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

Groupe sécurité Windows de l'OSSIR
7 février 2005

Le principe du moindre privilège appliqué aux systèmes Windows

Jean-Baptiste Marchand
<Jean-Baptiste.Marchand@hsc.fr>

Plan

- Rappels sur le modèle de sécurité de Windows NT
- Pourquoi le modèle de sécurité de Windows NT est trop souvent inopérant ?
- Techniques pour restreindre le contexte de sécurité
- Outil runas
- Limites
- Références

Rappels sur le modèle de sécurité

- Modèle de sécurité de Windows NT : 3 services de sécurité
  - Authentification
  - Autorisation
  - Audit
- Authentification, réalisée localement ou de façon distribuée, à l'aide de protocoles d'authentification réseau (NTLM, Kerberos V)
- Autorisation
  - Contrôle d'accès, via des DACL (ACL discrétionnaires)
  - Vérification des privilèges, lorsque le contrôle d'accès n'est pas adapté
- Audit
  - Traçabilité des événements sécurité à destination de l'administrateur
  - Après définition d'une politique d'audit sécurité (9 catégories dans les systèmes Windows récents)

Service d'authentification

- L'authentification se déroule en 3 phases
  - Validation de l'authentification
  - Création d'une session de connexion (*logon session*)
  - Création du contexte de sécurité initial
- Validation de l'authentification
  - Locale si un compte local est utilisé (via la LSA locale, qui utilise la base SAM locale)
  - Via un contrôleur de domaine si un compte de domaine est utilisé

Sessions de connexion

- Différents types de sessions de connexion
  - Différentes façons d'utiliser un même compte
  - 5 types de sessions de connexion
    - Interactive, via le réseau, en tant que service, en tant que batch, via Terminal Services (à partir de Windows XP)
  - La LSA locale vérifie qu'un compte donné possède le droit de connexion adéquat pour établir un **type de session donné**
  - Exemples
    - Ex 1 : par défaut, seuls les administrateurs peuvent s'authentifier de façon interactive (via Winlogon) sur un contrôleur de domaine
    - Ex 2 : un compte de machine ne peut pas être utilisé pour s'authentifier interactivement, même si on connaissait le mot de passe d'un compte de machine SERVEUR$...
  - **Droits de connexion** apparaissent au même endroit que les **privilèges** Windows, ce qui est trompeur...

Sessions de connexion spéciales

- Un système Windows créé au démarrage des sessions de connexion spéciales
  - LOCALSYSTEM (0x3e7) et ANONYMOUS LOGON
  - NETWORK SERVICE (0x3e4) et LOCAL SERVICE (0x3e5), à partir de Windows XP
  - Il n'y a pas de compte *derrière* ces sessions de connexion !
- Outil : LogonSessions (Sysinternals)

Contexte de sécurité

- Dans Windows, le contexte de sécurité est défini par processus ou par thread
- Contenu dans une structure appelée un jeton (*security token*)
- Un jeton = des SIDs + des privilèges
  - SID viennent à la fois de la LSA locale (pour les groupes locaux) et de la LSA d'un contrôleur de domaine (groupes du domaine)
  - Privilèges viennent uniquement de la LSA locale car ils n'ont de signification que localement
- Les SID vont être examinés lors du contrôle d'accès
- Les privilèges, activés le temps d'être utilisés, donnent le droit de mener des actions d'administration spécifiques sur le **système local**
- Outil : onglet *Security* de Process Explorer (jeton des processus)

Contrôle d'accès

- Windows gère les ressources du système sous la forme d'objets, qui sont sécurisables via des descripteurs de sécurité
- Types d'objets classiques protégés par des permissions
  - Fichiers, clés de base de registre, tubes nommés, ...
  - Processus, Threads, ...
  - Outil : ZAccessMan
- Certains composants de Windows gèrent également de la sécurité sur des objets privés (pas des objets du système d'exploitation)
  - Permissions sur les services gérées par le SCM
  - Contenu de la SAM, objets de la LSA, entrées dans Active Directory, partages de fichiers, ...

Privilèges

- Privilèges utilisés lorsque le contrôle d'accès ne convient pas
  - Ex : privilège de sauvegarde donné par défaut à *Backup operators*
- Exemples de privilèges critiques pour la sécurité
  - Chargement d'un pilote de périphériques (*driver*)
  - Déboggage d'un processus n'appartenant pas à l'utilisateur
  - Prendre la possession d'un objet
  - Sauvegarde et restauration (permet de contourner totalement le contrôle d'accès)
- Les membres du groupe administrateurs ont tous les privilèges sur le système local
- Quelques groupes applicatifs dont l'existence est prévue pour l'affectation des privilèges (*Backup operators*, *Server operators*, *Print operators*, ...) ou les ACLs par défaut (*Network Configuration Operators*)

Pourquoi le modèle de sécurité de Windows est trop souvent inopérant

- Si l'utilisateur est administrateur (ou membre d'un groupe administratif telle que *Power users* ou *Backup operators*)
  - Le contrôle d'accès ne joue peu ou plus son rôle
  - Une grande partie ou tous les privilèges sont affectés
- Un code malveillant s'exécutant sous l'identité d'un utilisateur ayant un contexte de sécurité élevé peut, par exemple :
  - Installer un nouveau service et exécuter du code sous l'identité LOCALSYSTEM
  - Charger un pilote de périphériques pour installer un rootkit noyau
  - Arrêter n'importe quel service (ex : suite de sécurité type anti-virus)
  - Lire les accréditations stockées en cache dans un domaine Windows et tenter de découvrir des comptes du domaine
  - ...

D'où vient cette facheuse manie d'être administrateur local ?

- Applications ne fonctionnant pas dans un contexte de sécurité non privilégié
  - Les développeurs Windows sont (étaient ?) traditionnellement administrateurs de leur poste, de sorte que l'application est "prévue" pour fonctionner dans un contexte privilégié
  - Absence de compréhension du modèle de sécurité de Windows NT de la part des développeurs
  - En train d'évoluer progressivement
    - http://pluralsight.com/wiki/default.aspx/Keith.GuideBook.HowToDevelopCodeAsANonAdmin
- Outils de diagnostic pour les applications qui ne fonctionnent pas correctement dans un contexte non privilégié
  - Filemon, Regmon, Process Explorer, ...
  - Ajustement des permissions peut résoudre le problème mais tâche pouvant s'avérer fastidieuse

Techniques pour restreindre le contexte de sécurité

- Ne plus être administrateur local et utiliser runas pour lancer des programmes sous une autre identité, typiquement plus privilégiée
- Rester administrateur mais utiliser des jetons restreints pour lancer des applications à risque (à partir de Windows 2000)
  - Ex : navigateur web ou client de messagerie
  - Permet de "désactiver" des SID dans un jeton (*deny-only* SID), de supprimer des privilèges ou spécifier des SID restreignant (*restricting* SIDs)
  - Option *Run this program with restricted access* (W2K3) ou *Protect my computer and data from unauthorized program activity* (WXP)
  - *Software Restriction Policies*, disponible à partir de Windows XP, qui permet d'imposer cette configuration pour des exécutables du système
  - Ex : DropMyRights de Michael Howard

Illustration du modèle de sécurité avec une identité non administrateur

- Un utilisateur uniquement membre du groupe *Utilisateurs* ne peut :
  - Pas modifier la configuration réseau
  - Pas manipuler les processus qui ne lui appartiennent pas
  - Lire uniquement une partie restreinte de la base de registres
  - Pas manipuler la plupart des services
  - ...
- Ilustrations

Outil runas

- runas
  - programme fourni à partir de Windows 2000, permettant de lancer un programme sous une identité différente de celle de l'utilisateur connecté
  - Repose sur un service (*Secondary Logon*), qui tourne SYSTEM et gère la création d'une nouvelle session de connexion
  - Nécessite la connaissance du login et du mot de passe du compte utilisé pour ouvrir la nouvelle session
- Utilisations
  - runas.exe en ligne de commande
    - Options /profile (défaut) et /noprofile, /savecred (déconseillé), /netonly
  - Shift+right click sur un exécutable dans l'explorateur Windows
    - *Option Run this program with restricted access* permettant de lancer le processus avec un jeton restreint

Utilisation de runas en pratique

- Une instance de cmd.exe lancée en tant qu'administrateur permet de
  - Gérer le système à l'aide des outils en ligne de commande
    - net.exe, sc.exe, wmic.exe, netsh.exe, ...
  - Possibilité de lancer des consoles d'administration MMC pour administrer le système local
    - services.msc, eventvwr.msc, gpedit.msc, ...
- Cas spécifique de l'Explorateur Windows
  - Problème : la tentative de lancement d'une nouvelle instance de l'Explorateur Windows via explorer.exe affiche un explorateur dans le contexte de sécurité de l'utilisateur authentifié interactivement
  - Solution : activer l'option *Launch folder windows in a separate process* **dans le profil du compte cible** (typiquement, *administrator*)
  - Permet d'utiliser l'interface de l'explorateur Windows (modification d'ACL, entrées du panneau de contrôle tel que ncpa.cpl, ...)

Extension à l'Explorateur : PrivBar

- PrivBar est une extension à l'explorateur Windows qui met en évidence le contexte de sécurité sous lequel tourne une instance d'explorer.exe
  - Permet de visualiser rapidement le contenu du jeton de l'instance explorer.exe
  - Signalétique permettant de distinguer rapidement les instances et leurs contextes de sécurité
- A télécharger à partir du blog d'Aaron Margosis
  - http://blogs.msdn.com/aaron_margosis/archive/2004/07/24/195350.aspx

Limites

- Attaques de type Shatter
  - Attaques sur l'interface graphique, permettant, au moins en théorie d'exploiter les fenêtres affichées dans la Winstation de l'utilisateur non privilégié pour exécuter du code dans un contexte de sécurité plus élevé

Références

- The Non-Admin blog - running with least privilege on the desktop
  - http://blogs.msdn.com/aaron_margosis/
- Local Administrator / Power User Hall of Shame
  - http://www.threatcode.com/
- Browsing the Web and Reading E-mail Safely as an administrator
  - http://msdn.microsoft.com/library/en-us/dncode/html/secure11152004.asp
  - http://msdn.microsoft.com/library/en-us/dncode/html/secure01182005.asp
- AppVerifier (SecurityChecks, LUA Privilege Predictor)
  - http://www.microsoft.com/technet/security/secnews/articles/sec_tools_for_appverifier.mspx

Références : modèle de sécurité

- Modèle de sécurité de Windows NT
  - *Programming Windows Security*. Keith Brown. Addison-Wesley
  - *The .NET Developer's Guide to Windows Security*. Keith Brown.
    - http://pluralsight.com/wiki/default.aspx/Keith.GuideBook.HomePage
  - Improving the Granularity of Access Control for Windows 2000. Mike Swift and Al.
    - http://www.cs.washington.edu/homes/mikesw/papers/tissec.pdf
  - Improving the Granularity of Access Control in Windows NT. Mike Swift and Al.
    - http://www.cs.washington.edu/homes/mikesw/papers/win2kacl.pdf
  - Modèle de sécurité des systèmes Windows (HSC)
    - http://www.hsc.fr/ressources/articles/mod_sec_win/