УДК 004.056.57

О.Л. Левошко[1]
Кіровоградський національний технічний університет

Використання дескрипторних таблиць та системних викликів для захисту від руткітів

Вступ. Програми, призначені для маскування мережевих з'єднань, процесів і дискових файлів, а також інших програм називають руткітами. Існують два типи руткітів: перші створюють нові файли або модифікують вже існуючі, другі обмежуються модифікацією оперативної пам'яті.

Основна частина. Маскування системних програм базується на модифікації структур даних або системного коду. Таблиці глобальних та локальних дескрипторів (GDT/LDT) зберігають базові адреси, межі та атрибути селекторів. Для захисту від руткіта створюється новий селектор з базою, відмінною від нуля, з подальшим його завантаженням до одного з сегментних регістрів. Побічним ефектом даного прийому стає поява нових селекторів в таблиці дескрипторів.

Частина руткітів модифікує таблицю дескрипторів переривань (IDT), що дозволяє їм перехоплювати будь-які переривання та виключення, в тому числі і системні виклики. Модифікація IDT дозволяє руткіту перехоплювати такі виключення, як, наприклад, загальне виключення захисту (General Protection Fault), звернення до сторінок та апаратні переривання. Для контролю вмісту таблиці переривань використовується процесорна команда SIDT, оскільки перехопити її виконання руткіт не в змозі. Руткіт може модифікувати таблиці дескрипторів переривань, замінюючи вектор 80h на свій власний код. Переривання INT 80h передає управління на функцію system_call. Руткіт або читає вектор 80h через SIDT, або знаходить system_call евристичним шляхом, оскільки вона містить досить характерний код. Вставивши на початок (або середину) цієї функції команду переходу на своє тіло, руткіт буде отримувати управління при будь-якому системному виклику. Отже, потрібно вилучити код функції system_call з пам'яті, порівнявши його з оригіналом, який можна взяти з дистрибутивного диска. Після виконання системного виклику управління отримує функція ret_from_sys_call, що йде слідом за system_call. Її перехоплюють багато руткітів. Команда SYSENTER передає управління з третього кільця прикладного рівня на ядерний рівень, використовуючи спеціальні MSR-регістри: IA32_SYSENTER_CS містить селектор сегмента, IA32_SYSENTER_EIP - адресу переходу, IA32_SYSENTER_ESP - нове значення регістра ESP при переході на ядерний рівень. Відобразити вміст регістрів MSR можна командою RDMSR, яку руткіт також не може перехопити.

Висновок. Отже, для захисту від руткіта потрібно контролювати код системних функцій system_call та ret_from_sys_call та використовувати швидкий механізм системних викликів, реалізований командами SYSENTER/SYSEXIT, SIDT, RDMSR .

---

[1] старший викладач кафедри програмного забезпечення