Vorbemerkung und Abgabe

Es gilt weiterhin die Vorbemerkung von Übungsblatt 1. Die Abgabe erfolgt online unter folgender Adresse: **https://svs.informatik.uni-hamburg.de/submission/for/gss15-3**.

Die Übungen zu diesem Blatt finden vom 18.05.2015 bis 21.05.2015 statt.

Aufgabe 1: Rechnersicherheit

1. **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.

   a) **(Pflicht; 2 Punkte)** Erläutern Sie stichpunktartig den Zweck der jeweiligen Technik.

   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.

   c) **(Pflicht; 1 Punkt)** Die Absicherung eines Systems mittels einer Zugriffskontrolle setzt hingegen immer auch eine vorherige Zugangskontrolle voraus. Warum?

   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.

2. **Biometrische Techniken: EasyPASS** – An mehreren deutschen Flughäfen wird die (teil-)automatische Passkontrolle und Einreise in die Bundesrepublik Deutschland anhand des biometrischen Reisepasses ermöglicht – eine manuelle Bearbeitung durch einen Grenzbeamten kann dadurch im Falle einer positiven Passkontrolle entfallen. Am *EasyPASS*-Automat müssen die Reisenden dazu ihren elektronischen Pass selbständig einscannen. Mittels einer Kameraaufnahme wird anschließend das Gesicht des Passagiers mit dem auf dem Pass digital gespeicherten Lichtbild verglichen. Bei Fragen zur Bedienung stehen Grenzbeamte zur Verfügung. Fotostrecke: **http://www.spiegel.de/fotostrecke/fotostrecke-47853.html**

   a) **(Optional)** Informieren Sie sich über die biometrischen Techniken im elektronischen Reisepass der Bundesrepublik Deutschland und die Funktionsweise von EasyPASS.

   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.

3. **Biometrische Techniken: Tippverhalten** – Im Rahmen eines Forschungsprojekts wurde ein Authentfizierungssystem entwickelt, welches Nutzer beim Einkauf in einem Online-Shop anhand des Tippverhaltens authentifiziert. Dabei wird die Tatsache ausgenutzt, dass praktisch jeder Nutzer die Tasten auf der Tastatur auf charakteristische Weise anschlägt (Zeitabstände zwischen den Anschlägen sowie häufige Tippfehler). Die Tippmuster werden direkt im Browser von einem Flash-Applet oder per JavaScript aufgezeichnet und verschlüsselt per SSL an den Webshop übertragen. Die bislang besten Ergebnisse werden erzielt, wenn für alle Nutzer immer derselbe Authentifizierungs-Satz (z. B. *It never rains in southern California*) verwendet wird. Bei der erstmaligen Registrierung im Online-Shop geben die Nutzer eine Tipp-Probe ab, indem sie etwa 20 mal den Authentifizierungs-Satz eingeben. Daraus errechnet das System ein charakteristisches Profil des Nutzers, welches im Online-Shop hinterlegt wird. Bei späteren Besuchen des Online-Shops gibt der Nutzer dann zur Authentifizierung seinen Benutzernamen ein und tippt den Authentifizierungs-Satz erneut ab. Der Online-Shop überprüft dann die übermittelte Tipp-Probe mit der Datenbank.

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

   b) **(Optional)** Welche Gegenmaßnahmen könnte der Betreiber ergreifen?

4. **Realisierung eines Online-Tickets** – Ein Kino möchte den Online-Verkauf von Eintrittskarten realisieren, die sich der Kunde an seinem PC ausdruckt. Jede Eintrittskarte soll einen Strichcode enthalten, der bei der Einlasskontrolle eingelesen wird.

   a) **(Optional)** Denken Sie sich eine Realisierung aus. Beschreiben Sie dazu, wie und wo die Barcodes erzeugt werden, welche Daten wo gespeichert werden und wie das Kino prüft, ob ein Ticket gültig ist.

   b) **(Optional)** Definieren Sie ein plausibles Angreifermodell für Ihre Implementierung.

   Hinweis: Die Aufgabenstellung besitzt einige Freiheitsgrade, d.h. Sie können, wenn Sie wollen, auch mehrere Angreifermodelle und Designvorschläge machen.

Aufgabe 2: Timing-Attack

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.

```
1
2 boolean passwordCompare(char[] a, char[] b)
3 {
4   int i;
5
6   if(a.length != b.length) return false;
7
8   for(i=0; i<a.length && a[i]==b[i]; i++);
9
10  return i == a.length;
11 }
```

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*
2. **(Optional)** Erläutern Sie in 1-2 Sätzen, warum hier ein Timing-Angriff möglich ist.
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?
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.

Aufgabe 3: Real-World-Brute-Force Angriff

Ein Arbeitsbereich hat ein Upload-Tool zur Einreichung von Übungsblättern entwickelt, das unter **http://svs.informatik.uni-hamburg.de/abgabe/** erreichbar ist. Nach dem Upload erhält man einen Sicherheitscode, mit dem man eine überarbeitete Fassung hochladen kann. Bei der Entwicklung war der Schutz des Systems gegen Brute-Force-Angriffe ein wichtiges Entwurfsziel. Dadurch soll verhindert werden, dass eine Übungsgruppe die Lösung einer anderen Gruppe, welche bereits etwas hochgeladen hat, einsehen kann. Dies soll durch die hohe Länge des Sicherheitscodes verhindert werden.

(Optional) In dieser Aufgabe sollen Sie auf Basis Ihrer Beobachtungen des unter o.a. URL erreichbaren Systems eine möglichst genaue Abschätzung der effektiven Sicherheit des Sicherheitscodes durchführen. Ermitteln Sie dazu anhand der vom System zur Verfügung gestellten Informationen, wie lange es im Mittel mindestens dauern würde, bis Sie Zugriff auf die Lösung von mindestens einer anderen Gruppe hätten, wenn der Webserver konstant 1000 Anfragen pro Sekunde beantworten würde. Dokumentieren Sie, wo erforderlich, Ihre Annahmen.

Hinweise:

1. Bitte überlasten Sie den Server nicht mit einem Brute-Force-Angriff! Diese Aufgabe ist analytisch zu lösen.
2. Um den verwendeten Zeichenvorrat im Sicherheitscode möglichst genau abschätzen zu können, sollten Sie sich im Abgabe-Tool durch Hochladen einiger Dateien selbst eine kleine Menge (max. 20) von Sicherheitscodes erzeugen.
3. Als Ausgangspunkt für Ihre Analyse bietet es sich an, die Ratewahrscheinlichkeit für einen einzelnen Sicherheitscode zu ermitteln.