michael.engel [☺] hs-coburg.de

Praktikum zur Vorlesung
Malware-Analyse und Reverse Engineering
Sommersemester 2017

Praktikumsaufgabe 7

Besprechung am Donnerstag, 1. Juni 2017

Hinweise: Praktikumsaufgaben müssen in der vorgegebenen Zeit in den eingeteilten Gruppen bearbeitet werden. Die Lösung der Aufgaben wird bewertet, jedes Gruppenmitglied soll Fragen zu den einzelnen Aufgaben beantworten können. Bei Unklarheiten und Rückfragen wird selbstverständlich eine Hilfestellung gegeben.

7.1 Systemaufrufe in Assembler

Schreiben Sie ein Programm **in x64 (x86 64 bit)-Assembler**, das die folgende Aufgabe mit Hilfe von Linux-Systemaufrufen implementiert. Verwenden Sie dazu die entsprechenden Linux-Systemaufrufe, z.B. `read` und `write` und lesen Sie die entsprechenden man-pages (z.B. `man 2 read`), um die Aufrufkonventionen herauszufinden. Eine Vorlage für Ihr Assemblerprogramm finden Sie in moodle.

1. Einlesen eines Zeichens `c` von der Tastatur (Standardeingabe `stdin`)
2. Bestimmen eines Wertes n: n = Anzahl Kommandozeilenparameter (in C: “argc”). Dieser liegt bei Programmstart auf dem Stack (siehe Vorlage)
3. `n`-mal das Zeichen `c` ausgeben.
4. Programm beenden mit Systemaufruf `exit` und return-Code = Anzahl ausgegebener Zeichen.

Hinweis: Sie können Assembler-Code wie folgt in ein ausführbares Programm übersetzen:

```
$ gcc -c meinprog.S            # erzeugt Objektdatei meinprog.o (oder Fehlermeldungen...)
$ ld -o meinprog meinprog.o    # linkt meinprog.o zu ausführbarem Programm meinprog
$ ./meinprog                   # ...und startet es
```

Noch ein Hinweis: Systemaufrufe überschreiben Registerwerte – daher wichtige Werte vor einem Systemaufruf in Speicherzellen oder auf dem Stack sichern!
Ein Überblick über die verfügbaren Linux-Systemaufrufe und deren Aufrufnummern ist auf https://filippo.io/linux-syscall-table/ zu finden. Parameterbeschreibungen dazu finden sich in den entsprechenden Linux-manpages. Das Dokument unter https://www.cs.cmu.edu/~fp/courses/15213-s07/misc/asm64-handout.pdf hilft bei Fragen zu Maschinenbefehlen weiter.

7.2 Selbstmodifizierender Code – Hausaufgabe

In Moodle finden Sie eine ZIP-Datei `smc.zip`, die ein selbstmodifizierendes Programm `smc` enthält. Finden Sie heraus, was das Programm berechnet! Nehmen Sie dabei Tools wie `objdump` und `gdb` zur Hilfe.