Aufgabe 1: Rechnersicherheit

Zugangs- und Zugriffskontrolle – In der IT-Sicherheit unterscheidet man zwischen den Systemfunktionen Zugriffskontrolle und Zugangskontrolle. Informieren Sie sich über die beiden Techniken, die auch im Rahmen der Vorlesung behandelt werden und beantworten Sie folgende Fragen.

1.a (Pflicht; 2 Punkte) Erläutern Sie stichpunktartig den Zweck der jeweiligen Technik.
Die Zugriffskontrolle regelt, welche Person welche Operation auf Ressource ausführen darf.
Die Zugangskontrolle regelt, welcher Partner ein Betriebsmittel in Anspruch nehmen darf.

1.b (Pflicht; 2 Punkte) Ist es sinnvoll, ein System mit einer Zugangskontrolle auszustatten, jedoch keine Mechanismen zur Zugriffskontrolle zu implementieren? Begründen Sie Ihre Antwort mit einem Beispiel.
Ist es notwendig, ein System mit einer Zugangskontrolle zu schützen, so macht es durchaus auch Sinn, eine Zugriffskontrolle einzubauen, da man nicht nur überprüfen will, wer die Ressource verwenden darf, sondern auch wie jemand diese Ressource verwenden darf. Der Fall dass man nur den Zugang kontrollieren will und jede Person dann alle Rechte hat, ist eher untypisch. Ein klassisches Beispiel ist eine Datenbank. Hier wird der Zugang durch eine Benutzerkennung kontrolliert. Gleichzeitig kann hier jedoch auch der Zugriff kontrolliert werden. (Welche Tabellen können gelesen/verändert werden)
Es gibt jedoch auch Gegenbeispiele, bei der nur eine Zugangskontrolle verwendet wird. Z.B. der Zugang zum Internet: Hier wird von dem Provider lediglich eine Zugangskontrolle gemacht. Wenn man jedoch Zugang zum Internet hat, hat man dort auch uneingeschränkten Zugriff auf alle Systeme.

1.c (Pflicht; 1 Punkt) Die Absicherung eines Systems mittels einer Zugriffskontrolle setzt hingegen immer auch eine vorherige Zugangskontrolle voraus. Warum?
Eine Zugriffskontrolle kontrolliert nur, ob der angegebene Nutzer auch die Berechtigung hat, eine bestimmte Tätigkeit auszuüben. Erst durch die Authentifizierung (Zugangskontrolle) wird der Anwender an einen Account gekoppelt und kann nicht alle Account ausprobieren bis der Zugriff gewährt wird.

1.d (Pflicht; 2 Punkte) File-Sharing-Dienste (z.B. Dropbox) ermöglichen es ihren Nutzern, einzelne Ordner mit Hilfe eines Share-this-Folder-Links anderen Nutzern freizugeben (Zugriffskontrolle auf Ordner-Ebene). Mit dem Link kann jeder auf den freigegebenen Ordner zugreifen, auch wenn er kein Konto bei dem jeweiligen Dienst hat. Ein Benutzerkonto (Login) beim File-Sharing-Dienst ist zum Zugriff auf den Ordner jedoch nicht erforderlich. Scheinbar widerspricht diese Situation also der Aussage in der vorherigen Teilaufgabe. Nehmen Sie hierzu Stellung.

Auch hier findet eine Zugangskontrolle statt. Der Ordner kann nur aufgerufen werden, wenn der Ordner „freigegeben“ wurde und man den entsprechenden Link hat. Hier wirkt der Link wie Zugangskontrolle und Zugriffskontrolle gleichzeitig. Der Link ist in der Regel so gestaltet, dass es nicht möglich ist ihn zu erraten (durchzuprobieren). Außerdem muss der Angreifer wissen, dass dieser Ordner freigegeben wurde.

2.b (Optional) Da inzwischen die meisten Bürger eine Webcam in ihrem Computer/Laptop eingebaut haben, wird nun im Rahmen einer eGovernment-Initiative vorgeschlagen, den elektronischen Reisepass auch zu Hause zu verwenden. Die Bürger sollen dadurch Dienstleistungen von Behörden über das Internet wahrnehmen können. Hierzu sollen Lesegeräte an die Bürger ausgegeben werden, mit denen der Pass ausgelesen wird. Mit der Webcam wird dann ein Foto des Benutzers aufgenommen und mit dem Lichtbild des Passes verglichen. Im Falle einer erfolgreichen Überprüfung werden die Informationen aus dem Pass an die Behörden-Website weitergeleitet (Authentifizierung) und der Dienst kann in Anspruch genommen werden. Nennen Sie zwei potentielle Schwachstellen dieser Realisierungsidee.

Das Einscannen des Passes setzt logischerweise seinen Besitz voraus, weshalb man Zugriff auf das auf ihm enthaltene Bild hat und dieses in die Webcam halten kann anstatt sein eigenes Gesicht. Eventuell kann es darüber hinaus möglicherweise sogar mit seinem eigenen Gesicht funktionieren, wenn die beiden Gesichter (das auf dem Pass und das des Benutzers) sich stark ähneln und das Webcam-Foto bei schlechtem Licht aufgenommen wird oder mit visuellen Veränderungen (z.B. Make-Up).

3.a (Optional) Welche Schwachstellen weist das oben beschriebene System auf? Stellen Sie das Angreifermodell auf, das diesem Systementwurf offenbar zugrunde liegt.

Beherrscht ein Benutzer das 10-Finger-Tippsystem oder ein vergleichbares System, das gewissen Regeln folgt, so ist es ziemlich wahrscheinlich, dass mehrere Benutzer das gleiche Tippverhalten haben und so die Wahrscheinlichkeit ziemlich hoch ist, sich in einen fremden

Account einzuloggen. Außerdem stellen 20 Durchläufe des Authentifizierungs-Satzes kein wirklich repräsentatives Ergebnis für das Tippverhalten eines Benutzers dar.
Ein Angreifer könnte an der Stelle eingreifen, an der das Tippmuster von einem Applet aufgezeichnet wird, wofür er auf jeden Fall genug Zeit hat, da der Satz 20 mal wiederholt werden muss.

3.b (Optional) Welche Gegenmaßnahmen könnte der Betreiber ergreifen?
Zuerst könnte er die Anzahl an Durchläufen des Authentifizierungs-Satzes erhöhen, sodass die Analyse des Tippverhaltens eindeutiger und zutreffender wird. Natürlich kann er als alternative Möglichkeit sich einzuloggen das Passwort hinzuziehen oder es gar zu dem Satz hinzufügen, jedoch verwirft das den Ansatz des Einloggens ohne ein Passwort.

Aufgabe 2: Timing-Attack

1. (Pflicht; 6 Punkte) Überprüfen Sie, ob diese Methode anfällig für Timing-Angriffe ist. Schreiben Sie hierzu ein kleines Java-Programm Timer.java, das die Laufzeit der Methode passwordCompare bei zwei identischen Passwörtern sowie bei zwei unterschiedlichen Passwörtern ermittelt. Verwenden Sie zur Zeitmessung die Methode System.nanoTime() und überlegen Sie sich, wie Sie auch auf einem schnellen PC einen signifikanten Zeitunterschied herbeiführen können. Hinweis: Es empfiehlt sich, den Just-in-Time-Compiler von Java zu deaktivieren, indem Sie Ihr Programm wie folgt starten: java -Djava.compiler=NONE Timer
Der Zeitunterschied ist gering, jedoch kann bei sehr langen Passwörtern, die jedoch schon an den vorderen Stellen Unterschiede aufweisen „deutliche“ Zeitdifferenzen festgestellt werden. Die Funktion terminiert außerdem sofort (geringe Zeit), wenn die Passwörter unterschiedliche Längen haben

2. (Optional) Erläutern Sie in 1-2 Sätzen, warum hier ein Timing-Angriff möglich ist.
Der Timing-Angriff ist hier in zwei Fällen möglich.
Fall 1: Die Passwörter sind unterschiedlich lang. Terminierung sofort.
Fall 2: Passwörter sind gleich lang, aber nicht identisch. Die Funktion terminiert nach dem ersten unterschiedlichen Zeichen.

3. (Pflicht; 2 Punkte) Wie geht der Angreifer beim Timing-Angriff konkret vor, um das im TPM hinterlegte Passwort mit möglichst wenig Versuchen zu ermitteln, d. h. welche Passwörter probiert er der Reihe nach aus und wie entscheidet er, welches Passwort er als nächstes probiert?

Ein auf einem PC installiertes Online-Banking-Programm startet erst dann, wenn der Benutzer das korrekte Passwort eingegeben hat. Das Programm sendet das eingegebene Passwort an ein Trusted Platform Module (TPM), in dem das korrekte Passwort abgelegt ist. Das TPM überprüft das eingegebene Passwort und signalisiert der Banking-Software das Ergebnis. Im TPM kommt die folgendermaßen implementierte Methode zum Einsatz.

4. (Optional) Ändern Sie den Quellcode der Methode passwordCompare so ab, dass keine Timing-Attacks mehr möglich sind. Achten Sie auf möglichst kurzen und übersichtlichen Code.

```
boolean passwordCompare(char[] a, char[] b)
{
      boolean identical = true;

      int i;

      if(a.length != b.length)  {
            identical = false;
      }

      for(i=0; i<a.length; i++) {
            if(a[i]!=b[i])
            {
                  identical = false;
                  //no return or break here
            }
      }

      return identical;
}
```