Détection de Débogueur (Anti-Debug)

Localisation: 0x004049C7 **Fonction:** ?
Type: Evasion **Sévérité:** Moyen

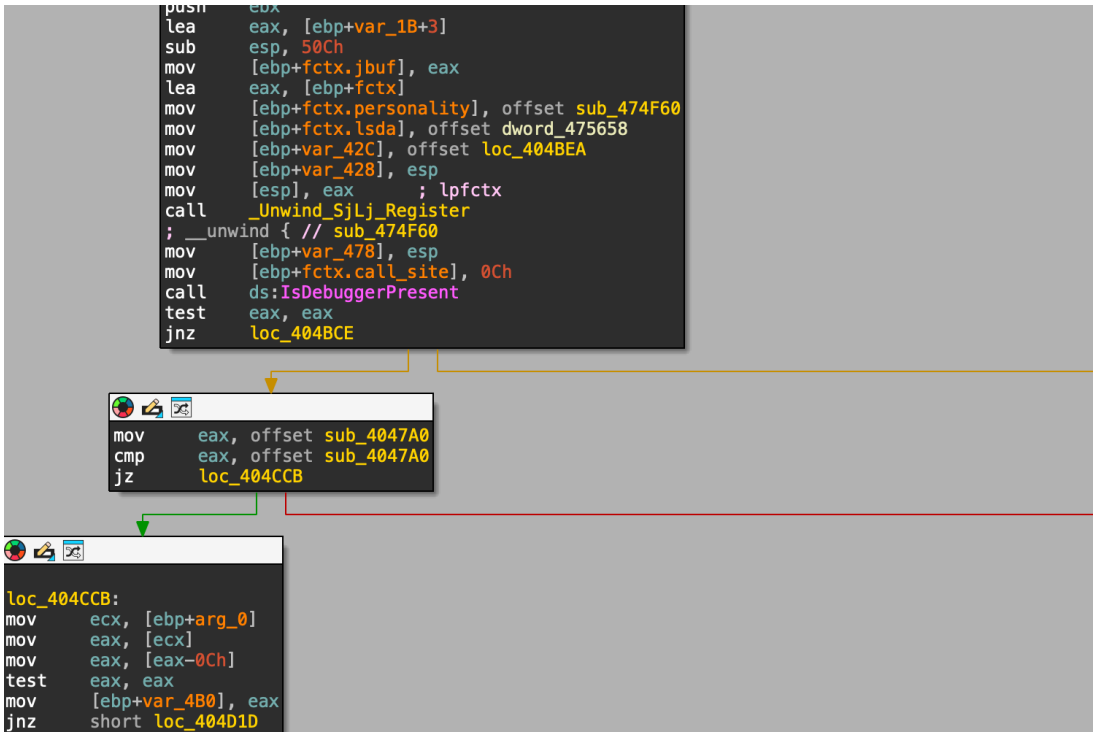
Code Assembleur

```
; Appel de l'API pour vérifier si un débogueur est attaché
call    ds:IsDebuggerPresent
test    eax, eax          ; Vérifie le résultat (1 = Debuggé, 0 = Normal)
jnz     loc_404BCE        ; Si EAX != 0, saut vers la sortie (Evasion)

; ... (Le payload s'exécuterait ici si EAX == 0) ...

loc_404BCE:               ; Bloc de sortie prématurée
mov     esp, [ebp+var_478]
lea     eax, [ebp+var_450]
mov     [esp+518h+hWnd], eax
call    _Unwind_SjLj_Unregister ; Nettoyage contextuel
lea     esp, [ebp-0Ch]
pop     ebx
pop     esi
pop     edi
pop     ebp
retn                      ; Fin de la fonction sans exécuter le payload
```

Screenshot



Analyse

Le programme implémente une technique d'évasion basique via `IsDebuggerPresent`. Il vérifie le drapeau `BeingDebugged` dans le PEB. Si un analyste est détecté (le registre `EAX` n'est pas nul), l'instruction `jnz` redirige le flux d'exécution vers `loc_404BCE`.

Ce bloc de code restaure simplement la pile et les registres avant de quitter la fonction via `retn`. Cela empêche l'exécution du code malveillant (manipulations graphiques GDI) situé plus bas dans la fonction, rendant l'analyse dynamique silencieuse si la protection n'est pas contournée.