

9. Blatt: Network Protocols and Architectures WS 10/11

Aufgabe 1: (100 Punkte) *Protokol Implementierung: Gateway Control Protocol*

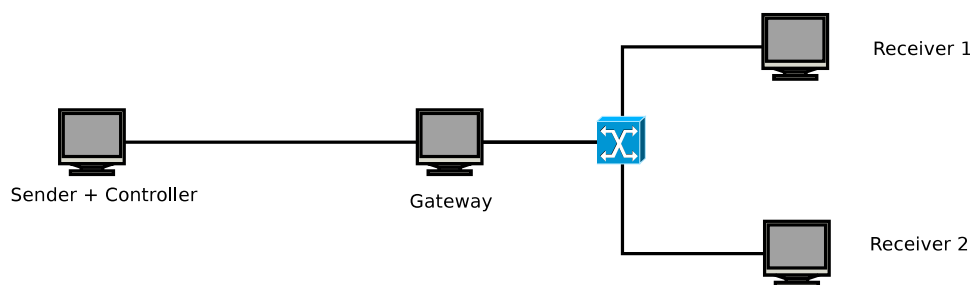


Abbildung 1: Experimental Setup

Dieses Übungsblatt beschäftigt sich mit der Implementierung des UDP basierten Gateway Control Protokolls (GCP), das im 7. Übungsblatt eingeführt wurde.

Der Gateway besitzt hierbei die folgende Funktionalität; (i) Nachrichten vom Typ *MSG* werden an den Zielhost weitergeleitet und (ii) das Weiterleiten von *MSG* Nachrichten durch den Gateway ist durch spezielle Kontrollnachrichten steuerbar. Eine Weiterleitung erfolgt dabei jedoch nur, wenn dies zuvor durch ein *OPEN* mit Angabe der IP und des Ports des Zielhosts auf dem Gateway freigegeben wurde. Falls eine Nachricht eintrifft und die Weiterleitung nicht zuvor gestattet wurde (*BLOCKED*), verwirft der Gateway diese einfach (*DROP*). Um die Implementation möglichst einfach zu halten, kann angenommen werden, dass die Rückrichtung (Receiver \rightarrow Sender) niemals blockiert und immer offen ist.

Es soll weiterhin angenommen werden, dass die Kommunikation über verlustfreie Kanäle erfolgt, so dass das Protokoll keine Neuübertragungen beachten muss. Wie in der obigen Abbildung angedeutet soll der Gateway Unterstützung für mehrere Sender und Empfänger bieten. Des Weiteren werden die Datennachrichten (*MSG*) sowohl die Adresse (IP+Port) des Empfängers, als auch des Absenders enthalten. Dies vereinfacht die Implementierung, da die Weiterleitung von Datennachrichten auf dem Gateway zustandslos erfolgen kann.

Eine Implementierung von Sender + Controller (sender.pl), sowie des Receivers (server.pl) werden zur Verfügung gestellt, so dass lediglich der Gateway zu implementieren ist. Hinweise zur Benutzung der Programme kann durch Verwendung des `--help` Parameters abgerufen werden. Hierbei implementiert sender.pl den Client, der sowohl Daten- als auch Kontrollnachrichten sendet. Ein einfacher Echoserver wird von server.pl bereitgestellt. Eingehende Datennachrichten werden hierbei an den Sender zurückgeschickt.

Das GCP ist UDP basiert und verwendet die folgenden sieben Nachrichtentypen, die als reiner ASCII-Text übertragen werden:

- **Data:** `MSG|DST_IP|DST_UDP_PORT|SRC_IP|SRC_UDP_PORT|Message text`

Dies ist eine Datennachricht, die *Message text* enthält. Die IP-Adresse des Senders beträgt hierbei *SRC_IP* und dessen Port *SRC_UDP_PORT*. In ähnlicher Weise wird der Empfänger im *DST*-Feld spezifiziert. Beachte, dass in der Rückrichtung die *DST* und *SRC* Felder getauscht werden.

Beispiel: `MSG|127.0.0.1|12880|127.0.0.1|12881|Foo bar`

- **Soft state control: SOPEN|DST_IP|DST_UDP_PORT|LENGTH**
Diese Kontrollnachricht öffnet einen bidirektionalen Kommunikationskanal zu dem in den *DST*-Feldern angegebenen Empfänger für eine Dauer von *LENGTH* Sekunden. Wird der Zustand nicht innerhalb der angegebenen Zeitspanne erneuert, wird der Kommunikationskanal automatisch geschlossen und es werden keine weiteren Nachrichten mehr an diese Zieladresse weitergeleitet. Die Erneuerung des Zustands erfolgt entweder durch Daten- oder durch ein weiteres SOPEN. Bei einer Datennachricht wird der Timer zurückgesetzt.
Beispiel: SOPEN|127.0.0.1|12880|2
- **Soft state control: SCLOSE|DST_IP|DST_PORT**
Diese Kontrollnachricht schliesst einen zuvor durch SOPEN geöffneten Kanal.
Beispiel: SCLOSE|127.0.0.1|12880
- **Hard state control: HOPEN|DST_IP|DST_UDP_PORT**
Diese Kontrollnachricht öffnet einen bidirektionalen Kommunikationskanal zu dem in den *DST*-Feldern angegebenen Host. Der Kanal bleibt solange geöffnet, bis er durch HCLOSE geschlossen wird.
Beispiel: HOPEN|127.0.0.1|12880
- **Hard state control: HCLOSE|DST_IP|DST_UDP_PORT**
Schliesst einen Kommunikationskanal, der zuvor durch HOPEN geöffnet wurde.
Beispiel: HCLOSE|127.0.0.1|12880
- **Control: SUCCESS|DST_IP|DST_UDP_PORT|LENGTH**
Diese Kontrollnachricht wird vom Gateway an den Sender + Controller gesendet, um eine erfolgreiche Ausführung eines der nachfolgenden Befehle zu signalisieren: SOPEN (SCLOSE) or HOPEN (HCLOSE). Die Werte des vorherigen Befehls, z.B., DST_IP, DST_UDP_PORT, LENGTH, werden unmodifiziert in diese Nachricht kopiert. Sollte das Längenfeld (LENGTH) im Falle von HOPEN nicht existent sein, ist die Länge mit 0 anzugeben.
Beispiel: SUCCESS|127.0.0.1|12880|2
als Antwort auf SOPEN|127.0.0.1|12880|2
- **Control: FAIL|DST_IP|DST_UDP_PORT|LENGTH**
Diese Kontrollnachricht wird vom Gateway an den Sender + Controller gesendet, um eine nicht erfolgreiche Ausführung eines der nachfolgenden Befehle zu signalisieren: SOPEN (SCLOSE) or HOPEN (HCLOSE). Die Werte des vorherigen Befehls, z.B., DST_IP, DST_UDP_PORT, LENGTH, werden unmodifiziert in diese Nachricht kopiert. Sollte das Längenfeld (LENGTH) im Falle von HOPEN nicht existent sein, ist die Länge mit 0 anzugeben.
Beispiel FAIL|127.0.0.1|12880|2
Als Antwort auf SOPEN|127.0.0.1|12880|2

Implementiere den Gateway in einer Soft- und Hardstate Variante in einer Sprache deiner Wahl. Nutze hierzu das oben beschriebene Protokoll und dessen Semantik. Die Bewertung wird auf einer Menge von automatischen Tests beruhen (diese werden denen in client.pl ähnlich sein), die die korrekte Implementierung des Protokolls validieren.

Das Programm *muss* auf den von der IRB bereitgestellten Unix Rechnern lauffähig sein. Wir werden somit *keine* Windowsprogramme akzeptieren! Gebe den Quellcode in einer lesbaren Form ab.

Abgabe bis Donnerstag, den 13. Januar 2011 nur bis 13:55 h s. t.

- **Als PDF-Dateien (keine MS-Office- oder OpenOffice-Dateien):** Mittels ISIS hochladen (<https://www.isis.tu-berlin.de/course/view.php?id=3584>)
- **In Papierform:** Postfach im Telefonknochenhaus (Erdgeschoss, hinter dem Pförtner rechts)
- Gib auf deiner Lösung deinen Namen, deine Matrikelnummer **und** den Namen deines Tutors an.