Institut für
Organisation und Management
von Informationssystemen
Prof. Dr. H. P. Großmann

ulm university universität
uulm

Praktikum
Informations- und Medientechnik

Sommersemester 2012

Inhaltsverzeichnis

Allgemeine Hinweise **5**

Vorversuch, Teil 1: Aufbau und Konfiguration eines IP-Netzes **7**

Vorversuch, Teil 2: Anwendung eines Packet-Sniffers zur Netzwerkanalyse **11**

Versuch 1: Übertragungsparameter **15**

Versuch 2: Niedrige Protokolle / Broadcastprotokolle **19**

Versuch 3: Höhere Protokolle / Dialogbasierte Protokolle **23**

Versuch 4: VoIP-/Videokonferenz-System: Setup und Protokollanalyse **27**

Versuch 5: Videostreaming: Setup mit Multicast/Unicast **29**

Versuch 6: Quality of Service (QoS), Traffic Engineering **33**

Versuch 7: IPv6 **37**

Versuch 8: Network Simulator Version 2 (NS-2) **41**

Allgemeine Hinweise

- Die IP-Adressen im Praktikum sind folgendermaßen aufgebaut: 10.x.y.z
  Dabei ist x die Versuchsnummer, y das Subnetz im Versuch und z ein Rechner im Subnetz.
- Rechner mit mehr als einem Netzwerk-Interface dienen im Praktikum als Router. Die Interfaces nennen sich bei Linux eth0, eth1, ...
- Alle Praktikumsrechner befinden sich in der Domäne itm.e-technik.uni-ulm.de.
- Alle RFCs findet man im Internet unter http://www.ietf.org/rfc.html.

Vorversuch, Teil 1: Aufbau und Konfiguration eines IP-Netzes

Literatur zur Einarbeitung

- Andrew S. Tanenbaum: Computernetzwerke
- RFC0791: IP – Internet Protocol
- RFC0792: ICMP – Internet Control Message Protocol
- RFC0826: ARP – An Ethernet Address Resolution Protocol
- RFC1519: CIDR – Classless Inter-Domain Routing
- Manual-Pages: ifconfig, route, arp

Vorbereitungsfragen

1. Wie ist unter Linux eine Manual-Page aufgebaut?
   (Sollten Sie sich damit noch nie beschäftigt haben, loggen Sie sich auf einem UNIX/Linux-Rechner ein und schauen Sie sich eine Manual-Page an indem Sie z. B. `man man` eingeben.)
2. Wofür werden IP-Adresse, Subnetzmaske und Default-Gateway benötigt?
3. Wie ist eine Broadcast-Adresse aufgebaut und wofür wird sie benötigt?
4. Warum besitzt jeder Rechner eine IP- und eine MAC-Adresse? Überlegen Sie sich dazu zunächst wie ein Netzwerk funktionieren würde, das nur IP- bzw. MAC-Adressen verwendet.
5. Was versteht man unter dem Loopback-Device und welche Funktion hat es?
6. Wie verhält sich ein Rechner mit der Konfiguration 192.168.1.1/255.255.255.1 im Subnetz 192.168.1.0/24?
   (Hinweis: /24 gibt die von links gezählte Anzahl gesetzter Bits in der Subnetzmaske an und ist damit gleichbedeutend mit /255.255.255.0)
7. Welche (Unter-)Netze können mit 192.168.1.x/26 gebildet werden?
8. Erklären Sie kurz das Verfahren der Unterteilung der IP-Adressen in Klassen. Was für ein Netz hat die Uni Ulm? Ist die Anzahl der zur Verfügung stehenden IP-Adressen für die Uni Ulm ausreichend oder eher knapp?
9. Was versteht man unter CIDR (Classless Inter-Domain Routing)? Welche Probleme/Nachteile wurden durch die Einführung von CIDR behoben?
10. Welche Informationen beinhaltet eine Routing-Tabelle und warum wird sie gebraucht?

11. Wozu dient ICMP?
12. Wozu dient ARP bzw. RARP?
13. Sie wollen ein ICMP-Paket (Echo Request) aus dem Praktikumsnetz an 134.60.1.25 (www.uni-ulm.de) schicken. Beschreiben Sie den genauen Ablauf. Berücksichtigen Sie ARP und ICMP. Gehen Sie davon aus, dass sich zwischen Ihrem Rechner im Praktikum und www.uni-ulm.de 3 Router befinden.

Versuchsaufbau

Dieser Versuch kann an jedem Praktikumsrechner durchgeführt werden.

Durchführung

Konfiguration der Netzwerkkarte

Üblicherweise nimmt man zur Konfiguration der Netzwerkkarten Tools wie z. B. YaST zur Hilfe. Um die Funktionsweise eines IP-Netzes besser zu verstehen, sollen die Netzwerkkarten in diesem Versuch „von Hand“ konfiguriert werden. Beenden Sie dazu die laufende Konfiguration mit `/etc/init.d/network stop` bzw. `/etc/init.d/networking stop`.

Stellen Sie zunächst eine Netzkonfiguration zusammen (IP-Adresse, Netzmaske, Broadcast-Adresse, Gateway-Adresse und Routing-Informationen). Konfigurieren Sie dann das Netzwerkdevice eth0 und fügen Sie der Routing-Tabelle die Default-Route hinzu. Die benötigten Befehle lauten `ifconfig` und `route`. Schauen Sie in den Manual-Pages nach wie die Befehle aufgerufen werden müssen.

Testen Sie Ihre Konfiguration nun mit dem Befehl `ping` indem Sie einem beliebigen Rechner außerhalb Ihres Subnetzes (z. B. www.uni-ulm.de) ein Datenpaket schicken. Nehmen Sie auch hier die Manual-Page zur Hilfe.

Schauen Sie sich nun nochmal die Einträge der Routing-Tabelle an und erklären Sie deren Funktion.

Address Resolution Protocol (ARP)

Wie muss der Befehl `arp` aufgerufen werden, um die Einträge der ARP-Tabelle anzuzeigen oder zu löschen? Lesen Sie dazu ebenfalls die Manual-Page.

Schicken Sie den Rechnern des Subnetzes einen Ping (Broadcast-Ping). Betrachten Sie danach die Einträge der ARP-Tabelle. Welche MAC-Adressen sind gespeichert? Was sind die Vor- und Nachteile des Speicherns von MAC-Adressen in einem Cache? Wie lange sind solche Einträge sinnvollerweise vorhanden? Weshalb?

Schicken Sie nochmal einen Ping zum Router und betrachten Sie danach die Einträge in der ARP-Tabelle. Schicken Sie nun einen Ping an www.uni-ulm.de (134.60.1.25)

und betrachten Sie wiederum die ARP-Tabelle.
Was hat sich geändert? Wieso gibt es keinen ARP-Eintrag für den Webserver der Uni Ulm?

Internet Control Message Protocol (ICMP)

In der letzten Aufgabe haben wir die ARP-Tabelle beim Versenden von Ping-Paketen angeschaut. Ein Ping wird oft zur Messung der Verzögerungen, die ein Netzwerk aufweist, eingesetzt. Pingen Sie www.uni-ulm.de, www.heise.de, www.ecu.edu.au und www.hawaii.edu an. Wie lange dauert es nach Absenden des Pakets bis eine Antwort registriert wird? Kommentieren Sie die Verzögerungen.

Vorversuch, Teil 2: Anwendung eines Packet-Sniffers zur Netzwerkanalyse

Literatur zur Einarbeitung

- Andrew S. Tanenbaum: Computernetzwerke
- RFC0768: UDP – User Datagram Protocol
- RFC0791: IP – Internet Protocol
- RFC0792: ICMP – Internet Control Message Protocol
- RFC0793: TCP – Transmission Control Protocol
- RFC0826: ARP – An Ethernet Address Resolution Protocol
- RFC1034 / RFC1035: DNS – Domain Name System
- Einführung und Dokumentation zum Netzwerksniffer Wireshark (http://www.wireshark.org/)

Vorbereitungsfragen

1. Für welche Zwecke läßt sich Packet-Sniffing nutzen?
2. Was versteht man unter dem Promiscuous Mode und wofür wird dieser benötigt?
3. Was ist der Unterschied zwischen Hub, Switch und Router?
4. Wie verhält sich ein Packet-Sniffer am Hub im Gegensatz zu einem Switch?
5. Was müßte man theoretisch machen, um den Verkehr in einem geswitchten Netzwerk abzuhören?
6. Was unterscheidet das Protokoll Telnet von SSH?
7. a) Welche Informationen sind in einem Ethernet-Frame enthalten?
   b) Gibt es auch andere Ethernet-Frame Versionen?
8. Welche Informationen sind in einem IP-Paket enthalten? Was ist der Sinn des Feldes TTL (Time to Live)?
9. Warum hat man das DNS eingeführt? Wie würde man in einem Netz ohne DNS z. B. eine Webseite aufrufen?
10. Warum macht es keinen Sinn, einem Nameserver einen sich leicht zu merkenden Namen zu geben?
11. Worin liegt der Unterschied zwischen primären und sekundären Nameservern? Welche Vor- und Nachteile ergeben sich durch den Einsatz von sekundären Nameservern?

12. Welche Probleme entstehen durch die Verwendung von Caching-Nameservern, die erhaltene Antworten speichern und nicht bei jeder Anfrage erneut den zuständigen Nameserver fragen?

13. Welche Nachteile ergeben sich durch die Verwendung von dynamischen Adressen in einem Rechnernetz? Welche Möglichkeit gibt es diese Nachteile zu umgehen?

Versuchsaufbau

Dieser Versuch kann an jedem Praktikumsrechner durchgeführt werden.

Durchführung

Beispiel: Mitschreiben des Netzwerkverkehrs

Wir möchten einen ARP-Vorgang im Detail betrachten. Rufen Sie zunächst den Packetsniffer Wireshark bzw. Ethereal (Abbildung 1) auf und wählen Sie Capture →Options... (Abbildung 2). Pingen Sie nun einen Rechner Ihres Subnetzes an. Was zeigt Ihnen Wireshark alles an? Woran kann es liegen wenn Sie keine ARP-Pakete aufgenommen haben?

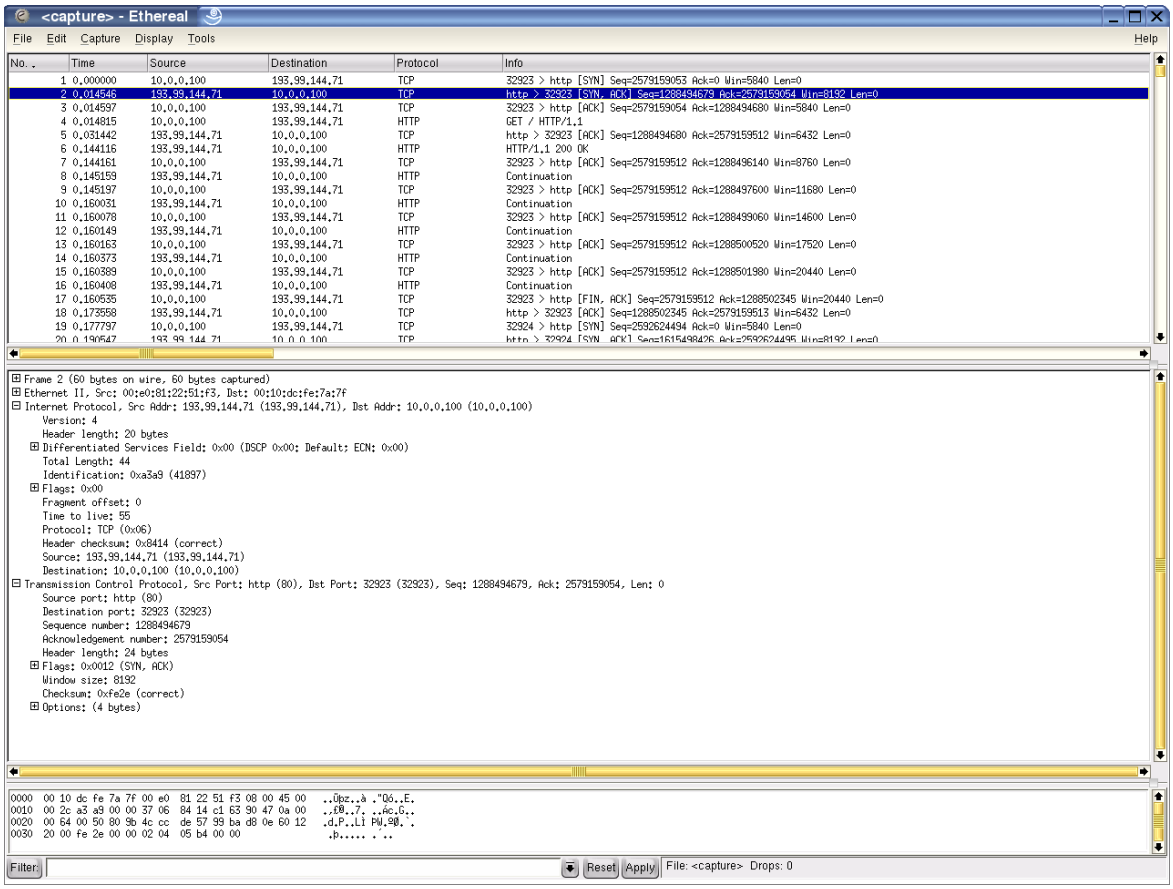


Abbildung 1: Screenshot des Wireshark Network Protocol Analyzer

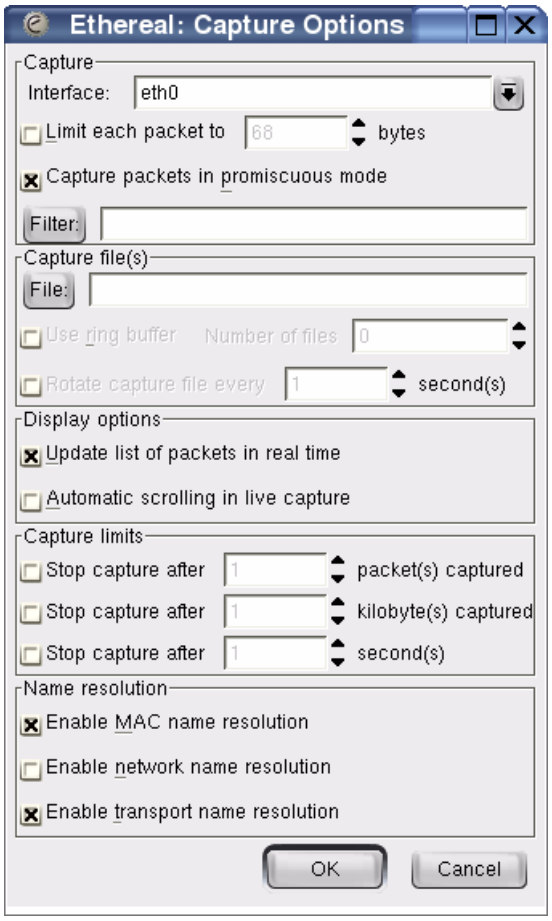


Abbildung 2: Screenshot des „Capture Options“ Dialogs

Analysieren Sie nun den Inhalt eines ARP-Pakets. Was für Typen von ARP-Paketen haben Sie protokolliert? Welcher Rechner verschickt welches Paket? An wen sind diese jeweils gerichtet? Welche Informationen werden übertragen? Was wird durch die übermittelte Information ermöglicht?
Wireshark kann den Hardware-Hersteller der Ethernet-Karte erkennen. Warum ist das möglich?

Erste Schritte mit Wireshark

- Rufen Sie eine (einfache) Webseite auf, schneiden Sie den Abruf mit und analysieren Sie diesen. Hier kann die Funktion „Follow TCP Stream“ weiterhelfen.

- Versuchen Sie bei einer Telnet-Session das Passwort zu finden (`telnet routeqos`; Benutzername: praktikum; Passwort: praktikum). Auch hier kann die Funktion „Follow TCP Stream“ weiterhelfen.

- Vergleichen Sie die Telnet-Sitzung mit einer SSH-Sitzung.

Internet Protocol (IP)

Schauen Sie sich den Header eines IP-Paketes an. Welche Informationen sind dort enthalten? Ist das Paket gültig oder wurde es fehlerhaft übertragen? Wieviele Bytes Daten enthält das Paket? Was sind das für Daten?

`traceroute` ist ein Programm, dass die gesamte Strecke (engl.: route) zu einem Ziel aufspürt (engl.: trace).
Starten Sie den Netzwerksniffer und analysieren sie den Weg der Datenpakete zum Rechner www.uni-ulm.de. Versuchen Sie anhand der protokollierten Datenpakete die Funktionsweise von `traceroute` zu verstehen.
Wiederholen Sie das Ganze mit dem Rechner www.hawaii.edu. Warum melden sich manche Rechner nicht und warum kann die Route trotzdem weiterverfolgt werden?

Transmission Control Protocol (TCP) und User Datagram Protocol (UDP)

Rufen Sie ein Webseite auf und analysieren die die TCP-Pakete. In welche Abschnitte lässt sich eine TCP-Verbindung einteilen? Wie sehen diese Abschnitte prinzipiell aus?

Analysieren Sie die von `traceroute` verschickten UDP-Pakete. Was fällt Ihnen hier im Vergleich zu TCP auf?

Domain Name Service (DNS)

Schicken Sie einen Ping an imap.uni-ulm.de und sniffen Sie mit. Wie schlüsselt das Betriebssystem imap.uni-ulm.de zu einer IP-Adresse auf? Welches Protokoll wird auf der Transportschicht für DNS eingesetzt und warum? Auf welchen Port hört der Domain Name Server?
Der IMAP-Server der Uni heißt nicht wirklich „imap“. Wie ist der Hostname dieses Servers und wie lautet seine IP-Adresse. Wie werden mehrere Namenseinträge für einen Rechner realisiert?
Der Eintrag des Servers besitzt ein Feld „Time to Live“. Was bewirkt dieser Eintrag? Wie groß ist diese Zeit?

Unter Umständen finden Sie in Ihren Netzwerkmitschnitten ICMP-Nachrichten die nicht von einem Ping-Kommando herrühren. Können Sie sich vorstellen wodurch diese veranlasst sind und welchen Zweck sie haben?

Versuch 1: Übertragungsparameter

Literatur zur Einarbeitung

- Andrew S. Tanenbaum: Computernetzwerke
- RFC0768: UDP – User Datagram Protocol
- RFC0791: IP – Internet Protocol
- RFC0793: TCP – Transmission Control Protocol
- Dokumentation zur Netzwerk-Emulation NIST Net (http://www-x.antd.nist.gov/nistnet/)
- Dokumentation zur Software Iperf zum Erzeugen/Messen von Netzwerktraffic (http://dast.nlanr.net/Projects/Iperf/)

Vorbereitungsfragen

1. Was versteht man unter der MTU und welche Bedeutung hat dieser Wert?
2. Wieviele Daten kann ein IP-Paket maximal in einem Ethernet-Netzwerk transportieren? Was passiert wenn mehr Daten verschickt werden sollen?
3. Wie wirken sich Paketverluste auf TCP- und UDP-Verbindungen aus?
4. Erklären Sie kurz die Funktion der Felder „Sequence Number“ und „Acknowledgement Number“ im TCP-Header. Was versteht man unter der ISN (Initial Sequence Number)?
5. Beschreiben Sie den TCP-Verbindungsaufbau. Geben Sie für jedes Paket die Sequence Number und die Acknowledgement Number an. Gehen Sie davon aus, das der Client die ISN zu 100 und der Server zu 5000 wählt.
6. Gehen Sie nun davon aus, dass Sie als Empfänger von Daten ein TCP-Paket mit folgenden Werten im Header verschickt haben.
   Sequence Number: 2721058443; Acknowledgement Number: 530153982; Flags: ACK; Window Size: 63712; keine Daten
   Was bedeutet dieses Paket für den Sender? Welche Sequence Number und Acknowledgement Number hat das nächste Paket des Senders?
7. Was versteht man unter Flusskontrolle und welche Bedeutung hat der Parameter „Window Size“ bei einer TCP-Übertragung?
8. Erklären Sie den Slow-Start-Algorithmus von TCP.
9. Wie kann die effektive Datenrate maximiert werden?

Versuchsaufbau

Dieser Versuch besteht aus drei Rechnern. Zwei davon sollen als Server (*server*) bzw. Client (*client*) eingesetzt werden. Beide sind an einen dritten Rechner (*packet*) angeschlossen, der als Router konfiguriert ist und auf dem die Netzwerk-Emulationssoftware NIST Net läuft. Dieser Rechner besitzt zusätzlich einen Uplink zum Masquerading-Server.

Durchführung

Fragmentierung von IP-Paketen

Schicken Sie von *client* einen Ping an *server* und geben Sie als „packetsize“ 8000 Byte an. Sniffen Sie den Ping mit und erläutern Sie was passiert.
Erklären Sie anhand dieses Beispiels die Funktion der Felder „Identification“, „Flags“ und „Fragment Offset“ im IP-Header.

Erzeugen/Messen von Netzwerktraffic

Stellen Sie mittels Iperf eine TCP-Verbindung zwischen dem Server und dem Client her. Geben Sie dazu auf dem Server das Kommando `iperf -s` und auf dem Client das Kommand `iperf -c server` ein.
Lesen Sie die effektive Datenrate ab und erläutern Sie deren Wert.

Variieren Sie den Parameter Window-Size (2k, 4k, 8k, 16k und 32k). Verwenden Sie dazu die Option `-w Window-Size` von Iperf.
Lesen Sie die Werte für die entsprechende effektive Datenrate ab, mitteln Sie über mehrere Messungen und tragen Sie die Ergebnisse graphisch auf. Erläutern Sie das so entstandene Diagramm. In Bereichen starker Änderung nehmen Sie gegebenenfalls weitere Messungen vor.

Netzwerk-Emulation

Bevor Sie die Netzwerk-Emulationssoftware NIST Net verwenden können müssen Sie mit `~/nistnet-<version>/Load.Nistnet` das benötigte Modul in den Kernel laden. Verwenden Sie die kommandozeilenbasierte Version von NIST Net. Sie können auch die grafische Version verwenden, diese ist jedoch nicht stabil. Vermeiden Sie es mit der Maus Zahlen zu markieren. Ein Absturz des grafischen Frontends bleibt in jedem Fall folgenlos für die Simulation.

- Stellen Sie empirisch fest in welcher Richtung die Übertragung von Datenpaketen durch Iperf erfolgt.
- Tragen Sie die resultierende Datenrate in Netzen mit 0, 3, 6, 9, 12, 15 % Paketfehlerrate graphisch auf. Führen Sie jeweils 5 Messungen durch und mitteln Sie die Teilergebnisse. Erläutern Sie das Ergebnis, was passiert auf TCP Ebene genau.

- Tragen Sie nun die resultierende Datenrate in Netzen mit 5, 10, 20, 50, 100, 200, 500 ms Verzögerung graphisch auf. Fertigen Sie drei Messreihen an, in denen die jeweils nur eine und einmal beide Transferrichtungen verzögern. Führen Sie hier nur 3 Messungen durch und mitteln Sie die Teilergebnisse. Erläutern Sie die Ergebnisse der Messreihen.

- Simulieren Sie eine DSL Datenübertragungsstrecke z. B. T-DSL 3000.
  - Auf welche Werte müssen Sie die maximale Up- und Downloadrate und die Verzögerungszeiten einstellen?
  - Wie groß ist die ideale Windowsize für Datenströme in jeweils eine Richtung?
  - Senden Sie bei ausgelasteter Datenstrecke außerdem ping Pakete in beiden Richtungen, wie groß ist die Verzögerungszeit.
  - Weiterhin versuchen Sie beide Datentransferrichtungen auszulasten. Wie verändert sich die Windowsize und wie die ping Zeiten.

  Analysieren Sie die Datenübertragung, fertigen Sie Schaubilder an.

- Über eine DSL Strecke (T-DSL 1000) soll nun in Downloadrichtung ein Datentransfer mit möglichst großer Windowsize gestartet werden. Nehmen Sie eine Zeitlang (ca. 30 s) den Datentransfer inklusive der Startphase mit Wireshark auf. Wie verändert sich die Windowsize in den TCP Paketen. Verursachen Sie nun mit NIST Net eine geringe Fehlerrate auf der Datenleitung (z. B. 0,1 %) und wiederholen die Messung. Erläutern Sie die Ergebnisse.

Versuch 2:
Niedrige Protokolle / Broadcastprotokolle

Literatur zur Einarbeitung

- Andrew S. Tanenbaum: Computernetzwerke
- RFC0768: UDP – User Datagram Protocol
- RFC0783 / RFC1350: TFTP – Trivial File Transfer Protocol
- RFC0791: IP – Internet Protocol
- RFC0951: BOOTP – Bootstrap Protocol
- RFC1034 / RFC1035: DNS – Domain Name System
- RFC2136: Dynamic Updates in the Domain Name System (DNS UPDATE)
- RFC2137: Secure Domain Name System Dynamic Update
- DNS HOWTO
- Dokumentation zu BIND9.3
- RFC2131: DHCP – Dynamic Host Configuration Protocol
- RFC2132: DHCP Options and BOOTP Vendor Extensions
- Preboot Execution Environment (PXE) Specification, Version 2.1 (ftp://download.intel.com/labs/manage/wfm/download/pxespec.pdf)

Vorbereitungsfragen

1. Welche Hierarchie steckt hinter dem DNS?
2. Wie erfolgt die Auflösung eines Hostnamens in eine IP-Adresse? Nennen Sie dabei die beteiligten Instanzen eines DNS-Systems.
3. Sind IP-Adressen hierarchisch angelegt?
4. Welche Probleme ergeben sich bei der Auflösung einer IP-Adresse in einen Domainnamen?
5. Wie lauten die wichtigsten DNS-Datensatztypen?
6. Was versteht man unter authoritative und non-authoritative Antworten?
7. Wie wird eine Zone definiert? Was sind die Schnittstellen zu anderen Zonen?
8. Was versteht man unter „dynamischen“ DNS? Wie funktioniert es und wozu wird es vorwiegend eingesetzt?

9. Wie kann dieses System sicher gestaltet werden ?
10. Was benötigt man für die dynamische Adresszuweisung? Warum muss die Anfrage des Clients per Broadcast geschehen?
11. Was versteht man unter „Leasing“?
12. Wie funktioniert BOOTP?
13. Welches Problem hatte die ursprüngliche BOOTP-Version?
14. Wofür wird DHCP verwendet und wie funktioniert es?
15. Erklären Sie die Bedeutung folgender Pakete:
    DHCP-DISCOVER, DHCP-OFFER, DHCP-REQUEST, DHCP-ACK, DHCP-NACK, DHCP-DECLINE und DHCP-RELEASE
16. Welchen Zweck erfüllt ein DHCP-Relay-Agent?
17. Was ist TFTP und wie funktioniert es?
18. Was versteht man unter PXE (Preboot Execution Environment)?
19. Wie läuft der Bootvorgang via PXE ab?

Versuchsaufbau

Dieser Versuch besteht aus zwei Rechnern (*kaiserschmarrn* und *apfelmus*), die beide an einen Hub angeschlossen sind, um jeweils den Traffic des anderen Rechners mitsniffen zu können. Beide Rechner befinden sich in der Subdomain *dnsversuch.itm.e-technik.uni-ulm.de*. Zum Starten des Rechners im normalen Modus muss zu Beginn des Startvorgangs SHIFT+TAB gedrückt werden.

Durchführung

Domain Name Service (DNS)

Pingen Sie www.uni-ulm.de an. Protokollieren Sie den Netzwerkverkehr mit und beschreiben Sie den Vorgang der Namensauflösung. Welche Daten werden bei einer DNS-Anfrage übertragen und welche Informationen schickt der Nameserver zurück?

Nun soll die hierarchische Struktur des DNS betrachtet werden. Mit dem Tool `dig` können Sie Anfragen an einen beliebigen DNS-Server stellen. Fragen Sie zunächst den Root-Nameserver a.root-servers.net was er über die Zone „.“, also die Wurzel der Domainhierarchie kennt. Geben Sie dazu `dig @a.root-servers.net . any` ein. Welche Informationen schickt Ihnen der Nameserver zurück?
Befragen Sie den Root-Nameserver über die nächste Hierarchiestufe (für deutsche Domains also „de.“) und über „uni-ulm.de.“. Sind die Antworten authoritative oder non-authoritative? Wie heißen die authoritative Server?
Fragen Sie sich weiter durch bis Sie einen Nameserver gefunden haben, der für die

Subdomain rz.uni-ulm.de zuständig ist und fragen Sie diesen nach der IP-Adresse des Rechners login.rz.uni-ulm.de.

Welcher Eintragungstyp (RR – Resource Record) wird für das Versenden von E-Mails benötigt? Welcher Mail-Server nimmt E-Mails entgegen, die an vorname.nachname@uni-ulm.de adressiert sind?

Rufen Sie `dig` nun so auf, dass Ihnen die IP-Adresse 134.60.237.254 in einen Domainnamen aufgelöst wird. Welche Einträge des DNS werden dazu abgefragt?

Konfiguration eines Domain Name Servers

Verwenden Sie für diesen Versuchsteil den Rechner *apfelmus* Auf diesem ist Name Server Software bind installiert. Die benötigten Konfigurationsdateien liegen im Verzeichnis `/etc/bind`. Konfigurieren Sie den Name Server nun zunächst so, dass er als „caching-only“ Server arbeitet. Testen Sie die Konfiguration mit dem Tool `dig` und durch mitsniffen der Pakete. Welche Antworten sind authorative?
Erweitern Sie die Funktionalität nun so, dass apfelmus ein Master Nameserver der Domaine *dnsversuch.itm.e-technik.uni-ulm.de* ist. Testen sie erneut die Konfiguration mit dig. Welche Antworten sind nun authorative? Konfigurieren Sie *kaiserschmarrn*, so dass er auf *apfelmus* als Name Server zurückgreift.

Dynamisches DNS

Im folgenden gehen wir davon aus, dass *kaiserschmarrn* häufig wechselnde IP-Adressen per DHCP zugewiesen bekommt und diese seinem DNS-Server dynamisch mitteilen möchte. Konfigurieren Sie den Server so, dass dieser Updates unterstützt. Achten Sie dabei darauf, dass die Datenintegrität während der Übertragung gewährleistet ist. Um ein Update eines Eintrags im Nameserver zu verursachen, benutzen Sie das Kommando `nsupdate`.

Dynamic Host Configuration Protocol (DHCP)

Schauen Sie sich die Adresszuweisung per DHCP mit Wireshark an. Starten Sie den Sniffer und führen Sie das Skript dhcp_on auf *apfelmus* aus. Dieses ändert die Konfiguration des Linux-Rechners von einer statischen auf eine dynamische IP-Adresse und startet das Netzwerkinterface neu.
Beschreiben Sie die DHCP-Anfragen und DHCP-Antworten. Welche Bedeutung haben die verschickten Pakete? Welche Informationen werden übertragen? Warten Sie eine Zeit lang und schauen Sie welche weiteren DHCP-Pakete verschickt werden. Welchen Sinn haben diese Pakete?

PC (Diskless Client) per DHCP/TFTP booten

Starten Sie nun Wireshark auf *apfelmus* und booten Sie *kaiserschmarrn* per DHCP/TFTP.
Wie unterscheiden sich die DHCP-Antworten von den vorherigen? Was passiert nachdem dem Rechner eine IP-Adresse zugewiesen wurde?
Welches Problem ergibt sich für TFTP durch die Verwendung von UDP? Wie wird dieses Problem gelöst?
Welche Dateien werden per TFTP übertragen? Wofür werden diese Dateien benötigt? Warum wird nach dem Übertragen der letzten Datei erneut eine DHCP-Anfrage gestartet? Was passiert danach?

Welche Rolle spielen die beteiligten Protokolle?

Versuch 3:
Höhere Protokolle / Dialogbasierte Protokolle

Literatur zur Einarbeitung

- Andrew S. Tanenbaum: Computernetzwerke
- RFC0172 / RFC0265: FTP – File Transfer Protocol
- RFC0821 / RFC2821: SMTP – Simple Mail Transfer Protocol
- RFC0793: TCP – Transmission Control Protocol
- RFC1738: URL – Uniform Resource Locators
- RFC1945: HTTP/1.0 – Hypertext Transfer Protocol
- RFC2068 / RFC2616: HTTP/1.1 – Hypertext Transfer Protocol
- RFC2396: URI – Uniform Resource Identifiers
- Dave Raggett, et al.: HTML 4.01 Specification, W3C Recommendation 24 December 1999 (http://www.w3.org/TR/html401/)
- Stefan Münz: SELFHTML, HTML-Dateien selbst erstellen (http://de.selfhtml.org/)

Vorbereitungsfragen

1. Wie sieht der Verbindungsaufbau, die Datenübertragung und der Verbindungsabbau per TCP aus?
2. Welche Bedeutung haben die Portnummern bei TCP und UDP?
3. Welche Informationen stecken in einer URL bzw. einem URI?
4. Welche Funktionalität bietet HTTP? Wie teilt der Webserver dem Client mit welcher Datentyp (Text, Grafik,...) übertragen wird?
5. Wie wird es technisch realisiert, dass auf einem Webserver (mit einer IP-Adresse) mehrere Domains gehostet werden können?
6. Welche Schwierigkeit ergibt sich im Zusammenhang von TLS und virtuellem Hosting?
7. Welche Syntax hat HTTP für das Abrufen einer Webseite?
8. Wofür wird das SMTP-Protokoll verwendet?
9. Wie kann es passieren, dass man eine Mail bekommt und im To-Feld die eigene Adresse nicht aufgelistet ist? Warum wird diese Mail trotzdem zugestellt?

10. Was versteht man unter dem SMTP-after-POP-Verfahren? Welcher Nachteil des SMTP-Protokolls wird damit umgangen?

11. Wie geschieht die Dateiübertragung bei FTP? Was bedeuten die Datenübertragungsmodi ASCII und Binary? Was versteht man unter Active FTP und Passive FTP? Welche Probleme können dabei bei FTP und NAT auftreten?

12. Welche Ports werden für HTTP, SSH, SMTP, IMAP, POP3 und FTP verwendet?

Versuchsaufbau

Dieser Versuch besteht aus zwei Rechnern (*apfelsaft* und *schorle*), die beide an einen Hub angeschlossen sind, um jeweils den Traffic des anderen Rechners mitsniffen zu können.

Durchführung

Abrufen einer Homepage „von Hand“

Das Tool `telnet` kann eine TCP-Verbindung zu einem Rechner auf einen bestimmten Port aufbauen. Stellen Sie eine Verbindung zu www.uni-ulm.de her und rufen Sie die Datei home.html „von Hand“ ab. Wie sieht die Statusantwort und Übertragung des Dokuments aus?

Sniffen einer Browser-Sitzung

Sniffen Sie eine Browser-Sitzung. Welche Anfragen stellt der Client an den Server? Welche Daten werden mit der Anfrage des Clients/Antwort des Servers sonst noch übertragen (User-Agent, Accept-Charset, Content-Type, ...)? Wie geschieht die Integration von Texten, Bildern, Videos, Dateien, ...?

Sniffen einer SSH-Sitzung

Stellen Sie nun eine SSH-Verbindung zu einem anderen Praktikumsrechner her (`ssh praktikum@10.x.y.z`; Passwort: praktikum) und sniffen Sie diese Sitzung mit. In welche Teile lässt sich die eigentliche SSH-Verbindung aufteilen? Welche Informationen sind sichtbar? Warum ist eine SSH-Sitzung einer Telnet-Sitzung vorzuziehen? Rufen Sie zum Vergleich (`telnet schorle`; Benutzername: praktikum; Passwort: praktikum) auf.

Simple Mail Transfer Protocol (SMTP)

Das SMTP-Protokoll ist vollständig textbasiert. Das hat den großen Vorteil, dass man eine SMTP-Sitzung auch selbst „von Hand“ mit Hilfe des Telnet-Programms durchführen kann. Dadurch lässt sich viel über SMTP lernen – unter anderem auch, wie Spammer falsche Absender-Adressen erzeugen.
Stellen Sie nun mit `telnet` eine Verbindung zum Mailserver der Uni (mail.uni-ulm.de) her und schicken Sie sich selbst eine Mail.

Geben Sie dazu folgendes ein:
Achten Sie bitte darauf, dass nach MAIL FROM: und RCPT TO: eine Ihrer eigenen E-Mail-Adressen steht.

```
HELO Hostname des Senders
MAIL FROM:<Absenderadresse>
RCPT TO:<Empfängeradresse>
DATA
Date: Datum
From: Absenderadresse
To: Empfängeradresse
Subject: Testmail
Hallo,
wie gehts?
.
QUIT
```

Welche Informationen sendet der Mail-Server nach dem Senden der einzelnen Befehle? Schauen Sie sich nun die soeben verschickte Mail (einschließlich Header) mit einem Mail-Client an und vergleichen Sie diese mit einer anderen Mail. Wiederholen Sie das Ganze nochmal und versuchen Sie, die Bedeutung der einzelnen Felder herauszubekommen. Wie reagiert der Mail-Server bei der Vorgabe eines falschen Hostnamens? Was bedeutet das für einen Spammer?

Analyse einer FTP-Sitzung

Starten Sie Wireshark und stellen Sie eine FTP-Verbindung zum FTP-Server der Uni Ulm her (`ftp ftp.uni-ulm.de`). Verwenden Sie als Benutzername „anonymous“ und geben Sie als Passwort Ihre E-Mail-Adresse ein. Mit dem Kommando `help` werden Ihnen sämliche FTP-Befehle angezeigt.
Versuchen Sie sich zunächst auf dem FTP-Server zurechtzufinden. Laden Sie nun eine beliebige Datei (z. B. eine README aus dem Verzeichnis /pub/mirrors/) herunter. Versuchen Sie das von Ihnen eingegebene Passwort zu finden. Wie läuft die Datenübertragung per FTP ab? Wann werden zusätzliche Verbindungen hergestellt?

Versuch 4: VoIP-/Videokonferenz-System: Setup und Protokollanalyse

Literatur zur Einarbeitung

- Andrew S. Tanenbaum: Computernetzwerke
- RFC0768: UDP – User Datagram Protocol
- RFC1889: RTP – A Transport Protocol for Real-Time Applications
- RFC2326: RTSP – Real Time Streaming Protocol
- RFC3261: SIP – Session Initiation Protocol
- http://www.cs.columbia.edu/~hgs/rtsp/faq.html: Der Zusammenhang zwischen H.323, RTP und RTSP wird hier im Ansatz erklärt.
- http://www.iec.org/online/tutorials/h323/: Auf dieser Webseite gibt es eine ausführliche Beschreibung zu H.323.
- Homepage von Asterisk, Software für eine private Vermittlungsstelle (PBX – Private Branch Exchange)
  (http://www.asterisk.org/)
- Dokumentation zum OpenH323 Gatekeeper
  (http://www.gnugk.org/)

Vorbereitungsfragen

1. Warum stellen Videokonferenzen und IP-Telefonie besonders hohe Anforderungen an das Netzwerk? Wie unterscheiden sich die Anforderungen von den „herkömmlichen“ Anforderungen (WWW, FTP und Videostreaming, ...)?
2. Wie groß dürfen die Verzögerungen bei der Datenübertragung höchstens sein, um beim Telefonieren nicht als störend empfunden zu werden?
3. Worin liegt für eine Videokonferenz bzw. die IP-Telefonie der Vorteil von UDP gegenüber TCP? Für welchen Teil der Videokonferenz wird allerdings TCP verwendet?
4. Welche unterschiedlichen Anwendungsbeispiele haben RTSP, H.323 und SIP? Welche Rolle spielt dabei RTP?
5. Wie hängen UDP, RTP, RTCP (RTP Control Protocol) und RTSP zusammen?
6. Wofür wird bei der IP-Telefonie Q.931 verwendet?
7. Welche weiteren Signalisierungsprotokolle kennen Sie?

8. Welche Aufgaben erfüllen Gatekeeper und Gateway bei VoIP?
9. Wie unterscheiden sich VoIP-Verbindungen mit bzw. ohne Gatekeeper?
10. Warum wird der IP-Telefonie eine große Zukunft vorausgesagt? Warum wird sie noch nicht flächendeckend eingesetzt?

Versuchsaufbau

Der Versuch besteht aus zwei Rechnern (*videotalk* und *videoshow*), an denen jeweils eine Kamera angeschlossen ist, einem H.323-Telefon (tiptel innovaphone 200) und einem SIP-Telefon (Siemens optiPoint 400 Std). Auf *videomax* läuft die PBX-Software Asterisk, deren Kommandointerface mit `asterisk -r` aufgerufen werden kann, und ein GNU Gatekeeper, dessen Statusinformationen über den Port 7000 abgefragt werden können.

Durchführung

Aufbau der Videokonferenzverbindung

Bauen Sie mit der Videokonferenzsoftware `Ekiga` eine Verbindung von einem Rechner zum anderen auf. Stellen Sie davor sicher, dass die Rechner **nicht** am Gatekeeper angemeldet sind. Zeichnen Sie mit Wireshark alle Pakete auf, bis die Verbindung zwischen den beiden Rechner hergestellt ist. Beenden Sie die Verbindung nach einiger Zeit wieder.

Erläutern Sie die Funktion der mitprotokollierten Datenpakete. Wie läuft die Anrufsignalisierung mit Q.931 ab?

Anmelden der Endgeräte am Gatekeeper

Melden Sie die beiden Rechner und das H.323-Telefon am Gatekeeper an. Protokollieren Sie mit Wireshark sämtliche Datenpakete bei der Registrierung und vom Aufbau bis zum Abbau einer Verbindung und beobachten Sie mit Hilfe des Statusport den Gatekeeper

Was für Datenpakete werden verschickt? Welche Bedeutung haben sie? Welche Informationen müssen übermittelt werden?

Telefonieren mit VoIP

Stellen Sie hintereinander in unterschiedlichen Kombinationen Verbindungen zwischen H.323- und SIP-Endgeräten her. Verwenden Sie Wireshark, das Asterisk Kommandointerface und den Gatekeeper Statusport, um die Abläufe zu protokollieren und zu analysieren.

Erklären Sie auch hier die Bedeutung der einzelnen Pakete. Welche Unterschiede sehen Sie zwischen H.323 und SIP?

Versuch 5:
Videostreaming: Setup mit Multicast/Unicast

Literatur zur Einarbeitung

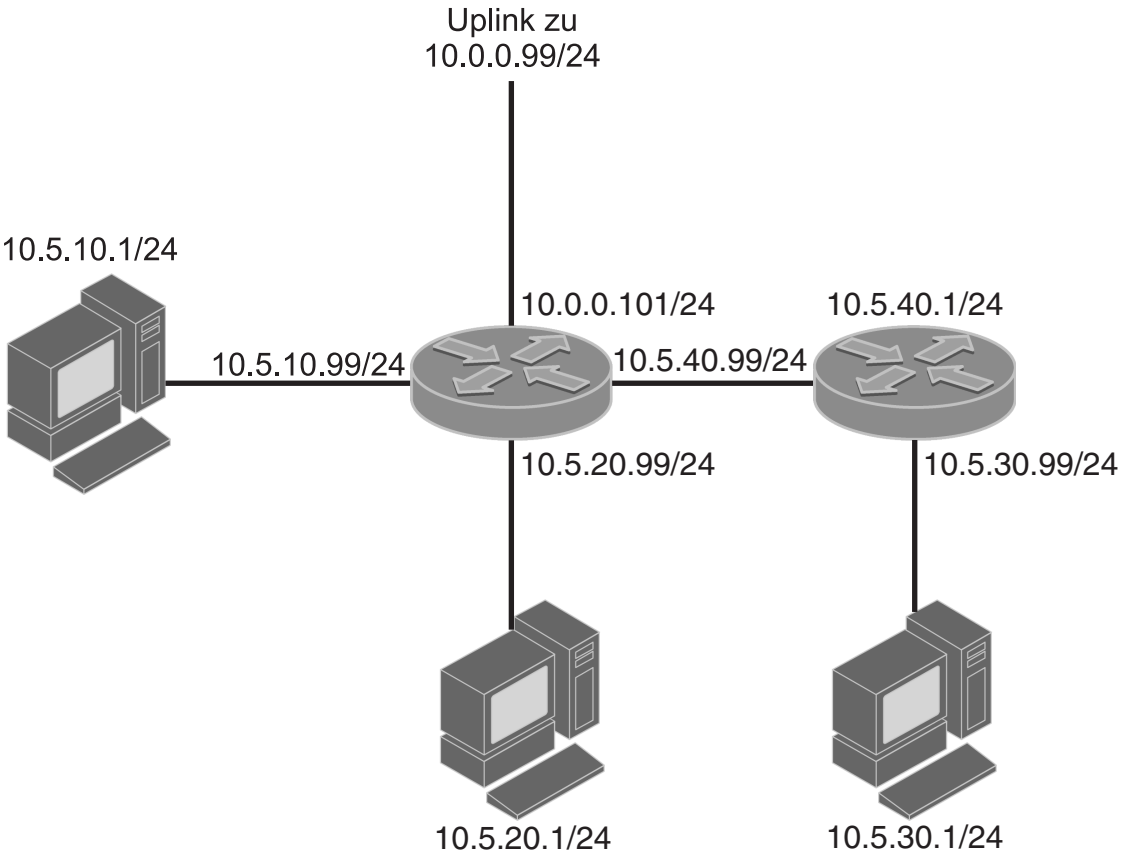
- Andrew S. Tanenbaum: Computernetzwerke
- RFC0966: Host Groups: A Multicast Extension to the Internet Protocol
- RFC1075: DVMRP – Distance Vector Multicast Routing Protocol
- RFC1112: Host Extensions for IP Multicasting
- RFC2117 / RFC2362: PIM-SM – Protocol Independent Multicast - Sparse Mode
- RFC2236: Internet Group Management Protocol, Version 2
- http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/ipmulti.htm
- Dokumentation zur Streaming-Software VLC
  (http://www.videolan.org/)
- Dokumentation zur Routing-Software
  MRT – Multi-threaded Routing Toolkit
  (http://sourceforge.net/projects/mrt/ und http://packages.debian.org/etch/mrt)
  und
  XORP – eXtensible Open Router Platform
  (http://www.xorp.org/)

Vorbereitungsfragen

1. Was versteht man allgemein unter Routing?
2. Was versteht man unter Unicast, Multicast und Broadcast?
3. Was sind Unterschiede, Vor- und Nachteile von Unicast und Broadcast bei der Verteilung von Multimediadaten?
4. Wie können die Nachteile vermieden werden und welche Vorteile bringt eine Multicast-Übertragung?
5. Welcher Adressbereich ist für Multicast vorgesehen? Warum werden weitere IP-Adressen benötigt, wenn doch ohnehin schon jeder beteiligte Rechner eine eigene IP-Adresse hat?
6. Welche besondere Bedeutung hat die TTL (Time to Live) bei einer Multicast-Übertragung?
7. An welche MAC-Adresse werden Multicast-Pakete adressiert? Erklären Sie das Zustandekommen der MAC-Adresse.

8. Wozu dient IGMP?
9. DVMRP basiert auf Reverse-Path-Forwarding. Erklären Sie kurz wie dabei ein Router den Verteilbaum bestimmt. Welchen Nachteil hat dieses Verfahren?
10. Erklären Sie kurz den Unterschied zwischen PIM Sparse Mode und PIM Dense Mode.
11. Welche Probleme ergeben sich bei Reliable Multicast?
12. Warum bieten viele ISP's (Internet Service Provider, z. B. T-Online) Multicast nicht an oder verbieten gar die Nutzung, obwohl es doch eigentlich Bandbreite spart?
13. Wie funktioniert Multicast-Routing? Benutzen Sie das Hilfsblatt für Skizzen.

Versuchsaufbau



Jeder der drei Arbeitsplätze liegt in einem eigenen Subnetz. Jeder Rechner benutzt den entsprechenden Router-Anschluss mit der IP 10.5.x.99 als Gateway. Der Router verbindet die drei Subnetze und stellt über einen Uplink zum Masquerading-Server 10.0.0.99 die Verbindung ins Internet her. Dieser Server verwendet für alle Pakete an die IP-Adressen 10.5.x.y die Adresse 10.0.0.101 des Routers als Gateway.

In diesem Versuch wird PIM Dense Mode und PIM Sparse Mode zum Einsatz kommen.

Durchführung

Überprüfen Sie die Konfigurationsdatei *mrtd.conf* der Routing-Software und starten Sie diese anschließend mit `mrtd`.

Internet Group Management Protocol (IGMP)

Sniffen Sie an einem der Rechner mit und schauen Sie sich an welche IGMP-Pakete verschickt werden. Wofür sind die Pakete an die Adresse 224.0.0.1? Benötigen Sie den Promiscuous Mode, um diese Pakete zu erhalten?

Unicast-Übertragung

Bauen Sie eine Unicast-Übertragung mit der Streaming-Software VLC auf. Verwenden Sie *ping* als Sender und *timestamp* als Empfänger. Was geschieht im Netz von *checksum*? Versuchen Sie die Übertragung auch durch *checksum* zu empfangen. Beobachten Sie zusätzlich den Verkehr in den Subnetzen von *timestamp* und *ping* vor und während der Übertragung.

Multicast-Übertragung

Bauen Sie von *ping* eine Multicast-Übertragung auf, verwenden Sie zur Konfiguration den Streaming-Assistent von VLC. Beobachten Sie den Netzwerkverkehr an *ping*, *checksum* und *timestamp*. Welcher Verkehr entsteht durch die Übertragung? Protokollieren Sie alle Pakete im Zusammenhang der Übertragung und versuchen Sie herauszufinden was sie bedeuten.

Beginnen Sie, den Stream auf einem Rechner zu empfangen. Was kann beobachtet werden?
Empfangen Sie den Stream auf dem zweiten Rechner. Protokollieren Sie, welche Pakete dazu verschickt werden und wie der Rechner der Gruppe beitritt (*). Welche Auswirkungen hat das auf das Netz des Senders?

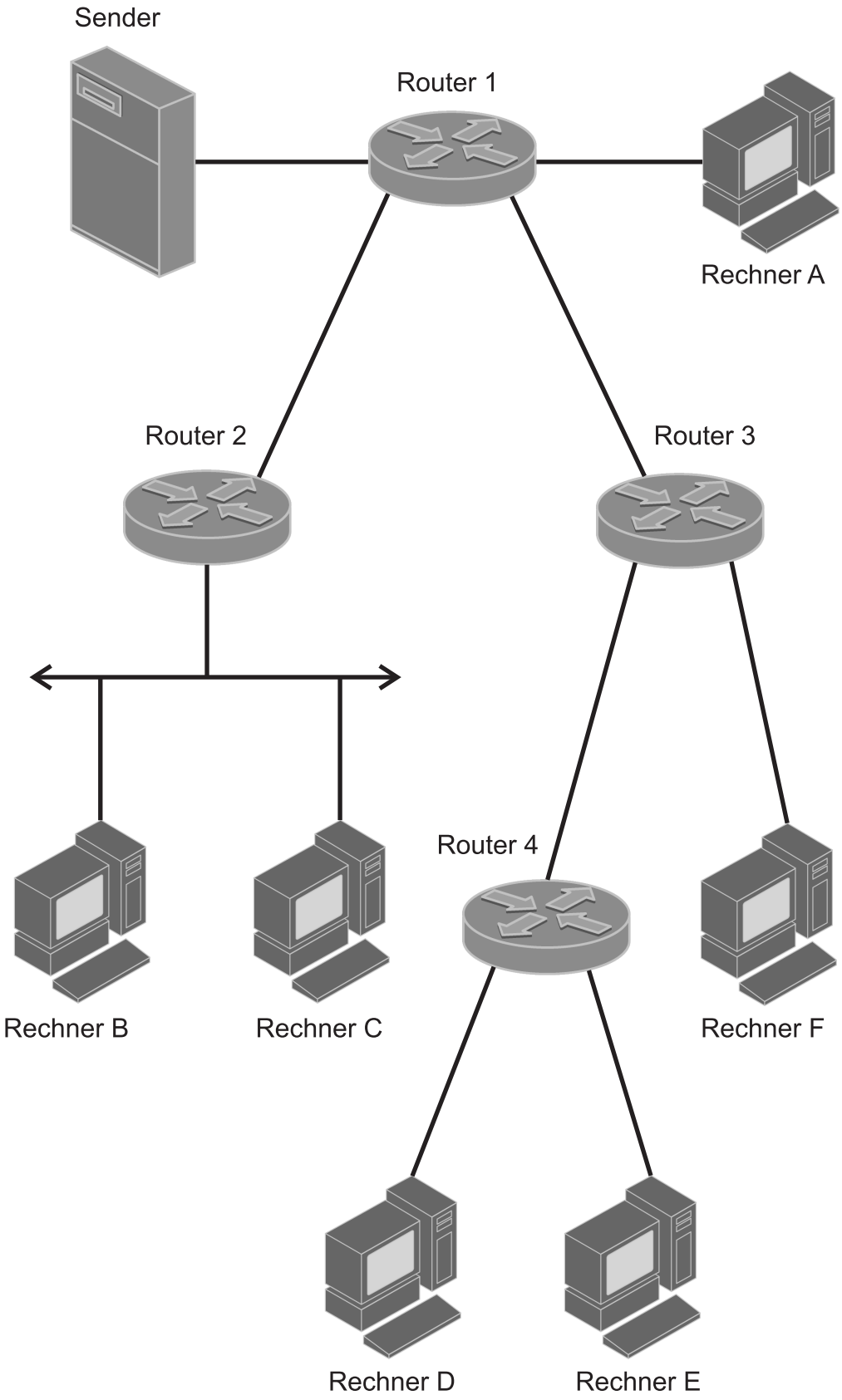
Beenden Sie den Empfang auf einem Rechner. Wie wird die Übertragung beendet?

Video On Demand

Versuchen Sie mit VLC Video On Demand zu realisieren. Gehen Sie dazu vor wie auf http://tldp.org/HOWTO/VideoLAN-HOWTO/vod.html beschrieben.
Hinweis: Auf *ping* läuft bereits ein Webserver (Web-Verzeichnis: /var/www/html/).
Klicken Sie nach einer Weile auf PAUSE und schauen Sie sich an wie der Empfänger dies dem Sender mitteilt.

Hinweis zur Auswertung: Die Auswertung sollte Grundlagen zu IP-Multicast erläutern und kurze Ergebnisse der Versuche enthalten. Schwerpunkt der Auswertung soll der mit (*) markierte Versuch sein.

Hilfsblatt zur Vorbereitungsfrage 13



Versuch 6:
Quality of Service (QoS), Traffic Engineering

Literatur zur Einarbeitung

- Andrew S. Tanenbaum: Computernetzwerke
- RFC0791: IP – Internet Protocol
- RFC1349: Type of Service in the Internet Protocol Suite
- RFC2212: Specification of Guaranteed Quality of Service
- RFC2474: Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers
- Manual-Pages: tc, tc-cbq, tc-htb
- Linux Advanced Routing & Traffic Control HOWTO, Chapter 9 (http://lartc.org/howto/)
- Dokumentation zur Software Iperf zum Erzeugen/Messen von Netzwerktraffic (http://dast.nlanr.net/Projects/Iperf/)

Hinweis für die Vorbereitung

Für die Durchführung dieses Versuchs ist es dringend erforderlich, das Linux Advanced Routing & Traffic Control HOWTO, Chapter 9 gelesen zu haben!

Vorbereitungsfragen

1. Was versteht man unter „Quality of Service“?
2. Nach welchen Gesichtspunkten kann man die verschiedenen Services in Serviceklassen (CoS – Classes of Service) aufteilen, um sie dann unterschiedlich behandeln zu können?
3. Welche speziellen Anforderungen, die manche Anwendungen an das Netzwerk stellen, versucht man mit QoS zu erfüllen?
4. Erklären Sie kurz die folgenden grundlegenden Verfahren: Admission Control, Traffic Shaping, Preferential Queuing, Selective Forwarding und TCP-Mechanismen (z. B. RED).
5. Was versteht man unter classless Queueing Disciplines? Nennen Sie Beispiele und erläutern Sie diese.
6. Was unterscheidet die classful Queueing Disciplines von den klassenlosen QDiscs? Nennen Sie auch hier Beispiele und erläutern Sie diese.
7. Erklären Sie kurz die beiden Konzepte DiffServ und IntServ.

8. Im Folgenden sehen Sie eine QoS-Konfiguration bei der das Tool `tc` verwendt wird. Versuchen Sie zunächst zu verstehen, welcher Traffic hier bevorzugt werden soll und zeichnen Sie dann die dazugehörige Baumstruktur.
   Welche Schwierigkeiten erkennen Sie bei der Konfiguration von CBQ?

```
tc qdisc add dev eth0 root handle 1: cbq bandwidth 10mbit avpkt 1000 cell 8

tc class add dev eth0 parent 1: classid 1:1 cbq bandwidth 10mbit                          \
   rate 8mbit weight 0.8mbit prio 8 allot 1514 cell 8 maxburst 20 avpkt 1000 bounded

tc class add dev eth0 parent 1:1 classid 1:3 cbq bandwidth 10mbit                         \
   rate 5mbit weight 0.5mbit prio 4 allot 1514 cell 8 maxburst 20 avpkt 1000
tc class add dev eth0 parent 1:1 classid 1:4 cbq bandwidth 10mbit                         \
   rate 3mbit weight 0.3mbit prio 2 allot 1514 cell 8 maxburst 20 avpkt 1000
tc class add dev eth0 parent 1:1 classid 1:5 cbq bandwidth 10mbit                         \
   rate 1kbit weight 0.1kbit prio 8 allot 1514 cell 8 maxburst 20 avpkt 1000

tc qdisc add dev eth0 parent 1:3 handle 30: sfq perturb 10
tc qdisc add dev eth0 parent 1:4 handle 40: sfq perturb 10
tc qdisc add dev eth0 parent 1:5 handle 50: sfq perturb 10

tc filter add dev eth0 parent 1:0 protocol ip prio 2 u32 match ip dport 1234 0xffff flowid 1:3
tc filter add dev eth0 parent 1:0 protocol ip prio 1 u32 match ip tos 0x08 0xff flowid 1:4
tc filter add dev eth0 parent 1:0 protocol ip prio 3 u32 match ip src 0.0.0.0/0 flowid 1:5
```

9. Wie löst die Uni Ulm das QoS-Problem?

Versuchsaufbau

Dieser Versuch besteht aus einem Linux-Router (*routeqos*) an den neben dem Uplink noch zwei weitere Subnetze angeschlossen sind. In einem Subnetz befindet sich ein Rechner (*parameter*) der als Videosteaming-Server verwendet werden soll. Das andere Subnetz (mit den Rechnern *address* und *control*) ist über einen 10Mbps-Hub angeschlossen. Dieser Hub soll das „Nadelöhr“ darstellen, um dann in einer Überlastsituation QoS zeigen zu können. Als FTP-Server kann mit einem beliebigen FTP-Client zum Uni Ulm FTP-Server (`ftp.uni-ulm.de`) verbunden werden.

Durchführung

TOS-Feld / DS-Feld

Sniffen Sie eine Webbrowser-Sitzung, eine SSH-Sitzung und einen FTP-File-Transfer. Schauen Sie sich nun das DS-Feld bei den drei Übertragungen an und interpretieren Sie das Ergebnis.

QoS mit CBQ

Im Folgenden sollen vier Fälle untersucht werden:

1. ohne QoS und ohne Netzüberlastung
2. ohne QoS und mit Netzüberlastung
3. mit QoS und mit Netzüberlastung

4. mit QoS und ohne Netzüberlastung

Schauen Sie sich zunächst den Idealfall (1) an. Starten Sie dazu auf *address* einen FTP-Download und interpretieren Sie die erreichte Übertragungsrate.
Starten Sie nun mit `gvlc` auf *parameter* den Videostreaming-Server, der einen UDP-Videostream an *control* senden soll. Mit dem gleichen Tool können Sie sich den Videostream an *control* auch anschauen. Ein Flood-Ping von *parameter* an *control* und *address* soll zusätzlich Traffic erzeugen. Verwenden Sie dazu den Befehl `ping -f -s 7000` *`IP-Adresse`*.
Die Überlastsituation (2) sollten Sie nun an den Störungen im Videostream und an der sinkenden Datenrate des Downloads erkennen.

Starten Sie nun auf *routeqos* das Skript, dass CBQ einrichtet (`./tc_cbq start`). Beobachten Sie das Verhalten des Downloads und die Qualität des Videostreams (3). Mit `./tc_cbq info` können Sie sich die Statistiken der einzelnen Klassen anschauen.

Stoppen Sie nun den Videostream und die Flood-Pings und überprüfen Sie anhand der FTP-Übertragungsrate, ob das eingerichtete QoS zu Performance-Einbußen führt (4) und vergleichen Sie mit dem Idealfall. Wenn Sie Zeit sparen wollen, starten Sie den Download neu.

QoS mit dem HTB

Konfigurieren Sie nun den HTB so, dass er den Traffic genauso behandelt wie die obige Konfiguration von CBQ. Gehen Sie von der Baumstruktur aus und richten Sie die QDiscs, Klassen und Filter ein. Schreiben Sie am Besten ein kleines Skript das die Befehle nacheinander ausführt.
Gehen Sie beim Testen Ihrer Konfiguration genauso vor wie bei CBQ.

Bandbreitenmessung mit Iperf

Zum Schluss soll die Aufteilung der Bandbreite auf die Klassen gemessen werden. Achten Sie darauf, dass die Messungen, die Sie mit Iperf durchführen mindesten 60 Sekunden dauern damit die Reaktionsgeschwindigkeit des HTB nicht zu allzu falschen Ergebnissen führt.
Bestimmen Sie zuerst die maximale Bandbreite der einzelnen Klassen und vergleichen Sie diese mit der konfigurierten Bandbreite. Führen Sie dann zwei Messungen gleichzeitig durch und setzen Sie bei der einen das TOS-Feld auf Bulk-Traffic und bei der anderen den Server-Port auf 1234. Wie wird die Bandbreite aufgeteilt? Führen Sie als letztes noch eine Messung mit Traffic aller drei Klassen durch. Wie wird die Bandbreite jetzt aufgeteilt?

Versuch 7:
IPv6 Tunneling und Adressvergabe

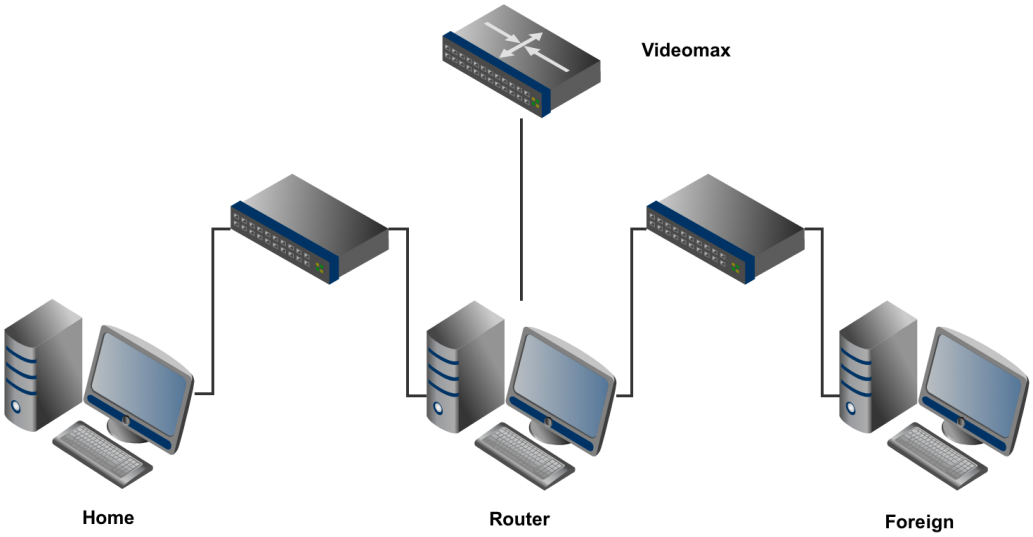
Literatur zur Einarbeitung

- Andrew S. Tanenbaum: Computernetzwerke
- RFC0791: IP – Internet Protocol
- RFC0792: ICMP – Internet Control Message Protocol
- RFC2460: Internet Protocol, Version 6 (IPv6) Specification
- RFC4861: Neighbor Discovery for IP version 6 (IPv6)
- RFC4862: IPv6 Stateless Address Autoconfiguration
- RFC4291: IP Version 6 Addressing Architecture
- SixXS, ein IPv6 Tunnelbroker
  (http://www.sixxs.net/main/)

Vorbereitungsfragen

1. Warum IPv6 welchen Problemen mit v4 wird damit begegnet?
2. Welche Unterschiede, Vor- und Nachteile gibt es zwischen IPv4 und IPv6?
3. Wie wird die Fragementierung bei IPv6 vorgenommen?
4. Vergleichen Sie den IPv6 Header mit dem IPv4 Header was hat sich verändert?
5. Man sagt es gäbe bei IPv6 mehr mögliche Adressen als bei dem Vorgänger. Wieviele Adressen kann man denn jedem mm^2 der Erdoberfläche zuweisen?
6. Was ändert sich in Bezug auf die Fragmentierung?
7. Mit welchen Methoden können IPv6 Adressen zugewiesen werden?
8. Welche Adressräume sind bei IPv6 bereits definiert?
9. Es gibt spezielle vordefinierte Multicastadressen bei IPv6, nennen Sie ein paar Beispiele.
10. Wie lassen sich IPv6 Pakete über ein IPv4 Netzwerk übertragen? Welche Protokolle existieren dafür? Welche Nachteile treten dabei auf?
11. Wie sieht das Paketlayout eines IPv6 Paketes aus, das per AYIYA in einem IPv4 Netz übertragen wird?
12. Wie wird ICMP, ARP und DHCP bei IPv6 realisiert?
13. Wie funktioniert DNS und Reverse DNS bei IPv6?

Versuchsaufbau



Durchführung

IP - Konfiguration

Zu Beginn der Versuche benötigen Sie nur den Rechner Router. Schauen Sie sich zum Anfang erst einmal die aktuelle Netzwerkkonfiguration des Rechners an. Dazu gehören zum Einen die Interfaces samt vergebenen IP-Adressen, zum Anderen die Routingtabelle.
Editieren Sie nun die Konfigurationsdatei **/etc/networking/interfaces** so, dass die Netzwerkkarten **eth1** und **eth2** folgende IPv6 Netzwerke verwenden:

XX Aktueller Tag

YY Aktueller Monat

Prefix 2a01:1e8:e104

eth1 Prefix:XX::1/64

eth2 Prefix:YYXX::1/64

anschließend starten Sie die Netzwerkdienste mit **/etc/init.d/networking restart** neu. Verifizieren Sie, mit den gleichen Befehlen wie oben, dass die Netzwerkparameter richtig gesetzt wurden.

IPv6 Tunneling

Eine der vielen Möglichkeiten zu testen ob man IPv6 Konnektivität besitzt bietet die Internetseite **www.kame.net**. Gehen Sie einmal mit einem Browser auf die Webseite und bewundern Sie die Schildkröte. Solange die Schildkröte nicht tanzt greift man per IPv4 auf die Seite zu.

Um für diesen Versuch ein IPv6 Netzwerk bereitstellen zu können verwenden wir den Tunnelanbieter **sixxs**. Dieser stellt ein Tool bereit, das unter Debian einen Tunnel zum nächstgelegenen Endpunkt aufbaut. Starten Sie den Dienst mit **/etc/init.d/aiccu start** . Welches Interface ist jetzt neu hinzugekommen und was hat sich an der IPv6 Routingtabelle geändert?
Um den funktionierenden v6 Zugriff zu verifizieren, rufen Sie noch einmal die Seite mit der Schildkröte auf.

Wie bereits geschrieben wird für den Zugang ein IP-Tunnel verwendet. Um diesen nachzuvollziehen führen Sie einen **ping** (bzw. **ping6**) auf www.heise.de aus. Was fällt auf in Bezug auf Antwortzeiten?

Um sich anzuschauen wie das Tunneling genau funktioniert eignet sich einmal mehr Wireshark. Starten Sie das Programm und schneiden Sie einen **ping6** oder ein **traceroute6** auf dem Interface **eth0** mit. Wie groß (in byte) ist der Overhead durch das Tunneling und welche Header tragen wieviel dazu bei? Welche IPv4 Adresse hat der Tunnelendpunkt und wie lange ist die Pingzeit zu diesem. Deckt sich diese mit den Erkenntnissen aus dem traceroute6?

(Aufgabe für Protokoll: Mit Googlemaps und der Routenplanerfunktion einmal die unterschiedlichen Wege von ipv4 und ipv6 kenntlich machen und den Streckenabschnitt, der durch das Tunneling transparent für einen ip6 Client ist, hervorheben oder beschreiben)

Stateless Adressvergabe

Als nächstes soll der Rechner **home** per IPv6 angebunden werden. Die Adresszuweisung soll stateless erfolgen. Dazu gibt es den Route Advertisement Daemon (radvd). Editieren Sie dessen Konfigurationsdatei **/etc/radvd.conf** auf **router** und zwar so, dass als Netzwerk (siehe) Prefix:XX::/64 verwendet wird. Starten Sie anschließend den Dienst mit **/etc/init.d/radvd start**.
Der Mechanismus der stateless Adressvergabe soll jetzt analysiert werden. Deaktivieren Sie dazu zuerst auf **home** das Interface **eth0** (**ifdown eth0**). Anschließend starten Sie wireshark auf router und loggen auf eth1 den Netzwerkverkehr mit. Jetzt aktivieren Sie das Interface auf **home** wieder. Welche Pakete sind an der Adressvergabe beteiligt? Was passiert sonst noch auf dem Interface bis die erste Anfrage von der zugewiesenen Adresse kommt? Wie kommt die Adresse von **home** zustande?

IPv6 Netzbereiche

In den Vorbereitungsfragen haben Sie hoffentlich die verschiedenen Adressbereiche und Multicastgruppen aufgezählt. Diese sollen nun in der Praxis einmal getestet werden. Vom Rechner home aus führen Sie dazu einen ping6 an folgende Adressen aus:

- globale Adresse des Routers
- linklokale Adresse des Routers (warum funktioniert das nicht ohne Weiteres?)

- alle Rechner (multicast Gruppe)
- alle Router (multicast Gruppe)

Statefull Adressvergabe

Um nun auch noch den Rechner **foreign** an das IPv6 Netz anzubinden soll dieser per statefull Adressvergabe eine Adresse bekommen. Editieren Sie die Konfigurationsdatei /etc/wide-dhcpv6/... so, dass eine Adresse aus dem Netzwerk Prefix:YYXX::/64 vergeben wird. Starten Sie anschließend den Dienst mit **/etc/init.d/wide-dhcpv6 start**. Analysieren Sie den Vergabevorgang wie in . Das Interface für wireshark ist jetzt allerdings **eth2**.

IPv6 und DNS

Welcher dns server wird auf **foreign** verwendet? In welchem Paket des Wiresharkmitschnitts ist dieser Server zugewiesen worden? Testen Sie die Funktion des DNS Servers indem Sie zuerst die vergebene IP Adresse für **foreign** herausfinden und anschließend einen reverse DNS lookup für diese Adresse an dem Server durchführen.

Versuch 8:
Network Simulator Version 2 (NS-2)

Literatur zur Einarbeitung

- Website von NS (http://www.isi.edu/nsnam/ns/index.html)
- NSNAM Wiki (http://nsnam.isi.edu/nsnam/index.php/Main_Page)
- The ns Manual (formerly ns Notes and Documentation) (http://www.isi.edu/nsnam/ns/ns-documentation.html)
- The GNU Awk User's Guide (http://www.gnu.org/software/gawk/manual/html_node/)

Vorbereitungsfragen

1. Wozu dienen Simulationen?
2. Was ist NS Version 2 und worauf basiert es?
3. Was bedeuten die Schlüsselwörter *proc*, *set*, *puts* in OTcl? Wozu dient ein *$*- und ein *#*-Zeichen?
4. Was für eine Aufgabe hat der *Network Animator* (NAM)?
5. Wozu dient der Ausdruck:
   ```
   set nf [open out.nam w]
   $ns namtrace-all $nf
   ```
6. Was beinhaltet eine `.nam` Datei und was kann damit gemacht werden? Bitte gehen Sie hier genauer auf das Dateiformat ein!
7. Was ist ein *node*? Wozu dient ein *link*?
8. Was versteht man unter *queue-type* und *queue-size*?
9. Welche verschiedenen Arten von *queues* gibt es? Erklären Sie bitte auch genau die Eigenschaften der verschiedenen *queues*.
10. Beschreibe mittels OTcl ein Netzwerk Grundgerüst, das aus 2 Knoten besteht und folgende Eigenschaften hat:
    a) Duplex Leitung mit Bandbreite von 1,5 mbps, 12 ms Delay
    b) Queue Typ: RED
    c) Queue Größe: 15
11. Wie sieht eine TCP (UDP) Verbindung in OTcl aus?
12. Erstelle für das zuvor erstellte Netzwerk eine FTP- und CBR- Übertragung von Knoten 1 nach Knoten 2.

13. Wie ordnen Sie für den NAM den Datenströmen eine Farbe zu? Mit welchem Befehl können Sie eine Orientierung der Knoten für den NAM festlegen?

14. Wie sollte eine finish Methode aussehen, die den Netzwerk Animator (NAM) startet?

15. Was versteht man unter Packet Loss? Aus welchen Gründen kann Packet Loss auftreten?

    Kann Packet Loss auch ohne spezielle Mechanismen vermieden werden?

Versuchsaufbau

Dieser Versuch besteht aus einem Rechner, der als Terminal zu *videomax* dient, auf dem der *Network Simulator Version 2* (NS-2) läuft.

Durchführung

Test der Simulation aus den Vorbereitungsfragen

Übertragen Sie Ihre Simulation, die Sie für die Vorbereitungsfragen erstellt haben auf *videomax*. Die Informationen zum Account auf *videomax* erhalten Sie von Ihrem Tutor. Testen Sie Ihre Simulation und beschreiben Sie den Ablauf.

Erste Simulation mit NS-2 und NAM

In Ihrem Home-Verzeichnis auf *videomax* befindet sich im Ordner *Aufgabe1* die Datei *a1.tcl*. Führen Sie diese mit dem Befehl `ns a1.tcl` aus. Beschreiben Sie die einzelnen Schritte die NS-2 ausführt, sowie das Ergebnis des NAM. Was fällt bei der Animation im NAM insbesondere auf?
Welche TCP-Variante wird in dieser Simulation eingesetzt? Was sind deren besondere Eigenschaften?

Einfache Änderungen an einer Simulation

Führen Sie zuerst im Ordner *Aufgabe2* die Datei a2.tcl aus und beschreiben Sie den Vorgang.

Jetzt ändern die den Typ des *Buffers* in **RED**. Was bewirkt dies und welche Auswirkungen hat diese Art von Warteschlange auf den Datenfluss?

Anschließend führen Sie bitte einen *Buffer* der Größe 10 ein. Was hat sich dadurch geändert?

Zuletzt setzen Sie die *TCP window size* auf 10. Beschreiben Sie erneut die Änderungen.

Auswertungsmöglichkeiten anhand von trace-Dateien

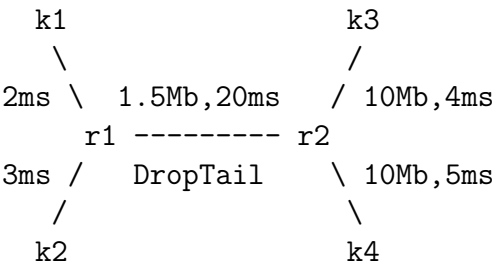
Mit Hilfe von zusätzlichen *trace*-Dateien können noch genauere Auswertungen durchgeführt werden. Erweitern Sie Ihr Skript aus Aufgabe 2 dahingehend, dass in einer weiteren *trace*-Datei, die zum Beispiel mit `out.tr` bezeichnet werden kann, alle Events der Simulation abgelegt werden.
Mit Hilfe von einfachen Skripten oder auch anderen Tools kann anhand der abgelegten Daten zum Beispiel die Anzahl der verlorenen Pakete (Packet Loss) und der Jitter der Pakete ausgerechnet werden. Dazu muss ein derartiges Skript die *trace*-Datei `out.tr` zeilenweise parsen und die entsprechenden Werte in Arrays ablegen. Viele NS-2 Benutzer verwenden zu diesem Zweck häufig die Skriptsprache **Awk** mit der sich sehr einfach Textdaten auswerten lassen. Um Ihnen eine Anhaltspunkt zu geben befindet sich im Verzeichnis *Aufgabe2* die *awk*-Datei *jitter.awk*. Mit diesem Skrip lässt sich der Jitter der FTP-Pakete aus Aufgabe2 berechnen. Auf videomax können Sie durch den Aufruf von *gawk* ein selbstgeschriebenes Skript starten.
Mit einem weiteren Skript *packloss.awk* lässt sich der Paketverlust der FTP-Verbindung messen. Bevor Sie die Simulation erneut starten setzten Sie die Window-Size der TCP-Verbindung auf Ihren Default-Wert zurück. Erhöhen Sie die Simulationszeit auf 60 Sekunden und messen Sie Jitter und Paketloss der FTP-Verbindung. Was können Sie aus den so gewonnenen Ergebnissen Schlussfolgern?

Grafische Auswertung einer Bandbreitenmessung einer Verbindung

In diesem Abschnitt soll die Simulation aus Aufgabe 2 um eine Bandbreitenmessung erweitert werden. Dazu kann die Member-Variable ***bytes_*** des Objekts *TCP-Sink* verwendet werden. Dieses zeigt an wieviele Bytes Ihr Ziel erreicht haben. Bei laufender Simulation kann damit die erreichte Übertragungsrate in Bit/s ausgerechnet werden. Werden die so ermittelten Werte in einer weitere Textdatei abgelegt, so lasssen sich diese Daten über einen Aufruf von zum Beispiel *xgraph* sehr einfach grafisch anzeigen. Ihre Aufgabe ist es nun eine Prozedur zu erstellen, die sich in regelmäßigen Abständen selbst ausführt und die aktuelle Bandbreite der FTP-Verbindung ausrechnet und in einer Datei (z.B. `ftp_a2.tr`) ablegt. In Ihrer *finish*-Prozedur kann das Tool *xgraph* ausgeführt und die gemessene Bandbreite grafisch am Bildschirm dargestellt werden. Was können Sie anhand dieser Ergebnisse erkennen?

Entwerfen einer eigenen Simulation

Sie haben folgende Netzwerkstruktur aus 4 Knoten und 2 Router gegeben:



Die Angaben beziehen sich auf Bandbreite und Delay. Bearbeiten Sie die folgenden Aufgaben. (Sie können den OTcl-Code aus Aufgabe 2 als Vorlage zur Bearbeitung dieser Punkte verwenden)

1. Erstellen Sie den OTcl-Quellcode der die Knoten des Netzwerks initialisiert und im weiteren alle Links wie oben beschrieben erzeugt.

2. Fügen Sie an den beiden Knoten k1 und k2 jeweils eine TCP Verbindung zu Knoten k3 hinzu. Diesen Agenten sollen per FTP Datenverkehr jeweils eine 500MB große Datei an k3 übertragen (Startzeit 0.5 bzw. 0.8). Es ist ratsam den Verbindungen verschiedene Farben zuzuordnen.

3. Lassen Sie einen *nam-trace* erstellen und verwenden Sie den NAM um ihn anzuzeigen.

4. Fügen Sie einen Buffer von 10 zwischen der Verbindung r1 und r2 ein.

5. Ergänzen Sie Ihre Simulation um einen Videostream von Knoten k1 nach k4. Dazu können Sie annehmen, dass ein Stream dieser Art die Eigenschaften von *CBR Traffic* über eine UDP-Verbindung hat. Setzten Sie dabei die folgenden Werte `packet_size_ 1000` und `rate_ 1mb` vom CBR Traffic.

6. Zwischen den Knoten k2 und k3 und zwischen k2 und k4 soll noch weiterer UDP-Traffic auftreten der weiteren irrelevanten Datenverkehr darstellen soll. Konfigurieren Sie die UDP-Agenten mit folgenden Einstellungen: CBR-Traffic, `packet_size_ 367` und `rate_ 0.35mb` sowie `packet_size_ 430` und `rate_ 0.56mb` Wiederholen Sie Schritt 3. Was können Sie beobachten?

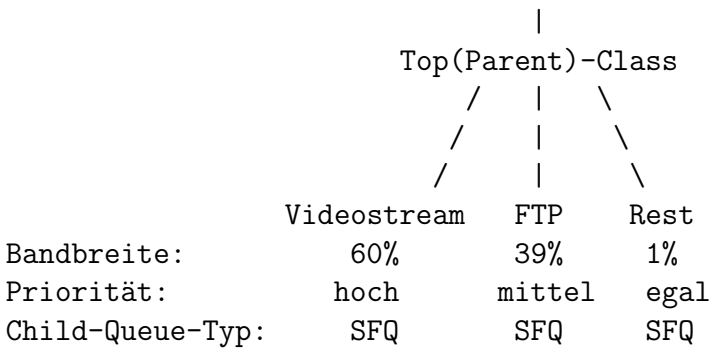
7. Messen bzw. errechnen Sie mit Ihren zuvor erstellten Skripten den Jitter und den PacketLoss des Videostreams. Ist die Qualität des Videostream für den Benutzer an Knoten k4 ausreichend? Was beobachten Sie bei der Übertragung der FTP-Downloads?

Erweiterung durch Quality of Service Verfahren

In den nächsten Aufgaben soll die Übertragungsqualität von bestimmten Verbindungen verbessert werden. Wir betrachten dazu das in der Aufgabe oben erstellte Netzwerk und das oben betrachtete Datenaufkommen. Dabei sollen die Pakete des Videostreams und der FTP-Downloads am Knoten r1 bevorzugt behandelt werden um die Qualität dieser Verbindungen zu verbessern. Dazu soll in dieser Aufgabe das Class-Based-Queuing (CBQ) Verfahren eingesetzt werden.
Kopieren Sie das oben erstellte Skript in eine neue Datei und bearbeiten Sie die folgenden Aufgaben:

1. Tauschen Sie die RED/DropTail Queue zwischen den Konten r1 und r2 durch eine CBQ-Queue (r1− r2) und eine DropTail-Queue (r2− r1).

2. Teilen Sie den Traffic in Knoten r1 in die folgende Traffic-Klassen ein.



Erstellen Sie den OTcl-Quellcode der diese Strukutr in der CBQ-Queue in Knoten r1 umsetzt. Dazu können Sie die Datei CBQ_Template.txt in ihrem Home-Verzeichnis auf videomax als Vorlage für diese Aufgabe verwenden. Testen Sie die Funktionsweise Ihrer Simulation durch grafische Analyse mit Hilfe des NAM.

3. Simulieren Sie dieses Szenario für eine Simulationszeit von einer Minute. Was sind die Beobachtungen? Was hat sich gegenüber der RED Queue geändert?

4. Messen Sie die Bandbreite, das Delay und den Jitter der Videostream-Pakete.

5. Wiederholen Sie die Bandbreitenmessung für die beiden FTP-Übertragungen zwischen den Knoten k1 und k3 sowie k2 und k4.