Rechnernetze Dokumentation

Sheraz Azad und Sven Marquardt

Wintersemester 2015/16

Inhaltsverzeichnis

ALLE BILDER UND TABELLEN MÜSSEN IN DEN TEXTFLUSS EINGEBUNDEN WERDEN

1 Schichtenmodell

1.1 Überlegungen für das Spiel "Vier gewinnt"

Die Kommunikation untereinander findet mittels einer Münze statt, welche hochgehalten wird, wobei wir den Binärcode verwenden. Zeigt die Münze Kopf stellt diese die 0 dar und zeigt die Münze Zahl stellt sie die 1 dar. Damit die beiden Positionen unterscheidbar sind, wird die Münze pro Position, also entweder Kopf (0) oder Zahl (1), jeweils für 2 Sekunden hochgehalten erst danach findet ein Positionswechsel statt.

Damit eine geregelte und sinnvolle Kommunikation zwischen den Kommunikationspartnern stattfinden kann, wurden Kommunikationsregeln festgelegt. Das Spiel "Vier gewinnt" hat 7 Reihen mit jeweils 6 Feldern, da wir bei diesem Spiel nur die Spaltenangabe brauchen um unseren Spielzug zu machen, wurden 3 Bits verwendet von 001 (1) bis 111 (7) welche die einzelnen Spalten darstellen.

HIER EIN BILD VON EINEM VIER GEWINNT SPIELFELD HIER FEHLT NOCH WIE MAN ENTSCHEIDET WER ALS ERSTES DRAN IST

Weitere Bitcodierungen für die Kommunikation sind:

Bitfolge	Bedeutung
001	Reihe voll
110	Gewonnen
101	Unentschieden
011	Weiter
100	Fertig
111	Nochmal bei Fehlübertragung

HIER EIN BEISPIEL FÜR DEN SPIELFLUSS DEN ENTWEDER ICH IN LÜBECK HABE

Tabelle 1: Hybrides Modell

5. Anwendungschicht	Das Spiel "Vier gewinnt"
4. Transportschicht	Wird nicht verwendet
3. Vermittlungsschicht	Wird nicht verwendet
2. Sicherungsschicht	Übersetzen der Kommunikation in Bits und Fehlererkennung
1. Bitübertragungsschicht	Übertragung von 0 und 1 durch Medium Münze

1.2 Vor- und Nachteile der Realisierung

Zweierteam mit dem die Analyse der Spielrealisierung gemacht wurde bestand aus Malte Grebe und Niklas Klatt.

Vorteile:
Aufgrund der Kommunikationsregeln ist das Spiel leicht zu verstehen und zu bedienen. Durch die ständige Überprüfung wird dafür gesorgt, das keine Fehler bei der Übertragung auftreten. Dadurch das ein Weiter (011) erwartet wird, gibt es Spielpausen und man kann in Ruhe sein Spielfeld aktualisieren.

Nachteile:
Ein Zug dauert ca. eine Minute, da jede Position zwei Sekunden gehalten wird. Spieler 1 oder 2 fängt zu früh mit der Übertragung vom nächsten Spielzug an, dadurch gibt es eine Fehlerübertragung die wiederholt werden muss.

Verbesserte Spielrealisierung:
Einführung einer Spielfeldsynchronisierung um sicherzustellen, das keine Fehler beim Eintragen der Positionen eingetreten sind.

1.3 Verbesserte Kommunikation durch Stifte

Es gibt zwei Varianten die Kommunikation druch Stifte zu verbessern.

Die **erste Variante** ist, das man einen waagerechten Sitft als 0 und einen senkrechten Stift als 1 interpretiert. Dadurch lassen sich die drei Kommunikationsbit leicht, schnell und eindeutig darstellen.

HIER EINE ZEICHNUNG MIT HALBWINKEL UND DIE BITS

ZU DEN WINKELN

Die **zweite Variante** ist, dass man die Stifte in bestimmten Winkel hinlegt. Hier können wir zum Beispiel sagen das wenn der Stift in einem 90 Grad Winkel liegt, dieser die Bitfolge 001 für Reihe voll darstellt. So können wir die verschiedenen Bitfolgen angeben und bräuchten mit dieser Variante sogar nur einen Stift statt drei.

Diese Fragestellung bezieht sich auf die Bitübertragungsschicht, da sie für die Übertragung von Informationen (Bits 0 und 1) zuständig ist.

1.4 Kommunikation durch Klatschen

Problem
Dadurch das alle Teams zeitgleich angefangen haben zu klatschen, konnte man nicht unterscheiden ob das Klatschgeräusch vom gegenüber sitzenden Kommunikationspartner kam, oder von einem Kommilitonen aus einer anderen Gruppe. Aufgrund dieser Tatsache sind bei allen Teams Fehler bei der Kommunikation entstanden.

Lösung
Auch hier gibt es zwei Lösungsansätze, die sich auf die Medienzugriffskontrollte aufbauen, in der dann nur eine Gruppe zur Zeit kommunizieren darf. Diese Möglichkeiten sind.

1. Ohne Koordinator: Jeder Gruppe im Raum wird eine zufällige Wartezeit in Sekunden zugeteilt, die sie abwarten müssen um kommunizieren zu können. Tritt der Fall auf das zwei oder mehrere Gruppen zur selben Zeit kommunizieren wollen, wird eine Wartezeit aus einem größeren Zeitintervall genommen um diesen Fall zu umgehen. Je nach Wichtigkeit könnte man hier den jeweiligen Gruppen eine Wartezeit aus einem kleinen Zeitintervall zu weisen, als dem Rest der Gruppen.

2. Mit Koordinator: Bei diesem Lösungansatz gibt es einen Koordinator im Raum, der die Anfragen der Gruppen, die kommunizieren wollen, an sich nimmt und stellt dann eine nach seinen Kriterien faire Reihenfolge fest, in der die Gruppen dann untereinander kommunizieren dürfen. Die Reihenfolge

hängt natürlich je nach Wichtigkeit der Gruppen ab und wird vom Koordinator behandelt.

Diese Fragestellung bezieht sich auf die Sicherungsschicht, da sie für die zuverlässige Übertragung von Informationen von einem Teilnehmer zum anderen Teilnehmer zuständig ist.

1.5 Kommunikation mit beliebigen Teilnnehmern

Wenn man davon ausgeht das jeder Teilnehmer dieselben Kommunikationsregeln hat, vergibt man jedem Teilnehmer eine eindeutige Adresse. Möchte man nun einen anderen Teilnehmer kontaktieren, muss man die zu übermittelende Nachricht adressieren. Die beinhaltenden Informationen der Nachricht bestehen aus Sender, Empfänger und Nachricht. Hierbei muss beachtet werden, das bevor man den Kontakt zu einem Teilnehmer aufnehmen möchte, vor Beginn des Spiels eine Kontaktaufnahme erfolgen muss die vom Empfänger bestätigt wird und erst dann kann das Spiel beginnen.

Diese Fragestellung bezieht sich ebenfalls auf die Sicherungsschicht, da sie für die zuverlässige Übertragung von Informationen von einem Teilnehmer zum anderen Teilnehmer zuständig ist.

1.6 Kommunikation mit bestimmten Teilnehmern

Auch hier bekommt jeder Teilnehmer eine **eindeutige** Adresse, wobei diese jedoch noch die Informationen Gebäude-, Raum-, Reihen- und Sitznummer beinhalten. Die Nachricht wird somit anhand dieser ausführlichen Informationen an den jeweiligen Teilnehmer gesendet.

Diese Fragestellung bezieht sich auf die Vermittlungsschicht, da jeder Teilnehmer aufgrund der eindeutigen Adresse mit jedem anderenen (bestimmten) Teilnehmer kommunizieren kann.

1.7 Datenfluss Problem

Problem
Bei einem worst-case-scenario erhält ein spezieller Teilnehmer soviele Informationen von anderen Teilnehmern, das er keinen Platz bzw keine Zeit mehr hat sich die Informationen zu notieren. Folglich gehen dadurch Informationen verloren und diese werden ein weiteres mal an denselben Teilnehmer gesendet, welches das Problem in die Länge zieht.

Lösung
Um den Informationsfluss zu stoppen oder zu kontrollieren schickt der spezielle Teilnehmer bei Bedarf eine Nachricht an die Teilnehmer das diese entweder langsamer senden sollen oder nur eine bestimmte Anzahl an Informationen senden dürfen. Bleibt das Problem weiterhin bestehen, schickt der spezielle diese Nachricht solange bis der Informationsfluss für ihn verarbeitbar ist. Außerdem können weitere spezielle Teilnehmer eingeteilt werden um den Informationsfluss an mehrere spezielle Teilnehmer zu verteilen und dadurch den Arbeitsaufwand eines speziellen Teilnehmers zu senken.

Diese Fragestellung bezieht sich auf die Transportsschicht, da es sich hierbei um eine Staukontrolle handelt, damit ein Teilnehmer seine Informationsrate zu schicken mindert, wenn das Netz überlastet ist.

2 Zuverlässige Datenübertragung

2.1 Messung der Häufigkeit von Rahmenverlusten

Jedes mal wenn der Client einen Rahmen sendet erhöht der Server einen Counter für einen empfangenen Rahmen um eins. Nachdem die Übertragung statt gefunden hat, kann der Server nun überprüfen ob es einen Rahmenverlust gab, indem er die Datei aufruft und die Differenz zwischen der Länge der Datei und dem Counter berechnet.
Rahmen insgesamt - Empfangene Rahmen = Verlorene Rahmen.
Wenn die Differenz null beträgt, dann ist kein Rahmenverlust aufgetreten.

2.2 Messung der Bitfehlerrate

2.3 Vorüberlegungen zum sicheren Protokoll

Um Rahmenverluse und Bitfehler zu kompensieren, gibt es viele Möglichkeiten, wobei folgende Maßnahmen in der zu diesem Pariktum zugehörigen Vorlesung und in anderen Vorlesungen wie Verteile Systeme vorgestellt wurden.

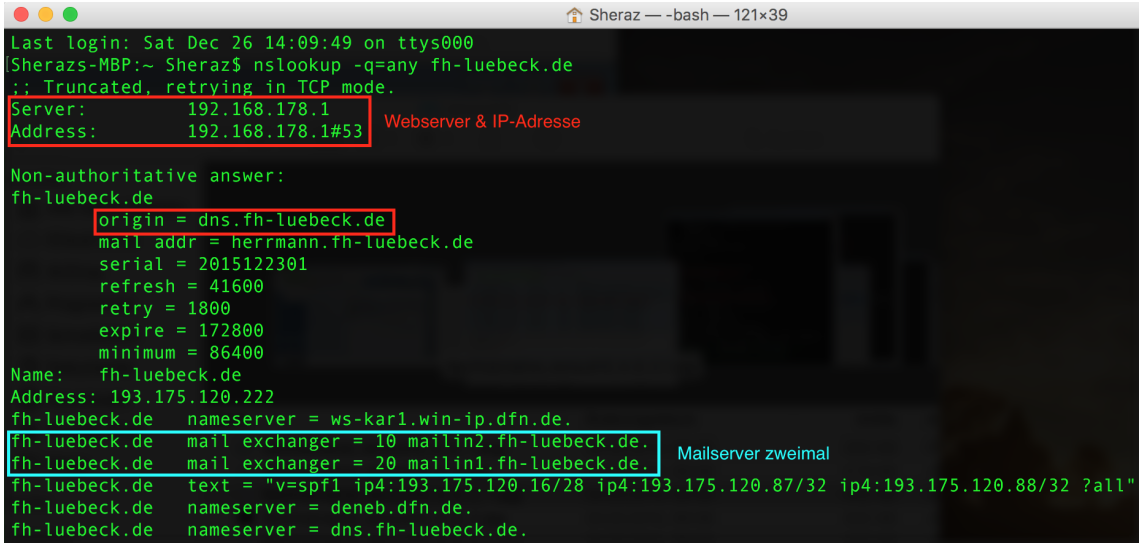
1. Die einfachste Möglichkeit ist einen **Timer** bei senden von Daten zu starten, welcher auf der Serverseite implementiert wird. Wenn nach einer bestimmten keine Rückmeldung vom Client empfangen wird, dann schickt der Server dieses Datenpaket noch einmal. Die Rückmeldung ist das Prinzip der Quittungen (Acknowledgements), welche entweder postiv (Datenpaket angekommen) oder negativ (Datenpaket nicht angekommen) sein können.

2. Passend zu dem Prinzip der oben genannten Quittungen, könnte man auch das dazu passende **Stop-and-Wait Protokoll** implementieren. Hierbei darf der Sender nur dann ein Datenpaket senden, wenn er von dem zuvor gesendeten Datenpaket eine Quittung vom Empfänger erhält. Falls die Quittung negativ ist wird das Datenpaket erneut gesendet, ansonsten das nächste Datenpaket welches an der Reihe ist, wie oben schon erwähnt wurde.

3. Um Duplikate zu verhindern kann man **Sequenznummern** verwenden. Wenn z.B. die Quittung vom Empfänger an den Sender verloren geht und dieser nach dem Tiemout das Datenpaket erneut sendet, hat der Empfänger redundante Datenpakete. Dieser kann dann anhand der Sequenznummern das er dieses Datenpaket schon empfangen hat, verwirft das zweite redundante Datenpaket und verschickt erneut eine positive Quittung an den Sender.

4. Eine weitere Methode ist es ein **Paritätsbit** (Parity byte) zu verwenden. Dieses Paritätsbit wird an die Bits des Datenpakets angehängt und berechnet sich aus der Summe aller Bits. Wenn die Summer der Bits gerade ist, ist das Paritätsbit "0", ist die Summe jedoch ungerade, ist das Paritätsbit "1". Das Problem mit dem Paritätsbit jedoch ist das Fehler eventuell nicht entdeckt werden. Haben wir zum Beispiel Bytefolge "01000001" ergibt sich das Paritätsbit "0" dafür. Entstehen hier jedoch zum Beispiel zwei Bitfehler wird aus der Bytfolge "10000010", die Bytefolge "10101010" aus dem sich ebenfalls das Paritätsbit "0" berechnen lässt. Allgemein bedeutet das, dass sich eine gerade Anzahl von Fehlern aufhebt und nur eine ungerade Anzahl von Fehlern das Paritätsbit ändert.

2.4 Verhinderung von Rahmenverlusten



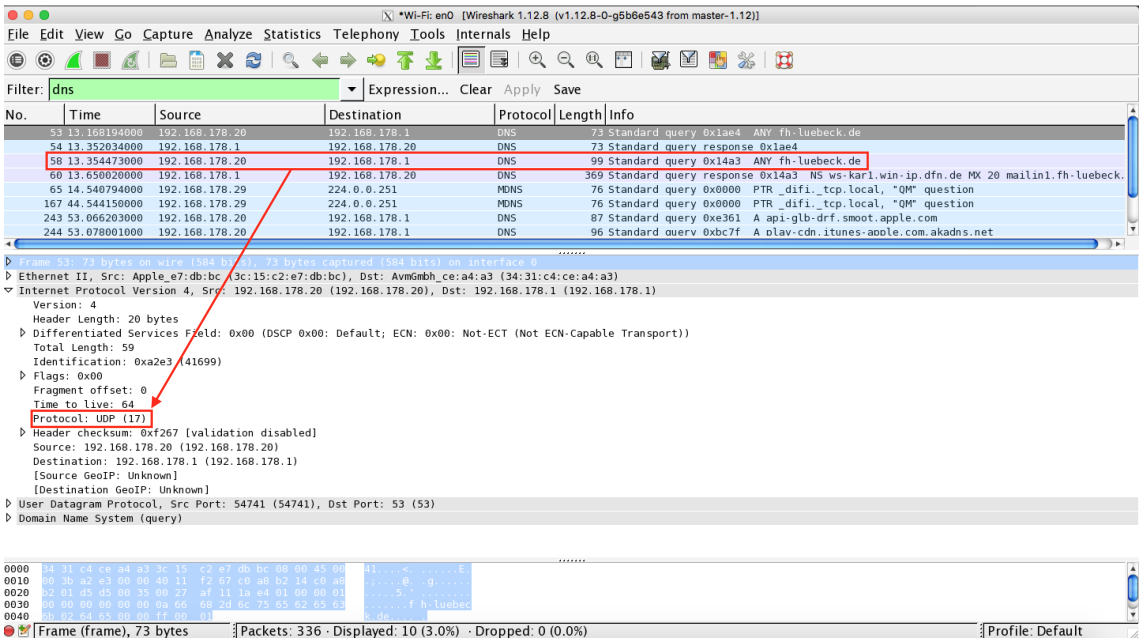
3 Anwendungsschicht und Tools

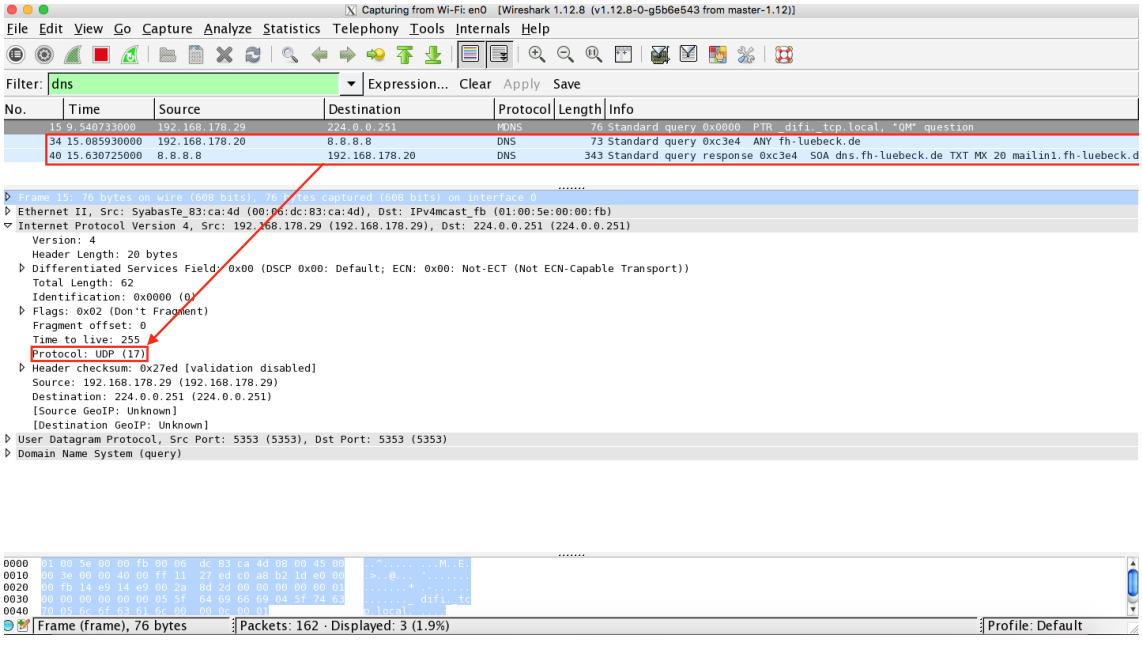
3.1 nslookup tool

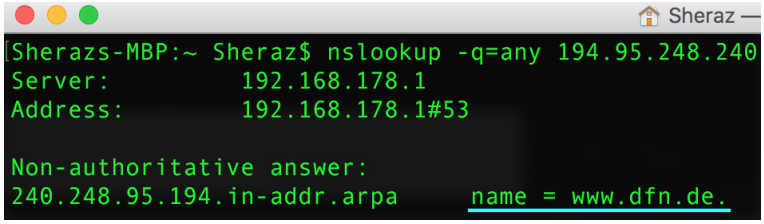
(Verwendet wurde das Terminal von einem MacBook Pro mid 2014 mit dem OS El Captian)

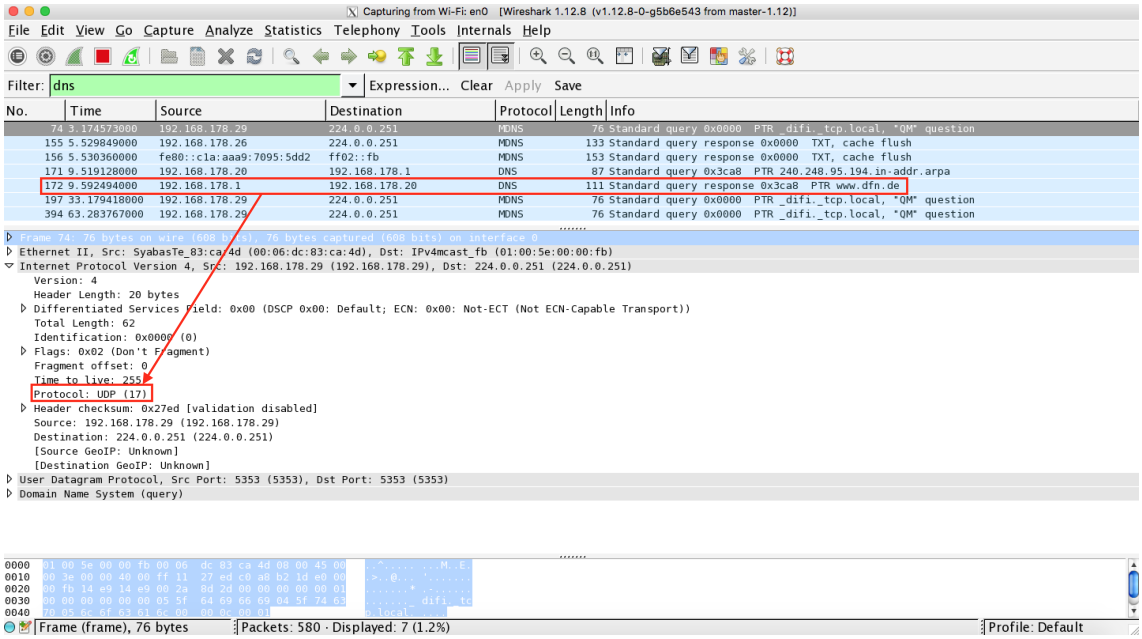
Bei dem Befehl **nslookup -q = any fh-luebeck.de** erhält man die IP-Adresse *192.168.178.1* für den Server und *fh-luebeck.de mail exchanger = 20 mailin1.fh-luebeck.de. und fh-luebeck.de mail exchanger = 10 mailin2.fh-luebeck.de.* für die Mail Server.

Aufgrund einer Aktualisierung konnte mit dem Befehl **nslookup -q = any fh-luebeck.de 8.8.8.8 / 8.8.4.4** konnte keine Unterschied weder am Macbook noch an den Laborrechnern festgestellt werden. Aus Neugier wurden Informationen bezüglich des Unterschiedes der beiden Anfragen, von Kommilitonen aus dem höheren Semester nachgefragt. Der offensichtliche Unterschied der beiden Anfragen ist, dass beim zweiten Befehl nicht nur der Hostname-Parameter sondern auch der Server-Parameter eingegeben wurde. 8.8.8.8/8.8.4.4 ist ein öffentlich zugänglicher DNS Server von Google, der auch Informationen über die FH-Lübeck gespeichert hat, welche wir dann über diesen öffentlichen DNS Server bekommen.


*Wi-Fi: en0 [Wireshark 1.12.8 (v1.12.8-0-g5b6e543 from master-1.12)]
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Filter: dns
Expression... Clear Apply Save
No. Time Source Destination Protocol Length Info
53 13.168194000 192.168.178.20 192.168.178.1 DNS 73 Standard query 0x1ae4 ANY fh-luebeck.de
54 13.352034000 192.168.178.1 192.168.178.20 DNS 73 Standard query response 0x1ae4
58 13.354473000 192.168.178.20 192.168.178.1 DNS 99 Standard query 0x14a3 ANY fh-luebeck.de
60 13.650020000 192.168.178.1 192.168.178.20 DNS 369 Standard query response 0x14a3 NS ws-kar1.win-ip.dfn.de MX 20 mailin1.fh-luebeck.
65 14.540794000 192.168.178.29 224.0.0.251 MDNS 76 Standard query 0x0000 PTR _difi._tcp.local, "QM" question
167 44.544150000 192.168.178.29 224.0.0.251 MDNS 76 Standard query 0x0000 PTR _difi._tcp.local, "QM" question
243 53.066203000 192.168.178.20 192.168.178.1 DNS 87 Standard query 0xe361 A api-glb-drf.smoot.apple.com
244 53.078001000 192.168.178.20 192.168.178.1 DNS 96 Standard query 0xbc7f A play-cdn.itunes-apple.com.akadns.net
Frame 53: 73 bytes on wire (584 bits), 73 bytes captured (584 bits) on interface 0
Ethernet II, Src: Apple_e7:db:bc (3c:15:c2:e7:db:bc), Dst: AvmGmbh_ce:a4:a3 (34:31:c4:ce:a4:a3)
Internet Protocol Version 4, Src: 192.168.178.20 (192.168.178.20), Dst: 192.168.178.1 (192.168.178.1)
Version: 4
Header Length: 20 bytes
Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00: Not-ECT (Not ECN-Capable Transport))
Total Length: 59
Identification: 0xa2e3 (41699)
Flags: 0x00
Fragment offset: 0
Time to live: 64
Protocol: UDP (17)
Header checksum: 0xf267 [validation disabled]
Source: 192.168.178.20 (192.168.178.20)
Destination: 192.168.178.1 (192.168.178.1)
[Source GeoIP: Unknown]
[Destination GeoIP: Unknown]
User Datagram Protocol, Src Port: 54741 (54741), Dst Port: 53 (53)
Domain Name System (query)
0000 34 31 c4 ce a4 a3 3c 15 c2 e7 db bc 08 00 45 00
0010 00 3b a2 e3 00 00 40 11 f2 67 c0 a8 b2 14 c0 a8
0020 b2 01 d5 d5 00 35 00 27 af 11 1a e4 01 00 00 01
0030 00 00 00 00 00 00 0a 66 68 2d 6c 75 65 62 65 63
0040 6b 02 64 65 00 00 ff 00 01
Frame (frame), 73 bytes
Packets: 336 · Displayed: 10 (3.0%) · Dropped: 0 (0.0%)
Profile: Default


Capturing from Wi-Fi: en0 [Wireshark 1.12.8 (v1.12.8-0-g5b6e543 from master-1.12)]
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Filter: dns
Expression... Clear Apply Save
No. Time Source Destination Protocol Length Info
15 9.540733000 192.168.178.29 224.0.0.251 MDNS 76 Standard query 0x0000 PTR _difi._tcp.local, "QM" question
34 15.085930000 192.168.178.20 8.8.8.8 DNS 73 Standard query 0xc3e4 ANY fh-luebeck.de
40 15.630725000 8.8.8.8 192.168.178.20 DNS 343 Standard query response 0xc3e4 SOA dns.fh-luebeck.de TXT MX 20 mailin1.fh-luebeck.d
Frame 15: 76 bytes on wire (608 bits), 76 bytes captured (608 bits) on interface 0
Ethernet II, Src: SyabasTe_83:ca:4d (00:06:dc:83:ca:4d), Dst: IPv4mcast_fb (01:00:5e:00:00:fb)
Internet Protocol Version 4, Src: 192.168.178.29 (192.168.178.29), Dst: 224.0.0.251 (224.0.0.251)
Version: 4
Header Length: 20 bytes
Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00: Not-ECT (Not ECN-Capable Transport))
Total Length: 62
Identification: 0x0000 (0)
Flags: 0x02 (Don't Fragment)
Fragment offset: 0
Time to live: 255
Protocol: UDP (17)
Header checksum: 0x27ed [validation disabled]
Source: 192.168.178.29 (192.168.178.29)
Destination: 224.0.0.251 (224.0.0.251)
[Source GeoIP: Unknown]
[Destination GeoIP: Unknown]
User Datagram Protocol, Src Port: 5353 (5353), Dst Port: 5353 (5353)
Domain Name System (query)
0000 01 00 5e 00 00 fb 00 06 dc 83 ca 4d 08 00 45 00
0010 00 3e 00 00 40 00 ff 11 27 ed c0 a8 b2 1d e0 00
0020 00 fb 14 e9 14 e9 00 2a 8d 2d 00 00 00 00 00 01
0030 00 00 00 00 00 00 05 5f 64 69 66 69 04 5f 74 63
Frame (frame), 76 bytes
Packets: 162 · Displayed: 3 (1.9%)
Profile: Default





Hinter der DNS *194.95.248.240* verbirgt sich die Website vom "Deutschen Forschungsnetz" (www.dfn.de).

3.2 Aufbau von Websiten

Beim Aufruf der beiden Webseiten *www.fh-luebeck.de* und *www.t-online.de* mit aktivierten AddOns, wurden bei beiden mittels *"IPvFox"* Hosts angezeigt die beim Aufruf der Webseiten als Ressourcen mit geladen wurden. Die Liste der Hosts von t-online.de war um ein dreifaches größer als der von fh-luebeck.

Nach und nach werden weitere Skripte mit *"NoScript"* erlaubt/zugelassen, wobei auf beiden Websiten nun einige Hosts mehr geladen werden. Außerdem

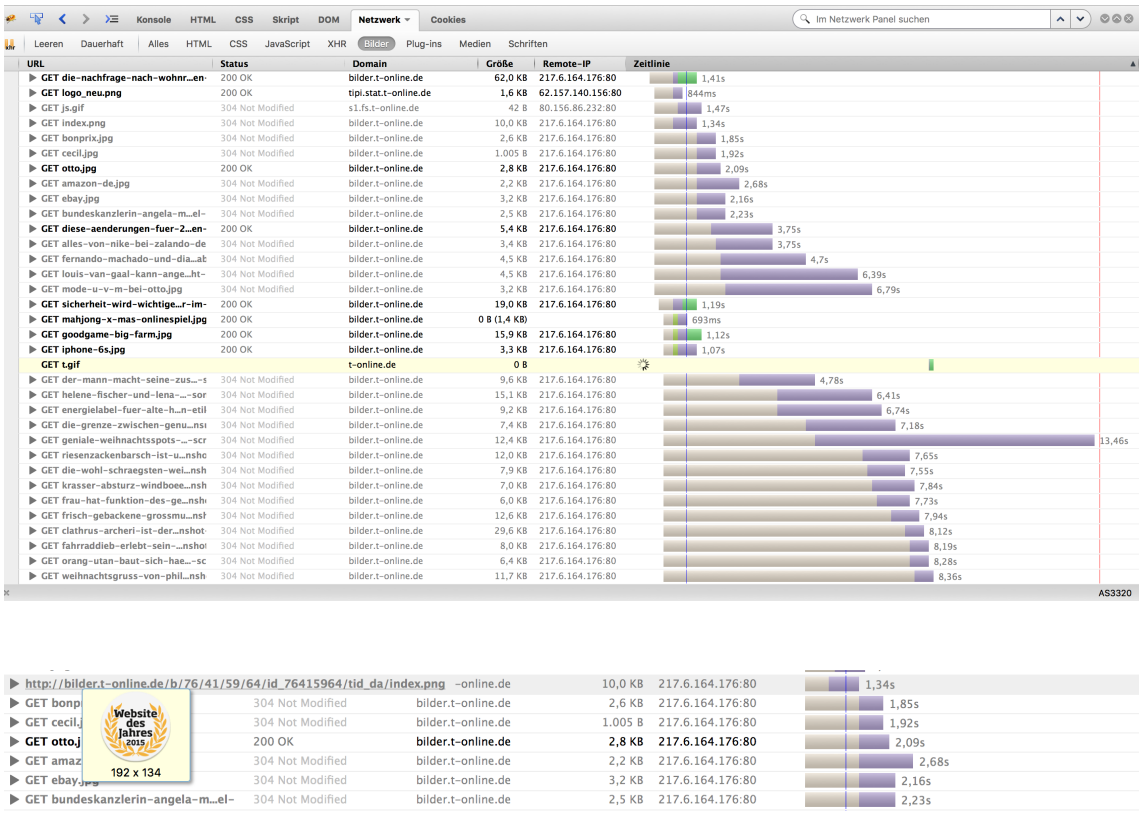


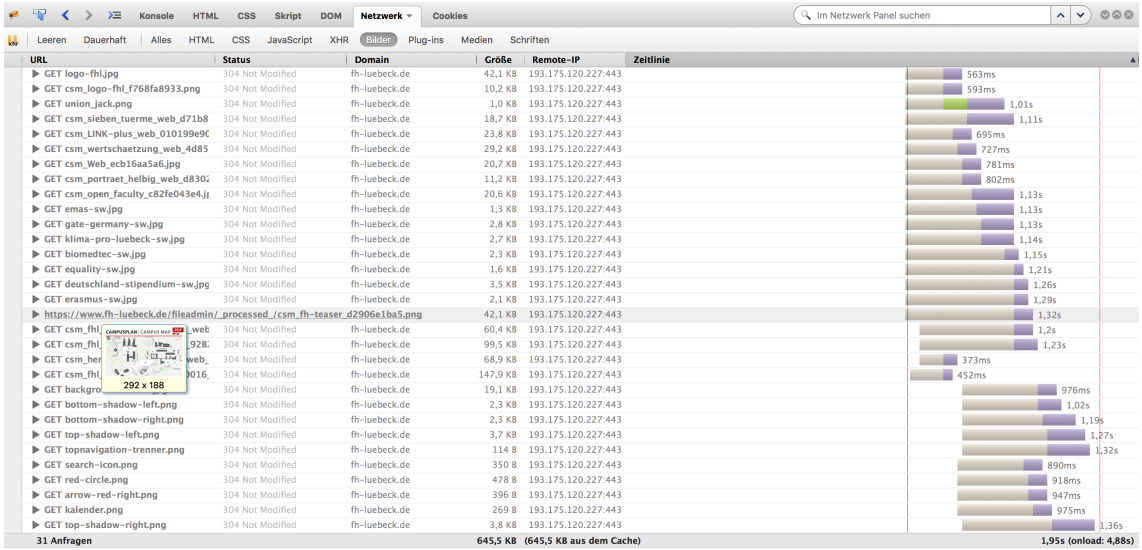
Abbildung 1: T-Online Website des Jahres 2015

haben wir mit *"Ghostery"* beide Websiten auf Tracker (Tracker dienen zur Analyse des Surfverhaltens eines Nutzers) untersucht. Auf t-online wurden vierzehn Tracker gefunden, wobei auf fh-luebeck erstaunlicherweise nur ein Tracker gefunden wurde.

3.3 Laden von Websiten

Bei erneutem Laden der beiden Website, dieses mal mit gestartetem AddOn *"Firebug"*, konnte festgestellt werden, dass es einige Inhalte oder Bilder gibt die t-online oder die fh-luebeck möglichst schnell bzw als erstes geladen haben möchte.

Wie man in der Abbildung 1 sehen kann, möchte T-Online natürlich dieses Bild



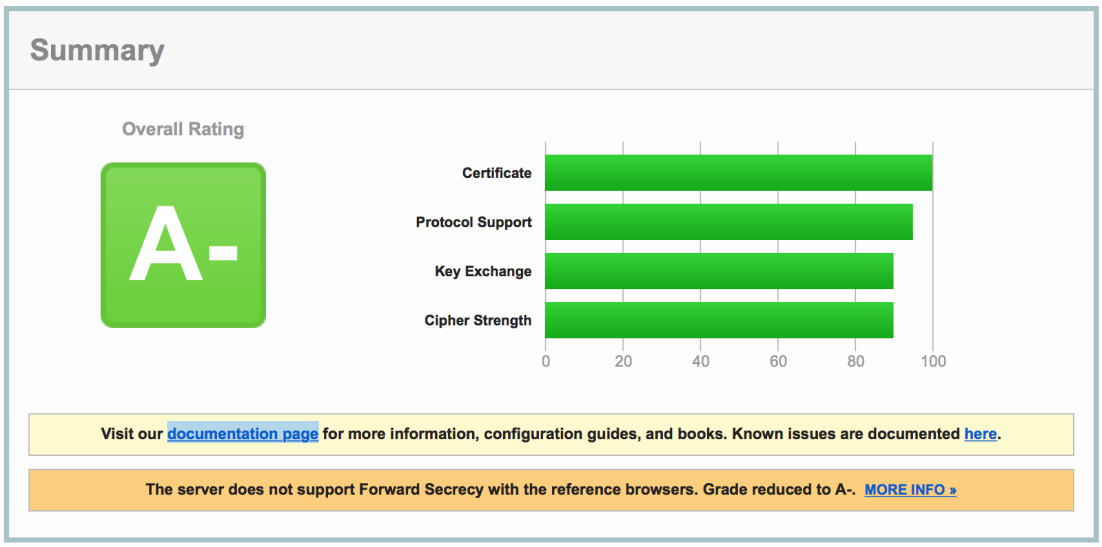
vorher laden als, ein Icon auf der unteren Hälfte der Website. Außerdem fällt auf das nachdem die für t-online "wichtigen" Inhalte/Bilder geladen wurden, die Website anfängt Inhalte/Bilder der Werbeagenturen wie (Otto, Bonprix, Ebay, Amazon, etc) zu laden.

Ähnliches Prinzip finden wir bei der Website *www.fh-luebeck.de*, hier werden auch erst das Logo und dann die Bilder für die Kategorie *"Aktuelles der Fachhochschule Lübeck"* geladen. Angesichts dieser Fakten kann man davon ausgehen das beide Websiten beim laden Ihrer Inhalte/Bilder Prioriäten setzen, in welcher Reihenfolge was geladen werden soll.

3.4 SSL/TLS im Browser

HIER DIE ANTWORT VON SVEN NOCH ABWARTEN WAS ER ZU DEM VERGLEICH DER BILDER SAGT

Mit der Website von *https://cc.dcsec.uni-hannover.de/* wurden die Browser Firefox und Safari getestet. Dabei konnte man eindeutig feststellen Außerdem können wir in die Suchleiste von Firefox *about:config* eingeben und so manuell SSL/TLS Einstellungen vornehmen.



3.5 SSL/TLS beim Server

Mit Hilfe der SSL Tools wurden die Seiten *www.signin.ebay.de* und *www.banking.haspa.de* untersucht. Hierbei lieferten beide das Ergebnis "A-", welches ein sehr unerwartetes und entäuschendes Ergebnis ist. Man hatte bei beiden Websiten ein Ergebnis von "A++" erwartet, da beide eine Möglichkeit für Online Transaktionen sind. Die Protkolle auf solchen Seiten sollten immer aktuell und im Schnitt "A+" haben.

3.6 Auswertung von Cookies

Im Laufe des Praktikums haben sich sehr viele Cookies angesammelt, viele davon sind von Dirttanbiertern die man, wenn man sich davor mit Cookies nie befasst hat, nicht kennt. Es haben sich ca. 80 Cookies angesammelt. Mit *"Cookie Monster"* kann Einstellungen vornehmen wie einzelne Cookies zu erlauben oder alle/keine Cookies erlauben. Außerdem kann man noch sehen welcher Cookie von welcher Website ist.

3.7 Diskussion AddOns

Firebug ist kein gutes AddOn für Netzwerkspezialisten oder Entwickler, aber für einen ”normalen” Anwender hat dieses AddOn wenig Sinn.
Fazit: Wir werden dieses AddOn weiterhin **nicht benutzen**.

No Script ist ein sehr sinnvolles Tool, da es nicht nur Sicherheit mit sich bringt sondern auch noch die Leistung steigert. Da man durch dieses Tool sinnlose Werbungen und Scripte deaktivieren kann, läd die Website viel schneller. Jedoch kommt es ab und zu vor das man eine Website besucht und ein bestimmtes Skript verwenden will welches aber von dem Tool blockiert wird, dies is jedoch kein großes Problem, da man manuell Skripte temporär oder für immer für die Webiste erlauben kann.
Fazit: Da wir vor dem Praktikum schon dieses AddOn kannten und verwendet haben, werden wir es auch in der Zukunft weiterhin **benutzen**.

Ghostey dient zum blockieren der Tracker und man hat immer den Überblick über die Tracker, da sie oben rechts auf dem Bildschirm beim laden der Website angezeigt werden.
Fazit: Wir kannten das AddOn vorher nicht und waren sehr beeindruckt und

werden es in der Zukunft **benutzen**.

CookieMonster lässt sich sehr einfach bedienen und man kann leicht Cookies ganz oder einzelnd erlauben und verbieten.
Fazit: Auch dieses AddOn war uns zuvor bekannt und wir werden es weiterhin **benutzen**.

DNSSec Validator prüft ob eine Website mit DNSSec geschützt ist. Man muss einige Einstellungen vornehmen, bevor man dieses AddOn zum laufen bekommt.
Fazit: Da wir keinen sinnvollen Anwendungsfall im Alltag dafür finden, werden wir dieses AddOn **nicht benutzen**.

3.8 Steganographie

Im Bild war die Datei ... enthalten und man konnte mit dem bloßen keinen einzigen Unterschied zwischen den beiden Bildern finden.

SVEN NOCHMAL FRAGEN WELCHE DATEI DA RAUS KAM UND WAS MAN MIT DER HISTOGRAMM FUNKTION GESEHEN HAT

4 Switch

5 Router

6 Transportschicht