à l'accès du réseau d'entreprise.

Ce bulletin sera paramétrable et on pourra demander :

- La présence du parefeu .
- La présence d'un anti-spyware.
- Un anti-virus à jour.
- La présence des mises à jour de sécurité

Si le bulletin n'est pas valide le client n'aura pas accès au réseau de l'entreprise.

Pas mal pensé !, j'ai hâte de tester ça.

Autre point important, la refonte du service Terminal Serveur :

La modification la plus important à mes yeux est la possibilité d'exécuter sur les postes de travail clients les programmes hébergés par le serveur comme si c'était des applications locales et non pas via le bureau virtuel.

Le procédé devient invisible pour les utilisateurs.

Intégration d'une passerelle Terminal Serveur :

La passerelle permettra d'accéder aux terminaux et programmes distants de n'importe quels endroits en étant protégés par un pare-feu.

Le protocole RDP 6.0 est alors encapsulé dans une connexion HTTPS.

Nouvelles stratégies de sécurités :

Permettant en outre de bloquer l'utilisation de clef USB ou de périphériques externes sur le poste client.

Et enfin le dernier point que j'ai retenu :

Le retour des BDC (Backup Domain Controller) !

En effet Microsoft introduit le **RODC** Read Only Domain Controleur

Sa principale caractéristique importante est que sa base ne possède pas de mot de passe, il faudra donc une connexion réseau pour l'identification, il sera possible aussi de définir des exceptions pour des mots de passe à répliquer.

C'est évidemment un condensé, il y a encore plein de nouveautés dans cet OS serveur, décidément très prometteur en terme de fonctionnalités.

IV - C++ Mixed code : Migration VC6-VC8

Cette session était clairement pour les développeurs C++,

Bon le message est fort et clair, le c++ n'est pas mort vive le C++ !

Après un bref rappel sur le fait que VC8 est compatible ISO C++ à 98 %

Une présentation rapide sur le C++/CLI nous a été faite,

L'objectif est clair, démontrer que notre " vieux " code peut être intégré et réutilisé dans une application .Net en C++/CLI.

Les avantages ? :

Évolution vers .Net et interopérabilité avec les autres langages.

Possibilité d'écrire une DLL en mode /clr :safe pour des produits comme SLQ serveur par exemple.

Un autre aspect qui me concerne particulièrement l'intégration de Windows forms dans une application MFC.

La aussi la simplicité du procédé est confondante.

Avec un bref rappel que l'option CLR peut être spécifiée par fichier du projet ou sur l'ensemble du projet.

Pour finir la session Eric Vernier nous a fait une démonstration d'un petit projet sur lequel il a travaillé :

La possibilité d'utiliser WPF dans une application MFC !!!

Il se pourrait bien que cette option soit intégrée dans le futur Visual Studio, à suivre, ce n'est pour l'instant qu'une initiative personnelle d'Eric Vernier.

Donc au final un bilan pour ma part assez positif : évolution du code, interopérabilité avec les autres langages .net , évolutions et ouverture des MFC accru sur le monde .net .

Le C++ et les MFC ont encore de beaux jours devant eux.

Liens intéressants :

<http://nico-pyright.developpez.com/tutoriel/vc2005/migrationcppnet/>

<http://nico-pyright.developpez.com/tutoriel/vc2005/interop/>

<http://dotnet.developpez.com/faq/cppcli/>

V - Introduction au HPC pour les développeurs

HPC ou le : High Performance Computing risque de devenir un enjeu important dans les années à venir.

Explications :

Dans les années qui ont suivi, la course à la fréquence sur les microprocesseurs n'a cessé d'augmenter la performance brute de nos applications et ce, sans changement dans le code.

Mais cette vitesse ne peut augmenter indéfiniment à cause de la surchauffe excessive du composant.

Sont ainsi apparues de nouvelles techniques visant à augmenter la puissance,

- Les machines multiprocesseurs souvent réservées aux serveurs.
- Plus récemment l'hyperthreading qui consiste à intégrer au sein du même microprocesseur plusieurs unités logiques.
- Enfin les multi-c#urs : plusieurs microprocesseurs physiques intégrés dans une seule puce.

C'est cette dernière technologie qui pour l'instant semble la plus prometteuse, et qui est développée activement par les deux fondeurs Intel et AMD.

Maintenant que le décor est planté, quelles sont les implications du coté de nos applications ?

Globalement les machines supportent beaucoup mieux le fait d'avoir plusieurs applications qui tournent en même temps, le niveau de performance obtenue est supérieur à celui d'une machine monoprocesseur.

En résumé elles tiennent mieux la charge.

Mais qu'en est-il pour une seule application ?

Hé bien c'est là que le bât blesse, une application classique mono thread ne tirera pas partie de cette architecture et ne tournera effectivement que sur un microprocesseur physique ou virtuel.

Alors pour tirer parti de cette architecture nos applications devront être multithread.

La cause est entendue mais la réalisation est délicate, entraînant des contraintes de coordinations des différents threads, ou tout simplement de partage de données entre thread.

Dans ce contexte plusieurs solutions visant à simplifier la programmation multithread existent

Comme **OPEN MP** qui permet de la parallélisation explicite.

C'est cette dernière méthode qui nous a été démontrée avec pour exemple le calcul du nombre PI#

Le principe de cette méthode est de placer directement des directives dans le code source pour indiquer les parties de code qui devront s'exécuter de manière parallèle.

Avec une démonstration à l'appui sur deux machines en cluster, du débogage distant dans Visual studio 2005 sur les machines du réseau !

Le résultat avec les outils liés à Visual Studio 2005 sont convaincants et performants.

L'impact sur le code est assez minime, mais à mon avis réservé à de petites portions de code ou à des programmes faisant beaucoup de calculs.

De ce fait je ne pense pas pouvoir appliquer cette technique dans mes applications courantes.

Une autre implémentation qui semble prometteuse est celle que **Herb Sutter** implémente dans le C++/CLI comme plateforme de test avec le projet **concur**.

Celui-ci permet de simplifier la mise en #uvre de code parallèle à l'aide d'un jeu de mots clefs venant étendre le langage#

Quand le projet sera validé, l'ensemble des langages .net disposeront de ces extensions...

A suivre.

Pour en savoir plus:

<http://chergui.developpez.com/cours/parallelisme/livre-openmp/>

<http://www.microsoft.com/france/Vision/PodcastMsdn.aspx?EID=d470a172-276e-4d3d-bd99-fb3bbd4510cf>

Création de Thread Win32 et MFC:

<http://c.developpez.com/faq/vc/?page=ProcessThread#MakeWorkerThread>

VI - Tout sur le nouveau fichier de format OpenXML

Le Nouveau format de Microsoft pour la gestion des documents tient ses promesses,

Cette session a été l'occasion de faire le point sur ce nouveau format et sur ce qu'il apporte.

En résumé :

- Gain de place de l'ordre de 50% en moyenne par rapport à un fichier binaire, effectivement les fichiers docx (par exemple) utilisent le format zip pour stocker les informations.

Quand on voit la place que la bureautique prend dans un serveur d'entreprise le gain n'est pas négligeable.

A ce sujet on pourra disposer d'un outil de migration en masse des documents situés sur un disque etc..

L'outil est encore en bêta d'après ce que j'ai compris.

- Accès relativement simple aux différents éléments constituant le fichier:

Celui-ci étant au format zip il est très aisé de le renommer, et d'accéder aux différents éléments le constituant.

La démonstration en a été faite pendant cette session, l'animateur a successivement changé les images ou encore la feuille de style associée à un document Word.

- Fiabilité accrue: qui n'a pas déjà eu un fichier .doc inutilisable ?

Le nouveau format apporte beaucoup plus de robustesse, si une des parties est endommagée : image, feuille de style etc, le fichier restera exploitable.

- Un format ouvert: comme il a été rappelé le format OpenXML a été soumis et standardisé par l'Ecma, un travail est cours pour la normalisation ISO.

De plus le format sera utilisable via une licence sans royalties.

Les programmeurs open source pourront aussi travailler avec OpenXML sans crainte.

- Interopérabilité entre les formats de documents :

Différents outils permettent aux utilisateurs d'office d'échanger des fichiers avec les différents formats de Word 2003 ou XP et d'utiliser le nouveau format OpenXML,

Ou d'échanger des documents avec le format ODF .

Ainsi sur SourceForge on trouve un add-in permettant la compatibilité entre ODF et Word.

<http://openxmldeveloper.org/archive/2007/02/02/1172.aspx>

Vous utilisez encore office 2003? qu'à cela ne tienne, il existe un pack de compatibilité Microsoft Office pour les formats de fichier Word, Excel et PowerPoint 2007

<http://www.microsoft.com/downloads/details.aspx?displaylang=fr&FamilyID=941b3470-3ae9-4aee-8f43-c6bb74cd1466>

Dans un autre registre Novell annonce la sortie d'une version d'OpenOffice.org disposant d'un composant de compatibilité avec OpenXML permettant ainsi le support croisé des formats Office (Doc, XLS), ODF (OpenDocument) et Open XML.

Un autre point en faveur d'OpenXML par rapport à ODF :

C'est sa capacité d'accepter des schémas métiers utilisateur pour remplir un document.

Enfin un élément important qui n'a pas été cité lors de cette session mais qui me vient à l'esprit :

Outre le fait que les développeurs vont pouvoir générer automatiquement des documents, ils auront la possibilité de ré-exploiter les informations contenues dans ces documents avec leurs applications, ce qui me semble très intéressant.

Le Site Microsoft sur OpenXML :

<http://www.microsoft.com/france/msdn/office/openxml/default.mspix>

Le site de l'Ecma pour OpenXML :

<http://www.ecma-international.org/memento/TC45.htm>

Autres liens pour les développeurs :

<http://grandfather.developpez.com/articles/openxml/structure/>

<http://grandfather.developpez.com/articles/openxml/php/lecture/>

<http://badger.developpez.com/tutoriels/dotnet/creer-fichier-word-openxml/>

VII - Conclusions

Voilà, j'espère que ces résumés sur les différentes sessions auxquelles j'ai pu assister vous seront utiles et vous donnerons envie d'aller plus loin.

Par manque de temps je n'ai pas pu assister à plus de sessions orientées développeur.

C'est bien là mon seul regret...

Ceci dit, malgré une affluence énorme : plus de 4000 personnes pour la première journée !, l'organisation était à la hauteur : vestiaire, hôtesse d'accueil, sac repas à midi, pack d'informations,

Tout ça pour un événement gratuit Bravo.