

Hans Weibel

Büro: TW 219

Telefon: 058 / 934 75 52

E-Mail: hans.weibel@zhaw.ch

WWW: <http://www.zhaw.ch/~wlan/>

BA15_wlan_1: Projektarbeit im Fachgebiet Kommunikation

Ermittlung der Performance von Netzwerkfunktionen am Beispiel von PRP

Bearbeitung durch: Mauro Guadagnini und Prosper Sebastian Leibundgut

Betreuung durch: Prof. Hans Weibel

Partner: Institute of Embedded Systems der ZHAW

Ausgabe: Montag, 9. Februar 2015 / Abgabe: 5. Juni 2015

1 Ausgangslage

Will man die Leistungsfähigkeit der Netzwerkanbindung eines Rechners beurteilen, bzw. den Ressourcenbedarf zur Erzielung einer bestimmten Leistung ermitteln, so bewegt man sich in einem sehr komplexen Feld. Die beobachtbare Leistung ergibt sich aus dem Zusammenspiel von Rechenleistung, Betriebssystem, Protokollsoftware, Netzwerkadapter, Netzwerk, Anwendungssoftware, Verhalten des Kommunikationspartners und zu guter Letzt dem Protokoll selbst. Es ist nicht immer einfach zu erkennen, welcher Faktor limitierend wirkt. Darüber hinaus ist zu berücksichtigen, dass Beobachtungs- und Messverfahren das Verhalten mitbeeinflussen können.

In der vorliegenden Arbeit sollen derartige Fragestellungen vorerst an einem konkreten Fall studiert und anschliessend verallgemeinert werden. Beim konkreten Fall handelt es sich um ein Ethernet-Redundanzprotokoll, das auf dem Host alleine implementiert wird (d.h. dass keine Unterstützung vom Netzwerk selbst geleistet werden muss). Es handelt sich um das vom InES mitentwickelte PRP (Parallel Redundancy Protocol). PRP ermöglicht eine unterbrechungsfreie hochverfügbare Kommunikation, wie sie von gewissen sicherheitskritischen Anwendungen verlangt wird. Das Verfahren besteht darin, dass zwei unabhängige Netzwerke verwendet werden und die Endknoten doppelt angebunden werden. Jedes Frame wird vom sendenden Knoten repliziert und auf beiden Netzwerken übertragen. Der Empfänger verarbeitet das zuerst eintreffende Frame und verwirft das Duplikat. Dazu wird in den End-

knoten ein Protokoll-Layer eingeführt, der vom InES als open Source verbreitet wird (siehe https://github.com/ZHAW-InES-Team/sw_stack_prp1). Es soll nun untersucht werden, wie sich die Anwendung des Verfahrens auf die Performance des Endknotens auswirkt bzw. welche zusätzlichen Ressourcen benötigt werden.

In der Verallgemeinerung soll aufgezeigt werden,

- wie Testverfahren und Hilfsmittel arbeiten und anzuwenden sind
- wie Testszenarien gestaltet sein müssen, dass die zu ermittelnden Eigenschaften der "Unit unter Test" von den Einflüssen der Testumgebung unterschieden werden können
- wie die Resultate zu interpretieren sind

2 Aufgabenstellung

Im Zentrum der Arbeit stehen Linux-Endsysteme (kann jedoch in Bezug auf PRP auch auf Windows ausgedehnt werden).

Die untenstehende Auflistung von Teilaufgaben gibt keine Chronologie vor. Es werden vermutlich mehrere Zyklen durchlaufen, in welchen die einzelnen Aspekte verfeinert und konkretisiert werden.

2.1 Studium von PRP und Aufbau Testumgebung

Es ist eine einfache PRP-Umgebung aufzubauen, die aus zwei Netzwerken A und B (je durch einen Ethernet-Switch repräsentiert) sowie zwei bis drei Endsystemen bestehen. Das Protokoll und seine Implementierung sind zu studieren.

Es sollen Konfigurationen definiert werden, in welchen die Leistungsfähigkeit der Implementierung bzw. der zusätzliche Bedarf von Ressourcen ermitteln werden können. Dabei sollen unter anderem Aspekte beachtet werden wie:

- Effekt von Laufzeitunterschieden zwischen Netz A und Netz B (konstante und schwankende Unterschiede, sprunghafte Änderungen).
- Zeitweiser Ausfall eines Netzwerkpfades.
- Abhängigkeit vom verwendeten Protokoll (TCP, UDP).
- Einfluss nicht entfernter Duplikate auf Funktion und Performance.
- Auswirkung auf Applikationen, wenn Frames out-of-sequence ankommen (wenn z.B. auf dem kürzeren Pfad Frames verloren gehen oder wenn Frames priorisiert werden).

2.2 Untersuchung von Tools

Es können zwei Ansätze unterschieden werden: Blackbox- und Whitebox-Messungen.

Für den Blackbox-Ansatz existieren diverse Tools, welche die Nutzung von Ressourcen bzw. Performanceparameter messen (z.B. ping, BWping, iperf, netperf, flowgrind, top, ntop, atop, sar, und viele andere mehr).

Voraussetzung für den Whitebox-Ansatz ist der Zugriff auf den Programmcode der zu untersuchenden Funktion. Mittels Profiling-Tools kann man einen Einblick in das Ausführungsverhalten des Codes gewinnen.

Neben Linux-Bordmitteln sind auch selbst entwickelte Hilfsmittel (z.B. Applikationen, die eine genau definierte Last erzeugen) und dedizierte Messgeräte in Betracht zu ziehen.

Mit Hilfe von geeigneten Kalibrier- und Vergleichsmessungen soll aufgezeigt werden, was die ermittelten Resultate aussagen bzw. wie sie zu interpretieren sind.

Es soll auch geklärt werden, welchen Einfluss die Offload-Mechanismen moderner Ethernet-Adapter haben.

2.3 Verhalten der PRP Implementierung

Es ist zu studieren, wie sich die Anwendung von PRP auf das Endsystem auswirkt, insbesondere auf die Belastung von CPU und Memory. Dabei sollen dual und single attached Endknoten verglichen werden.

2.4 Generalisierung

Die Messungen und die Interpretation der Resultate werden nur dann eine Aussagekraft haben, wenn gewisse Grundsätze und Rahmenbedingungen eingehalten sind. Diese Grundsätze und Rahmenbedingungen sind zu formulieren. Ebenso soll eine Bewertung der untersuchten Tools abgegeben werden.

2.5 Weiterführende Aspekte

Die zunehmende Virtualisierung von Processing, Storage und Network bringt eine neue Dimension in diese Betrachtungen. Wie sieht es aus, wenn Virtuelle Maschinen auf demselben Host miteinander kommunizieren? Was leisten die betrachteten Tools in dieser Umgebung?

3 Ziele

- Für einige ausgewählte Tools und Messmethoden herrscht Klarheit, welche Messungen sie ermöglichen und wie die Resultate zu interpretieren sind.
- Es liegt ein PRP- Testnetzwerk vor, welches erlaubt, die relevanten Szenarien auszumessen.
- Die InES-Implementierung von PRP ist in Bezug auf Leistungsfähigkeit und Ressourcenbedarf evaluiert.
- Es sollen einige grundsätzliche Aussagen gemacht werden darüber, wie und mit welcher Zuverlässigkeit man die Performance in physikalischen und virtuellen Konfigurationen messen kann.