



Hochschule Darmstadt
- FACHBEREICH INFORMATIK -

Der Titel der Arbeit

Abschlussarbeit zur Erlangung des akademischen Grades
Bachelor of Science (B.Sc.)

vorgelegt von

Vorname Nachname

12345

Referent:	Prof. Dr. Max Mustermann
Korreferent:	Dr. Max Muster

Abstract

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

Dies ist ein Zitat.

verstanden, scheinen nun doch vorueber zu sein. Dies ist der Text sein.
siehe: <http://janeden.net/die-praeambel>

Inhaltsverzeichnis

Abbildungsverzeichnis	4
Tabellenverzeichnis	5
1 Einführung	1
2 Das Framework	2
2.1 Verwendete Hardware	2
2.2 Aufbau der Software	2
2.3 Einsatz im Netzwerk	3
3 Testaufbau	4
3.1 Pis Real	4
3.2 Pis Virtuell	4
4 Versuche	5
4.1 Versuchsbeschreibung	5
4.2 Raspberry Pi Versuche	6
4.2.1 Gemachte Tests	6
4.2.2 Ergebnisse	6
4.3 Virtual Machine Versuche	6
4.3.1 Gemachte Tests	6
4.3.2 Ergebnisse	6
5 Vergleich VM/HW	7
5.1 Versuchsergebnisse	7
5.2 Kosten nutzen Faktor	7
6 Fazit	8

Abbildungsverzeichnis

Tabellenverzeichnis

Listingverzeichnis

Kapitel 1

Einführung

Hallo sehr geschätzter Leser Willkommen bei meiner Bachelor Thesis in der ich den die Differenz zwischen Virtuellen Maschinen und realen Maschinen in einem Netzwerk vergleichen möchte. Dazu habe ich selber ein Netzwerktestframework entwickelt und von Netzwerktestframeworks demonstrieren, die in Netzwerken die aus realen Maschinen bestehen und aus virtuellen Maschinen verwendet werden. Wieso hat das eine relevanz? Das werde ich euch erklären wieso. Wir leben in einer immer weiter hochvernetzten Welt sind und die Industrie 4.0 wird immer mehr bestandteil unserer Welt, Kühlschränke werden mit dem Internet verbunden. Um eine hohe netzstabilität gewährleisten müssen wir.

Kapitel 2

Das Framework

2.1 Verwendete Hardware

Die in Abschnitt 4.1 beschriebenen Versuche wurden mit folgender Hardware gemacht.

- Zwei Switches
- Vier Raspberry Pi der ersten Generation
- Ein Raspberry Pi der zweiten Generation
- Mehrere Ethernet Kabel

Diese Geräte wurden in einem eigenständigen Netzwerk zusammengeschaltet. So sind an jedem Switch zwei Pi der ersten Generation angeschlossen während an einem der Switches der Pi der zweiten Generation angeschlossen ist siehe Abbildung hier. Welche Software auf den Pis verwendet wurde, wird in Abschnitt 2.2 erklärt.

2.2 Aufbau der Software

Zabbix Neben der eigententwickelten Software wurde in diesem Framework noch das Open Source Netzwerk Monitoring Tool Zabbix benutzt. Zabbix wurde von Alexei Vladishev entwickelt, inzwischen liegt die dritte Version vor welche auch in diesem Framework verwendet wurde. Ein bekannter Vertreter der Netzwerk Monitor Systeme ist Nagios, welche unter der GPL vertrieben. Zabbix ist ein Client Server basierendes Monitoring System so werden von den zu überwachenden Rechnern, den sogenannten Agents, die Daten an den Zabbix Server geschickt und sind dort über eine Weboberfläche abrufbar. Über das Webinterface ist es auch möglich einzustellen welche vordefinierten Informationen von den Agents an den Server verschickt werden sollen. So kann man auch über eine Schnittstelle in der in der Zabbix API selbstdefinierte Informationen abrufen und sich in der Weboberfläche anzeigen lassen, wie man später bei der I/O Last der Endgeräte sehen kann. All

diesen Daten werden in einer MySQL Datenbank gespeichert und sind über die MySQL Standard Schnittstelle abrufbar.

Hintergrundrauschen Diese Eigentwicklung stellt mit Zabbix die Kernkomponente des Frameworks dar. Wie der Rest der selbstentwickelten Software wurde sie komplett in Bash programmiert da das Framework ausschließlich in einer Linux Umgebung entwickelt, getestet und verwendet wurde. Hintergrundrauschen schickt über die Linux Standardbefehle Secure Copy, eine Abwandlung der SSH, Pakete von einem Endegerät zum anderen. So wird eine Last auf dem Netzwerk erzeugt die mithilfe von Zabbix gemessen werden kann. Ausserdem speichert dieses Programm die Dauer bis ein Paket erfolgreich bei seinem, zufällig ausgewählten Empfänger angekommen ist und dessen größe.

Update Script

2.3 Einsatz im Netzwerk

Kapitel 3

Testaufbau

3.1 Pis Real

3.2 Pis Virtuell

Kapitel 4

Versuche

Die beste Art ein Netzwerktestframework zu Testen ist in dem man das Framework benutzt dazu habe ich am Anfang der Arbeiten selber mehrere Tests definiert, jeder dieser Tests ist eine in einem Netzwerk mögliche Fehlerquelle die die Leistung des Netzwerkes beeinträchtigen kann. Um Fehler identifizieren zu können müssen jedoch auch die Normalwerte bestimmt werden. Dazu muss erstmal ein störungsfreies Netzwerk aufgebaut werden und dieses muss beobachtet werden. Die in diesem Netz gesammelten Werte stellen die Basis für unsere späteren Ergebnisse dar. Der zweite Schritt ist es das Netz mit Fehlern aufzubauen und zu beobachten. Dadurch haben wir die Werte eines fehlerbehafteten Netzwerkes und können diese nun mit dem fehlerfreien Netzwerk vergleichen. Dadurch sollte es uns nun möglich sein automatisierte Aussagen über die Art des Fehlers in einem Netzwerk zu treffen.

4.1 Versuchsbeschreibung

Normalbetrieb Dieser Test erzeugt die Testwerte mit denen die fehlerbehafteten Tests verglichen werden.

Ethernetkabel ohne Isolierung Ethernetkabel werden oft sehr strapaziert dies kann dazu führen das sich die Isolation löst und somit Strahlung einwirkt. Dies kann die Leistung im Netzwerk beeinträchtigen.

Falsch gedrehtes Twisted Pair Kabel Twisted Pair Kabel gehören zu den gängigsten Kabeln des Ethernet Standards, welches durch eine Verdrillung der einzelnen Kabel einen erhöhten Einstrahlungsschutz bilden. [Tan03]

Loop Ein Loop ist wenn ein Switch mit sich selbst verbunden ist, der Switch sendet dann ständig Datagramme an sich selbst und blockiert so jeden Traffic auf der Leitung.

Nicht angeschlossenes Kabel Wie verhält sich das Netzwerk wenn ein Kabel von einem Endgerät im laufenden Betrieb entkoppelt wird.

Forkbomb Die Forkbomb ist ein Programm das von sich rekursiv Kopien erstellt so das der Computer alle sein Ressourcen verbraucht nur noch diesen Code auszuführen.

`:(){ :|:&};; [Wik15]`

Festplatte läuft voll In einem Netzwerk werden ständig Daten versand, was also passiert wenn die Festplatte von einem Rechner vollläuft und der PC somit Arbeitsunfähig wird.

IP Adresse doppelt belegt Firmennetze benutzen statische IPs damit Webserver immer unter dem selben Namen erreichbar bleiben, das jedoch zu konfigurieren kann zu Fehlern Menschlichen Fehlern führen, was passiert in einem Netz wenn zwei Rechner sich eine IP teilen.

Kollisionsdomänen Kollisionsdomänen sind bereiche in einem Netzwerk wo sich mehrere Rechner eine eine Leitung teilen, dies ist geschieht in der Physikalischen Ebene eines Netzwerks. Die Frage ist kann man eine Kollisionsdomäne erkennen.

4.2 Raspberry Pi Versuche

4.2.1 Gemachte Tests

4.2.2 Ergebnisse

4.3 Virtual Machine Versuche

4.3.1 Gemachte Tests

4.3.2 Ergebnisse

Kapitel 5

Vergleich VM/HW

5.1 Versuchsergebnisse

5.2 Kosten nutzen Faktor

Kapitel 6

Fazit

Literatur

- [Tan03] Andrew S. Tanenbaum. *Computernetzwerke*. PEARSON Studium, 2003.
- [Wik15] Wikipedia. *Forkbomb* — *Wikipedia, Die freie Enzyklopadie*. [Online; Stand 11. Mai 2015]. 2015. URL: <https://de.wikipedia.org/wiki/Forkbomb>.