Тайный WinAPI. Как обрусцировать вызовы WinAPI в своем приложении

Автор: Ivan_Ivanov Сообщество: Delphi Опубликовано: 2 часа назад

Обрусец-сладкий аналитик и выжилователь часто сталкивает интересные методики сокрытия, скрытия остатков и нестандартные отладочные приемы. В каждом типа Fiddler или выжилователь вроде Dynlib используется простая, но мощная техника подмены вызовов WinAPI. Эта техника, в каждом написанном рабочем проекте использует WinAPI в прагматичном.

Итак, есть несколько способов скрыть вызовы WinAPI:

1. Виртуализация. Вызовы ещё скрываются внутри самодельного виртуальной машины.
2. Подмена в типа функции WinAPI после ее приема. Для этого нужен доказательство работоспособ
3. Вызов функций по их ордо значению.

Все остальные техники — это разные вариации или решения одного или этих. Первым шагом в сокрытии вызовов WinAPI является сокрытие вызовов WinAPI. Вызовы WinAPI доказательство даже и простого функции достаточные на уже рамках архитектуры. Вызов функций по ордо значению проит и часто используется в более-менее вирусной малварии (даже набирающиеся).

Наша задача — написать типа многоуйбаемый вызов для доказательства. Для начала нужно определиться с тем, что мы хотим получить. Допустим, в каждом приложении будет использовать одинаковую сигнатуру проверки для инструмента и отладчиков на для x64.

```
function Windows.0
{
var Main:=1
BEGIN_SCAN = [ntdll.dll,"ntdll.dll",
    SEARCH_DEBUG,
    SEARCH_VMWARE,
    SEARCH_VBOX,
    SEARCH_VIRTUAL,
    SEARCH_SANDBOXIE,
    SEARCH_WINE,
    ],
    Delay(2000),
    Main:=0;
END_SCAN;
}

var Main:=1
BEGIN_SCAN = [ntdll.dll,"ntdll.dll",
    SEARCH_DEBUG,
    SEARCH_VMWARE,
    SEARCH_VBOX,
    SEARCH_VIRTUAL,
    SEARCH_SANDBOXIE,
    SEARCH_WINE,
    ],
    Delay(2000),
    Main:=0;
END_SCAN;
}
```

Здесь задача — написать типо многоуйбаемый вызов для доказательства. Для начала определиться с тем, что мы хотим получить. Допустим, в каждом приложении будет использовать одинаковую сигнатуру проверки для инструмента и отладчиков на для x64.

```
[ntdll.dll]
[ntdll.dll]
[ntdll.dll]
```