Rechnernetze Übung 1 Bericht

Name: Heidemann Joé Nicolas

s4jeheid 1634963

Aufgabe 2:

Mehrere Versuche in unterschiedlichen Netzen und mit unterschiedlichen Konfigurationen ließen keine Verbindung mit anderen Computer zu. Die Restriktionen meines Internetanbieters bezüglich der Funktionalitäten meines Routers erzeugten die meisten Fehlschläge.
Das Bearbeiten der Firewall erbrachte keine Lösung.

Aufgabe 3:

Ich habe die Anfragen mit Wireshark bemerkt, aber die Anfragen wurden nicht an den Port/TokenRingListener weitergeleitet, somit habe ich mich zwar durch Wireshark geklickt und viel aber nur folgendes Indiz für diese Hypothese gefunden:

Source	Destination	Protocol	length	Info
ROG-LAPTOP	host.docker.internal	UDP	100	58334 → 63714 Len=58

Eine Anfrage von meinem Laptop an meinen Desktop, aber die Destination hat Wireshark mit “resolve names” nicht als “DESKTOP” erkannt.

Hier ist ein lokal aufgebauter Ring. 4 Instanzen des Programms liefen simultan um diesen Ring zu erzeugen. den Verkehr zwischen den Knoten des Rings ließen sich in Wireshark im Interface “Adapter for loopback traffic capture” verfolgen. Wie In der Vorlesung erwähnt, ist dieses Interface die “Loopback-Adresse” oder auch “localhost” mit der Adresse = 127.0.0.1
Der einfachste Filter um nur UDP-Protokoll Verkehr angezeigt zu bekommen ist “udp”. Hinzu kommen einige Erweiterungen für die Filterung von spezifischen Sourceports oder destinationports sowie IPs. “||” entspricht dem logischen ODER, “&&” entspricht dem logischen UND.

Auffällig war die Paketgröße die sich beim Erweitern des Rings ebenfalls erhöhte. Die “Sequence” wird nicht nur in der Konsole ausgegeben sondern auch im Paket mitgesendet, folglich steigt die Paketgröße mit jedem neuen Knoten.


```
{"sequence":539,"ring":[{"ip":"192.168.0.53","port":57710},{"ip":"192.168.0.5
3","port":53651}]}
{"sequence":541,"ring":[{"ip":"192.168.0.53","port":57710},{"ip":"192.168.0.5
3","port":53651}]}
{"sequence":543,"ring":[{"ip":"192.168.0.53","port":57710},{"ip":"192.168.0.5
3","port":53651}]}
{"sequence":545,"ring":[{"ip":"192.168.0.53","port":57710},{"ip":"192.168.0.5
3","port":53651}]}
{"sequence":547,"ring":[{"ip":"192.168.0.53","port":57710},{"ip":"192.168.0.5
3","port":53651}]}
{"sequence":549,"ring":[{"ip":"192.168.0.53","port":57710},{"ip":"192.168.0.5
3","port":53651}]}
{"sequence":551,"ring":[{"ip":"192.168.0.53","port":49918},{"ip":"192.168.0.5
```

```
3","port":57710},{"ip":"192.168.0.53","port":53651}]}
{"sequence":554,"ring":[{"ip":"192.168.0.53","port":49918},{"ip":"192.168.0.5
3","port":57710},{"ip":"192.168.0.53","port":53651}]}
{"sequence":557,"ring":[{"ip":"192.168.0.53","port":49918},{"ip":"192.168.0.5
3","port":57710},{"ip":"192.168.0.53","port":53651}]}
{...}
{"sequence":605,"ring":[{"ip":"192.168.0.53","port":49918},{"ip":"192.168.0.5
3","port":57710},{"ip":"192.168.0.53","port":53651}]}
{"sequence":608,"ring":[{"ip":"192.168.0.53","port":49918},{"ip":"192.168.0.5
3","port":57710},{"ip":"192.168.0.53","port":53651}]}
{"sequence":611,"ring":[{"ip":"192.168.0.53","port":49918},{"ip":"192.168.0.5
3","port":55651},{"ip":"192.168.0.53","port":57710},{"ip":"192.168.0.53","port":
53651}]}
{"sequence":615,"ring":[{"ip":"192.168.0.53","port":49918},{"ip":"192.168.0.5
3","port":55651},{"ip":"192.168.0.53","port":57710},{"ip":"192.168.0.53","port":
53651}]}
{"sequence":619,"ring":[{"ip":"192.168.0.53","port":49918},{"ip":"192.168.0.5
3","port":55651},{"ip":"192.168.0.53","port":57710},{"ip":"192.168.0.53","port":
53651}]}
{"sequence":623,"ring":[{"ip":"192.168.0.53","port":49918},{"ip":"192.168.0.5
3","port":55651},{"ip":"192.168.0.53","port":57710},{"ip":"192.168.0.53","port":
53651}]}
```


Verbesserungsvorschlag:

Eine TimeToLive Einbauen die 2sek * Ring.size() lang auf ein Signal wartet.
Sobald ein Endpoint im Ring länger auf ein Signal warten muss. Sendet er ein “integrity-Signal” an seinen Folgeknoten, dieser muss bestätigt das “integrity-Signal” und sendet wiederum seinem Folgeknoten ein “integrity-Signal”. So baut sich der Ring, Knoten für Knoten wieder neu auf und fehlerhafte Knoten werden aus dem Ring entfernt. Die Umsetzung lässt natürlich Raum für Optimierungen, aber immer noch besser als alle laufenden Instanzen zu beenden und Endpoint für Endpoint neue IP und Port Adressen einzugeben.