

Stammvorlesung Sicherheit im Sommersemester 2014

Übungsblatt 6

Hinweis: Übungsblätter können freiwillig bei Jessica Koch, Raum 256, Geb. 50.34 („Info-Bau“) bis zur Übung am 14.7.14 zur Korrektur abgegeben werden. Die Korrektur dient nur der Selbstkontrolle; es gibt keine Punkte und keinen Klausur-Bonus.

Aufgabe 1. Wir betrachten ein Public-Key-Identifikationsprotokoll $(\text{Gen}, \text{P}, \text{V})$, bei dem ein Benutzer P einen Benutzer V davon überzeugen will, dass P einen diskreten Logarithmus $\log_g h$ modulo p , mit ungeradem primem $p \in \mathbb{N}$ und $g, h \in \mathbb{Z}_p^*$, kennt. Wir nehmen dazu an, dass beide Benutzer dem Protokoll folgen. Dabei sei der Beweisalgorithmus P , der Verifikationsalgorithmus V sowie der Protokollablauf folgendermaßen definiert:

- (1) Die Parametergenerierung $\text{Gen}(1^k)$ wählt zuerst ein ungerades primes $q \in \mathbb{N}$ und setzt $p := 2q + 1$, sodass p prim ist. Weiterhin sei $g \in \mathbb{Z}_p^*$ ein Element der Ordnung $q := \text{ord}(g)$. Wir ziehen zufällig gleichverteilt ein $s \in \mathbb{Z}_{\text{ord}(g)}$ und setzen $h := g^s \bmod p$. ($\text{ord}(g)$ gebe dabei die Ordnung von g in \mathbb{Z}_p^* an.) Anschließend wird ein Public-Key $pk := (\mathbb{Z}_p^*, g, h)$ und ein Secret-Key $sk := (\mathbb{Z}_p^*, g, s)$ ausgegeben.
- (2) $\text{P}(sk)$ wird mit Eingabe des Secret-Keys sk gestartet, wählt ein zufällig gleichverteiltes $r \in \mathbb{Z}_{\text{ord}(g)}$ und gibt $\text{out}_P := X := g^r \bmod p$ aus.
- (3) Der Verifikationsalgorithmus $\text{V}(pk, \text{out}_P)$ erhält als Eingabe den Public-Key pk und die P-Ausgabe $X := \text{out}_P$ aus (2); anschließend wird ein $b \in \{0, 1\}$ zufällig gleichverteilt gezogen und $\text{out}_V := b$ ausgegeben.
- (4) $\text{P}(sk, \text{out}_V)$ erhält zu sk die V-Ausgabe $b := \text{out}_V$ aus (3), berechnet $y := r + b \cdot s \bmod \text{ord}(g)$ und gibt $\text{out}_P := y$ aus.
- (5) $\text{V}(pk, \text{out}_P)$ erhält zu pk die P-Ausgabe $y := \text{out}_P$ aus (4), überprüft, ob $g^y \bmod p = h^b X \bmod p$ gilt. Falls ja, wird 1 ausgegeben; anderenfalls 0.

Wir sagen, dass V den Beweis von P akzeptiert, wenn nach (mehrfacher) Anwendung des Protokolls immer $\text{V}(pk, \text{out}_P) = 1$ in (5) gilt.

- (a) Zeigen Sie die Korrektheit von $(\text{Gen}, \text{P}, \text{V})$.
- (b) Falls V in (3) $b = 0$ zieht, ist y in (4) offensichtlich unabhängig von s . (Das bedeutet, dass jemand ohne Kenntnis von sk in (2) $g^r \bmod p$ und in (4) r senden würde, sodass V akzeptiert.) Falls V in (3) $b = 1$ wählt, ist dies nicht der Fall. Wie könnte jemand, der sk nicht kennt, in diesem Fall dennoch Elemente X in (2) und y in (4) erzeugen, sodass V in (5) 1 ausgibt? Wie groß ist die Erfolgswahrscheinlichkeit?
- (c) Geben Sie einen Simulator \mathcal{S} in der Rolle von P (analog zum Graph-Dreifärbbarkeitsbeispiel aus der Vorlesung) an.

Lösungsvorschlag zu Aufgabe 1.

- (a) Für $(pk, sk) := \text{Gen}(1^k)$, mit pk und sk wie in (1), $X := P(sk)$ in (2), $b := V(pk, X)$ in (3) und $y := P(sk, b)$ in (4) gilt

$$g^y \bmod p = g^{r+b \cdot s} \bmod p = h^b X \bmod p$$

und damit $V(pk, \text{out}_P^{(4)}) = \text{yes}$.

- (b) Wir wählen in (2) ein Bit $b' \in \{0, 1\}$ zufällig gleichverteilt, setzen $X := g^r h^{-b'} \bmod p$, für zufällig gleichverteiltes $r \in \mathbb{Z}_{\text{ord}(g)}$, und senden X an V . (Hinweis: Der Public-Key $pk = (\mathbb{Z}_p^*, g, h)$ ist uns bekannt. Da $g^r \bmod p$ ein zufälliges Element in \mathbb{Z}_p^* ist, ist auch $g^r h^{-b'} \bmod p$ ein zufälliges Element in \mathbb{Z}_p^* . Das heißt, der Wert X in (3) sieht in beiden Fällen, $b' = 0$ oder $b' = 1$, zufällig für V aus. V 's Wahl von b in (3) ist also unabhängig von b' .) Mit Wahrscheinlichkeit $\frac{1}{2}$ haben wir $b' = b$ gezogen. Sendet V uns in (3) ein Bit $b = b'$, setzen wir in (4) $y := r$ und senden dieses an V . V verifiziert

$$g^y \bmod p = h^b X \bmod p = h^b g^r h^{-b'} \bmod p = g^r \bmod p$$

und gibt yes aus. Falls $b' \neq b$ gilt, brechen wir ab. Mit Wahrscheinlichkeit $\frac{1}{2}$ gibt V in (5) den Wert yes aus.

- (c) Wir geben einen Simulator \mathcal{S} an, der ein Transskript $\langle (X, y), b \rangle$ ausgibt, sodass die Verteilungen $\langle P(sk), \mathcal{A}(1^k, pk) \rangle$ mit Angreifer \mathcal{A} und (die Ausgabe von) $\mathcal{S}(1^k, pk)$ ununterscheidbar sind. Der Simulator $\mathcal{S}(1^k, pk)$ agiere wie folgt beschrieben:

- (i) \mathcal{S} spielt Protokoll mit \mathcal{A} , in der Rolle von P .
- (ii) \mathcal{S} sendet $X := g^r h^{-b'} \bmod p$, für zufällig gleichverteiltes $r \in \mathbb{Z}_{\text{ord}(g)}$ und zufällig gleichverteiltes $b' \in \{0, 1\}$, an \mathcal{A} .
- (iii) \mathcal{S} hofft auf $b' = b$, nachdem \mathcal{A} ein Bit $b \in \{0, 1\}$ gesendet hat. Ist dies der Fall, wird das Protokoll fortgeführt. Anderenfalls, spult \mathcal{S} den Angreifer \mathcal{A} zurück und spielt das Spiel von vorn.
- (iv) \mathcal{S} sendet den Wert $y := r$ aus (ii) an \mathcal{A} . (Dieses passiert, falls \mathcal{A} in (iii) nicht zurückgespult wurde.)
- (v) \mathcal{S} gibt schließlich Transskript $\langle (X, y), b \rangle$ aus. (Es gilt $g^y \bmod p = h^b X \bmod p$.)

Wie wir in (b) sahen, ist \mathcal{A} 's Wahl von b in (iii) unabhängig von b' . Wir erhalten für die Wahrscheinlichkeit, dass $b' = b$ gilt, $\frac{1}{2}$. Das bedeutet für \mathcal{S} , dass nach k -maligem Zurückspulen die Wahrscheinlichkeit, dass \mathcal{S} terminiert, $1 - \frac{1}{2^{k+1}}$ ist.

Wir zeigen nun, dass die Ausgabeverteilung von $\langle P(sk), \mathcal{A}(1^k, pk) \rangle$ und $\mathcal{S}(1^k, pk)$ ununterscheidbar sind. Ein ehrliches V kann also nicht (effizient) sagen, ob V mit einem ehrlichen P oder mit \mathcal{S} das Protokoll führt. Wir wissen aus (b), dass X für $b' = 0$ oder $b' = 1$ zufällig verteilt ist. Sei nun $b := V(pk, X)$, konditioniert auf $b' = b$. Wie oben gesehen, ist die Wahrscheinlichkeit dafür $\frac{1}{2}$. Für einen Beweiser P sowie für einen Simulator \mathcal{S} ist für $b = 0$ der Wert y ein zufälliger Wert aus $\mathbb{Z}_{\text{ord}(g)}^*$; für $b = 1$ ebenso.

Aufgabe 2. Wir betrachten im Folgenden das Jacobi-Symbol $\left(\frac{a}{n}\right)$ für zwei natürliche Zahlen $a, n \in \mathbb{N}$. Dabei gilt, falls $n = p \in \mathbb{P}$ eine Primzahl ist, folgendes:

$$\left(\frac{a}{p}\right) := a^{\frac{p-1}{2}} \bmod p = \begin{cases} 1 & \text{wenn } a \text{ quadratischer Rest modulo } p \text{ ist} \\ -1 & \text{wenn } a \text{ quadratischer Nichtrest modulo } p \text{ ist} \\ 0 & \text{wenn } a \text{ ein Vielfaches von } p \text{ ist} \end{cases}$$

Desweiteren gelten folgende Rechenregeln für ungerades $n \in \mathbb{N}$:

- * $\left(\frac{a}{n}\right) = \left(\frac{a \bmod n}{n}\right)$, $\left(\frac{-1}{n}\right) = (-1)^{\frac{n-1}{2}}$
- * $\left(\frac{a_1 a_2}{n}\right) = \left(\frac{a_1}{n}\right) \left(\frac{a_2}{n}\right)$, $\left(\frac{a}{mn}\right) = \left(\frac{a}{m}\right) \left(\frac{a}{n}\right)$

- (a) Berechnen Sie die Jacobi-Symbole von $\left(\frac{15}{35}\right)$, $\left(\frac{32}{33}\right)$, $\left(\frac{17}{39}\right)$

- (b) Zeigen Sie, dass -1 für $n = pq$ mit Primzahlen p und q für die $p \equiv q \equiv 3 \pmod{4}$ gilt, kein quadratischer Rest mod n ist, jedoch das Jacobi-Symbol 1 ergibt. (Solche Zahlen n nennt man Blum-Integer.)
- (c) Wir betrachten nun folgendes Identifikationsprotokoll, das auf der Schwierigkeit der Berechnung von Quadratwurzeln in der Gruppe \mathbb{Z}_n^* für $n = pq$ Blum-Integer besteht, sofern n nicht leicht zu faktorisieren ist:

Eine vertrauenswürdige Instanz veröffentlicht einen RSA-Modulus $n = pq$, wobei n ein Blum-Integer ist. Ein Prover P möchte einem Verifier V gegenüber beweisen, dass er in Besitz eines Geheimnisses ist. Dazu wählt P geheime $s \xleftarrow{\$} \mathbb{Z}_n$, $t_1 \xleftarrow{\$} \{-1, 1\}$ und veröffentlicht $v = t_1 \cdot s^2 \pmod{n}$.

Protokoll:

1. P wählt $r \xleftarrow{\$} \mathbb{Z}_n$, $t_2 \xleftarrow{\$} \{-1, 1\}$, berechnet $x = t_2 \cdot r^2 \pmod{n}$, sendet x an V .
2. V wählt $b \xleftarrow{\$} \{0, 1\}$, sendet b an P .
3. P berechnet $y = r \cdot s^b \pmod{n}$, sendet y an V .
4. V überprüft, ob $y^2 = \pm x \cdot v^b \pmod{n}$ gilt.

- (i) Zeigen Sie die Korrektheit des Protokolls für ehrlichen P und V .
- (ii) Geben Sie die Erfolgswahrscheinlichkeit für einen unehrlichen P an, falls das Protokoll k mal durchgeführt wird.
- (iii) Zeigen Sie durch Angabe eines Simulators, dass die Zero-Knowledge-Eigenschaft gilt.
- (iv) Welche Information würde V über v lernen, falls wir das Protokoll ohne die zufälligen Vorzeichen t_1, t_2 durchführen würden? (Überlegen Sie dazu, was P gegenüber V beweist, falls es keine Vorzeichen gibt)

Lösungsvorschlag zu Aufgabe 2.

- (a) $\left(\frac{15}{35}\right) = \left(\frac{5}{35}\right) \left(\frac{3}{35}\right) = \left(\frac{5}{7}\right) \left(\frac{5}{5}\right) \left(\frac{3}{7}\right) \left(\frac{3}{5}\right) = 0$
 $\left(\frac{32}{33}\right) = \left(\frac{-1}{33}\right) = (-1)^{\frac{33-1}{2}} = (-1)^{16} = 1$
 $\left(\frac{17}{143}\right) = \left(\frac{17}{13}\right) \left(\frac{17}{11}\right) = \left(\frac{4}{13}\right) \left(\frac{6}{11}\right) = \left(\frac{2}{13}\right) \left(\frac{2}{13}\right) \left(\frac{2}{11}\right) \left(\frac{3}{11}\right) = 1 \cdot (2^5 \pmod{11})(3^5 \pmod{11}) = 1 \cdot (-1) \cdot 1 = -1$
- (b) $\left(\frac{-1}{n}\right) = \left(\frac{-1}{p}\right) \left(\frac{-1}{q}\right) = (-1)^{\frac{p-1}{2}} (-1)^{\frac{q-1}{2}} = (-1)(-1) = 1$, da $\frac{p-1}{2}$ bzw. $\frac{q-1}{2}$ jeweils ungerade sind ($p \equiv q \equiv 3 \pmod{4}$) und somit -1 kein quadratischer Rest bzgl p und q und somit auch nicht bzgl n ist.
- (c) (i) Wenn P das Geheimnis s kennt, wird er V davon überzeugen können, da

$$y^2 = (rs^b)^2 = r^2 s^{2b} = t_2 x \cdot t_1 v^b = \pm x v^b \pmod{n}$$

- (ii) Falls ein Angreifer \mathcal{A} sich als P ausgeben möchte ohne s zu kennen, muss er V entsprechende x und y senden, so dass $y^2 = x v^b$. Da er die Wahl von b nicht kennt muss er im Vorfeld ein Bit b' raten um x passend aufzusetzen, indem er ein zufälliges y zieht und $x = y^2 v^{-b'} \pmod{n}$ berechnet. Dieses x schickt er an V und dieser zieht zufällig ein $b \xleftarrow{\$} \{0, 1\}$ und sendet dies an \mathcal{A} . Dieser sendet y an V und dieser verifiziert, ob $y^2 = x v^b = y^2 v^{-b'} v^b$ gilt. Der Angreifer ist erfolgreich, wenn $b' = b$. Dies tritt mit Wahrscheinlichkeit $1/2$ auf und somit hat \mathcal{A} bei k Protokolldurchläufe eine Erfolgswahrscheinlichkeit von $(1/2)^k$.
- (iii) Ein Simulator S , der s nicht kennt, kann gültige Protokolltranskripte simulieren für alle möglichen V , indem er folgendes ausführt:
1. S wählt zufälliges Bit b' , zufälliges $r \xleftarrow{\$} \mathbb{Z}_n$ und berechnet $x = r^2 v^{-b'} \pmod{n}$. Er sendet x an V .
 2. V sendet zufälliges Bit b an S .
 3. Falls $b' = b$ sendet S $y = r$ (siehe (ii)) ansonsten löscht er das Transkript und startet die Simulation von vorne.

- (iv) Durch das Vorzeichen handelt es sich bei $v = t_1 s^2$ im Falle von $t_1 = 1$ um einen quadratischen Rest mit Quadratwurzel s oder für $t_1 = -1$ um keinen quadratischen Rest (siehe (b)). Würde man das Vorzeichen weglassen, würde die Information, dass v zur Menge der quadratischen Reste mod n gehört von P bewiesen werden. Das zweite Vorzeichen t_2 dient dazu, dies auch bei der Überprüfung von V zu verschleiern.

Aufgabe 3. Achtung: Aufgabe wurde geändert! Es gibt keine Set-Befehle und Write-Rechte implizieren **keine** Read-Rechte!

Gegeben sei das folgende System im Bell-LaPadula-Modell:

- Subjektmenge $\mathcal{S} = \{\text{Alice, Bob, Carol}\}$
- Objektmenge $\mathcal{O} = \{D_1, D_2, D_3, D_4\}$
- Menge der Zugriffsoperationen $\mathcal{A} = \{\text{read, write, append, execute}\}$
- Menge der Sicherheitsstufen $L = \{\text{Verwaltung, Lehre, Forschung, Präsidium}\}$ mit der partiellen Ordnung definiert durch $\text{Lehre} \leq \text{Verwaltung} \leq \text{Präsidium}$ und $\text{Lehre} \leq \text{Forschung} \leq \text{Präsidium}$
- Zugriffsmatrix M gegeben durch

	D_1	D_2	D_3	D_4
Alice	r,w,a	r	r,w,a	r,x
Bob	r,w,a	r,w,a	r,w,a	r,x
Carol	r	r	r,w,a	r,w,a,x

- Zuordnung der Sicherheitsstufen $F = (f_s, f_c, f_o)$ gegeben durch

	f_s	f_c		f_o
Alice	Verwaltung	Verwaltung	D_1	Verwaltung
Bob	Forschung	Lehre	D_2	Lehre
Carol	Präsidium	Forschung	D_3	Forschung
			D_4	Präsidium

Geben Sie für die folgende Liste von Zugriffsanforderungen und Anfragen zur Änderung der Sicherheitsstufe an, ob das System Zugriff gewährt beziehungsweise die Sicherheitsstufe ändert oder nicht. Berücksichtigen Sie erfolgreiche Zugriffe und Änderungen der Sicherheitsstufe für die nachfolgenden Zugriffsanforderungen. Geben Sie für abgelehnte Anfragen als Begründung an, welche Eigenschaft(en) verletzt würde(n), wenn der Zugriff beziehungsweise die Änderung der Sicherheitsstufe erlaubt würde. Dabei bezeichne (s, o, l) eine Zugriffsanforderung von Subjekt s auf Objekt o mit der Zugriffsart l . Gehen Sie von einem leeren Initialzustand aus.

1. $(\text{Alice}, D_1, \text{a})$
2. $(\text{Bob}, D_2, \text{w})$
3. $(\text{Bob}, D_3, \text{r})$
4. $(\text{Carol}, D_3, \text{r})$
5. $(\text{Carol}, D_2, \text{w})$
6. $(\text{Alice}, D_2, \text{r})$
7. $(\text{Bob}, D_4, \text{r})$
8. $(\text{Bob}, D_2, \text{a})$

Lösungsvorschlag zu Aufgabe 3.

	Anforderung	Zugriff erteilt/verweigert
1.	$(Alice, D_1, \mathbf{a})$	erteilt
2.	(Bob, D_2, \mathbf{w})	erteilt
3.	(Bob, D_3, \mathbf{r})	erteilt, $f_C(Bob) = f_O(D_3) = \text{Forschung}$
4.	$(Carol, D_3, \mathbf{r})$	erteilt, $f_C(Carol)$ bleibt unverändert
5.	$(Carol, D_2, \mathbf{w})$	verweigert (*- & ds-Eigenschaft)
6.	$(Alice, D_2, \mathbf{r})$	erteilt, $f_C(Alice)$ bleibt unverändert
7.	(Bob, D_4, \mathbf{r})	verweigert (ss-Eigenschaft)
8.	(Bob, D_2, \mathbf{a})	verweigert (*-Eigenschaft)