

Veranstaltungseinheit Protokolle

Bemerkung: Diese Übung benutzt viele Netzwerkaufrufe, die je nach Firewall nicht funktionieren können. Insbesondere aus dem ZHAW Netzwerk können gewisse Aufrufe geblockt werden. Bei Timeouts oder Verbindungsfehlern versuchen Sie die Übung von ihrem Netzwerk zu Hause oder über einen persönlichen Hotspot. In diesem Fall reichen Screenshots zur Abgabe.

DNS Namen auflösen

Ziel der Übung

- Sie verstehen die Prinzipien des DNS
- Sie können einen DNS Aufruf mit Hilfe von Konsolen Tools selbst ausführen

Aufgabe(n)

Das Programm `host` auf der Betriebssystem Kommandozeile dient dazu DNS Aufrufe zu tätigen. Eine Anleitung wird angezeigt mit dem Kommando `man 1 host` oder online, z.B. <http://linux.die.net/man/1/host>

Benutzen Sie dieses Kommando um die IP Adresse des Servers `zhaw.herokuapp.com` zu bestimmen.

Das Programm `host` können Sie später gebrauchen, um Probleme im Zusammenhang mit DNS zu isolieren und debuggen.

Erwartete Resultate und Abgabe

- Ausgabe der IP Adresse auf der Betriebssystem Kommandozeile
- Abgabe während dem Labor

GET Requests mit curl

Ziel der Übung

- Sie kennen das Kommando `curl` und können damit HTTP GET Request absetzen

Aufgaben

Das Programm `curl` auf der Betriebssystem Kommandozeile dient dazu, HTTP Requests zu tätigen. `curl` ist kein vollwertiger Browser, jedoch können damit einzelne HTTP Requests isoliert abgesetzt werden. `curl` wird oft zum debugging eingesetzt. Eine Anleitung wird angezeigt mit dem Kommand `man 1 curl` oder online, z.B. <http://linux.die.net/man/1/curl>

- Benutzen Sie das Kommando `curl` um das Tasklisten API auf dem Server `zhaw.herokuapp.com` anzusprechen. Eine Taskliste befindet sich unter der URL `http://zhaw.herokuapp.com/task_lists/demo`
- Verwenden Sie für einen zweiten Aufruf anstatt den Hostname `zhaw.herokuapp.com` die IP Adresse, welche Sie in der vorherigen Übung bestimmt haben
- Ein Aufruf der URL `http://zhaw.herokuapp.com/non_existing` wird vom Server mit einem HTTP Status 404 beantwortet. Verifizieren Sie dieses Verhalten, indem sie die URL mit `curl` aufrufen und zusätzlich die Option `-I` angeben um die Antwort Header auszugeben.

Erwartete Resultate und Abgabe

- Abgabe während dem Labor

POST Requests mit curl

Ziel der Übung

- Sie können mit dem Kommando `curl` HTTP POST Requests absetzen
- Sie können mit dem Kommando `curl` ein REST API auf einem Server ansteuern und JSON Daten übergeben

Aufgaben

Mit dem Programm `curl` können auch HTTP Post Requests abgesetzt werden. Sobald mit dem `-d` Parameter Daten übergeben wurden, schickt `curl` diese per HTTP Post an die angegebene URL. Der Dienst auf `http://zhaw.herokuapp.com` verwertet POST Requests auf die Adresse `http://zhaw.herokuapp.com/task_lists/` indem er eine neue Taskliste anlegt. Als Daten können sie eine Taskliste im JSON Format mitgeben, z.B.:

```
{"tasks": [{"title": "Buy milk", "done": false}]}
```

Die Antwort des Servers enthält eine Identifizierung der erstellten Taskliste (Feld `id`). Diese kann benutzt werden, um die Taskliste später per HTTP GET wieder abzurufen. Die neue Taskliste ist erreichbar über die URL `http://zhaw.herokuapp.com/task_lists/:id` (wobei `:id` ersetzt werden muss mit der konkreten Identifizierung der erstellten Taskliste).

- Erstellen Sie mit dem Kommando `curl` eine neue Taskliste.
- Sie können durch Veränderung der Daten im JSON auch kompliziertere Tasklisten erstellen.
- Rufen Sie die neu erstellte Taskliste ebenfalls mit dem Kommando `curl` ab.
- Beachten Sie, dass in der Shell spezielle Zeichen wie `{ }` und `"` escaped werden müssen. Am besten schliessen sie den gesamten Wert für den `-d` Parameter in einfachen Anführungszeichen (`'`) ein.

Erwartete Resultate und Abgabe

- URL zur neuen, mit `curl` erstellten Taskliste
- Abgabe während dem Labor

Sockets Javascript

Ziel der Übung

- Sie können mit Javascript einfache TCP/IP Anwendungen schreiben
- Sie können Javascript unabhängig vom Browser einsetzen

Aufgabe

Im Browser sind mit Javascript aus Sicherheitsgründen nur sehr limitierte Netzwerkzugriffe möglich. Serverseitiges Javascript kennt diese Einschränkungen nicht. Das `net` Modul (<http://nodejs.org/api/net.html>) von `node.js` erlaubt TCP/IP Verbindungen mit javascript zu benutzen.

In dieser Übung sollen die vorhergehenden Aufgaben nicht mehr mit den Kommando Zeilen Befehlen sondern mit selbst geschriebener Software implementiert werden.

- Installieren Sie `node.js` <http://nodejs.org/>
- Lesen Sie sich in die Applikationsentwicklung mit `node.js` ein, insb. in das Modul `net` (<http://nodejs.org/api/net.html>) und das Modul `DNS` (<http://nodejs.org/api/dns.html>)
- Benutzen Sie das Modul `DNS` um den Hostnamen aus der ersten Übung aufzulösen

- Benutzen Sie das Modul net oder das Modul http um einen GET Request wie in der zweiten Übung abzusetzen
- Benutzen Sie das Modul net oder das Modul http um einen POST Request wie in der dritten Übung abzusetzen

Erwartete Resultate und Abgabe

- Abgabe während dem Labor