1. 實驗日期:

2020/04/30

1. 實驗名稱:

Exp.9: 路徑MTU之發現

Exp.10: Network Disturbance網路干擾分析

Exp.11: Error Control錯誤控制

1. 實驗相關技術資訊:

1. What is the path MTU discovery? Please briefly describe how it works.

PMTUD的目標是找出可以從來源發送到目的地的最大可能封包大小，對於大數據傳輸，這將是發送數據的最有效方式。

PMTUD的工作原理是發送不同大小的數據包，並在IP header中設置Do not Fragment（DF）位，這告訴路徑上的路由器如果發現無法處理數據包的鏈路，則丟棄該數據包。路由器也應該發回ICMP錯誤消息，說它們已經丟棄了數據包。來源將獲得這些ICMP錯誤，並意識到相關的數據包太大，並且回報數據包大小。

不幸的是，從PMTUD首次定義以來，互聯網已成為一個更加嚴峻的地方。許多人使用ICMP數據包進行拒絕服務（DoS）攻擊。結果，許多人阻止所有傳入的ICMP數據包，這打破了PMTUD，因為來源無法再得到及時的反饋。來源確實看到它的數據包丟失並且在它們未被確認時可以超時，但是這大大增加了使用PMTUD的代價，並且實際上使數據傳輸比不使用PMTUD時更慢。

2. ARQ is a traditional technique for error control. Please compare the following ARQ schemes with examples.

1. Idle RQ (or Stop-and-Go)
2. Selective Repeat
3. Go-back-N

**自動重傳請求**（Automatic Repeat-reQuest，ARQ）是OSI模型中[數據鏈路層](https://zh.wikipedia.org/wiki/%E6%95%B0%E6%8D%AE%E9%93%BE%E8%B7%AF%E5%B1%82)和傳輸層的錯誤糾正協議之一。它通過使用確認和超時這兩個機制，在不可靠服務的基礎上實現可靠的信息傳輸。如果發送方在發送後一段時間之內沒有收到確認幀，它通常會重新發送。ARQ可能包括停止等待ARQ協議和連續ARQ協議，錯誤檢測（Error Detection）、正面確認（Positive Acknowledgment）、超時重傳（Retransmission after Timeout）和 負面確認及重傳（Negative Acknowledgment and Retransmission）等機制。

**停止並等待ARQ協議（stop-and-wait）**

停止並等待協議的工作原理如下：

1. 發送點對接收點發送數據包，然後等待接收點回復ACK並且開始計時。
2. 在等待過程中，發送點停止發送新的數據包。
3. 當數據包沒有成功被接收點接收時候，接收點不會發送ACK.這樣發送點在等待一定時間後，重新發送數據包。
4. 反覆以上步驟直到收到從接收點發送的ACK.

發送點的等待時間應當至少大於傳輸點數據包發送時間（數據包容量除以發送點傳輸速度），接收點ACK接收時間（ACK容量除以接收點傳輸速度），數據在連接上的傳送時間，接收點檢驗接收數據是否正確的時間之和。在實際應用當中，等待時間是這個和的2到3倍。

這個協議的缺點是較長的等待時間導致低的數據傳輸速度。在低速傳輸時，對連接頻道的利用率比較好，但是在高速傳輸時，頻道的利用率會顯著下降。

**連續ARQ協議（Continuous ARQ）**

為了克服停止並等待ARQ協議長時間等待ACK的缺點。這個協議會連續發送一組數據包，然後再等待這些數據包的ACK.

**回退N重傳(Go-Back-N)**

* 接收點丟棄從第一個沒有收到的數據包開始的所有數據包。
* 發送點收到NACK後，從NACK中指明的數據包開始重新發送。

**選擇重傳(Selective Repeat)**

* 發送點連續發送數據包但對每個數據包都設有個一個計時器。
* 當在一定時間內沒有收到某個數據包的ACK時，發送點只重新發送那個沒有ACK的數據包。

效果上SR > Go-Back-N > Stop-and-wait

參考資料:

<https://zh.wikipedia.org/zh-tw/%E8%87%AA%E5%8A%A8%E9%87%8D%E4%BC%A0%E8%AF%B7%E6%B1%82>

<https://wapbaike.baidu.com/item/PMTU>