TECHNISCHE UNIVERSITÄT DRESDEN

Faculty of Computer Science Institute for System Architecture, Operating Systems Group

Betriebssysteme und Sicherheit, WS 2012/13

6. Aufgabenblatt – Sicherheit

Geplante Bearbeitungszeit: zwei Wochen

Aufgabe 6.1

Erläutern Sie die für ein Schutzsystem wesentlichen Begriffe Schutzmatrix, ACL und Capability. Geben Sie für die beiden Listenstrukturen jeweils die Form eines Eintrags an. Diskutieren Sie Vor- und Nachteile.

Aufgabe 6.2

In einem System gebe es zwei Nutzer A und B, die beide einer Gruppe G angehören, sowie eine Datei D. Konstruieren Sie eine Rechtezuteilung einerseits mittels ACL, andererseits mittels Capabilities, die folgendes bewirkt:

- Mit Ausnahme der Nutzer A und B darf jeder die Datei D lesen und ausführen.
- Die Mitglieder der Gruppe G dürfen zusätzlich auch schreibend auf D zugreifen.
- Nutzer B darf die Datei nur lesen.
- Nutzer A hat keinerlei Zugriff auf D.

Legen Sie zunächst für die beiden Listenformen die jeweilige Struktur eines Eintrags fest (dabei sind Wildcards zugelassen). Sollten bei der Interpretation einer Liste bestimmte Voraussetzungen erforderlich sein oder andere Probleme auftreten, so beschreiben Sie diese.

Aufgabe 6.3

Die folgenden Fragen und Aufgaben beziehen sich auf die „klassische“ Sicherheitsarchitektur von Unix.

(a) Welche Objekte, Subjekte und Operationen spielen hier eine Rolle?

(b) Werden Zugriffssteuerlisten (ACL) oder Capability-Listen eingesetzt?

(c) Welche Attribute sind den beim Öffnen einer Datei beteiligten „Objekten“ (im allgemeinen Sinn) zugeordnet und wie wird die Entscheidung gefällt, ob eine Datei geöffnet werden darf?

(d) Formulieren Sie die in Aufgabe 2 geforderte Rechtezuteilung mittels des Unix-Rechtesystems. Welche Probleme treten dabei auf?

(e) Verdeutlichen Sie ein weiteres Problem, das mit dem Rechtesystem von Unix verbunden ist, anhand des Änderns eines Passworts in einem Unix-Betriebssystem. Die verschlüsselten Passwörter sind in einer Datei `passwd` gespeichert, die von jedem gelesen, aber nur mit Hilfe eines speziellen (gleichnamigen) Programms geschrieben werden kann. Die relevanten Spezifikationen für diese beiden Objekte lauten:

```
rw- r-- r-- root root /etc/passwd
rwx r-x r-x root root /usr/bin/passwd
```

Beschreiben Sie zunächst das Problem, das beim Ausführen des Programms auftritt, und anschließend die Lösung dieses Problems.

Aufgabe 6.4

Welche Aufgaben haben kryptographische Systeme? Wie lassen sich solche Systeme klassifizieren? Auf welche prinzipielle Weise erreichen sie ihr Ziel? Erläutern Sie dabei auch die Begriffe „symmetrische und asymmetrische Systeme“ sowie „hybride Verschlüsselung“.

Aufgabe 6.5

Der RSA-Algorithmus (Algorithmus von RIVEST/SHAMIR/ADLEMAN, 1977) hat folgende Struktur:

(I) Wähle zufällig zwei Primzahlen $p, q \neq 2$ mit $p \neq q$, annähernd gleiche Stellenzahl.

(II) Bilde $n = pq$. In diesem Fall gilt für die EULERsche Funktion φ: $\varphi(n) = (p-1) \cdot (q-1)$.

(III) Wähle c mit $3 \leq c < \varphi(n)$, so dass $\text{ggT}\left(c, \varphi(n)\right) = 1$.

(IV) Berechne d mit $cd \equiv 1 \bmod \varphi(n)$ und $1 < d < \varphi(n)$.

(V) Verteile den Modul n sowie c als öffentlichen Schlüssel und d als privaten Schlüssel.

Eine Nachricht $x < n$ wird durch $x^c \equiv y \bmod n$ mit $0 \leq y < n$ verschlüsselt und entschlüsselt durch $y^d \equiv x \bmod n$.

(a) Worin liegt das Sicherheitsrisiko dieses Algorithmus?

(b) Begründen Sie die einzelnen Restriktionen in (I) und (III). Welche Aussage macht die EULERsche Funktion? Welche Bedeutung hat Schritt (IV) für das Vorgehen?

(c) Demonstrieren Sie den RSA-Algorithmus an dem folgendem Beispiel:

Der Modul sei $n = 55$, der öffentliche Schlüssel sei $c = 7$. Verschlüsseln Sie damit die Nachricht $x = 2$. Berechnen Sie den privaten Schlüssel d, entschlüsseln Sie die verschlüsselte Nachricht und zeigen Sie deren Übereinstimmung mit x.

Hinweise:

- Benutzen Sie zur Berechnung des ggT zweier Zahlen a, b mit (o.B.d.A.) $a > b$ sowie der Summendarstellung nach dem Erweiterten EUKLIDischen Algorithmus eine Tabelle folgender Form:

	a	b	
a	1	0	
b	0	1	
x_i	s_i	t_i	y_i

mit

$$y_i = -\left(x_{i-1} \operatorname{div} x_i\right)$$
$$x_{i+1} = x_{i-1} \bmod x_i$$
$$s_{i+1} = s_i \cdot y_i + s_{i-1}$$
$$t_{i+1} = t_i \cdot y_i + t_{i-1}$$

div bezeichnet die ganzzahlige Division, mod den dabei auftretenden Rest.

In obiger Tabelle beginnt i mit 0, die Formeln gelten ab $i = 1$ (zuerst ist also y_1 in der „b-Zeile“ zu berechnen). Der Algorithmus bricht ab bei $x_i = 0$, und dann gilt: $x_{i-1} = \text{ggT}(a, b) = s_{i-1}a + t_{i-1}b$.

Die letzte Gleichung gilt auch in jeder Zeile, was zur Rechenkontrolle genutzt werden sollte.

- Zerlegen Sie zum Entschlüsseln die verschlüsselte Nachricht y in Primfaktoren und versuchen Sie dann, in y^d geeignete Potenzen abzuspalten, deren Produkt modulo 55 gleich 1 oder -1 ist.