

Übungsblatt 5 – Dynamisches Routing, Traceroute & Application Layer

Nachdem Sie nun komplexere Netzwerke aufgesetzt haben und den Verkehr Ihrer Netzwerke analysiert haben, betrachten wir in der kommenden Übung die dynamische Umsetzung des Routings, sowie die Anwendungsschicht. Viele dieser *Application Layer*-Protokolle nutzen Sie bereits ohne es bewusst wahrgenommen zu haben. Da Sie als angehende Netzwerkprofis aber nicht nur daran interessiert sind Dinge zu nutzen, sondern deren Aufbau zu verstehen, soll mit der Übung zum Application-Layer diese Lücke ein Stück weit kleiner werden.

Aufgabe A – Routing & Traceroute

- 1.) Im wesentlichen gibt es zwei fundamentale Routing-Algorithmen. Dies sind das Distanz-Vektor- und Link-State-Routing. Diese ermöglichen es den kürzesten Weg durch einen Graphen zu finden (Shortest-Path-Problem).

Für gewöhnlich wird für das Distanz-Vektor-Routing der Bellman-Ford-Algorithmus verwandt, das Link-State-Routing nutzt den Dijkstra-Algorithmus.

a.) Recherchieren Sie wie das Link-State-Routing unter Nutzung des Dijkstra-Algorithmus funktioniert.

b.) Recherchieren Sie wie das Distanz-Vektor-Routing unter Nutzung des Bellman-Ford-Algorithmus funktioniert.

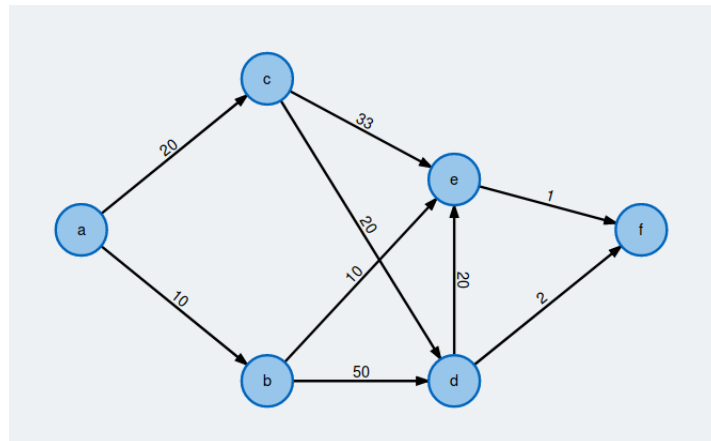
c.) In welchen Protokollen finden diese beiden Protokollen Verwendung? Ist diesen Protokollen etwas gemein?

d.) **Fakultativ:** Warum wird keines der beiden Verfahren für das Exterior-Gateway-Protokoll (EGP) genutzt?

e.) Erläutern Sie die fundamentalen Unterschiede beider Lösungsansätze.

f.) Diskutieren Sie ob der Bellman-Ford-Algorithmus für das Link-State-Routing und der Dijkstra-Algorithmus für das Distanz-Vektor-Routing genutzt werden könnte.

- 2.) Gegeben sei folgender Graph:



Finden Sie den kürzesten Weg vom Knoten *a* zum Knoten *f*!

- Nutzen Sie zunächst den Dijkstra-Algorithmus.
- Anschließend soll der Bellman-Ford-Algorithmus genutzt werden.

Aufgabe B – Traceroute

- 1.) Lesen Sie die folgenden Artikel:

<https://en.wikipedia.org/wiki/Traceroute>,
<https://linux.die.net/man/8/traceroute>.

Beantworten Sie anschließend folgende Fragen:

- Wofür wird Traceroute genutzt?
- Wie wird Traceroute umgesetzt, d.h. wie läuft eine „Routen-Verfolgung“ ab?
- Welche Limitationen ergeben sich aus der Umsetzung?
- Dokumentieren Sie die Syntax, sowie die Bedeutung von Traceroute beispielhaft.

- 2.) Lesen Sie folgendes Paper zu Paris-Traceroute von der ACM International Measurement Conference 2006:

<http://conferences.sigcomm.org/imc/2006/papers/p15-augustin.pdf>

- Warum ist eine „neue“ Traceroute-Applikation notwendig?
- Nennen Sie drei Topologie-Anomalien die durch Paris-Traceroute erkannt werden können.
- Recherchieren Sie wie *paris-traceroute* zu nutzen ist! Notieren Sie sich entsprechen die Kommandos und deren Bedeutung.

Aufgabe B – Domain Name System (DNS)

Das Domain Name System ist ein dezentrales System (verteilte Datenbank nach der Client-Server-Architektur), dessen primäre Aufgabe die Adressauflösung von Domain Name(n) zu IP-Adresse(n) ist. M.a.W. DNS bietet eine Abbildung von Domainname auf IP-Adresse ¹.

In der dritten Übung wurde das DNS bereits kurz angeschnitten, da Ihre Netzwerke im letzten Schritt einen Uplink ins Internet erhalten haben und auch Domain Namen auflösen konnten.

Nun schauen wir uns das DNS und einige Tools etwas genauer an.

1.) Rekapitulieren Sie Ihr Wissen zu DNS!

- a.) Auf welchem Layer des OSI-Modells arbeitet DNS?
- b.) Welches Transportprotokoll nutzt DNS?
- c.) Auf welchem Port laufen DNS-Server standardmäßig?

2.) Nennen und Erklären Sie die Komponenten des DNS-Systems.

- a.) Was ist gemeint wenn von einem Client-Server-Modell die Rede ist?
- b.) Was wird unter dem Begriff DNS-Resolver verstanden?
- c.) Was ist ein DNS-Root-Server, was ist ein TLD-Server und was ein Domain-Server?
- d.) Was wird unter dem Konzept der Domain und was unter einer Zone verstanden?
- e.) Was ist ein Stub im Kontext von DNS?
- f.) Was ist ein Bind-Server?
- g.) Recherchieren Sie welche wichtigen Ressource-Types DNS kennt.

3.) Erläutern Sie die Auflösung einer DNS-Anfrage.

- a.) Welche beiden Möglichkeiten einer Namensauflösung gibt es? Wie kann eine Name aufgelöst werden?
- b.) Wie erfolgt die jeweilige Auflösung eines DNS-Requests?
- c.) Verdeutlichen Sie sich anhand eines Beispiels, wie ein DNS-Request bearbeitet wird.
- d.) DNS bietet theoretisch eine rekursive und iterative Namensauflösung, praktisch wird eine Mischung aus beiden Verfahren angewandt. Recherchieren Sie, wie diese Auflösung aussieht.

¹Bzw. als Inverse – die Abbildung von IP-Adresse auf Domainnamen (Reverse-Lookup)

4.) Recherchieren Sie kurz wie die Tools

- host
- dig
- nslookup.

zu nutzen sind.

- a.) Erläutern Sie kurz was jedes der oben genannten Tools leistet.
- b.) Nennen Sie für jedes Tool geeignete Einsatzgebiete/ Szenarien.
- c.) Recherchieren Sie die Syntax, sowie Semantik der Tools.
- d.) Notieren und kommentieren Sie sich entsprechende Beispiele.

Aufgabe C – HTTP(S) & HTML

Kein anderes Protokoll ist für das World-Wide-Web so wichtig wie *HTTP*. In diesem Teil sollen Sie recherchieren, wie die bunten Seiten in Ihren Browser kommen.

- 1.) Recherchieren Sie zunächst was *HTTP* ist.
Eine gute Anlaufstelle wäre Tanenbaums Computer Networks Chapter 7.3 – The World Wide Web. Jede andere Quelle ist ebenso gut!
- 2.) Erläutern Sie die Funktionsweise von *HTTP*.
- 3.) Auf welcher Schicht des OSI-Modells ordnen Sie *HTTP* ein?
- 4.) Auf welchen Port laufen meistens Webserver? Auf welchem Port läuft die verschlüsselte Variante *HTTPS*?
- 5.) Wie sieht ein typischer *HTTP*-Header aus?
- 6.) Nennen Sie alle *HTTP*-Methoden. Notieren Sie sich was diese machen und wie deren Aufruf aussieht.
- 7.) Mittlerweile gibt es unterschiedliche Versionen von *HTTP*. Vornehmlich sind dies 1.1 und *HTTP/2* – nennen Sie einige fundamentale Unterschiede.
- 8.) Machen Sie sich kurz klar, welche Aufgabe SSL/TLS übernimmt.
(Hinweis: An dieser Stelle genügt ein grobes Verständnis!)
- 9.) Recherchieren Sie wie die Tools *telnet*, *netcat* und *openssl s_client* für das Abfragen von *HTTP* genutzt werden können.

Wie sieht die Syntax zum Verbinden auf einen Server aus? Notieren Sie sich entsprechend Beispiele. Sie sollten als Vorbereitung auf die Laborübung sich auch das dazugehörige Arbeitsblatt anschauen, sodass Sie zielgerichtet nach entsprechenden Beispielen suchen können.

- 10.) Optional: Suchen Sie sich ein Tutorial zu *openssl s_client* heraus. Durchlaufen Sie entsprechendes Tutorial.
Bspw.: <https://tinyurl.com/y9nnaz6a>
- 11.) Recherchieren Sie was *STARTTLS* bedeutet. Worin unterscheidet sich *STARTTLS* von *HTTPs*?
- 12.) Recherchieren Sie kurz was ein kryptografisches Zertifikat ist. Wozu werden diese im Zusammenhang mit *HTTPs* genutzt?
- 13.) Erläutern Sie wie Sie sich Zertifikate mit *openssl* anschauen können.

Aufgabe D – E-Mail mit POP3, IMAPv4 & SMTP

Das Simple Mail Transfer Protokoll (SMTP) wird, wie der Name schon sagt, zum Austausch von E-Mails in Computernetzwerken genutzt. Primär wird es zum Weiterleiten von Mails zwischen Server genutzt. Auf Ihren Endgeräten kommt zumeist *IMAP* oder *POP3* zum Einsatz.

- 1.) Recherchieren Sie zunächst was sich hinter den Akronymen POP3, IMAPv4, sowie SMTP verbirgt.
- 2.) Erläutern Sie im groben welche Aufgaben die oben genannten Protokolle übernehmen.
- 3.) Auf welcher Ebene des OSI-Modells arbeiten die Protokolle?
- 4.) Machen Sie sich im groben klar, wie diese Protokolle arbeiten.
- 5.) Worin unterscheiden sich POP3 und IMAP?
- 6.) Auf welchen Ports arbeiten die drei Protokolle?
- 7.) Auf welchen Ports arbeiten die drei Protokolle mit Verschlüsselung?
- 8.) Recherchieren Sie wie IMAP, POP3 und SMTP via Kommandozeile genutzt werden können. Notieren Sie sich entsprechende Kommandos, sowie deren Bedeutung!