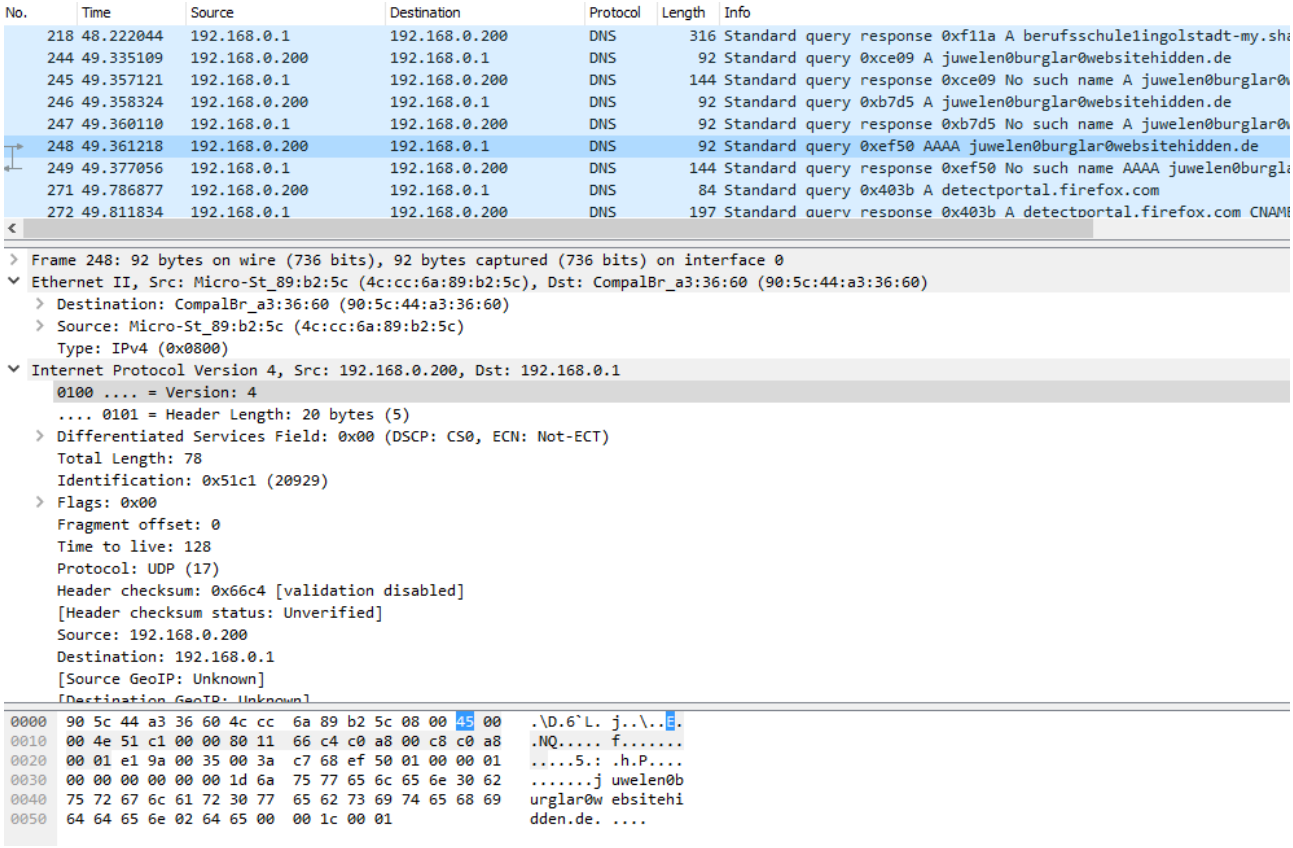
Einführung in und Funktionsweise des ISO OSI-Modells

Der Sniffer

Mit einem Snifferprogramm z.B. Wireshark kann man dem ISO OSI Modell bei der Arbeit zusehen!!?!



1. Wireshark starten.
   Starten Sie im Wireshark die Aufzeichnung mit dem Icon ganz links und tragen Sie DNS als Filter ein.
2. Nun Firefox starten und die im Bild oben „markierte Internetseite“ anwählen.
3. Gehen Sie zurück zu Wireshark, stoppen Sie die Aufzeichnung und speichern Sie diese im Tauschordner.
4. Wählen Sie im oberen Fenster dasjenige Frame aus mit der Größe von 92B und Protokoll DNS.
5. Sollte es nicht geklappt haben, müssen Sie 1 Zeichen in der URL verändern.
6. Beschreiben Sie dann zunächst den Inhalt der 3 Fenster eines Snifferprogramms.

1. Fenster oben: _____

_____

2. Fenster Mitte _____

_____

3. Fenster unten _____

_____

Hexdump

1. Ergänzen Sie die Ziffern des Hexdumps. Umranden Sie die Bereiche der einzelnen OSI-Schichten farbig.

Byte	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
00																
10																
20																
30																
40																
50																

2. Markieren Sie die Bereiche der OSI-Schichten. Markieren Sie die Quell-MAC-, die Target-MAC-, die Sender-IP-, die Empfänger-IP-, die Source-Port- und die Ziel-Port-Adresse und notieren Sie deren Eigenschaften.

Byte	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
00																
10																
20																
30																
40																
50																

_____

_____

_____

3. Markieren Sie die Bereiche der OSI-Schichten. Ergänzen Sie für jede Schicht den Namen des „Paketes“ und seine Byte-Größe in Klammern. (Kontrolle : Die Tabelle hat insgesamt : 6 x 16 Byte = 96 Byte !)

Byte	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
00																
10																
20																
30																
40																
50																

Das OSI Modell (Kommunikation aufgeteilt in Schichten)

Wie eben gesehen, zerteilt das OSI-Modell die Netzwerkkommunikation in verschiedene Bereiche.

Auch die zwischenmenschliche Kommunikation kann in verschiedene Bereiche / Teile unterteilt werden. Da dabei auch kommuniziert wird, eignet Sie sich sehr gut zum Vergleich mit dem OSI-Modell.
Notieren Sie einige Teilbereiche der Kommunikation zwischen 2 Menschen:

_____

Man kann ganz allgemein unterscheiden:
1. Was ist zu tun also z.B. der Brotzeitdienst oder der Verschlüsselungsdienst. Daher sprechen wir hier von **Diensten**.
2. Wie ist der Dienst auszuführen z.B. im Supermarkt für die Kollegen und Kolleginnen einkaufen oder den Cäsarcode verwenden. In diesem Fall sprechen wir von **Protokollen**.

Was versteht man unter einem Dienst / einem Protokoll (in eigenen Worten)?

_____

RFC – Request for Comment durch die IETF (Internet Engineering Task Force):

_____

In der Netzwerkkommunikation gibt es 7 verschiedene Bereiche mit genau definierten Diensten (z.B. Anwendungen Daten übermitteln) zu deren Aufgabe jeweils anhand von **Protokollen** (z.B. http / https um Webseiten zu übermitteln) bestimmt ist. Die 7 **Protokoll**gruppen des ISO OSI-Modells sehen Sie auf der nächsten Seite.
Ein vereinfachtes Netzwerkmodel mit lediglich 4 Schichten ist das TCP/IP- oder DoD-Model.
Welche 3 Vorteile hat die Aufteilung in Schichten?

_____

_____

_____

_____

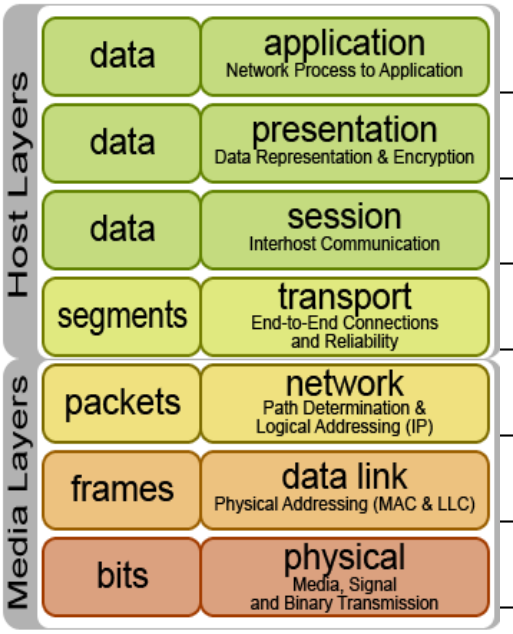
_____

_____

Kurzbeschreibung:

Switch: _____

Router: _____



dt. Name

DoD-Model
(dt./engl.)

Um sich die vielen Namen des OSI-Models besser merken zu können, gibt es einige Merksprüche. Notieren Sie sich passende auf ein extra Blatt.
Schreiben Sie die Abk. der Protokolle aus und beschreiben Sie deren Aufgabe in Stichpunkten. Notieren Sie auch deren OSI- und Port-Nrn.

Protokoll	Aufgabe	OSI-/ Port-Nr.
HTTPS		
UDP		
TCP		
SMTP		
POP3		
IMAP		
SSH		
IP		
Ethernet (IEEE-Name: 802.3)		
DNS		
RIP		
FTP		
DHCP		
OSPF		
Word / Firefox		

Verwendete Abkürzungen:

ISO: _____

OSI: _____

DoD: _____

IEEE: _____

Notieren Sie stichpunktartig die Aufgaben der 7 Schichten

Der Hex-Dump

Zeilen-/ Bytezähler	Hex-Dump	ASCII-Decoded
0000	7c 4f b5 59 07 f2 00 22 15 7e 8c 75 08 00 45 00	|O.Y..." .~.u..E.
0010	00 28 71 0b 40 00 80 06 00 00 c0 a8 02 de 53 95	.(q.@... ......S.
0020	7c 60 d6 b8 00 50 41 71 4a 99 73 9b 48 9a 50 10	|`...PAq J.s.H.P.
0030	01 04 93 96 00 00	.....

Zur Übertragung müssen alle Informationen angepasst (codiert) werden, damit sie mit Bits übertragen werden können.. Bei Zeichenfolgen z.B. ist das der ASCII-Code.

Codier Übung

Schreiben Sie die 1. 6 Buchstaben Ihres Familiennamens auf:

_____ _____ _____ _____ _____ _____

Codieren Sie diesen mit der ASCII-Tabelle in Hex-Ziffern:

_____ _____ _____ _____ _____ _____

Rechnen Sie diese in Binär-Code um:

_____ _____ _____ _____ _____ _____ _____ _____ _____ _____ _____ _____

Jetzt können Sie schon codieren und verstehen auch, warum hex oft cooler als bin ist!?!!

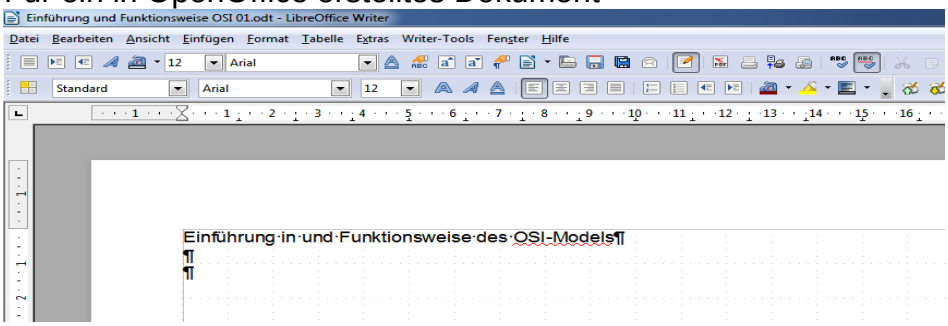
Und nun machen Sie es gleich auch noch anders herum:
Notieren Sie für die Byte-Nummern 0x1, 0x29 und 0x1c aus dem obigen Hex-Dump die enstprechenden Hex-Werte und die dazugehörigen ASCII-Zeichen:

_____

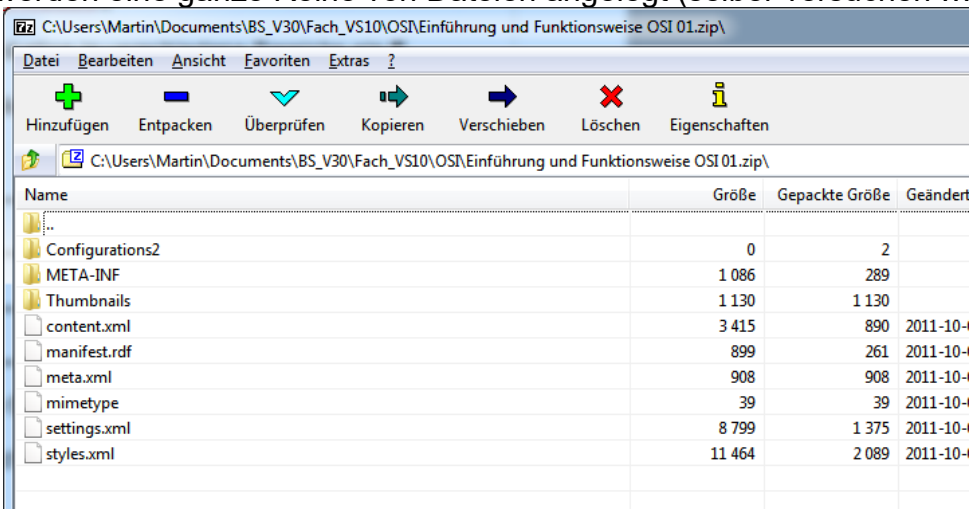
Maßnahmen zur effizienten Datenübertragung durch Komprimierung von Informationen

Es wäre nun cool (=effizient), wenn alle nötigen Zusatzinformationen, welche die Protokolle mitschicken müssen, mit möglichst wenig Bits vom Sender zum Empfänger übertragen werden können, damit möglichst viel Zeit (Bandbreite) für die eigentlichen Nutzdaten (z.B. der Inhalt einer Website) übrig bleibt.
Auch bei vielen Programmen müssen viele Informationen zusätzlich zu den zu speichernden Daten mit abgespeichert werden. Welche Informationen z.B. in einem mit einem Textverarbeitungsprogramm erstellten File so alles zusätzlich gespeichert werden, können Sie sich ansehen.

Für ein in OpenOffice erstelltes Dokument



werden eine ganze Reihe von Dateien angelegt (selber versuchen … in zip umbenennen)



mit z.B. folgenden Inhalten (irgendwo steht auch der eingegebene Textes)

```
<?xml version="1.0" encoding="UTF-8"?>
- <office:document-meta office:version="1.2" xmlns:grddl="http://www.w3.org/2003/g/data-view#" xmlns:ooo="http://openoffi
xmlns:dc="http://purl.org/dc/elements/1.1/" xmlns:xlink="http://www.w3.org/1999/xlink" xmlns:office="urn:oasis:names:t
  - <office:meta>
      <meta:creation-date>2011-10-09T21:10:06.04</meta:creation-date>
      <meta:document-statistic meta:non-whitespace-character-count="42" meta:character-count="47" meta:word-count="6" meta:p
      <meta:editing-duration>P0D</meta:editing-duration>
      <meta:editing-cycles>1</meta:editing-cycles>
      <meta:generator>LibreOffice/3.4$Win32 LibreOffice_project/340m1$Build-203</meta:generator>
    </office:meta>
</office:document-meta>
```

… für bestimmte Zusatzinformationsbereiche sind also Dateien angelegt worden, diese enthalten jede Menge Informationen (z.B. „generator“), welche genau (hier z.B.: „LibreOffice/3.4...“) beschrieben werden.
Bei einem Netzwerkprotokoll müssten derartige Informationen als Zahlen und zwar möglichst wenige Zahlen codiert werden.

Machen wir nun OpenOffice zu einem Netzwerkprotokoll und überlegen, wie Informationen der beiden oben aufgelisteten Zeilen beginnend mit „meta: document-statistic“ und „generator“ möglichst effizient übertragen werden können.
Der schlaue Martin schlägt vor: 42 476 111.
Na ja, die 1. fünf Ziffern, das kann ja jeder, das geht so:

Aber die drei Einser, was hat der sich nur dabei gedacht? Da ist bestimmt ein Trick dabei!
JEP … Trick 17:

Zur Übermittlung von Informationen werden also auch einzelne Bits (Flags, z.B. dumm = 0 oder <der Herr Lehrer> = 1) verwendet, diese sind oft zu Gruppen von 8 Bit (1 Byte, 2 Hex-Ziffern) zusammengefasst.

Quelle: dilbert.com

Data Encapsulation (doch kein Witz also? ☹)

Beim ISO OSI-Model gibt es 7 verschiedene Schichten. Natürlich kennt sich jedes Protokoll/jede Schicht ausschließlich mit seinen/ihren eigenen Informationen aus. Wenn schon Bits da sind, die übertragen werden müssen, hängt es/sie die eigenen Infos einfach vorne (Header) oder hinten (Trailer) an. Die bereits vorhandenen Daten sind aus Sicht des Protokolls unverständlich, wie für uns z. B. Chinesisch oder Plattdeutsch.
Diesen Vorgang, dass Protokolle nacheinander ihre eigenen Daten als Header und ggfs. als Trailer anfügen, nennt man Encapsulation. Auf Seite 2 dieses Geheftes kannst Du Dir das Ergebnis eines solchen Vorgangs ansehen. Die Daten der einzelnen Schichten hast Du dazu (hoffentlich) farbig markiert.
Mache nun auf einem extra Blatt eine entsprechende Zeichnung und verwende Encapsulation, Header, Trailer, SDU und PDU durch entsprechende Beschriftung.