SoSe16

1 Multiple Choice

Aufgabe 1
a) Bitte geben Sie für die nachfolgenden Multiple-Choice-Fragen eine Begründung für Ihre Wahl an. Bitte beachten Sie, dass keine Punkte vergeben werden, falls die Begründung nicht Ihre angekreuzte Lösung unterstützt. (8 P.)

1. Die Caesar-Chiffre ist eine Substitutionschiffre. **WAHR**
   Begründung: Es werden einzelne Buchstaben vertauscht nach einem bestimmten Schema.
   (Alternativ: Jeder Buchstabe wird durch exakt einen anderen Ersetzt)

2. Die Vigenère-Chiffre ist eine poly-alphabetische Chiffre. **WAHR**
   Begründung: Vigenère-Chiffre ist wie der Caesar-Chiffre (monoalphabetisch), bis auf die Tatsache, dass dabei mehrere Alphabete (poly-) verwendet werden.
   (Alternativ: Jeder Buchstabe wird durch verschiedene andere ersetzt in Abhängigkeit von seiner Position)

3. Das One-Time-Pad ist eine poly-alphabetische Chiffre. **WAHR**
   Begründung: Bei Polyalphabetischen Verfahren wird ein Buchstabe durch verschiedene andere ersetzt, durch welchen genau, wird Beispielsweise mit einem Codewort bestimmt. So auch beim One-Time-Pad. Dort gilt zusätzlich, dass das Schlüsselwort der Länge des Klartextes entspricht.

4. Die Enigma wurde mit Hilfe des Kasiski-Tests geknackt.
   Begründung: **FALSCH**, Alan Turing hat die Enigma geknackt, dies gelang dadurch das einige Fakten vorhanden waren wie z.b. das der Buchstabe nie mit dem selben Chiffriert wurde und andere “regelmäßigkeiten” die es ermöglichten den Enigma zu knacken.
   (Vigenere-Chiffre wurde durch Kasiski Test geknackt)

2 Schutzziele

Aufgabe 2
a) Erklären Sie den Unterschied zwischen Angriffssicherheit (Security) und Betriebssicherheit (Safety). (3 P.)
Security: Schutz gegen Angriffe, meist mit Schadabsicht.
Authentizität, Integrität, Vertraulichkeit, Verfügbarkeit, Verbindlichkeit
Safety: Schutz gegen Fehler von "innen", Gegenmaßnahmen: testen und verifizieren
Ausfallsicherheit

b) Gegeben sind die aufgelisteten Szenarien. Nennen Sie für jedes Szenario das verletzte Schutzziel hinsichtlich der Angriffssicherheit. (6 P.)

1. Bei der Sicherheitsanalyse eines Onlineshops wird festgestellt, dass das aktuelle System anfällig gegen Replay-Attacken ist, bei der eine Bestellung ohne Einwilligung des Benutzers ein zweites Mal getätigt werden kann.
Schutzziel: **Authenticity**

2. Die Chefetage macht sich Sorgen um die Sicherheit der Geschäftsgeheimnisse. Die IT-Abteilung meldet, dass DES genutzt wird, um alle sensiblen Daten zu verschlüsseln. Die Chefetage ist beruhigt.
Schutzziel: **Confidentiality**

3. Eve hat einen Artikel in einem Onlineshop bestellt. Beim Erhalt der Rechnung weigert Eve sich zu bezahlen und behauptet den Artikel nicht gekauft zu haben. Dem Betreiber des Onlineshops ist es nicht möglich zu beweisen, wer der Käufer war.
Schutzziel:**Non-Repudiation**

4. Eve manipuliert die Kommunikation in einem unverschlüsselten Netzwerk und ändert die IP Adressen in Antwortpaketen zur Namensauflösung (DNS).
Schutzziel: **Authenticity: Schutz vor Fälschung von Daten (Spoofing:manipulieren/ vortäuschen)**

3 Hash-Funktionen

Aufgabe 3
a) Erklären Sie die folgenden Begriffe für Hash-Funktionen (6 P.):
1. Einwegfunktion (“pre-image resistance”)
f= x->y und f^-1 und muss in polynomialzeit berechenbar sein. Es darf praktisch nicht möglich sein, zu einem gegebenen Hashwert x0 eine Nachricht m0 zu finden, deren Hashwert x0 ist, das heißt für die h(m0) = x0 gilt.

2. Kollisionsresistent (“collision resistance”)
Nachricht m1 und m2 können sich unterscheiden, aber haben den selben Hashwert.
Wenn es schwer ist m1 und m2 zu finden mit h(m1) = h(m2) “praktisch unmöglich”

3. Schwach kollisionsresistent (“second pre-image resistance”)
Zu einem gegebenem m1 ein m2 finden mit gleichem Hashwert. h(m1) = h(m2)
m1 und m2 haben unterschiedlichen Hash und sind so nicht zu entziffern wenn man einen Hash “errät”

b) Zeigen oder widerlegen Sie: Ist h eine kollisionsresistente Hashfunktion, so ist h ebenfalls eine schwach kollisionsresistente Hashfunktion. (4 P.)

Man nehme **m1**
Schwach Kollisionsresistent: Angreifer bekommt ein festes **m1**, zu dem er ein **anderes m2** finden muss mit selben Hash.
Kollisionsresistent: Angreifer kann sich **m1** und **m2** aussuchen, solange Sxie verschieden sind aber den selben Hash haben.

Daher ist es anzunehmen, dass Kollisionsresistent auch schwach kollisionsresistent ist, da der Angreifer einfach ein **m1** auswählen kann und das **m2** errechnen, bis er den Hash hat.

Alternative:
Wenn h nicht schwach kollisionsresistent wäre, dann könnte man eine beliebige Kollision (x, x′) effizient bestimmen, indem man ein festes x wählt und dazu eine Kollision berechnet.

Kollision:
- Paar (x, x') mit $x \neq x'$ aber $h(x)=h(x')$
- Bei Byteparität z.B. `0011011` und `1001110`

4 Shamir Secret-Sharing

Aufgabe 4

a) Der ”Dealer” hat ein Geheimnis k, das er auf n andere Personen, den ”Shareholdern”, aufteilen möchte, sodass mindestens t der n Shareholder nötig sind, um das Geheimnis k wiederherstellen zu können.

Gegeben: t = 2, n = 7, p = 11, k = 5

Der Dealer wählt: f(x) := 5 + 3x

1. Berechnen Sie die Schlüsselverteilung. (4.5 P.)

Shares $S = \{s_1, s_2, s_3, s_4, s_5, s_6, s_7\}$ t

$s_1 = (1, f(1) \ mod \ p) = (1, \ 8 \ mod \ 11) = (1, 8)$

$s_2 = (2, f(2) \ mod \ p) = (2, \ 11 \ mod \ 11) = (2, 0)$

$s_3 = (3, 3)$

$s_4 = (4, \ 6)$

$s_5 = (5, \ 9)$

$s_6 = (6, 1)$

$s_7 = (7, 4)$

2. Berechnen Sie das Se cret Recovery für s2 und s4. (3.5 P.)

Recovery für $S' = \{s2, s4\} = \{(x_1, y_1), (x_2, y_2)\} = \{(2, 0), \ (4, 6)\}$

Wir berechnen $l_i(0) = \prod_{j=1, j \neq i}^{2} \frac{0 - x_j}{x_i - x_j} \ mod \ p$

$l_1(0) = \frac{-x_2}{x_1 - x_2} = \frac{-4}{2-4} = 2$

$l_2(0) = \frac{-x_1}{x_2 - x_1} = \frac{-2}{4-2} = [-1 \ mod \ 11] = 10$

Daraus können wir k herleiten mit:

$k = f(0) = L(0) = \Sigma y_i \cdot l_i(0)$

Wenn wir die Werte einsetzen, bekommen wir das Geheimnis:

$k = 0 \cdot 2 + 6 \cdot 10 = [60 \ mod \ 11] = 5$

5 Erweiterter Euklidischer Algorithmus

Aufgabe 5

a) Welche Voraussetzung muss erfüllt sein,so dass Sie den erweiterten euklidischen Algorithmus nutzen können? (2 P.)

$ggT(a,b) = z$ muss darstellbar sein als $z = x * a + y * b$

+

b) Bestimmen Sie x, y ε Z derart, so dass 933 · x + 233 · y = 5 ist. Bitte geben Sie die Berechnung so detailliert wie möglich an. Sie müssen die Voraussetzung hier nicht überprüfen. (6 P.)

933 = 4 * 233 + 1

233 = 233 * 1 + 0

=> 1 = 1 * 933 - 4 * 233 | *5

5 = 5 * 933 - 20 * 233

Berechnung mit dem erweiterten Euklidischen Algorithmus:

Iteration	0	1	2	3
Rest	933	233	1	0
Divisor	-	4	233	
X	1	0	1	
Y	0	1	4	

Damit ergibt sich: X= (-1)^2* 1 = 1 und Y = (-1)^(2+1) * 4= 4: 933*1+ 233*(-4) = 1.
Multiplizieren mit 5 ergibt: X = 5 und Y = -20

c) Was ist das inverse Element von 233 in $Z_{933}*$? (2 P.)

<u>Kurze und richtige Lösung</u>

$1 = a * 933 + b * 233$ gesucht ist b

Voraussetzung: ggt(933,233)=1

<u>Erweiterter Euklidischer Algorithmus</u>

$933 = 4 * 233 + 1$ => Voraussetzung erfüllt $1 = 933 - 4 * 233 \ (I)$

$I: \ 1 = 1 * 933 - 4 * 233$ => Inverses Element von 233 in Z_{933}^* ist -4 mod 933 = 929

Anmerkung: -4 ist aber nicht zwischen 0 und 932, daher wäre es glaube zumindest in Mathe1 nicht zu Ende gerechnet? Keine Ahnung wie da der Prof hier ist. Jap, hab vergessen das modulo 933 zu nehmen - danke!

Lösungsweg 1 :

Richtig ist 933-4 = 929 (stand auch mal da, wurde aber anscheinend gelöscht?), lässt sich auch einfach überprüfen: 233 * 4 = 932, also auch 932 mod 933, währenddessen 929 * 233 = 216457, welches mod 933 dann 1 entspricht.

Ausführlich:

$1 \equiv 233 * x \pmod{933}$

$x \equiv 1/233 \equiv 233^{-1} \pmod{933}$

Nun kommt aus b) diese Zeile: 1 = 1 * 933 - 4 * 233

Da wir mit mod 933 rechnen, ist es egal wie oft die 933 in unserem Term vorkommt, damit gilt auch: $1 \equiv$ -4 * 233 (mod 933). Nun fällt auf, dass dieser Term genau dem gesuchten entspricht. Also wissen wir jetzt unser x:

$233^{-1} \pmod{933} \equiv -4$ Jetzt muss noch die -4 zu mod 933 angepasst werden. Daher ergibt sich $-4 \equiv 929 \pmod{933}$. Das multiplikative Inverse ist also 929. Sagt auch WolframAlpha.

Lösungsweg 2: ACHTUNG FALSCH!!

Wir müssen zeigen, dass ein x existiert für das gilt

$1 \equiv 233 * x \pmod{933}$

Das Inverse Element ist nach Aufgabenteil b : y = 4

Wir müssen nur noch zeigen, dass

$ggt(933, 4) = 1$

$933 = 233 * 4 + 1$

Somit ist das gesuchte x=4, denn

4*233 mod 933 = 1 **falsch!!!!**

4*233 = 932

932 mod 933 = 932

Falsch - Begründung:

233*4 = 932

932 mod 933 = 932

932 mod 933 != 1 mod 933

P.S. : An den Roten aus Lösungsweg 1 der meinte "4" wäre falsch :

Wir rechnen hier in einer multiplikativen Gruppe modulo 933 (Link dazu) Also ist sowohl 929, also auch 4 ein inverses. **falsch!!!!**

929 mod 933 != 4 mod 933

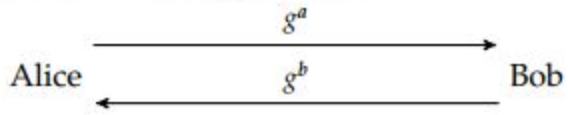
aber **929 mod 933 = - 4 mod 933**

Genau wie jede andere Zahl x die dem schema x * 233 modulo 933 = 1 entspricht.

Ich habe mal unsere Lösungswege klar getrennt, sind aber beide richtig. **Nein, Lösungsweg 2 ist falsch.** Und es tut mir leid, ich hatte deinen alten Lösungsweg entfernt da er absolut nicht nachvollziehbar war.

6 Schlüsselübertragung und Protokolle

Das Diffie-Hellman-Protokoll hat die folgende Form:



$g \in G$ ist ein Erzeuger einer zyklischen Gruppe G der Ordnung p, wobei p prim ist, und $a \in \mathbb{Z}_p^*$ zufällig von Alice gewählt wird, und $b \in \mathbb{Z}_p^*$ zufällig von Bob gewählt wird.

Aufgabe 6

a) Was ist der Unterschied zwischen Schlüsseltransport- und Schlüsselgenerierungsprotokollen? (2 P.)
Schlüsseltransportprotokolle dienen zur vermittlung von Sitzungsschlüsseln über eine Trusted Third Party, also eine vertrauenswürdige Zentrale. (Needham-Schroeder)
Schlüsselgenerierungsprotokolle dienen zur dezentralen Schlüsselgenerierung zwischen zwei Kommunikationspartnern. (Diffie-Hellman)

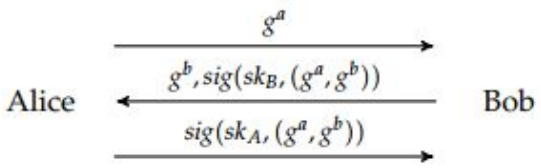
b) Ist das Diffie-Hellman-Protokoll ein Schlüsseltransport- oder ein Schlüsselgenerierungsprotokoll? (1 P.)
Schlüsselgenerierungsprotokoll

c) Skizzieren Sie einen Man-in-the-Middle-Angriff auf das Diffie-Hellman-Protokoll, in dem Sie
• die Beschriftungen der Pfeile im folgenden Bild ergänzen,
• angeben, welche Schlüssel Alice und Bob für die folgende Kommunikation benutzen werden, und
• angeben, wie der Angreifer diese berechnen kann. (5 P.)

Alice benutzt nach diesem Angriff den Schlüssel : g^{ac} .
Bob benutzt den Schlüssel g^{bc} .
Der Angreifer kann diese wie folgt errechnen:
Achtung : Nicht einfach die Buchstaben aus der Folie kopieren, Bob hat hier den Schlüssel g^b und nicht g^d.
In den Folien verwendet der Angreifer auch zwei verschiedene g^x, das dient nur der Einfachheit.

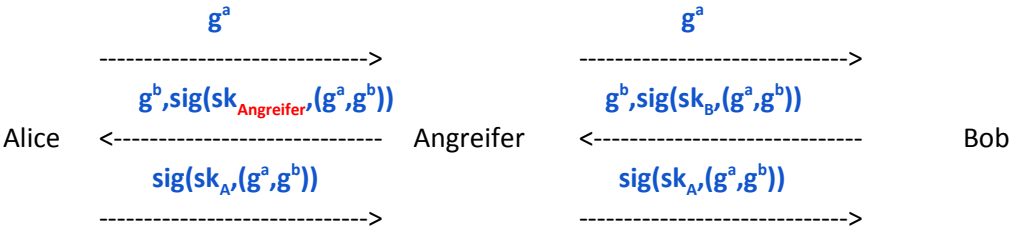
Der Angreifer berechnet so zwei Schlüssel :
Zwischen Alice und Angreifer : g^{ac} und zwischen Bob und Angreifer g^{bc}

d)
Das folgende Protokoll beschreibt eine signierte Schlüsselgenerierung zwischen Alice und Bob im Stile des Station-To-Station-Protokolls, allerdings ohne Verschlüsselung der Signatur in der zweiten und dritten Nachricht.



Skizzieren Sie einen Man-in-the-Middle-Angriff auf dieses Protokoll, in dem Sie die Beschriftungen der Pfeile im folgenden Bild ergänzen. Am Ende dieses Angriffs nehmen Alice und Bob einen gemeinsamen Schlüssel $K = g^{ab}$ als authentifiziert an, den der Angreifer nicht kennt. Allerdings glaubt Alice, diesen Schlüssel mit dem Angreifer ausgehandelt zu haben, während Bob glaubt, diesen Schlüssel mit Alice ausgehandelt zu haben.

(Wie beim Diffie-Hellman-Protokoll gilt: $g \in G$ erzeugt G; $a, b \in \mathbb{Z}_p^*$ werden zufällig von Alice bzw. Bob gewählt; der Signaturschlüssel von Alice, sk_A, ist nur Alice bekannt; sk_B nur Bob; beide kennen die zugehörigen (öffentlichen) Verifikationsschlüssel vk_A und vk_B.) (6 P.)



Danke an den anonymen roten $sk_{Angreifer}$, hier die neue Erklärung :
Die Aufgabe ist ziemlich verwirrend gestellt. Der Angreifer redet mit Alice ganz “legitim”, tut vor Bob aber so als wäre er Alice. Dabei erfährt er nie was a und b ist, bzw. g^{ab}
Da Alice denken soll, dass sie mit dem Angreifer redet muss der Angreifer die Nachrichten an Alice mit seinem eigenen Schlüssel signieren, das geht weil er g^a und g^b kennt.
An Bob leitet er aber einfach alles weiter was von Alice kommt.
Im Endeffekt ist die Authentizität verletzt, da Bob denkt es würde mit Alice reden.

7 Netzwerksicherheit

Aufgabe 7 Firewalls
a) Ordnen Sie die folgenden Schichten im OSI-Model an: Application Layer, Data Link Layer, Network Layer, Physical Layer, Presentation Layer, Session Layer, Transport Layer. Nutzen Sie die vorgegebene Tabelle. (3.5 P.)

Application Layer
Presentation Layer
Session Layer
Transport Layer
Network Layer
Data Link Layer
Physical Layer

Eselsbrücke : PLEASE **D**O **N**OT **T**HROW **S**ALAMI **P**IZZA **A**WAY

b) Auf welchem OSI-Layer ist eine Firewall, die ihre Entscheidungen anhand von Meta-Paketdaten (Quelle, Ziel, Port, Größe) trifft und keine Kenntnis des Protokolls hat?
Bitte kreuzen Sie genau eine Möglichkeite an. (1 P.)

Falsch :

- ☒ Application Layer
- X Transport Layer
- ☐ Data Link Layer
- ☐ Keine der Angegebenen

Es ist das **Transport Layer**, da Application Layer bedeutet es ist schon an TCP/UDP over IP vorbei und nur noch der Inhalt des Pakets landet bei der Application/Anwendung die es empfängt.
Transport Layer bedeutet im Allgemeinen Fall : IP Protocol und darauf TCP oder UDP. In dessen Header findet sich die Hops Anzahl, Source/Dest IP, TCP/UDP Port und Länge des Paketinhalts.

c) Bitte benennen Sie je ein Vor- und Nachteil einer Proxy-Firewall gegenüber einer zustandsbasierten Firewall (2 P.)

Vorteil : Proxy Firewalls erlauben Sicherheit auf einer höheren Ebene, zum Beispiel Schutz gegen gefährliche oder blockierte JavaScript Dateien in Websites oder gegen EMails von Spam/blockierten Adressen.
Nachteil : Proxy Firewalls müssen für jeden Anwendungsfall bzw. Jedes Application Layer Protokoll (zB. HTTP/FTP/SSH) zugeschnitten sein. Man benötigt im worst-case für jedes Protokoll eine eigene Firewall. Zudem kostet die extra Überprüfung zusätzlich Rechenleistung und Zeit.

d) Ist die folgende Aussage korrekt? Trojaner sind Programme, die neben der augenscheinlichen Funktionalität versteckte schadhafte Funktionalität haben.

Ja ein Trojaner ist eine Software die sich einschleust in ein System und versteckte, vom Nutzer ungewollte Funktionen ausführt sobald es z.b. die Rechte von Windows erhält oder installiert wurde.

Alternativ : Trojaner erben ihren Namen vom “Trojanischen Pferd”, da sie Schadcode enthalten aber dem Nutzer als vertrauenswürdig erscheinen oder ihm erst gar nicht auffallen. Sie werden über Sicherheitslücken des Systems (Würmer), Downloads (In kostenlosen Anwendungen versteckt) oder Emails verbreitet (“Rechnung.pdf.exe”).

e) Erklären Sie, was die einzelnen Teile der Befehle bedeuten und was sie machen. (4 P.)
iptables -A INPUT -p tcp –dport 993 -j ACCEPT
iptables -A OUTPUT -p tcp -sport 993 -j ACCEPT

Iptables fügt die Regel zu INPUT hinzu, das im Protokoll TCP auf Destination Port 993 alles akzeptiert (-j) wird
Iptables fügt die Regel zu Output hinzu, das im Protokoll TCP auf source port 993 alles akzeptiert (-j) wird

Es gibt drei sogenannte “Ketten” oder engl. “Chains” die standardmäßig enthalten sind.
Das wären “INPUT, OUTPUT, FORWARD”.
INPUT - Alles was an den Rechner geschickt wurde
OUTPUT - alles was von Anwendungen die auf dem Rechner laufen verschickt wurde
FORWARD - alles was an diesen Rechner reinkommt, aber weitergeleitet bzw. geroutet werden soll (Also Ziel IP ist ein anderer Rechner)
Man kann Regeln an eine Kette anhängen mit “ **-A** “ und auch löschen mit “**-D**”.

Man kann die Regel eingrenzen durch verschiedene Eigenschaften (in beliebiger Reihenfolge) :
Protokolle :“ -p udp | tcp | icmp ”
Source IP (Absender IP) : “ -s *127.0.0.1*”
Destination IP (Ziel IP) : “ -d *8.8.8.8*”
Source Port : “ -sport *21*“
Destination Port : “ -dport *80*”

Zum Schluss gibt man die Aktion an die durchgeführt werden soll :
“ -j ACCEPT | DROP | REJECT “
ACCEPT - Lässt das Paket durch
DROP - Verwirft das Paket ohne dem Sender etwas mitzuteilen
REJECT - Verwirft das Paket, sagt dem Sender aber, dass es abgelehnt wurde

Quellen : http://www.selflinux.org/selflinux/pdf/iptables.pdf und
https://wiki.ubuntuusers.de/iptables2/

F: Welche Adressen sind mit 192.168.100.0/24 gemeint**?**
A : Das heißt die ersten 24 bits der IP Adresse (in Binärdarstellung) sind fest (Netzwerkanteil) und die restlichen 8 bit (IP adresse besteht aus 32 bit) können beliebig sein (Hostanteil).
Also sind mit 192.168.100.0/24 folgende Adresse gemeint: 192.168.100.0 bis 192.168.100.255
Hier ein Online Rechner : http://jodies.de/ipcalc?host=192.168.100.0&mask1=24

f) Gegeben ist der folgende Auszug von iptables.

```
root@ubuntu:/home/test# iptables -L
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT     tcp  --  anywhere             anywhere            tcp dpt:http
ACCEPT     tcp  --  192.168.100.0/24     anywhere            tcp dpt:ssh

Chain FORWARD (policy DROP)
target     prot opt source               destination
ACCEPT     tcp  --  anywhere             anywhere            tcp dpt:http
ACCEPT     tcp  --  anywhere             anywhere            tcp spt:http
DROP       tcp  --  anywhere             anywhere            tcp dpt:http-alt

Chain OUTPUT (policy DROP)
target     prot opt source               destination
ACCEPT     tcp  --  anywhere             anywhere            tcp spt:http
ACCEPT     tcp  --  anywhere             192.168.100.0/24    tcp spt:ssh
```

1. Bitte geben Sie bei den folgenden Paketinformationen jeweils an, ob das Paket akzeptiert (ACCEPT) oder verworfen (DROP) wird. Das System hat die IP 192.168.100.10/24 und alle Pakete laufen über dieses System. Setzen Sie ein Kreuz in die Spalte ACCEPT oder DROP für das jeweils zutreffende. (*Hinweis: http = Port 80; ssh = Port 22; http-alt = Port 8080; * = alle*) (2.5 P.)

Hinweis : "policy DROP|ACCEPT" gib an was passiert wenn keine passende Regel für eine Kette gefunden wurde. Die "Ketten" sind in der Aufgabe oben drüber erklärt.

Quelle	Ziel	Quell-Port	Ziel Port	ACCEPT	DROP
*	8.8.8.8	9000/tcp	53/tcp		**FORWARD policy**
192.168.100.1	192.168.100.10	1000/tcp	**80/tcp**	**_ INPUT Regel 1**	
192.168.101.1	192.168.100.10	1021/tcp	**22/tcp**		**INPUT p0LiCy**
192.168.100.1	2.2.2.2	5100/tcp	**80/tcp**	**FORWARD Regel 1**	
2.2.2.2	192.168.100.1	**80/tcp**	5100/tcp	**FORWARD Regel 2**	

2. Folgender Befehl wird auf dem System ausgeführt: iptables -P INPUT ACCEPT.
Welche Pakete werden nun akzeptiert oder verworfen? (2.5 P.)

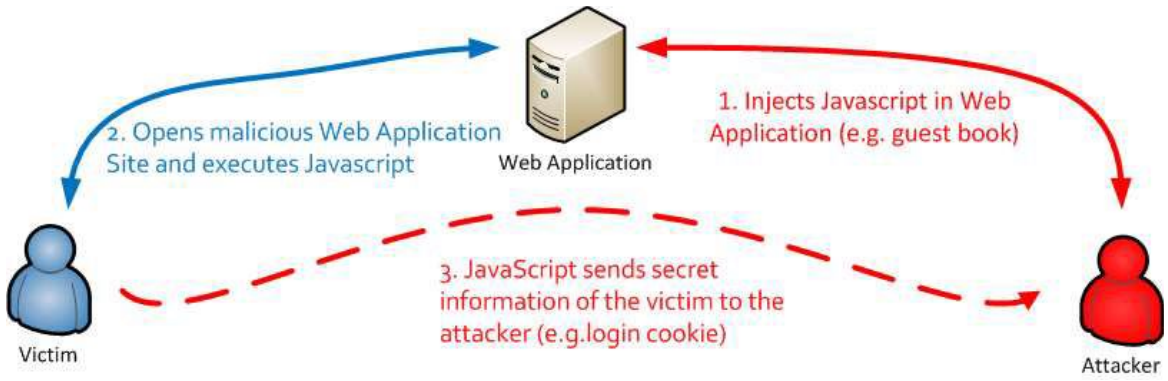
Bemerkung : Die policy von INPUT wurde also geändert, sodass alles angenommen wird.

Quelle	Ziel	Quell-Port	Ziel Port	ACCEPT	DROP
*	8.8.8.8	9000/tcp	53/tcp		**FORWARD policy**
192.168.100.1	192.168.100.10	1000/tcp	80/tcp	**INPUT Regel 1**	
192.168.10**1**.1	192.168.100.10	1021/tcp	**22/tcp**	**INPUT policy(ACCEPT)**	
192.168.100.1	2.2.2.2	5100/tcp	80/tcp	**FORWARD Regel 1**	
2.2.2.2	192.168.100.1	80/tcp	5100/tcp	**FORWARD Regel 2**	

*Fazit : SSH wird jetzt von allen IP Adressen angenommen und nicht mehr von **192.168.100.0/24***

8 Web Sicherheit

Aufgabe 8
a) Skizzieren Sie einen **Persistent-XSS-Angriff.** Erklären Sie die einzelnen Schritte in Ihrer Skizze. Bitte schreiben Sie **keinen Fließtext** ! Nutzen Sie die vorgezeichneten Boxen um die Parteien zu benennen und zeichnen Sie zusätzlich die Verbindungen ein. Bitte verdeutlichen Sie auch die korrekte Reihenfolge des Angriffs. (3 P.)



b)
Geben Sie 2 Voraussetzungen an, die einen Persistent-XSS-Angriff begünstigen. (2 P.)

1) Serverseitige Software mit Sicherheitslücken
2) Userseitige veraltete Browsersoftware

- Kein HTML/JS Escaping (Filtern) von User eingaben in Formularen
  - BB-Codes in Foren für Bilder oder Links :
    [img] img.blabla.net/blabla.png [img] wird zu ****
    [img] img.blabla.net/blabla.png” onclick=”alert(‘hello i bims a xss’); [/img]
    Wird zu ****
  - Wenn man URLs für Bilder eingeben kann, und die bilder mit css eingebunden werden statt HTML : ** **
    Kann man als url “javascript:alert(‘hi’);” eingeben, was zu folgendem führt :
    ** **
- Kein HTML/JS Escaping (Filtern) von User eingaben in URL Parameter
  ww.google.de/**search.php?q=<script>alert(“hi”);</script>**

c) Sie befinden sich auf dem Anmeldeformular IhrerWeb Applkation https://flug-cased.de/login.php. Sie versuchen sich in Ihr System mit dem Benutzernamen karl und dem falschen Kennwort test' einzuloggen. Folgende Fehlermeldung erscheint:
1064: You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near "test"' at line 1; SQL:
SELECT * FROM users WHERE username='karl' and password='test"

1. Nennen Sie zwei für einen Angreifer interessante Eigenschaften, die man von dieser Fehlermeldung ableiten kann. (3 P.)

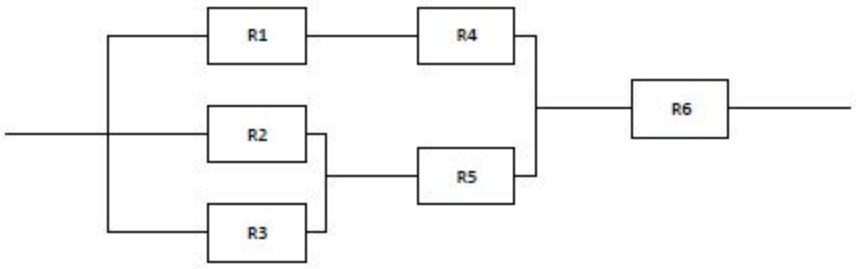
1. Eigenschaft : **Die MysQL Fehlermeldungen werden nicht unterdrückt vom System**
2. Eigenschaft : **Gefährliche Symbole aus Benutzereingaben werden nicht gefiltert**

2. Bitte nennen Sie für jede gefundene Eigenschaft mindestens eine Gegenmaßnahme. Geben Sie jeweils ein Beispiel an. (3 P.)

1. Gegenmaßnahme : **Alle Fehlermeldungen unterdrücken : (PHP)** `error_reporting(0);`
2. Gegenmaßnahme : **Userinput mit regex säubern oder bei PHP mysqli statt mysql bibliothek verwenden.**

9 Reliability

Gegeben sei ein System, dessen Komponenten in der dargestellten Art und Weise voneinander Abhängen.



Aufgabe 9

a) Geben Sie die Reliability-Funktion des Gesamtsystems in Abhängigkeit der jeweiligen Reliability-Funktionen der Komponenten an. Hinweis: Die Formel muss nicht ausmultipliziert werden. Geben Sie auch mögliche Teilergebnisse an. (6 P.)

R(t) = (1 - (1 - R1*R4) * (1 - (1 - (1 - R2)*(1 - R3)) * R5)) * R6

b) Geplante Obsoleszenz: Das Smart TV geht zuverlässig nach genau sieben Jahren kaputt, niemals eher, niemals später. Wie lautet die Reliability-Funktion für das Smart TV? (Eine Zeiteinheit entspricht einem Jahr.) (3 P.)
RSTV(t) =
Bin mir nicht sicher, glaube aber dass es stimmt : R(t) = P(t = 7)

c) Zwei Bit, 1 und 1, wurden mit dem Parity-Bit 1 übertragen. Es gab maximal einen Bitfehler. War die Übertragung erfolgreich? Begründen Sie die Antwort. (1.5 P.)

Wenn wir die zwei Bits “11” übertragen dann soll uns das Parity Bit helfen eine fehlerhafte übertragung zu erkenne. Wie ihr aus der Tafel ablesen könnt ist das Parity bit bei einer erfolgreichen Übertragung (1 1) gleich 0. Aber wie ihr ebenfalls ablesen könnt, ist das Parity bit auch bei einer Fehlerhaften Übertragung (0 0) auch gleich 0. Aber dafür müssten zwei bitfehler auftreten.
Bei nur einem Bitfehler (1 0) oder (0 1) ist das Parity bit gleich 1, und somit wird dabei der Fehler erkannt. Also ist die Übertragung bei maximal einem Fehlerhaften bit erfolgreich.

Paritätsbit (gerade)

X	Y	$P = X \oplus Y$
0	0	0
0	1	1
1	0	1
1	1	0

d) Erklären Sie den Unterschied zwischen Fault Avoidance und Fault Recovery. (2 P.)
-Fehlervermeidung (fault avoidance)

Design des Systems stellt sicher, dass Fehler nicht auftreten
Beispiel: Testen, Verifikation, ...

-Wiederherstellung aus Fehlerzustand (fault recovery)

Strategien, um ein System bei Auftreten eines Fehlers wieder in einen korrekten Systemzustand zu bringen

Klausur WS 15

Einführung Aufgabe 1:

a) Die Vorlesung unterscheidet Sicherheit in zwei Kategorien. Nennen Sie diese und erläutern Sie ihre Abgrenzung, indem Sie nennen, wogegen die beiden Sicherheitskategorien schützen. (2 P.)
Kategorie I: Betriebssicherheit(safety) Schutz gegen: Fehler innerhalb des Systems
Kategorie II: Angriffssicherheit(security) Schutz gegen: aktive Angreifer

b) Gegeben sind die aufgelisteten Szenarien. Nennen Sie für jedes Szenario, welche Sicherheitskategorie (aus Aufgabenteil a) beeinträchtigt ist. (3 P.)

1. Die unübersichtliche Benutzeroberfläche des Email Programms hat dazu geführt, dass der Benutzer eine Email mit vertraulichen Informationen versehentlich unverschlüsselt verschickt hat.

Sicherheitskategorie:
Safety, da das System deshalb kompromittiert werden.

2. Der neue Prozessor ist wesentlich schneller als seine Vorgängerversion, jedoch gibt es eine Befehlssequenz, die den Prozessor überhitzt und zum Absturz bringt. Der Hersteller hat dieses Problem erkannt und nach umfangreichen Tests festgestellt, dass diese Befehlssequenz im Normalbetrieb nie ausgeführt wird.

Sicherheitskategorie: **Security, es ist eine Sicherheitslücke vorhanden die von Angreifern genutzt werden kann.**

c) Nennen Sie 2 Unterschiede zwischen sogenannten “Nation-State-Adversaries”, und privatwirtschaftlichen Angreifern in Bezug auf ihr Verhalten und ihre Ziele. (2 P.)
Nation-State-Adversaries: Entitäten, großes Budget, Ziel: Vorteil/Sicherheit für das Land
Privatwirtschaftliche Angreifer: Firmen, kleineres Budget, Ziel: Vorteil gegen Konkurrenz

-

2 Mathematische Grundlagen der asymmetrischen Kryptographie

Aufgabe 2 Sei n = pq für zwei Primzahlen p und q.
a) Gegeben p und q, wie berechnet man die eulersche Phi-Funktion ϕ auf n? (1 P.)
$\phi(n)$ = **(p-1)*(q-1)**

b) Gegeben a, b $\in$ Z, erhält man durch den erweiterten euklidischen Algorithmus x, y $\in$ Z, sodass- xa + yb = ggt(a, b)
kann man mit dem erweiterten euklidischen Algorithmus das Inverse zu e bestimmen, also d sodass de $\equiv$ 1 (mod $\phi(n)$)? Zeigen Sie die Korrektheit der Vorgehensweise.
(5 P.)

Bestimmen Sie weiterhin (unter Benutzung des erweiterten euklidischen Algorithmus) das Inverse zu e = 3 für den Fall, dass $\phi(n)$ = 10. (3 P.)

c) Sei $\phi(n)$ bekannt. Leiten Sie eine quadratische Gleichung her, also eine Gleichung der Form $p^2 + rp + s = 0$, mit der *n* faktorisiert werden kann, d.h. p, q sind die Lösungen der Gleichung. (10 P.)

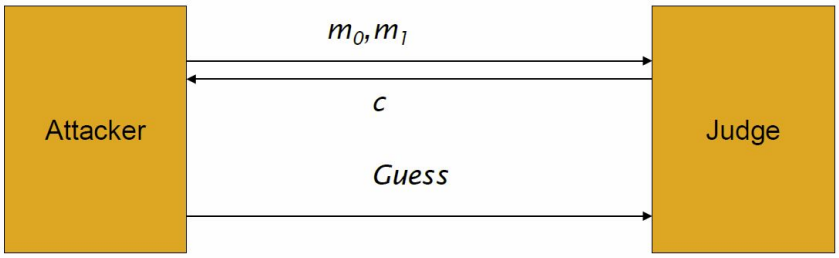
(x-p)(x-q)=0 hat p,q als Lösungen. Umformen:
$0 = x^2 + (-p-q)x + pq$ n=pq
$0 = x^2 + (-p-q)x + n$ phi(n)=(p-1)(q-1)=pq-p-q+1 $\Rightarrow$ -p-q=phi(n)-n-1
$0 = x^2 + (phi(n)-n - 1)x + n$

3 Asymmetrische Verschlüsselungsverfahren

Aufgabe 3
a) Nennen Sie einen Vorteil von asymmetrischer Kryptographie gegenüber symmetrischer Kryptographie. (1 P.)
Keine geheime Kommunikation für Schlüsselübergabe notwendig.

b) Nennen Sie einen Nachteil von asymmetrischer Kryptographie gegenüber symmetrischer Kryptographie. (1 P.)
Aufwendiger/langsamer vor allem bei großen Datenmengen.

c) Semantische Sicherheit:
Die Skizze (4.Vorlesung: „Kryptographie: asymmetrische Verschlüsselung") visualisiert das Spiel für semantische Sicherheit zwischen Angreifer („Attacker") und Richter („Judge").



Wie hat der Richter *c* berechnet ? (2 P.)
Der Richter wählt m0 oder m1 aus und verschlüsselt diese zu c.

Was muss der Angreifer bestimmen, um das Spiel zu gewinnen, also was schickt der Angreifer an den Richter in seiner Vermutung („Guess")? (2 P.)
Der Angreifer muss bestimmen, welche Nachricht (m0 oder m1) der Richter verschlüsselt hat. Wenn seine Vermutung richtig ist, hat er das Spiel gewonnen.

d) Zeigen oder widerlegen Sie: RSA ist semantisch sicher. (6 P.)
~~**Hier nein, weil der Attacker einfach die beiden Nachrichten mit RSA verschlüsseln kann (pubKey) und mit dem empfangenen Chiffrat C vergleichen kann. -> Chiffrat wird RSA(m0) oder RSA(m1) gleichen und er kann den Guess jedesmal richtig treffen**~~
RSA ist nicht semantisch sicher. N gehört zum öffentlichen schlüssel und Primfaktorisierung ist eindeutig lösbar (wenn auch ohne shor-algo nicht in polynomieller zeit). Daraus kann der Angreifer auch auf d kommen. Er ist also im Semantikspiel nicht auf raten angewiesen.
Der Angreifer hat das Semantikspiel nicht gewonnen weil der öffentliche Schlüssel öffentlich ist, sondern weil es bei asymmetrischen verfahren auf den pk ankommt.

e) Sie haben in der Vorlesung zwei Ansätze für die Verwaltung von Zertifikaten kennengelernt:

das *Web-of-Trust* (WOT) und die *Public Key Infrastructure* (PKI). Beide Ansätze haben Vor- und Nachteile. Nennen Sie zwei dieser Vorteile, zwei Nachteile, oder einen Vorteil und einen Nachteil. Geben Sie dabei an, worauf er sich bezieht (schreiben Sie zum Beispiel: Vorteil WOT:...). (4 P.)
WOT:
Vorteil: dezentral
Nachteil: Authentizität nicht gewährleistet

PKI:
Vorteil: Authenthizität, Sicher mehr oder weniger
Nachteile: Alles wird auf einmal hochgenommen wenn PKI atk wird

4 MAC & Verschlüsselung

Im Allgemeinen ist nicht sichergestellt, dass MACs die Nachricht, die sie kodieren, geheimhalten. Im schlimmsten Fall ist davon auszugehen, dass eine gegebene MAC den kompletten Inhalt der Nachricht enthält. Nehmen wir also an, dass $mac' : K \times M \to T$ eine MAC ist, die schwer zu fälschen ist. Dann ist folgende MAC $mac' : K \times M \to T \times M$ auch schwer zu fälschen:

$$mac(k,m) = (mac'(k,m),m).$$

Sei weiterhin enc : K x M -> C eine sichere Verschlüsselungsfunktion, d.h. für alle $m \in M$ gilt, dass es schwer ist, ohne $k \in K$ zu wissen, aus enc(k,m) Informationen über *m* zu ermitteln.
Für *mac*, *mac'* und *enc* nehmen wir an, dass die Schlüssel gleichverteilt zufällig aus K gezogen Werden.

Aufgabe 4
a) Geben Sie die Encrypt-and-MAC-Konstruktion *e&m* zu *mac* und *enc* inklusive ihrem Typ an: (4 P.)
e&m : K x K x M -> **C x T x M**
e&m (kenc, kmac,m) = **(enc(kenc, m), mac'(kmac, m), m)**

b) Demonstrieren Sie, dass *e&m* die Vertraulichkeit einer Nachricht *m* selbst dann nicht garantiert, wenn die Schlüssel aus *K* nur dem Sender und dem Empfänger einer Nachricht bekannt sind. (4 P.)

Aufgrund der Konstruktion von mac(k,m) = (mac'(k,m),m) enthält (enc(kenc, m), mac(kmac, m)) = (enc(kenc, m), mac'(kmac, m), m) trivialerweise m in Klartext.

5 Zugriffskontrolle

Aufgabe 5 Gegeben sei eine DAC-Richtlinie ausgedrückt in einem statischen “Access Matrix Model” bestehend aus

- einer Menge von Subjekten S*,
- einer Menge von Objekten O,
- einer Menge von Rechten R*, und
- einer Abbildung M : S* x O* -> 2^{R*} , der Zugriffsmatrix, welche die Zugriffsrechte statisch beschreibt, das heißt M ist zu jedem Zeitpunkt gleich.

Definieren Sie ein RBAC-System, das die DAC-Richtlinie, welche durch S,O, R und M ausgedrückt wird, simuliert. Zur Erinnerung, ein RBAC-System besteht aus

- einer Menge von Subjekten S,
- einer Menge von Rollen R,
- einer Menge von Zugriffsrechten für Objekte P, sowie einer
- Zuordnung von Benutzer zu Rollen sr : S -> 2^R und einer
- Zuordnung von Rollen zu Zugriffsrechten pr : R -> 2^P .

a) Definieren Sie S und R. Hinweis: Da die DAC-Richtlinie keine explizite Einteilung der Subjekte in Kategorien vornimmt, muss man hier mit dem arbeiten, was über die DAC-Richtlinie bekannt ist. (4 P.)

S := S*

R := S*.O*

Traut sich keiner ran? Das ist Abgefuckt, dass kann einfach keiner :D Dann kommt das hoffentlich mal nicht dran :D

b) Definieren Sie die Zuordnung von Benutzer zu Rollen sr : S -> 2^R , sodass diese konsistent mit S und R ist. (2 P.)

sr(s) := {

c) Definieren Sie die Menge der Zugriffsrechte für Objekte P, so, dass Rechte für den Zugriff auf ein Objekt (im Sinne der DAC-Richtlinie) damit ausgedrückt werden können. (2 P.)

P :=

d) Definieren Sie die Zuordnung von Rollen zu Zugriffsrechten pr : R -> 2^P entsprechend der Zugriffsmatrix M. (3 P.)

pr(r) := {

e) Angenommen, jeder Benutzer nimmt immer alle ihm verfügbaren Rollen gleichzeitig wahr, d.h. *session = { (s, sr(s)) : s ∈ S}* Zeigen Sie, dass dann das RBAC-System die DAC-Richtlinie ausdrückt, d.h. für alle r ∈ R, s ∈ S und o ∈ O gilt

r ∈ M(s, o) genau dann, wenn ∃Rj ∈ R: (s, Rj) ∈ session ^ (r, o) ∈ pr(Rj).
(8 P.)

6 Authentifizierung

6 Authentifizierung
Aufgabe 6 Passwörter sind textbasierte Geheimnisse, die es erlauben, einen legitimen Benutzer zu authentifizieren.
a) Warum sollte man Passwörter nicht im Klartext speichern? (1 P.)
Zugriff auf Datenbank liefert alle Passwörter (welche oft für mehrere Logins verwendet werden)

b) Warum genügt es, den Hashwert eines Passworts zu speichern, um einen legitimen Benutzer zu authentifizieren? (2 P.)
Hashes vergleichen ist wegen der kollisionsresistenz Hashfunktion ausreichend.

c) In der Praxis fügt man häufig dem Hashwert noch einen Salt s hinzu.
Was versteht man in diesem Zusammenhang unter Salt? (1 P.)
Salt ist eine zufällig gewählte Zeichenfolge die vor der Hashfunktion an den Klartext angehängt wird.

Ein Eintrag für den Benutzer b mit dem Passwort p und Salt s soll gespeichert werden, unter Benutzung der Hashfunktion h. Wie lautet der Eintrag? Bitte einsetzen.
(3 P.)

(b, **h**(p+s), **s**)

Müsste es hier nicht (b, h(p+s), s) heißen? - In Vorlesung 5 Folie 27 heißt es: "Zur Verbesserung der Sicherheit verwendet man Salts: das Passwort wird mit einem zufälligen Wert (Salt) kombiniert und dann gehasht; zudem wird der Salt-Wert gespeichert"

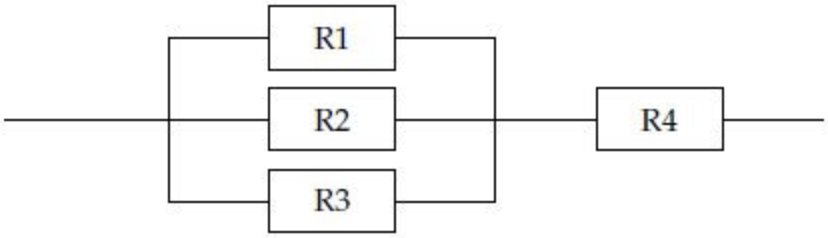
d) Warum ist es besser, statt einer Hashfunktion eine Schlüsselableitungsfunktion wie bcrypt zu benutzen? (2 P.)
Erschwert das Erstellen von Rainbowtables durch den deutlich erhöhten Zeitaufwand/Rechenaufwand (verglichen mit normalen Hashfunktionen).

e) Welche Eigenschaft(en) von Hashfunktionen stellt/stellen sicher, dass es schwer ist, auch nur einen Teil des Passworts aus dem Hashwert des Passworts zu errechnen?
(4 P.)

- **Kollisionsresistenz ("collision-resistent"),** Richtig
- Schwache Kollisionsresistenz ("second-pre-image resistant") ,
- **Eigenschaft der Einwegfunktion ("pre-image resistant")** Richtig oder
- keine der genannten Eigenschaften.

7 Reliability

Gegeben sei ein System, dessen Komponenten in der dargestellten Art und Weise voneinander abhängen.



Aufgabe 7
a) Geben Sie die Reliability-Funktion des Gesamtsystems in Abhängigkeit der jeweiligen Reliability-Funktionen der Komponenten an. (3 P.) Hinweis: Die Formel muss nicht ausmultipliziert werden.
$R_{ges}(t)$ = (1 - (1 - R1) * (1-R2) * (1-R3)) * R4

b) Geplante Obsoleszens: Das iTelefon der Firma Birne geht zuverlässig nach genau drei Jahren kaputt, niemals eher, niemals später. Wie lautet die Reliability-Funktion für das iTelefon? (Eine Zeiteinheit entspricht einem Jahr.) (3 P.)
$R_{iTel}(t)$ =e^(-1/3t) lambda= 1 Fehler/3 Jahre

Ich bin nicht sicher aber denke mal wie folgendes
R (ges) = die Summe von 1 bis 3 (1,i) binomialform(R(t) hoch i). (1 - R(t)) hoch 3 - i

Kann das jemand bitte iwie schritt für schritt erklären wie man das macht ?

Ich stimme für diese Aufgabenstellung ebenfalls der orangenen Antwort zu, aber hier der Lösungsweg, wie man auf die schwarze Funktion kommen würde.
Es gilt $MTTF = \int_0^\infty R(t)\,dt$ und die Aufgabenstellung (abgesehen von den "niemals wieder…") gibt an
MTTF=3. Also ist eine Lösung für R(t) zu finden mit der Gleichung $3 = \int_0^\infty R(t)\,dt$ (Rest ist simple Analysis)

Alternative Antwort (Bitte um Prüfung): Ist hier nicht eher eine sehr einfache unstetige Funktion gemeint, wegen den beiden "*niemals*"?
R(t) = 1 für x in [0,3), und
R(t) = 0 für x in (3,unendlich).

c) Zwei Bit, 0 und 1, wurden mit dem Parity-Bit 1 übertragen. Es gab maximal einen Bitfehler. War die Übertragung erfolgreich? Begründen Sie die Antwort. (1 P.)
Nein,
11 xor = 0
00 xor = 0 nicht gleich 1

Alternative Antwort (Bitte um Prüfung): Nicht ganz klar, da nicht genauer spezifiziert ist, um welche Art von Parität es sich handelt: ”*Odd parity*” oder “*even parity*”? Im Fall von gerader Parität war die Übertragung erfolgreich, aber im anderen Fall nicht.

Ich denke es geht hier um die gerade Parität wie in Vorlesung 12 Folie 37 beschrieben. Hierbei muss der Empfänger der Nachricht prüfen ob x xor y xor p = 0, dann war die Übertragung erfolgreich. Das ist hier der Fall 0 xor 1 xor 1 = 0.