Netzwerktests

Erstellung eines Modells für die Kategorisierung von Netzwerktests

- Ausarbeitung des Vortrags vom 07. Februar 2002
  im Rahmen des Seminars Pleiten, Pech und Pannen in der Informatik (Wintersemester 2001/2002)
  01. März 2002

Eberhard Karls Universität Tübingen
Wilhelm-Schickard-Institut für Informatik
Arbeitsbereich Technische Informatik

- Betreuer: Priv. Doz. Dr. habil. Thomas Kropf
  Ausarbeitung: Horst Rechner

Inhaltsverzeichnis

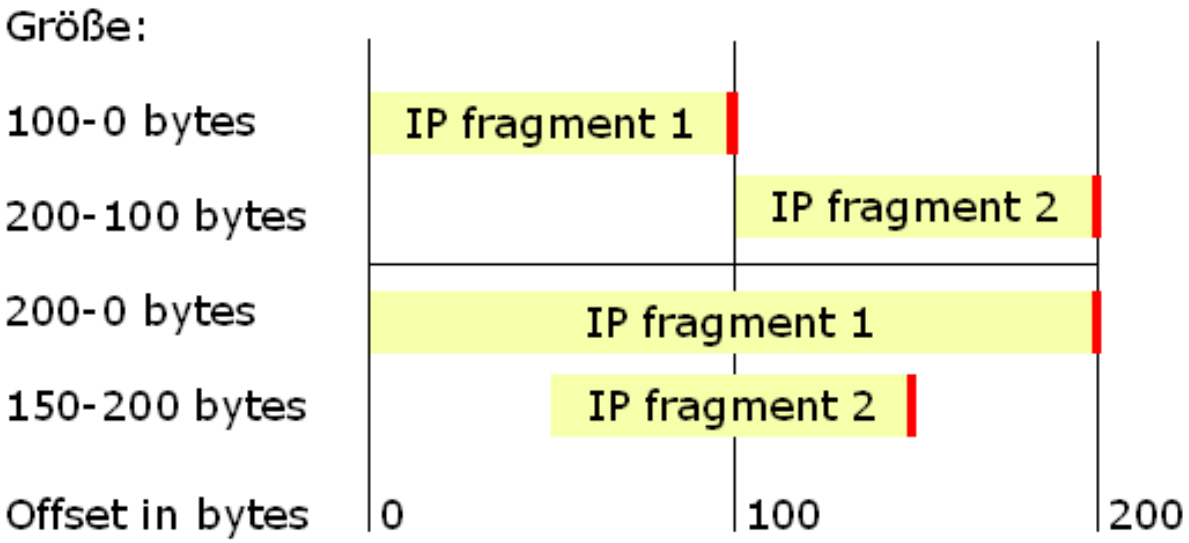
1. Motivation

1.1 Teardrop DoS

Im Jahr 1998 fand im Internet eine Attacke auf Server statt, die einen Fehler im TCP/IP Stack des jeweiligen Betriebssystems (Linux und Windows) ausnutzte. Betroffen davon war unter anderem die eCommerce Site www.amazon.com.

In Abbildung 1 wird der Implementationsfehler des Stacks veranschaulicht, der die Maschinen zum Absturz brachte. Bei der Übertragung von Daten mit dem IP Protokoll werden diese in einzelne Pakete oder Fragmente zerlegt, die nacheinander übermittelt werden. Diese IP-Fragmente tragen im Header jeweils die Information der Position des einzelnen Fragments bezüglich des gesamten Datums.

Den Fehler, welcher der Teardrop DoS ausnutzte, ist, das die Größe des allokierten Puffers für die Zwischenspeicherung der Fragmente bei der erneuten Zusammensetzung des Datums aus der Positionsangabe (dem Offset) berechnet wurde.
Im oberen Teil des Schaubildes sind 2 Fragmente dargestellt, so wie sie normalerweise übermittelt werden. Die Größe wird für Packet 1 aus dem Offset des ersten Packets (100) und 0 berechnet. Somit ergibt sich eine Größe von 100 Bytes. Beim zweiten Packet aus dem Offset der zweiten Packets minus dem des ersten, also 200 - 100 = 100 Bytes.
Der untere Teil des Schaubilds zeigt eine Fragmentierung, die den Teardrop Fehler ausnutzt. Das zweite Packet hat einen Offset, der kleiner ist als der des ersten Packets, und somit wird negativer Speicher allokiert, was den TCP/IP Stack zum Absturz brachte. Eine genaue Beschreibung des Fehlers für den Linux Stack findet sich bei [CPTIT].



(Abbildung 1: Teardrop DoS)

1.2 Kosten des Angriffs

Die Kosten eines DoS Angriffs sind schwer zu kalkulieren, da hier nicht nur der Schaden durch nicht zustande gekommene Bestellungen bei eCommerce Sites in die Rechnung mit eingeht, sondern

auch Image- und Folgeschäden.

Eine Erhebung des Computer Security Institute (FBI) von 2001 [CSI99] gab den direkten finanziellen Schaden durch Ausnützen von Sicherheitslücken bei Universitäten und Firmen mit 378 Millionen $ bei 186 Rückmeldungen (im Jahre 2000: 276 Millionen $ bei 249 Rückmeldungen) an. Diese Zahl schließt den Schaden durch sämtliche Angriffe auf Netzwerke und Computer der befragten Organisationen von außen über das Internet (ca. 70% der Angriffe) und von innen ein.
Da Angriffe von außen nur zu einem Teil durch Denial of Service Attacken stattfinden, können diese Zahlen auch nur als grober Richtwert angesehen werden, zumal hier auch nur ein Bruchteil von Universitäten und Firmen Auskunft über registrierte Sicherheitsprobleme gegeben haben.

Eine der ersten wissenschaftlichen Studien, die sich ausschließlich mit DoS Attacken beschäftigt haben, wurde 2001 von der University of California vorgestellt [Moore01], und gibt die Anzahl der Attacken auf Internetrechner im gemessenen Zeitraum von 3 Wochen mit 12805 an.

In dieser Studie sind sowohl Flooding Attacken (z.B. SYN Flood, DDoS), wie auch die Ausnutzung von Protokollschwächen enthalten.

1.3 Zu klärende Fragen

Die Fragen, die in dieser Ausarbeitung gestellt werden, sind:

- Können wir diesen Fehler kategorisieren und kann man darüber hinaus ein allgemeines Schema aufstellen, welches Netzwerktests kategorisiert?
- Hätte dieser Fehler durch Netzwerktests verhindert werden können?

2. Modell für die Kategorisierung von Netzwerktest

Unter den Begriff Netzwerktests fallen viele Tests, die zur genaueren Betrachtung zuerst in ein Schema eingeordnet werden sollen.

Beispiele für Netzwerktest sind Tests bezüglich der

- Konformität der Implementation des TCP/IP Stacks gegenüber der IETF RFCs (Internet Engineering Task Force Request for Comments)
- erreichbaren Übertragungsraten bei der Verwendung von bestimmten Medien
- Fehleranfälligkeit von Applikationen (ISO/OSI Layer 7) bei Nichteinhaltung von Standards

Ein mögliches Schema ist ein Schaubild, welches durch eine Aspekt, eine Layer und eine variable Achse aufgespannt wird.

2.1 Aspekt-Achse

Die Aspekt-Achse gibt den Aspekt an, welcher getestet werden soll.

- Conformance
- Error
- Performance

Unter **Conformance Tests** fallen alle Tests, die gegenüber einem bestehenden Standard durchgeführt werden können. In Fall von TCP/IP könnten dies die RFCs der IETF sein, die gegenüber der eigentlichen Implementation des Stacks abgeprüft werden. Synonym können hier auch die Ausdrücke Compliance oder Functionality Tests verwendet werden.
Das Ergebnis eines solchen Tests ist ein boolescher Wert.

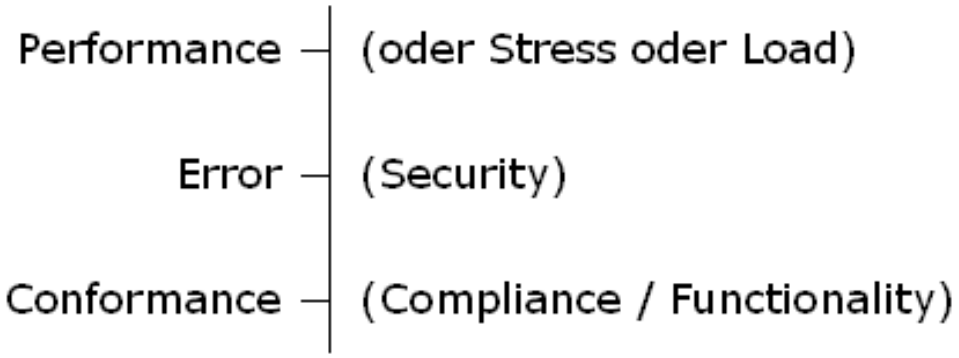
Bei **Fehler Tests** wird die Abweichung in Bezug auf Standards überprüft. Im Prinzip ist der Ausgang dieser Tests auch wieder ein boolescher Wert, nur ergibt sich hier ein Problem. Im Gegensatz zur Einhaltung von Standards kann bei Fehlertests nicht einfach ein Regelwerk Punkt für Punkt abgeprüft werden, sondern es müssen alle mögliche Abweichungen getestet werden, was den Aufwand von Fehler Tests gegenüber Conformance Tests erhöht.

Am Beispiel des Teardrop DoS müsste sowohl ein negativer Wert des zu allokierenden Speichers überprüft werden, wie auch ein zu großer Wert, welcher bereits bestehende Daten im Speicher überschreiben würde.
Also ergibt sich bei diesem Test schon einen doppelt so hoher Aufwand, wobei abzusehen ist, dass, je komplexer der zu prüfende Aspekt wird, desto höher wird der Aufwand für Error Tests. Außerdem ist bei Fehler-Tests immer auch die Kreativität des Testers gefragt, alle möglichen Abweichungen vom Standard zu erkennen.

Da die Fehleranfälligkeit auch immer mit der Sicherheit eines Systems verknüpft ist, kann hier synonym auch der Ausdruck Security Test verwendet werden.

Performance Tests beschreiben die Leistung einer Netzwerkkomponente bezüglich einem Richtwert. Dieser Aspekt ist dann schwierig zu testen, wenn man die Leistung einer Applikation z.B. von SMTP testen möchte, da die Leistung der darunterliegenden Protokolle mit in die Rechnung einfließen, oder wenn überhaupt keine Referenzwerte existieren. Bei SMTP spielen Variablen, wie die darrunterliegenden Protokolle, Leistung des Servers, usw. hinein. Im Beispiel "Email" wird die Schwierigkeit dieses Tests veranschaulicht.

In diesen Aspekt fallen auch Last und Überlast Tests.

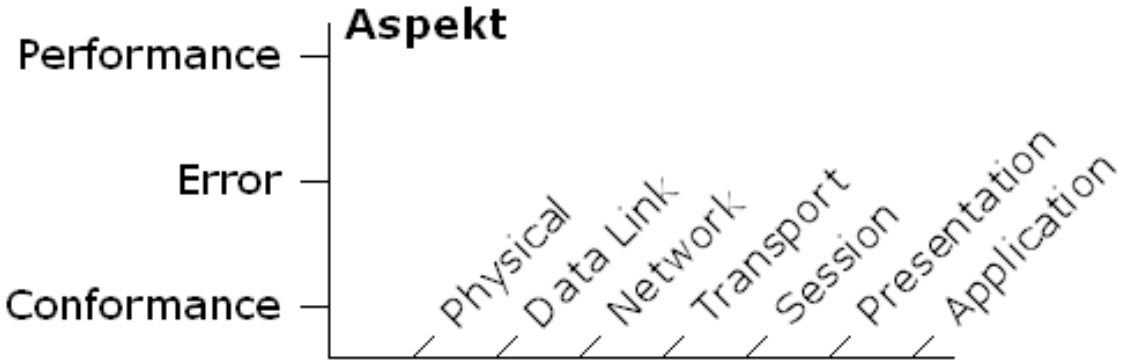


(Abbildung 2: Aspekt-Achse)

2.2 Layer-Achse

Um ein Netzwerk in seine Komponenten zu zerlegen, wird das Standard 7 Schicht ISO/OSI Modell

verwendet [Lewis97]. Somit kann der jeweilige Test auf seiner Schicht identifiziert werden. Im Teardrop Beispiel wäre dies das Network Layer mit der Implementation des IP Protokolls.

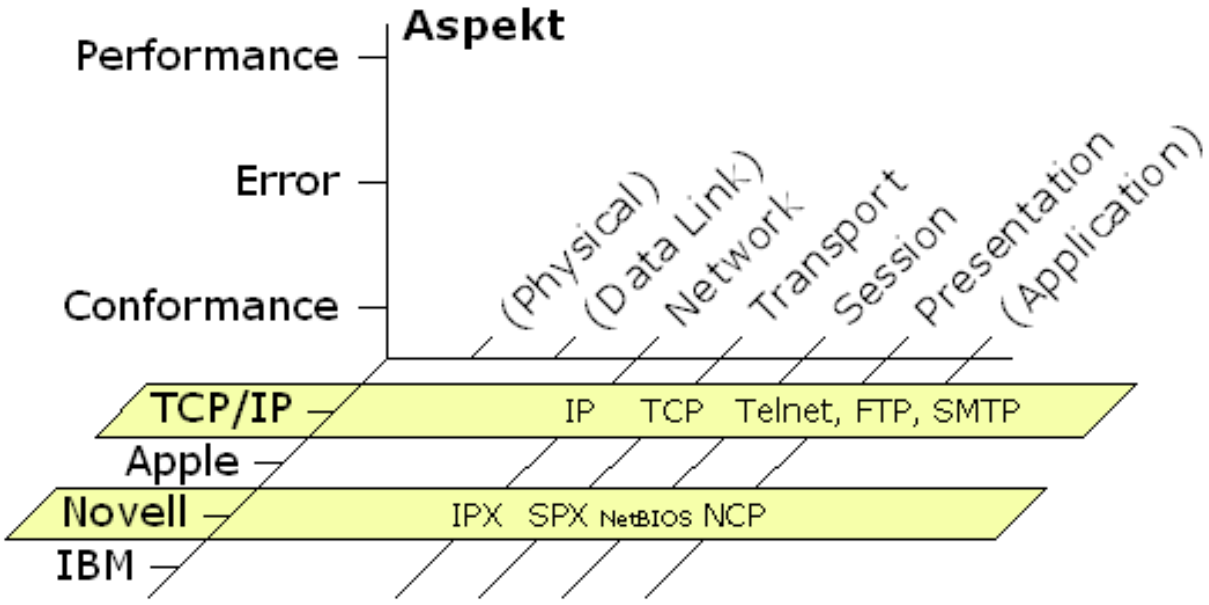


(Abbildung 3: Layer-Achse)

2.3 variable Achse

Auf der variablen Achse wird der jeweilige Network Stack abgetragen. Da die meisten Stacks [Lex] nur auf den Layern 3-6 implementiert sind, muss diese Achse variabel gehalten werden, und bei Tests auf den Layern 1, 2 und 7 jeweils angepasst werden.

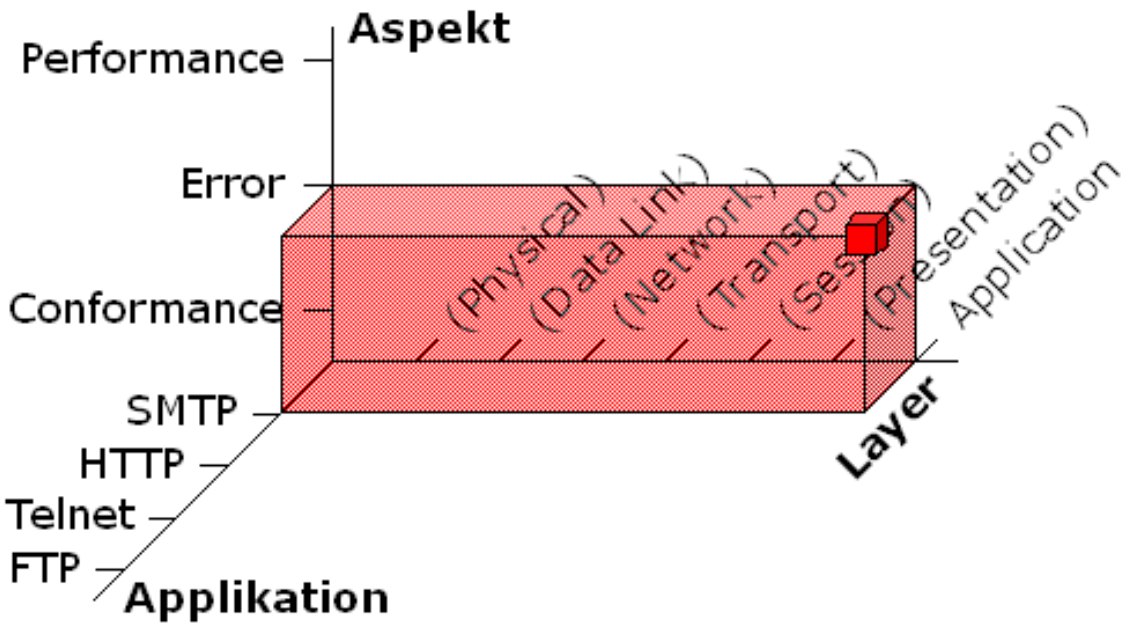
Für ein bestimmtes Layer-Variablen Tupel bekommt man dann das jeweilige Protokoll, wie z.B. für die Implementation des Apple Protokoll auf dem Network Layer das Datagram Delivery Protocol (DDP).



(Abbildung 4: variable Achse)

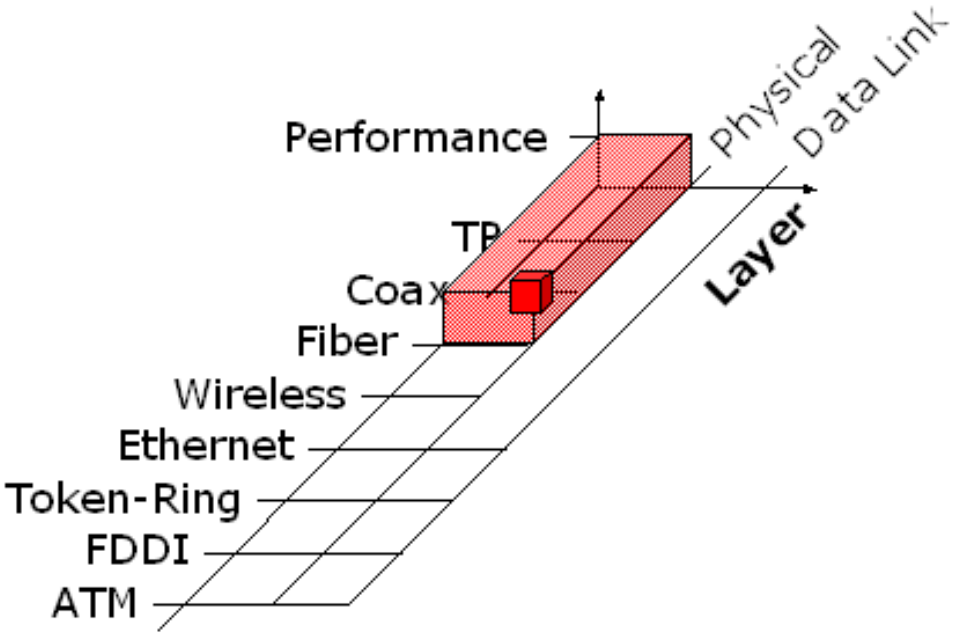
2.4 Layer 1,2 und 7

Bei den Layern 1,2 und 7 muss jeweils die variable Achse angepasst werden. Folgende Schaubilder sollen hierfür Beispiele liefern.



(Abbildung 5: Anpassung der variablen Achse auf das Applikations Layer)

In Abbildung 5 wurde ein Test abgetragen, der eine Implementierung des SMTP Protokolls auf die Abweichung bezüglich der RFCs testet (also z.B. sendmail).



(Abbildung 6: Anpassung der variablen Achse auf Layer 1 und 2)

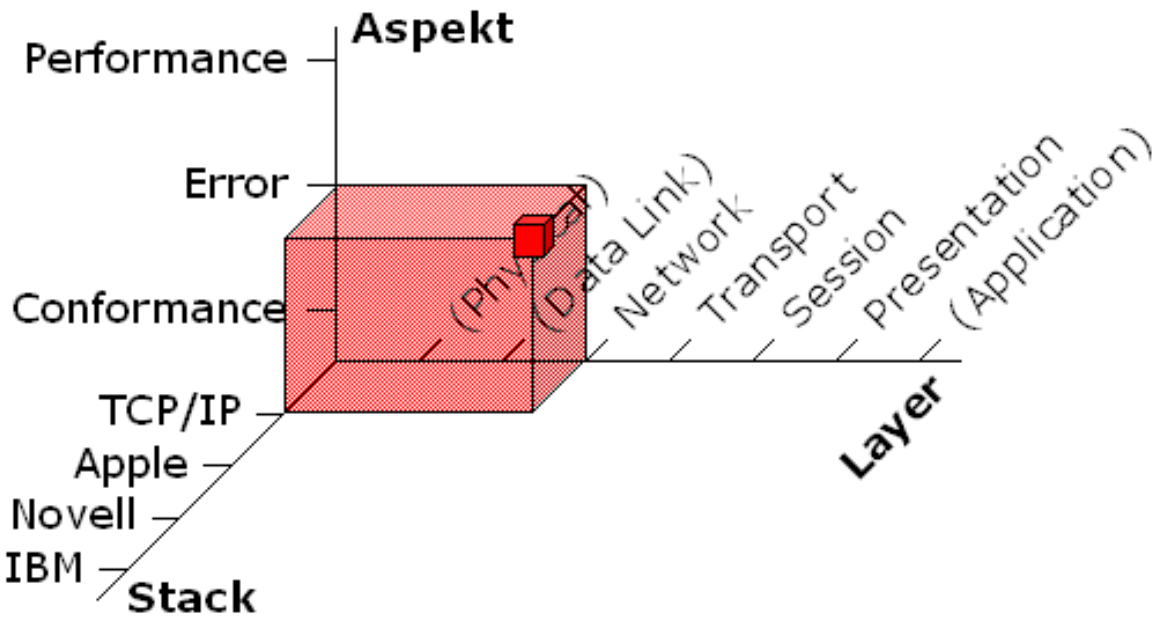
Hier wurde ein Test abgetragen, der die Konformität einer Fiberoptik bezüglich ihren Spezifikationen des Herstellers durchführt.

2.5 Einordnung des Teardrop DoS in das Modell

Wir bekommen für jeden Test einen Punkt im Schaubild, der den Test klassifiziert, aber nicht eindeutig identifiziert.

Die Anordnung der Punkte auf der Aspekt- und Layer-Achse ist so, das für die Funktion eines bestimmten Tests die jeweiligen näher dem Ursprung des Schaubilds liegenden Tests zuvor durchgeführt werden müssen. Natürlich wird dies ist der Praxis nicht gemacht, da bei einem Performance Einbruch im Applikations-Layer (z.B. langsamer Dateitransfer bei FTP) nicht die Konformität des physikalischen Trägers bezüglich seinen Spezifikationen nachgeprüft wird. Jedoch garantiert dieses sukzessive Verfahren das Finden des Fehlers.

In unserem Teardrop Beispiel würden wir folgendes Schaubild bekommen:



(Abbildung 7: Einordnung des Teardrop Beispiels)

Also findet unser Test auf dem Network Layer des TCP/IP Stacks statt (IP), und testet das IP-Protokoll gegen die Verletzung der RFC.

Der entsprechende Abschnitt für diesen Fehler findet sich in RFC 760 [IETF80] des IETF:
"The fragment offset field of the second new internet datagram is set to the value of that field [the offset field] in the long datagram [the unfragmented one] plus NFB [the number of fragment blocks].".
Wie schon in Abschnitt 2.1 angesprochen, hätten hier die Auswirkungen jeder möglichen Abweichung von diesem RFC überprüft werden müssen, um den Fehler zu finden.

Dies hätte die Anwendung eines Packet Generators / Analyzers bedurft, welcher korrupte Pakete aussendet, und die Rückmeldung des sich unter Test befindlichen TCP/IP Stacks ausliest.

Als Beispiel für ein Testsystem, welches in diesem Fall hätte angewandt werden können, wurde eine Software gewählt [Ixia02], welche Protokolle automatisch auf Ihre Konformität und Fehleranfälligkeit prüft. Auch hier wieder das Problem, dass die durchgeführten Tests für Fehleranfälligkeit nur durch die Kreativität der Programmierer bestimmt wird.

Ixia ANVL Test Suites

- IP Test Suites
  IP RIP (v1 and v2) Gateway / OSPFv2 (RFC1583/2328) / BGP4 (RFC1771) / RMON (RFC1757/RFC1513)
- PPP Test Suites
  Basic PPP (with tests for LCP, PAP, and CHAP) / IPCP (RFC1332) / Multilink PPP (RFC1717/RFC1990) / VJ Test Suite (TC/IP, RFC1144) / Spanning Tree (IEEE 802.1d)
- Multicasting Test Suites
  IGMP (RFC2236v2) / DVMRP (IETF Draft 3) / PIM (sold as one unit) / Sparse Mode - IETF Draft#1v2 / Dense Mode - IETF Draft#3v2
- TCP Test Suites
  Core (RFC 793, 1122) / Advanced (RFC 2001, 2581, 1191, 2385) / Performance (RFC 1323, 2018)
- VPN Test Suites
  PPTP (IETF Draft 2) / L2TP (RFC2661) / IPSec AH (RFC2402/2401) / IPSec ESP (RFC2406) / IPSec IKE (RFC2409/2408) / L2TPSec (RFC2661)

Es werden die RFCs Punkt für Punkt durchgegangen, und die Rückmeldungen überprüft. Ist der Test erfolgreich verlaufen, wird eine Eins vermerkt. Dies kann in Software passieren, da hier keine zeitkritischen Performance Tests durchgeführt werden. Bei Performance Tests kommen entweder Rechner Arrays oder Testhardware zum Einsatz.

2.6 Schwächen des Modells

Im Rahmen des Seminars wurde nicht überprüft, ob jedes Tripel in diesem Modell einen sinnvollen Test ergibt. Das heißt, kann das Schaubild komplett gefüllet werden? Existiert z.B. ein Test, der die Performance von Ethernet testet, und ist, falls ein solcher existiert, dieser Test überhaupt von Interesse?

Ein weiterer Schwachpunkt ist die variable Achse, die eine andere Herangehensweise an die Kategorisierung nahe legt. Ein Tripel aus Mengen, die die jeweiligen Achsen repräsentieren, könnte hier eine mögliche Lösung sein. Jedoch geht bei diesem alternativen Modell der Einschlusscharakter der Aspekt- und Layer-Achsen verloren.

3. Weiteres Beispiel: Email

Alle bisherigen Beispiele hatten immer einen Standard oder Referenzwert, welcher den positiven Ausgang des Tests definierte. Dies ist aber nicht immer der Fall. Bei Performance Tests ist das Ergebnis des Tests nicht so einfach festzulegen.

Wie findet man hier Referenzwerte?

Ein Beispiel [Acter02]:

Der Administrator eines Firmennetzes erhält von einem einem Kollegen folgende Beschwerde: "Meine Email ist langsam!"

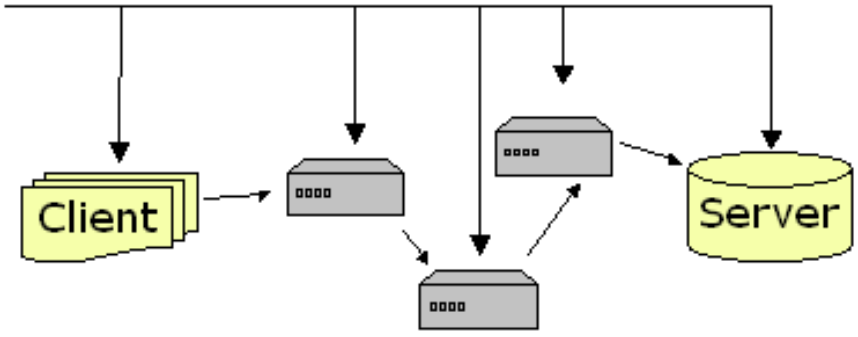
Dieser Satz verlangt in unserem Schema nach einem Performance Test, welcher auf dem Applikationslayers des TCP/IP Stacks durchgeführt wird. Also testet man SMTP auf seinen Durchsatz und die Latenzzeit.

(Abbildung 8: Beteiligte Netzwerkkomponenten)

Die Beteiligten Netzwerkkomponenten sind zum einen der Client des Mitarbeiters, das Firmen LAN, und der Email-Server. Um die Analyse durchzuführen betrachtet man das System aus 2 Perspektiven. Das eingesetzte Werkzeug ist wieder ein Protokoll Analyzer, der zum einen den Fluss der übermittelten Pakete mit Zeitmarken versieht, und so eine Laufzeitanalyse erlaubt, und zum anderen den Durchsatz von Paketen messen kann, die von den Netzwerkkomponenten übermittelt werden.

3.1 Netz

Im Idealfall wird bei der Analyse des Netzes an jede Komponente ein Agent des Protokollanalyzers gekoppelt, sodass ein Paket, welches der Client zum Server schickt (und vice versa) während seines gesamten Weges betrachtet werden kann. Mit dieser Methode können Verzögerungen bei der Weiterleitung des Packets leicht identifiziert werden.



(Abbildung 9: Analyse des Netzwerks)

Da das Netz normalerweise im Betrieb, also unter Last, getestet wird, sollte dieser Test wiederholt gemacht werden, da Einflüsse wie

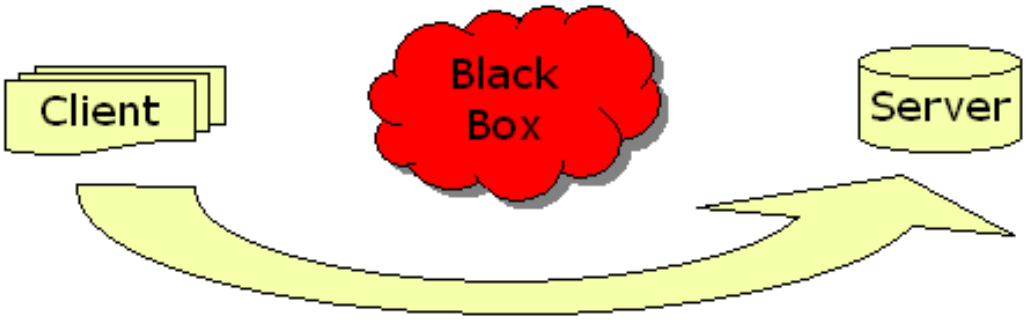
- Netzwerklast

- Kollisionen
- Retransfers
- Unterschiedliche Routen der Pakete (bei größere Netzen)

die Messung beeinflussen werden.

3.2 Client und Server

Der zweite Schritt wäre der Test des Client und des Servers. Es wird nun angenommen, dass das dazwischenliegende Netz keinen Einfluss auf den Datenstrom hat (Black Box), so als wäre der Client direkt mit den Server verbunden.



(Abbildung 10: Analyse des Clients & Servers)

In diesem Schritt können mit einem Protokollanalyzer Größen wie Think Time von Client und Server bei Anfragen, und Datenrate des Servers identifiziert werden. Auch hier sollte der Test wiederholt werden, da Einflüsse wie

- momentane Serverlast

die Messung beeinflussen.

3.3 Zusammenfassung

In diesem Beispiel sollte verdeutlicht werden, dass bei einem Performance Test eines Netzes viele Komponenten mit hineinspielen, und der Test somit aufwendig zu implentieren und auszuwerten ist. Die Auswertung wird nicht selten mit Erfahrungswerten durchgeführt, und nicht mit Referenzdaten der einzelnen Komponenten.

4. Ausblick

Eine weitere Domäne für Netzwerktests sind Sprachdienste über Best-Effort Netzwerke, wie VoIP. Hier interessiert vor allem die Performance, die auch Quality of Service genannt wird. Sprachdienste stellen besondere Ansprüche an die Performance, die mit folgenden Eckdaten verdeutlicht werden sollen:

- Verfügbarkeit und garantierter Durchsatz
  z.B. Muss bei dem Audioprotokoll G.723.1, welches bei Videokonferenzen eingesetzt wird, ein

effektiver Durchsatz von 6.4 kbps (UDP: 13.3 kbps) garantiert werden. Und dies über die ganze Länge des Gesprächs.
Da diese Gespräche über das Internet geführt werden, muss eine unterschiedliche Priorität für die Pakete festgelegt werden, die Massendaten, wir News niedriger Einstufen als Echtzeitdaten. Eine solche Prioritarisierung ist in IPv6 implementiert.

- geringe Latenz (< 250 ms) und Änderung der Latenz (Jitter)
  Um ein möglichst natürliches Sprachgefühl zu erhalten, sollte Latenz und Jitter so gering wie möglich gehalten werden.

Das hier erstellte Modell kann auch diese Tests einordnen, wobei eine Aufsplittung der Kategorie Performance in abgefragten Unterpunkte wie Durchsatz, Latenz und Jitter eine feinere Auflösung und Identifikation der Tests bietet.

5. Quellenverzeichnis

- [Acter02]
  White Papers
  Acterna, LLC
  http://www.acterna.com/technical_resources/white_papers/index.html
- [Burg00]
  Week 15: TCP/IP security
  Prof. Mark Burgess, University College Oslo, Faculty of Engineering, Norway
  http://www.iu.hio.no/~mark/lectures/sysadm/html/SA15.eng.html
- [CSI99]
  Computer Security Institute and Federal Bureau of Investigation
  1999 CSI/FBI Computer Crime and Security Survey, Computer Security Institute publication, 1999.
- [CPTIT]
  -
  Corporate Persuasion Through Internet Terrorism
  http://63.105.33.158/security/denial/w/teardrop.dos.html
- [Ixia02]
  ANVL™ Automated Network Validation Library
  Ixia
  http://www.ixiacom.com/products/paa/anvl/anvl_protfamilies.php
- [Evans02]
  Notes on Texas Instruments Processors
  Brian L. Evans, The University of Texas, Austin
  http://www.ece.utexas.edu/~bevans/courses/realtime/lectures/23_DSPs/texasInstruments.html
- [IETF80]
  DoD Standard Internet Protocol
  Information Sciences Institute, University of Southern California, 1980
  http://www.ietf.org/rfc/rfc0760.txt
- [Lewis97]
  James Bond Meets The 7 Layer OSI Model, 1997
  Richard Lewis
  http://www.pe.net/~rlewis/Resources/james.html
- [Lex]
  Protocol Stacks in Relationship to the OSI Model

Lexicon Consulting
http://www.lex-con.com/osimodel.htm

- [Moore01]
  Inferring Internet Denial-of-Service Activity
  David Moore, Geoffrey Voelker und Stefan Savage, CAIDA University of California, San Diego
  http://www.caida.org/outreach/papers/2001/BackScatter/

6. Abbildungsverzeichnis

Verbesserungsvorschläge und Kommentare bitte an:
Horst.Rechner@gmx.de