Hochschule
Albstadt-Sigmaringen
Albstadt-Sigmaringen University

DIGITALE FORENSIK
Modul 6: Betriebssysteme
Übungen zu SB 1
Hans-Georg Eßer, 15.04.2012

Theorie

1. Welche Übergänge gibt es zwischen den Prozesszuständen „bereit“, „laufend“ und „blockiert“? Geben Sie für jeden *nicht möglichen* Übergang an, warum es einen solchen nicht gibt.

2. Nennen Sie den wichtigsten Unterschied zwischen Threads und Prozessen.

3. Warum kann es einen Prozesszustand „swapped“ geben, aber keinen gleichnamigen Thread-Zustand?

4. Benutzer *anton* hat es geschafft, unter Linux eine Kopie des Programms *ls* zu erzeugen, die der Benutzerin *berta* gehört und bei der das SUID-Bit gesetzt ist. Er hat das Recht, diese Datei zu starten. Was kann *anton* mit dieser Datei anfangen?

5. Warum speichert das Ext3-Dateisystem die so genannten erweiterten Attribute (z. B. ACLs) nicht im Inode einer Datei?

6. Emulation ist im Vergleich zu Virtualisierung sehr langsam. Was ist ein entscheidendes Feature, das nur mit Emulation möglich ist, so dass Emulation eine sinnvolle Technik ist?

Praxis

7. Booten Sie die virtuelle Maschine (FOM-Debian-Mini in VirtualBox) und melden Sie sich als Benutzer `root` (Passwort `root`) an.
- Erzeugen Sie mit `useradd -m testing` einen neuen Account (mit Namen `testing`).
- Setzen Sie mit `passwd testing` das Passwort (auf `testing`; zweimal blind eingeben).
- Betrachten Sie mit `grep testing /etc/shadow` den Eintrag in der Shadow-Datei. Welche Methode hat Linux verwendet, um den Hash zu erzeugen?
- Mit
```
SALT=$( grep testing /etc/shadow | cut -d'$' -f3 )
PASS=testing
```
kopieren Sie aus der Shadow-Datei-Zeile das verwendete Salt in die Variable `$SALT` und das für den Account vergebene Passwort (`testing`) in die Variable `$PASS`.
Geben Sie dann das Kommando
```
python -c "import crypt; print crypt.crypt ('$PASS',
    '\$6\$$SALT')"
```
(alles in einer Zeile) ein und vergleichen Sie die Ausgabe mit der Zeile aus der Shadow-Datei.
- Inwiefern haben Sie mit diesen drei Befehlen die Grundlage für ein Programm geschaffen, das (mit Hilfe eines Dictionaries) eine Brute-Force-Passwortattacke gegen die Shadow-Datei durchführen kann?

8. Schreiben Sie auf Basis von Aufgabe **7** ein Python-Programm, das die Datei `passes.txt` mit
```
z = file("passes.txt").readlines()
```
einliest und für jeden der dort gespeicherten Passwortkandidaten obigen Test durchführt. Sie können dabei am Anfang des Programms den vollständigen „Passwort“-String aus der Shadow-Datei in einer Variablen speichern. Für jedes Passwort soll das Programm ausgeben, ob der Test erfolgreich war. Zum Testen erzeugen Sie eine kleine Beispieldatei (`passes.txt`) mit einigen Passwörtern, darunter auch `testing`. Sie können das Programm zur Vereinfachung auch mit einem Bash-Shell-Skript kombinieren, das wie Aufgabe **7** Informationen mit `grep` und `cut` aus Dateien extrahiert.