Protéger ses applications contre l'élévation de privilèges

Nicolas RUFF
Ingénieur Chercheur en sécurité
EADS-CCR

Introduction

- Les enjeux de la sécurité locale
  - Qui peut vouloir élever ses privilèges ?
    - Les virus exécutés par des utilisateurs non administrateurs
    - Les utilisateurs des terminaux en libre service
    - Les utilisateurs de clients légers (type Terminal Server)

- Dans l'idéal, l'utilisateur ne devrait pas être administrateur local de son poste
  - Microsoft en a pris conscience et travaille sur des solutions
    - Commande RunAs
    - Jeton "Restricted"
    - Groupe "network operators" pour les nomades
    - Futur Visual Studio 2005
  - Rappel : il n'est pas nécessaire d'être administrateur local pour déboguer un processus !

Objectifs de la présentation

- Présenter les erreurs "classiques" dans les applications
  - Méthodes d'élévation de privilèges
    - Utilisateur -> Administrateur
    - Administrateur -> SYSTEM
  - Avec des exemples issus de cas réels
- Donner des solutions
  - En général simples
  - Mais requièrent la plus grande attention
    - Un simple guillemet oublié peut ouvrir une faille !

Plan

- Hypothèses
- Techniques d'élévation de privilèges
  - L'interface graphique
  - Les interfaces de communications
    - Canaux nommés
    - LPC
    - Divers
    - Recommandations générales
  - L'ordre de recherche des exécutables
- Conclusion
- Bibliographie
- Remerciements

Hypothèses

- **Les attaques physiques sont exclues de l'étude**
  - **Si l'utilisateur peut redémarrer le poste sur un support externe, il pourra en prendre le contrôle**
  - **Exemples**
    - **Supports bootables : CD, disquette, clé USB, disque FireWire, …**
    - **Outils : BartPE, ERD Commander, Auditor, …**
- **Le système est protégé "à l'état de l'art"**
  - **En particulier, à jour des correctifs de sécurité**

Attaques
L'interface graphique

Attaques
L'interface graphique (1/10)

- **Contexte**
  - Une famille d'attaques appelée "shatter attacks"
  - "Découverte" en 2002 (?)

- **Conditions d'existence**
  - Un "objet" (en général graphique) …
    - … appartient à un processus privilégié (ex. SYSTEM)
    - … accepte des messages provenant d'un processus utilisateur
    - … crée des processus ou redirige son flot d'exécution en fonction du contenu d'un message
      - Volontairement (ex. WM_TIMER)
      - Involontairement (ex. "buffer overflow")

Attaques
L'interface graphique (2/10)

- **Exemple n°1 (trivial mais vrai)**
  - Un processus SYSTEM affiche une popup
  - L'utilisateur presse F1
  - Dans le menu d'aide il choisit "Jump to url … CMD.EXE"
  - Le shell qui apparaît est SYSTEM
- **Cf. UTILMAN.EXE, qui peut être invoqué en appuyant 5 fois sur la touche "shift"**
  - Corrigé par MS03-025

Attaques
L'interface graphique (3/10)

- **Exemple n°2 : les fonctions de callback**
  - **Message WM_TIMER envoyé à une fenêtre**
    - **WPARAM TimerID, LPARAM lpfnCallback**
  - **La fonction de callback est appelée avec les droits de la cible**
    - **Sauf si le message est intercepté par l'application (rare)**
  - **Comportement corrigé par MS02-071**
    - **Mais il existe d'autres messages utilisant des fonctions de Callback**
    - **Ex. LVM_SORTITEMS, LVM_SORTITEMSEX, EM_SETWORDBREAKPROC**

Attaques
L'interface graphique (4/10)

- **Exemple n°3 : les pointeurs de données**
  - Certains messages fournissent un pointeur vers la zone de sortie des résultats
  - Il est alors possible d'écrire dans la mémoire avec les droits de la cible
  - Ex. message HDM_GETITEMRECT
    - lParam pointe vers une structure de type RECT
  - Attaque réelle sur les styles Windows XP
    - "CommCtrl 6.0 Button Shatter Attack"
  - Cette attaque est en général "multi-stages"
    - Un message met en place la structure cible
    - Un autre message provoque l'écriture

Attaques
L'interface graphique (5/10)

- **Exemple n°4 : pointeurs de fonction dans des zones de données "user defined"**
  - Exemple : les données privées d'une fenêtre
    - API SetWindowLong**() / SetWindowLongPtr()**
    - **Message GWL_USERDATA**
  - Le service Messenger est vulnérable
    - Corrigé par MS04-032
  - **L'API AfxOldWndProc423() serait également vulnérable**
    - **Utilisée pour sous classer des fenêtres non-MFC dans une fenêtre MFC**

Attaques
L'interface graphique (6/10)

- Le problème global des objets graphiques
  - Ils appartiennent au bureau de l'utilisateur
    - … qui peut modifier leur comportement
    - … qui peut leur envoyer des messages

- Il ne faut pas leur faire confiance !
  - Les cases grisées peuvent être réactivées
  - Les fenêtres masquées peuvent être affichées
    - Ex. contournement du mot de passe dans KAV v5.0.149, v5.0.153
  - Le contenu du texte masqué peut être obtenu
    - Ex. outils Revelation, Asterisk Logger
  - La taille limite de texte peut être modifiée
    - Ex. un message WM_USER envoyé au service "Still Image" provoque un "buffer overflow" (corrigé par MS00-065)

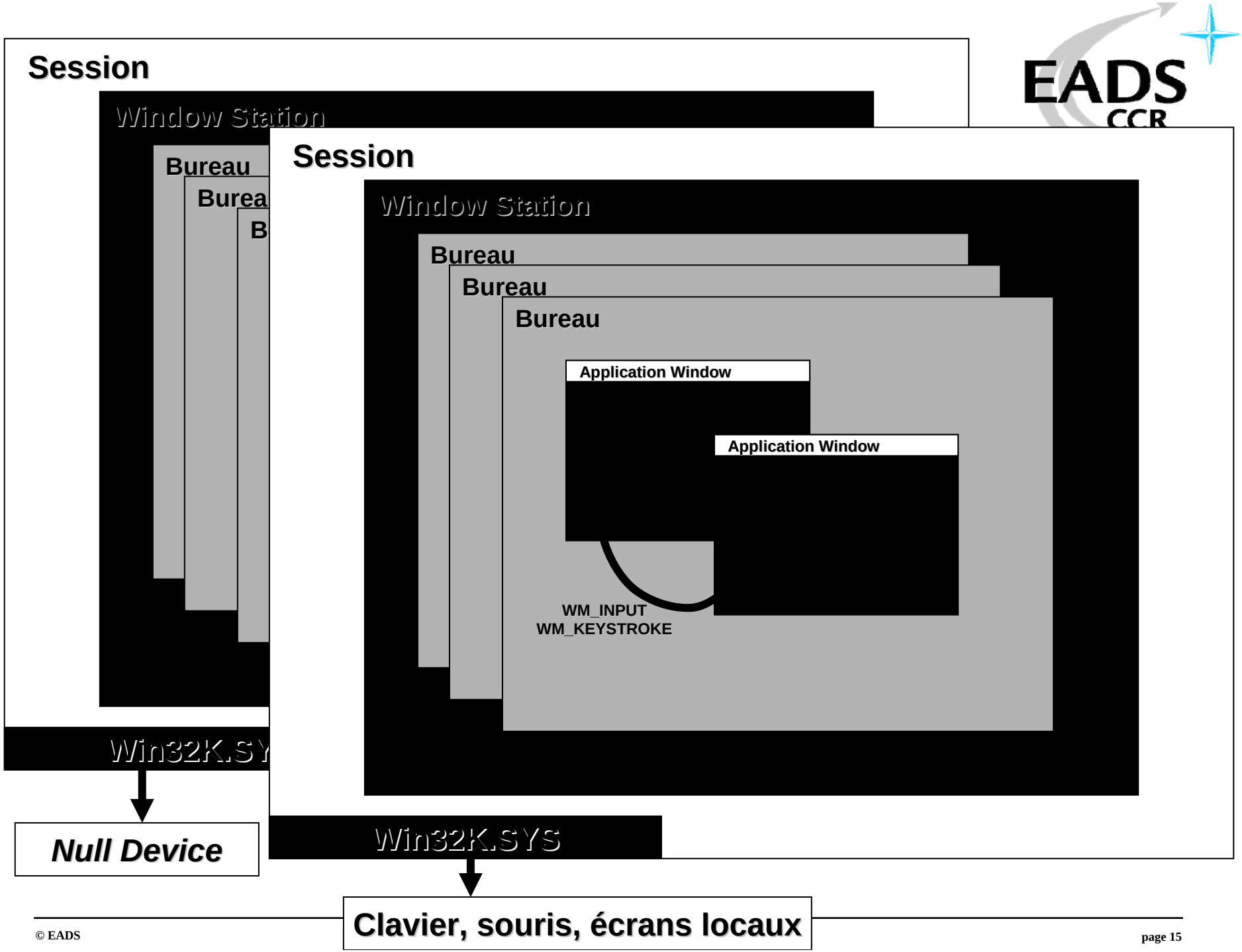
Attaques
L'interface graphique (7/10)

- De nombreuses applications sont concernées
  - Kerio Personal Firewall 2.1.4
  - Sygate Personal Firewall Pro 5.0
  - McAfee VirusScan 7.0
  - WinVNC 3.3.6
  - ...

- Y compris des applications Microsoft
  - Le fameux UTILMAN
  - NetDDE crée une fenêtre appartenant à WINLOGON
  - La fenêtre "MM Notify Callback" appartient à WINLOGON
  - Le service "Messenger" crée une fenêtre appartenant à CSRSS
    - Ex. net send localhost "hello" affiche une fenêtre SYSTEM

Attaques
L'interface graphique (8/10)

- **Une complexité supplémentaire : les sessions multiples**
  - **Principe : accès graphique simultané par plusieurs utilisateurs**
  - **Utilisé par :**
    - **"Remote Desktop"**
    - **"Terminal Server"**
    - **"Fast User Switching"**
    - **Outils tiers tels que "Superior SU"**
  - **Implémentation :**
    - **Fenêtre applicative ∈ Bureau ∈ *Window Station* ∈ Session**


Session
Window Station
Bureau
Session
Window Station
Bureau
Bureau
Bureau
Application Window
Application Window
WM_INPUT
WM_KEYSTROKE
Win32K.SYS
Null Device
Win32K.SYS
Clavier, souris, écrans locaux

Attaques
L'interface graphique (9/10)

- **Session**
  - **Conserve l'état global de l'interface graphique**
  - **Reçoit les entrées/sorties**
- ***Window Station*** **(**ex. winsta0)
  - **Gère le presse-papiers**
  - **Une seule *Window Station* est active sur la console à un instant T**
- **Bureau**
  - **Plusieurs bureaux existent simultanément**
    - **Ex. applications, SAS (**winsta0\winlogon**), écran de veille**
  - **Seules les fenêtres du même bureau peuvent communiquer entre elles**

- **Outil**
  - **LogonSessions de SysInternals**

Attaques
L'interface graphique (10/10)

- **Recommandations**
  - **Les objets graphiques doivent être créés dans le contexte de sécurité de l'utilisateur**
    - **Principe de séparation des privilèges**
    - **Attention au RevertToSelf()**
  - **Ne pas envoyer de données sensibles dans l'espace utilisateur**
    - **Mots de passe masqués**
    - **Fenêtres masquées**
  - **Ne pas rendre les services "interactifs" sauf exception**
    - **Rappel : contrôle de l'interactivité**
      - **Par service : flag SERVICE_INTERACTIVE_PROCESS (0x100)**
      - **Globalement : par la clé HKLM\SYSTEM\CCS\Control\Windows\NoInteractiveServices**
      - **Vérification : par la commande "sc query type= interact"**
  - **Si besoin d'interactivité, utiliser un bureau différent**

Attaques
Les interfaces de communication

Attaques
Les canaux nommés (1/5)

- **Rappel : qu'est-ce qu'un canal nommé ?**
  - Canal de communication interprocessus (IPC)
    - Local ou distant (via SMB)
  - Implémenté sous forme de "filesystem driver"
    - Pilote NPFS.SYS
    - Accès via "\\<machine>\pipe\<nom de canal>"
    - Avantages
      - Sémantique identique à celle d'un système de fichiers
      - Bénéficie du système d'ACLs
  - Outils
    - Pas d'outil Microsoft
    - Outils tiers PIPELIST, PIPEACL

Attaques
Les canaux nommés (2/5)

- **Contexte**
  - Une famille d'attaques basée sur l'usurpation d'un canal nommé
  - "Découverte" en 2000 (?)
    - MS00-053 : canal nommé prédictible utilisé par le SCM

- **Conditions d'existence**
  - Un client avec un niveau de privilège élevé (ex. SYSTEM) se connecte sur un canal nommé
  - Ce canal n'existe pas initialement ou limite le nombre d'instances simultanées
  - Le nom du canal est prédictible
  - Le niveau d'impersonation n'est pas limité par le client
    - Rappel des niveaux de QoS :
      - Anonymous, Identification, Impersonation, Delegation
    - Ces niveaux sont passés dans le paramètre *dwFlagsAndAttributes* de CreateFile() par exemple

Attaques
Les canaux nommés (3/5)

- **Les services pouvant être démarrés par un utilisateur non privilégié sont des cibles intéressantes**
  - **Identifiables par la commande "sc sdshow"**
  - **Utilisent souvent des canaux nommés**
    - **cf. outils PIPELIST, PIPEACL**

- **Exemples réels exploitables via l'API ImpersonateNamedPipeClient()**
  - **MS00-053 : canal nommé prédictible utilisé par le Service Control Manager**
    - **\\.\pipe\net\NtControlPipeX**
    - **X est incrémenté à chaque démarrage de service**
  - **MS01-031 : canal nommé prédictible au démarrage du service Telnet**
  - CAN-2003-0496 : abus de la procédure 'xp_fileexist' dans SQL Server
    - **Permet d'obtenir les droits du processus SQL Server**
    - **Corrigé par l'ajout du privilège SeImpersonatePrivilege dans Windows 2000 SP4+ (cf. Q821546)**

Attaques
Les canaux nommés (4/5)

- **Ça se complique**
  - **La connexion à un canal nommé et l'ouverture d'un fichier utilisent la même API CreateFile()**
    - **Ex. la commande "runas cmd.exe" est traitée par le service RunAs**
    - **… donc la commande "runas \\.\pipe\hacker" permet (théoriquement) de gagner les droits SYSTEM !**
  - **Théoriquement l'impersonation n'est possible qu'après la première lecture/écriture**
    - **Différence entre \\.\pipe et \\machine\pipe : la contrainte est levée dans le deuxième cas !**

Attaques
Les canaux nommés (5/5)

- Le serveur vérifie-t-il le code de retour de ImpersonateNamedPipeClient ?
  - Si le client demande SecurityAnonymous, l'appel échoue
  - Dans ce cas le code continue à s'exécuter dans le contexte précédent (SYSTEM ?)
- Si un nouveau processus enregistre le même canal
  - Les connexions supplémentaires au delà du nombre d'instances simultanées sont servies par ce nouveau processus

■ Recommandations
- Ne pas tomber dans les pièges précédents
- Créer les instances avec le flag FILE_FLAG_FIRST_PIPE_INSTANCE
  - Windows 2000 SP2+, Windows XP
- Ne pas abuser du privilège SeImpersonatePrivilege
  - Windows 2000 SP4+ (Q821546), Windows XP SP2

Attaques LPC (1/2)

- **Quelques failles classiques dans les LPC**
  - **Droits d'accès trop laxistes**
  - **Vulnérabilités dans l'implémentation (noyau et clients)**
    - **Numéros de messages prédictibles**
    - **Appel à NtImpersonateClientOfPort() permettant de spécifier n'importe quel processus cible**
    - **Non utilisation du handle retourné par NtAcceptConnectPort()**
    - **Etc. etc.**
    - **Cf. travaux de Todd Sabin sur Windows NT4**

Attaques LPC (2/2)

- Vulnérabilités dans l'implémentation noyau des LPC
  - Différentes exploitations de l'API NtImpersonateClientOfPort()
  - Corrigé par MS00-003 et MS00-091
- Vulnérabilité applicative : NTLMSSP
  - Problème d'index non vérifié
  - Port NtLmSecuritySupportProviderPort
  - Corrigé par MS01-008
- Droits d'accès laxistes : DebPloit
  - Lors d'un déboguage, le Session Manager (SMSS.EXE) communique par LPC
  - Port DbgSsApiPort
  - Ce port possède une ACL incorrecte
    - "Tout le monde" peut s'y connecter avec ZwConnectPort(DbgSsApiPort)
  - Corrigé par MS02-024

Attaques Divers (1/1)

- **Les communications avec le noyau**
  - **Communication Ring3/Ring 0 via l'API DeviceIoControl()**
  - **En mode "Neither I/O", le buffer passé en paramètre peut être n'importe quelle adresse mémoire**
    - **Le driver doit opérer ses propres contrôles …**
  - **Exemple de pilote exploitable : NAV 2002**
  - **Cf. travaux de S. K. Chong**

- **Les espaces mémoire partagés**
  - API OpenFileMapping() / MapViewOfFile()
  - **Les droits d'accès à des espaces mémoire partagés peuvent être trop laxistes**
  - **Ces espaces sont parfois utilisés pour des communications Ring0/Ring3 (!)**
  - **Exemple : MS05-012 "COM Structured Storage Vulnerability"**
  - **Cf. travaux de Cesar Cerrudo**

- **Les RPC**
  - **Non présenté ici mais attention néanmoins …**

Attaques
Recommandations générales (1/1)

- **Recommandations**
  - **Vérifier que les points de communication (même non documentés) sont sécurisés**
    - **ACL (en local)**
    - **Authentification du client + ACL (à distance)**
    - **Dans le doute, ne pas utiliser les ACLs par défaut**
  - **Ne pas faire confiance aux données provenant d'un niveau de privilège inférieur**
    - **Même si ces données ne sont pas théoriquement "visibles" de l'utilisateur**

Attaques
La recherche d'exécutables

Attaques
La recherche d'exécutables (1/5)

- **Contexte**
  - **Nommer un fichier n'est pas une opération triviale …**
    - **Compatibilité 8.3**
    - **Problème des séparateurs (espaces, / vs. \)**
    - **Caractères localisés et noms Unicode (préfixe \\?\)**
    - **Liens symboliques et "hard links"**
    - **Alternate Data Streams**
  - **L'ordre de recherche des EXE et DLL peut être configuré**

- **Conditions d'existence**
  - **Un exécutable lancé avec des privilèges élevés (typiquement un service) est désigné par un chemin ambigu**
  - **Un exécutable lancé avec des privilèges élevés charge des dépendances "au mauvais endroit"**

Attaques
La recherche d'exécutables (2/5)

- **Erreurs classiques**
  - **Chemins contenant un espace**
    - **Ex. HKLM\SYSTEM\CurrentControlSet\Services\MonService**
      - **KO : ImagePath=C:\Program Files\Mon Service\MonService.exe**
      - **OK : ImagePath="C:\Program Files\Mon Service\MonService.exe"**
    - **Dans le premier cas, "C:\Program.exe" est exécuté s'il existe**
    - **Exemple d'application vulnérable : MS AntiSpyware Beta 1 ☺**
    - **La pire erreur serait d'installer Windows dans un chemin contenant des espaces ☺**
      - **Ex. "C:\Win XP"**

Attaques
La recherche d'exécutables (3/5)

- **Ordre de recherche des EXEs**
  - **Répertoire courant**
  - **%SystemRoot%\System32**
  - **%SystemRoot%**
  - **Reste du %PATH%**
  - **Note : sous Unix, ajouter "." dans le PATH de l'administrateur est considéré comme une faille de sécurité …**
- **D'où le problème des "chemins incomplets"**
  - **Ex. clé Shell = "Explorer.exe"**
    - **Lors du démarrage du système, %SystemDrive% est le répertoire courant**
    - **Si le fichier C:\Explorer.exe existe, il sera utilisé comme shell utilisateur par WINLOGON**
    - **Correctif** MS02-064 : le groupe "tout le monde" ne peut plus créer de fichiers dans la racine

Attaques
La recherche d'exécutables (4/5)

- **Ordre de recherche des DLLs**
  - **Si la clé "HKLM\System\CCS\Control\Session Manager\KnownDLLs" existe**
    - **%SystemRoot%\system32**
    - **Répertoire de l'exécutable**
    - **Répertoire courant**
    - **%SystemRoot%**
    - **%PATH%**
  - **Sinon**
    - **Répertoire de l'exécutable**
    - **Répertoire courant**
    - **%SystemRoot%\system32**
    - **%SystemRoot%**
    - **%PATH%**
  - **Cf. Q164501**

Attaques
La recherche d'exécutables (5/5)

- Ordre de recherche des DLLs (suite)
  - Clé SafeDllSearchMode (Windows 2000 SP3+)
    - Toujours rechercher %SystemRoot%\system32 et %SystemRoot% en premier
  - Fichier .exe.local dans le répertoire de l'application
    - Force le chargement des DLLs à partir du répertoire courant

- Recommandations
  - La plupart des problèmes se posent lorsque l'utilisateur peut *ajouter* des fichiers dans le répertoire courant de l'application

Conclusion

- Créer des applications exécutées sous un compte privilégié impose un devoir de rigueur
- Les pièges sont nombreux et méconnus
  - De plus, le comportement du système varie selon la version de Windows et sa configuration
- *Tous* les pièges n'ont pas été abordés ici
  - Le service AT lance des tâches sous le compte SYSTEM
  - L'installation de pilotes d'imprimante permet de charger des modules noyau
  - Etc.
- Mais j'espère que cette présentation vous sera profitable !

Bibliographie

- **Shatter Attacks**
  - **Shatter Attacks - How to break Windows**
    - **http://security.tombom.co.uk/shatter.html**
  - Shattering By Example
    - http://www.security-assessment.com/Papers/Shattering_By_Example-V1_03102003.pdf
  - **Win32 Message Vulnerabilities Redux**
    - **http://www.idefense.com/idpapers/Shatter_Redux.pdf**
  - **Shooting the Messenger**
    - **http://www.blackhat.com/presentations/bh-usa-04/bh-us-04-moore/bh-us-04-moore-whitepaper.pdf**

Bibliographie

- Canaux nommés
  - Discovering and Exploiting Named Pipe Security Flaws for Fun and Profit
    - http://www.blakewatts.com/namedpipepaper.html
    - Les travaux de Blake Watts sont la référence dans le domaine
  - Pipes
    - http://msdn.microsoft.com/library/default.asp?url=/library/en-us/ipc/base/pipes.asp
- RPC
  - Writing a Secure RPC Client or Server
    - http://msdn.microsoft.com/library/en-us/rpc/rpc/writing_a_secure_rpc_client_or_server.asp

Bibliographie

- LPC
  - http://www.bindview.com/Support/RAZOR/Advisories/2000/adv_NTPromotion.cfm
  - http://www.bindview.com/Support/RAZOR/Advisories/2000/LPCAdvisory.cfm
  - http://www.bindview.com/Support/RAZOR/Advisories/2001/adv_NTLMSSP.cfm

- DeviceIoControl()
  - http://www.bellua.com/bcs2005/asia05.archive/BCSASIA2005-T04-SK-Windows_Local_Kernel_Exploitation.ppt
  - http://sec-labs.hack.pl/papers/win32ddc.php

  - Exemple NAV 2002
    - http://www.scan-associates.net/papers/navx.c
  - En chinois dans le texte ☺
    - http://www.xfocus.net/articles/200306/545.html

Bibliographie

- **Espaces mémoire partagés**
  - **http://www.bellua.com/bcs2005/asia05.archive/BCSASIA2005-T05-Cesar-Windows_IPC_Exploitation.ppt**
- **Writing Secure Code, 2nd Ed.**
  - **Michael Howard, David LeBlanc**
  - **http://www.microsoft.com/mspress/books/5957.asp**