Introduction C Windows

Ecole de cybersécurité

Introduction

- Architecture de windows
- Exemple de programme C

Architecture Windows

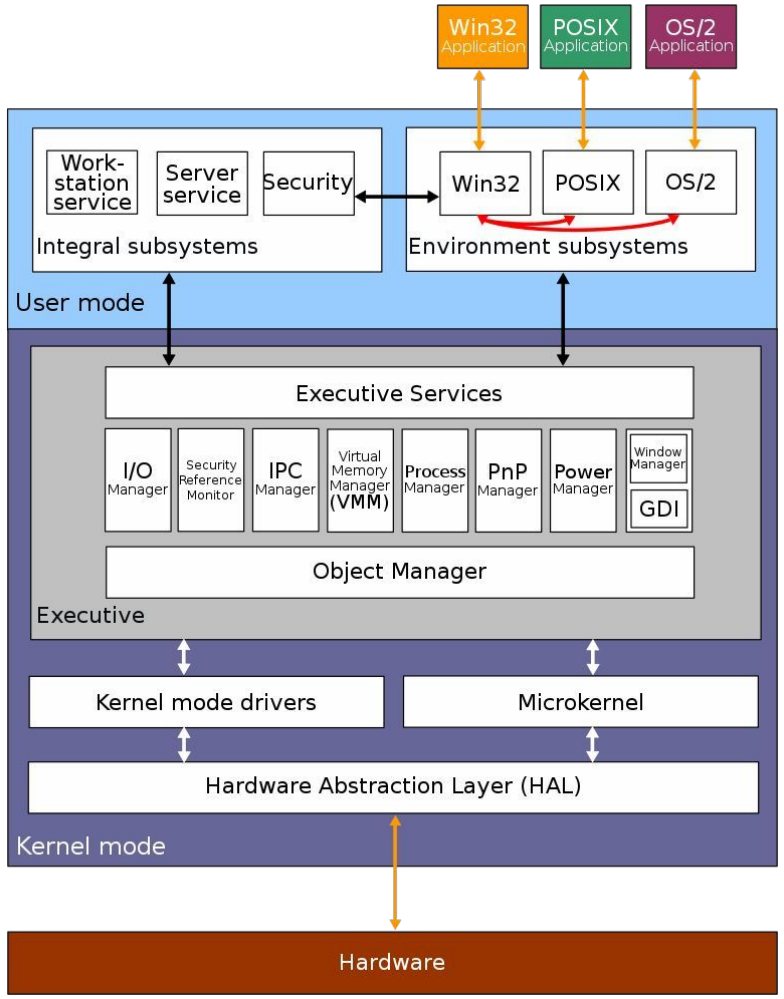
Modèle de système d’exploitation :

Multi-tâche, Multi-utilisateur, Multi-processeur, Multi-plateforme.

Programmation événementielle :

Paradigme de programmation fondé sur les événements.

Architecture de windows



Architecture de windows

Différentes couches:

- Sous-systèmes d’environnement
- Exécutif
- Noyau
- Couche d’abstraction matérielle
- Pilote de périphérique

Architecture de windows

Sous-systèmes d'environnement :

> Processus en charge de restituer auprès des applications utilisateur un environnement d'exécution dédié, incluant une ligne de conduite visuelle, une combinaison de caractéristiques programmatiques, et, in fine, une version spécialisée des services fondamentaux du système.

Exemples de sous-système : Windows 32/64 bits (mode graphique et à interface caractère), Posix, OS/2, XBOX, et d'autres.

Architecture de windows

Ecole de cybersécurité

Exécutif :

Sur-ensemble du noyau Windows intégrant les services de base qu'offre le système d'exploitation, tels que le gestionnaire de processus et de threads, le gestionnaire de mémoire virtuelle, le moniteur de références de sécurité, le système d'E/S et le gestionnaire de cache.

Architecture de windows

Noyau :

Supervise la manière dont le système d'exploitation s'alimente auprès des ressources incorporées à l'ordinateur (essentiellement le ou les processeurs et la mémoire). Le noyau fournit l'ordonnancement et la ventilation des threads, le traitement des interceptions (trappes, interruptions et exceptions), et la synchronisation multi processeur.

Architecture de windows

Ecole de cybersécurité

Couche d’abstraction matérielle :

Fournit l’interface de bas niveau vers la plateforme matérielle sur laquelle Windows s’exécute, isolant de la sorte le noyau des caractéristiques internes de l’ordinateur (interfaces d’E/S, contrôleurs d’interruption, mécanismes de communication multi processeur et autres).

Architecture de windows

Pilote de périphérique :

Module mode noyau chargeable qui fait l'interface entre le système d'E/S et le matériel concerné (ou éventuellement un autre pilote). Les pilotes de périphériques sous Windows ne manipulent pas directement les périphériques matériels, mais s'en remettent pour ce faire à la couche d'abstraction prévue à cet effet.

Processus système fondamentaux

Processus d'ouverture de session :

Processus mode utilisateur, exécutant **Winlogon**.exe, chargé de capturer le nom et le mot de passe de l'utilisateur, de les relayer (pour vérification) à l'autorité de sécurité locale (**Lsass**.exe) et de créer le processus initial de la session utilisateur.

Sous-système Windows :

Implémente le code mode utilisateur du sous système d'environnement Windows (**Csrss**.exe). Gère les fenêtres et les éléments graphiques de Windows.

Exécutif

Gestionnaire de processus :

Composant de l’exécutif en charge de superviser le cycle de vie des processus et des threads, depuis leur création jusqu’à leur démantèlement.

Gestionnaire de mémoire :

Composant de l’exécutif qui implémente la mémoire virtuelle paginée à la demande, en donnant à chaque processus l’illusion de disposer d’un très large espace d’adressage, et d’en être de surcroît le seul détenteur.

Exécutif

Gestionnaire de configuration :

Composant de l’exécutif responsable du de la mise en oeuvre et du suivi du Registre Windows.

Gestionnaire d’alimentation :

Composant de l’exécutif qui coordonne les événements électriques de l’ordinateur et génére des notifications d’E/S de gestion électrique aux pilotes de périphériques concernés.

Exécutif

Gestionnaire d’objets :

Composant de l’exécutif chargé de créer, protéger, suivre et supprimer les objets.

Moniteur de références de sécurité (SRM, Security Reference Monitor) :

Composant de l’exécutif en charge du maintien et de la stricte application des stratégies de sécurité sur l’ordinateur local. Il sécurise les ressources du système d’exploitation, en assurant la protection et l’audit des objets.

Exécutif

Système d’E/S :

Composant de l’exécutif responsable de l’acheminement des requêtes d’E/S aux pilotes ou périphériques d’E/S.

Concepts clés: Objets et handles

Objet :

Structure de donnée centralisant les propriétés définitoires et les caractéristiques actuelles d'une ressource. Chaque processus Windows est par exemple représenté par un bloc EPROCESS, lequel contient toutes les informations de contrôle requises pour la gestion de tels éléments.

Concepts clés: Objets et handles

Handle :

Identificateur d’objet. Un processus reçoit un handle vers un objet lorsqu’il le crée ou l’ouvre.

ex: **CreateFileW** retourne un HANDLE pas un “file descriptor”

Interfaces de programmation

MSDN:

- Vue globale
- Application de bureau
- Par Header C

3 Groupes d’API:

- Windows
- Native
- Noyau

Interfaces de programmation

API Windows :

Interfaces de programmation 32/64 bits intégrées aux systèmes de la gamme Microsoft Windows. Couvrant un large éventail de fonctionnalités, l'API Windows constitue la manière privilégiée pour du code mode utilisateur de solliciter le système en vue d'une tâche prédéfinie :

- créer un nouveau processus (**CreateProcess**)
- ouvrir un fichier (**OpenFile**)
- émettre une requête d'E/S (**DeviceIoControl**), etc.

Interfaces de programmation

API native :

Services natifs du système d'exploitation appelable depuis le mode utilisateur. Par exemple, le service interne sur lequel s'appuie la fonction Windows **OpenFile** pour ouvrir un fichier est **NtOpenFile**.

API noyau :

Sous-routines appelables exclusivement à partir du mode noyau.

Quelques remarques sur le STYLE de programmation windows

- Notation Hongroise

  p, b, lp, sz, … plus de détail ici

  pour dénoter le type avec le nom de la variable
- Typedef

  UINT, LPCSTR, … chaque type est redéfinie de manière explicite
- Fake Qualifier

  IN, OUT, CALLBACK, APIENTRY, WINAPI juste une macro (ex: #define IN) pour dénoter l'intention d'usage du paramètre

Processus et threads

Processus :

Définie l’espace d’adressage virtuel (i.e. PDBR, PDE, PTE) et données associées de contrôle nécessaires pour l’exécution d’un ensemble de threads.

Processus et threads

Thread :

Entité d’un processus que Windows ordonnance pour l’exécution. Windows emploie pour l’ordonnancement des threads un schéma adaptatif par priorités : les threads peuvent être préemptés par d’autres threads de priorité supérieure. Le système peut en conséquence répondre rapidement à des événements extérieurs, favoriser ou au contraire défavoriser certains éléments de l’exécution.

Processus et threads

Job :

Collection de processus manipulable comme un tout.

Fibre :

Variante de l’entité thread qui permet à une application de concrétiser ses propres stratégies d’exécution (multitâche coopératif).

Processus et threads

Service :

Logiciel généralement chargé lors du démarrage du système qui n’est pas lié à un utilisateur interactif. Les services, analogues des démons UNIX, implémentent souvent le second sous ensemble d’un mode de transaction client/serveur.

Exemples de programme en C

3 types d'exécutable sous windows:

CLI: Interface de commande

Desktop: Interface graphique

Native: Pilote de périphérique

- Flag dans le fichier PE
- Présence d’un symbol main, winmain ou DriverEntry.

Exemples de programme en C

Pour développer en C sous windows:

Ms Build Tools

Pour lancer une ligne de commande avec le compilateur:

Compiler from command line

```
z:\> cl.exe /I . *.c
```

Exemples de programme en C


```
1.  #include <windows.h>
2.  #pragma comment(lib, "user32.lib")
3.  int WINAPI WinMain(HINSTANCE hInst, HINSTANCE hInstPrev,
                       PSTR cmdline, int cmdshow)
4.  {
5.  return MessageBox(NULL, "hello, world", "caption", 0);
6.  }
```

z:\> cl.exe messagebox.c

Exemples de programme en C


```
1.  #include <windows.h>
2.  #pragma comment(lib, "user32.lib")
3.  int WINAPI WinMain(HINSTANCE hInst, HINSTANCE hInstPrev,
                       PSTR cmdline, int cmdshow)
4.  {
5.  return MessageBox(NULL, "hello, world", "caption", 0);
6.  }
```

z:\> cl.exe messagebox.c

Exemples de programme en C

z:\> cl.exe minimal.c

z:\> cl.exe hello.c