

software name

HSR Studienarbeit Network Unit Testing

Network testing

David Meister, Andreas Stalder

4. Oktober 2016

1 Änderungsgeschichte

Datum	Version	Änderung	Autor
27.09.16	1.0	Erstellung erster Version	dm/as

Tabelle 1: **Änderungsgeschichte**

Inhaltsverzeichnis

1	Änderungsgeschichte	2
2	Einführung	5
2.1	Zweck	5
2.2	Gültigkeitsbereich	5
2.3	Referenzen	5
3	Einleitung	7
3.1	Testing Motivation	7
4	Device Tests	7
4.1	Scope	7
4.2	Nutzen	7
4.3	Beispiele von Test Cases	7
5	Circuit Tests	7
5.1	Scope	7
5.2	Nutzen	7
5.3	Beispiele von Test Cases	7
6	Routing Tests	7
6.1	Scope	7
6.2	Nutzen	7
6.3	Beispiele von Test Cases	7
7	Traffic Tests	7
7.1	Scope	7
7.2	Nutzen	7
7.3	Beispiele von Test Cases	7
7.3.1	QoS	7
7.3.2	Throughput	7
8	Application Tests	8
8.1	Scope	8
8.2	Nutzen	8

	software name
8.3 Beispiele von Test Cases	8
8.3.1 DNS	8
8.3.2 DHCP	8
8.3.3 Webservice	8
8.3.4 SSH	9

2 Einführung

2.1 Zweck

Dieses Dokument stellt den Projektplan für unser Studienarbeit dar, es dient zur Planung, Steuerung und Kontrolle.

2.2 Gültigkeitsbereich

Dieses Dokument ist über die gesamte Projektdauer gültig. Es wird in späteren Iterationen angepasst. Somit ist jeweils die neuste Version des Dokuments gültig und alte Versionen sind obsolet.

2.3 Referenzen

Noch keine.

3 Einleitung

3.1 Testing Motivation

3.2

4 Device Tests

4.1 Scope

4.2 Nutzen

4.3 Beispiele von Test Cases

5 Circuit Tests

5.1 Scope

5.2 Nutzen

5.3 Beispiele von Test Cases

6 Routing Tests

6.1 Scope

6.2 Nutzen

6.3 Beispiele von Test Cases

7 Traffic Tests

7.1 Scope

7.2 Nutzen

7.3 Beispiele von Test Cases

7.3.1 QoS

7.3.2 Throughput

Test Output ist ein wichtiger Punkt im Netzwerktest. Hier gibt es zwei Möglichkeiten. Den Link-Throughput, sowie den End-to-End-Throughput.

Link-Throughput

End-to-End-Throughput

8 Application Tests

8.1 Scope

Mit den Application Tests, werden Netzwerkservices getestet. Diese können zum Beispiel DNS, DHCP, Webservice usw. sein. Um sicherzustellen, damit bei einer Firewallumstellung alle Service noch funktionieren, schickt man Anfragen an die Services.

8.2 Nutzen

So kann man sicherstellen, dass alle Komponenten noch korrekt miteinander kommunizieren und das nichts geblockt wird.

8.3 Beispiele von Test Cases

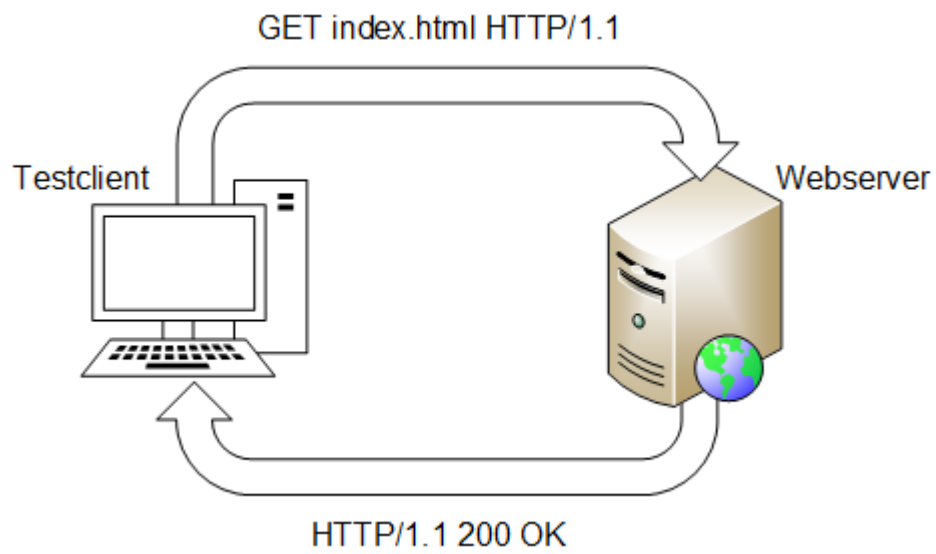
8.3.1 DNS

8.3.2 DHCP

8.3.3 Webservice

Ein Webservice kann sehr leicht getestet werden. Man kann dem Server eine Anfragen mit senden und weiss was die Antwort sein muss. Beispiel anfragen sind:

- GET www.example.com/index.html HTTP/1.1
- GET www.example.com/user/12 HTTP/1.1



8.3.4 SSH