

アドホックネットワーク構築

学籍番号 045713C:大城 和也
グループ B

提出期限:平成 17 年 7 月 12 日 (火)

グループメンバー

045711G : 上原 裕亮
045712E : 大城 和輝
045715K : 大城 悠
045727C : 杉山 千秋
045728B : 砂川 祐樹
045729K : 住谷 聡
045731B : 高良 史朗
045760E : 与儀 健人

1 実験目的

アドホックネットワークとは何かを学ぶとともに、実際に iBook を用いて 20-30 台規模のアドホックネットワークを構築し、実践を交えてネットワーク構築の基礎を理解する。

2 実験方法

2.1 olsrd、NTTCP のソースを入手し、各自のノート PC にインストール

実験の web ページからソースをダウンロードし、説明に習ってインストールを行う。

2.2 マルチホップが起きる環境を作成せよ。

マルチホップを起こさせるために距離をとるか、ipfw でルーティングテーブルの設定を変える。

2.3 工学部1号館709室での映像をライブカメラで撮影し、実験室のノードで確認する

工学部1号館の709室にwebカメラを取り付けたPCを置き、その映像をブロードキャストする。その映像をアドホックのネットワークを活用して実験室で確認する。そのためには途中でホップすることが必要となるので途中でノードをおくようにする。

2.4 その他

その他ではサークル上のネットワークの構築が主な実験内容であり、ipfwでルーティングテーブルの設定を行う事によりサークル上にノードをつなく。

3 結果と考察

3.1 olsrd、NTTCPのソースを入手し、各自のノートPCにインストール

olsrd、NTTCPをダウンロードし、ターミナルを用いてインストールした。

olsrdについて

olsrdとはOLSRプロトコルを用いるアプリケーションである。OLSRとはOptimized Link State Routingの略称であり、ルーティングプロトコルの一種である。一般にアドホックネットワークのプロトコルはReactive型とProactive型に別れ、OLSRDはProactive型である。これは通信を行う前に経路を確立しておくものである。そのため、通信要求が発生するとすぐに通信できるという利点を持つ。あらかじめ経路表を作成するために常にパケットを送出して周辺に存在するノードの確認を行う(Helloメッセージ)。OLSRの特徴として挙げられるのはフラッドイングを効率よく行う事であり、それはMRP集合というノードの集まりを作る事で行えるようになっている。

nttcpについて

nttcpはnew tcpのことでありtcpの改良版である。これらはネットワークのパフォーマンスを測るものであり、通信の速度をはかるものと考えれば良い。オプション-iを用いる事でそのノードはホストとなり、-T hostnameでそのホストとのスループットを測る事ができる。これを用いて実験を行う。

3.2 マルチホップが起きる環境を作成せよ。

マルチホップとはあるノードとノードの距離が離れている、あるいは制限されていて直接通信できない際に、その途中のノードを中継として経由して通信する事を言う。

よってマルチホップが起きる環境を作成するには無線が届かないところまでお互いのノードの距離をとるか、ルーティングケーブルを設定して互いのパケットを破棄するようにすればよい。

今回は工学部 709 室の実験の際には物理的距離をとることで、サークル上のネットワークの構築の実験では ipfw コマンドを用いてマルチホップが起こることを目指した。

3.3 工学部 1 号館 709 室での映像をライブカメラで撮影し、実験室のノードで確認する

先程も述べたようにこの実験では、実際に距離をとる事でマルチホップが起きるようにしている。今回私たちが使用した AirMAC の無線が届く距離は約 40m である、これは実験室から 709 室までの距離よりより長い。しかし、壁などの障害によりその距離は著しく低下する。そのため、マルチホップが必要となってくるのである。

次に今回の私たちの配置を図 1-図 8 で表す。



図 1: 709 室



図 2: 7 階の階段前



図 3: 6 階の階段前



図 4: 5 階-6 階の中間地点



図 5: 5 階の階段前

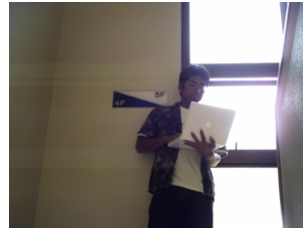


図 6: 4 階-5 階の中間地点

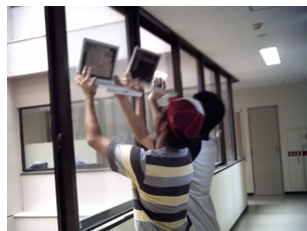


図 7: 4 階の階段前

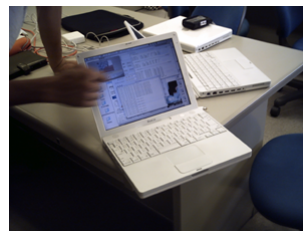


図 8: 403 号室 (実験室)

ちなみに、これらの階段というのは実験室に一番近い階段のことを言う。さらに、写真がきれいには写ってはいないが 6 階と 7 階の中間地点にも一つのノードがある。

初めに、ping コマンドを用いて目標の端末間の通信が可能かどうかを調べた。ちなみに ping コマンドは相手に制御用のデータを送り、その応答が帰ってくるかを調べるものである。

```
-----  
[Satoshi-SUMIYA:~] j04029% ping 192.168.1.15  
PING 192.168.1.15 (192.168.1.15): 56 data bytes  
64 bytes from 192.168.1.15: icmp_seq=0 ttl=62 time=21.697 ms  
64 bytes from 192.168.1.15: icmp_seq=1 ttl=62 time=9.504 ms  
64 bytes from 192.168.1.15: icmp_seq=3 ttl=62 time=17.578 ms  
64 bytes from 192.168.1.15: icmp_seq=4 ttl=62 time=6.429 ms  
64 bytes from 192.168.1.15: icmp_seq=5 ttl=62 time=112.848 ms  
64 bytes from 192.168.1.15: icmp_seq=6 ttl=62 time=375.811 ms  
64 bytes from 192.168.1.15: icmp_seq=7 ttl=62 time=4.577 ms  
^C  
--- 192.168.1.15 ping statistics ---  
11 packets transmitted, 7 packets received, 36% packet loss  
round-trip min/avg/max = 4.577/78.349/375.811 ms  
-----
```

time とはデータを送信してからの返ってくるまでの往復の時間を示している。これを考えるとかなりのばらつきがあるのが分かる。これより、通信が不安定なのが読み取れる。

さらに、図の配置のときに実験室のノードが通ったルートを traceroute コマンドを用いて調べた。

```
-----  
[Satoshi-SUMIYA:~] j04029% traceroute 192.168.1.15  
traceroute to 192.168.1.15 (192.168.1.15), 30 hops max, 40 byte packets  
 1 * * 192.168.1.12 (192.168.1.12) 11.555 ms  
 2 192.168.1.31 (192.168.1.31) 33.696 ms 19.31 ms 9.194 ms  
 3 192.168.1.11 (192.168.1.11) 16.286 ms 13.367 ms 10.318 ms  
 4 192.168.1.15 (192.168.1.15) 23.414 ms 11.435 ms 16.87 ms  
-----
```

この結果を見て分かるように今回用いたノード (PC) の数が 9 つなのに比べて、実験室から 709 室までに経由したノードは僅か 4 つだけとなっている。これは OLSR プロトコルでの MPR 集合に選ばれないで除外されたと考えられる。つまり、例を挙げれば 192.168.1.29(以後 29) と 192.168.1.15(以後 15)の間には一つのノードがあるのだが 29 は 15 と直接通信ができるのでプロトコルはそれを経由しないようにしたのである。他のノードにも同じ事が言える。つまり、距離の近さが全員を経由しないような状況を作り出したのである。

次に実験室のノードから 709 室のノードのスループットを示す。

```
-----  
[Satoshi-SUMIYA:~] j04029% nttcp -T 192.168.1.15  
nttcp-l: couldn't get dataport from receiveing side "192.168.1.15" (=192.168.1.15)  
) [Satoshi-SUMIYA:~] j04029% nttcp -T 192.169.1.15  
      Bytes  Real s  CPU s  Real-MBit/s  CPU-MBit/s  Calls  Real-C/s  CPU-C/s  
1 8388608 216.61  0.12   0.3098   559.2405   2048    9.45   17066.7  
1 8388608 216.69  0.20   0.3097   335.5443   5178   23.90   25890.0  
-----
```

今回、スループットを測る際には nttcp を使ってもなかなかつながりませんでした。ちなみに私の位置 (図 5) からスループットを測定したときは

```
-----  
[Kazuya-OSHIRO:~] j04013% -T 192.168.1.15  
      Bytes  Real s  CPU s  Real-MBit/s  CPU-MBit/s  Calls  Real-C/s  CPU-C/s  
1 8388608 102.96  0.14   0.6518   479.3490   2048   19.89   14628.6  
1 8388608 103.04  0.34   0.6513   197.3790   5118   49.67   15052.9  
-----
```

となっており、私の方が 709 室に近いのか MBit/s の値が大きくなっているのが確認できる。さらに、直接スループットを測定した場合は下のようになった。

```
-----
[Kazuya-OSHIRO:~] j04013% nttcp -T 192.168.1.15
      Bytes Real s CPU s Real-MBit/s CPU-MBit/s Calls Real-C/s CPU-C/s
1 8388608 4.89 0.09 13.7329 745.6540 2048 419.10 22755.6
1 8388608 4.91 0.26 13.6736 258.1110 4020 819.09 15461.5
-----
```

ここでの速度は 13.7329 となっておりかなり速い、つまりホップを繰り返す事により極端に通信の速度が低下しているのが分かる。これは無線という事、さらにホップを数回行うという事によりネットワーク自体の安定性が低いと考えられる。

3.4 その他

3.4.1 サークル上のネットワークの構築

初めに結果から。結果として火曜日の班はサークル上のネットワークを構築する事はできなかった。飛び飛びでそれらしきものができていたので時間があればきっと構築できたと思う。

ちなみに私の周りでは三人だけでできたのですが…。三人だと設定しなくてもサークルになるので意味はない。とりあえず、設定とその時に経過していることを示す。

```
-----
[Kazuya-OSHIRO:~] j04013% sudo ipfw show
00100      127      9564 deny udp from 192.168.0.30 to 192.168.0.255 698 via
en100200    98      7388 deny udp from 192.168.0.7 to 192.168.0.255 698 via en1
00300       6      436 deny udp from 192.168.0.51 to 192.168.0.255 698 via
en100400    0          0 deny udp from 192.168.0.43 to 192.168.0.255 698 via
en100500    0          0 deny udp from 192.168.0.29 to 192.168.0.255 698 via
en100600    0          0 deny udp from 192.168.0.28 to 192.168.0.255 698 via
en100700   61      4584 deny udp from 192.168.0.31 to 192.168.0.255 698 via
en100800   47      2740 deny udp from 192.168.0.19 to 192.168.0.255 698 via
en100900   36      2772 deny udp from 192.168.0.18 to 192.168.0.255 698 via
en101000    0          0 deny udp from 192.168.0.22 to 192.168.0.255 698 via
en101100    0          0 deny udp from 192.168.0.24 to 192.168.0.255 698 via
en165535 2295879 2056240542 allow ip from any to any
-----
```

```
-----
[Kazuya-OSHIRO:~] j04013% traceroute 192.168.0.19
traceroute to 192.168.0.19 (192.168.0.19), 30 hops max, 40 byte packets
 1 192.168.0.12 (192.168.0.12) 28.225 ms 24.538 ms 7.41 ms
 2 192.168.0.19 (192.168.0.19) 11.923 ms * 5.889 ms
-----
```

これは ipfw を使う事によって 192.168.0.12 と 192.168.0.19 以外のあの時実験室にあったノードのパケットを拒否した設定である。初めは全てを拒否するコマンドをした後、この二つのアドレスを許可したのだがそれでは何故か

互いのノードが認識されなくなったので強引的に作成しようとした。この様に設定した。

traceroute をしてみたらたった 2 人ではあるがホップを行い通信する事ができた。このようにサークル上にするのはまだまだだったが今回、私は ipfw を用いて制御する事はできた。

3.4.2 ネットワークを使用するアプリケーションの利用

今回、アドホックネットワークで iChat でのボイスチャットを試してみた。成功して少し面白かった。他にも iTunes における曲の共有や、ネットワーク上でのファイルの共有ができる事を確認した。このようにアドホックを用いる事でインフラの無い小さなネットワークを作成し、その間だけでのネットワークアプリケーションが使える。

3.4.3 アドホックネットワークの活用例の検討

- 一つ目、現在、アドホックネットワークで期待されているのは災害時に携帯やインターネットなどの連絡網が麻痺したときに非常用の連絡網として活躍する事が挙げられる。これにより、救助隊の位置や場所を把握したりすることにより救助に役立てるということである。
- 二つ目、自動車をノードとし、アドホックネットワークを構築する。これにより隣接する車や信号機の状況などを確認できるようにし、交通事故の回避や、渋滞の緩和などが期待できる。さらに、その情報を携帯などの端末を用いて歩行者にも知らせればさらに事故は減るであろう。
- 三つ目、それは学校の中でネットワークを構築するといったものである。特に大学ではプレゼンや授業の資料配りに活用できるであろう。
- 四つ目、団地やマンション内だけの掲示板や回覧板の役目をするネットワークをアドホックで作成する。これは多少三つ目と被っているだろうか…。
- 五つ目、イベント会場などの一時的なネットワークとして活用する。これにより、イベントの場所などの情報を客に提供したり、従業員の位置の確認なども可能となる。
- 六つ目、ゲームのネットワーク対戦などに用いる。これはもう既に PSP で実装されているのかな？従来の通信よりはるかに用途が増すと考えられる。

いくつか提案したがこれらの案にもセキュリティ面などの問題が残るのが現状である。

4 その他 (感想)

今回はネットワークの構築という事でした。アドホックというものを初めて知りましたが以外と面白いもので iChat でのボイスチャットには少なからず驚きました。残念だったのがやはり、おまけとしてのサークル上のネットワークができなかった事。… 欲しかったな 10 点 (ボソッ)。TA がもう少し時間くれたらできたかもな～。とまあ今回もとりあえず仕上げる事ができましたがやっぱりレポートを始めるのが遅くてちょっと苦しみました。次はなるべく早く仕上げられるように心がけたいと思います。

参考文献

- [1] <http://internet.watch.impress.co.jp/www/column/wp2p/wp2p06.htm>,”
第 6 回 OLSR (Optimized Link State Routing) プロトコル”
- [2] <http://www.h7.dion.ne.jp/matsu/index.html>,” ファイヤープロジェクト”
- [3] <http://www.rsch.tuis.ac.jp/mizutani/online/latex/index.html#general>,”
文書整形システム LaTeX 入門”
- [4] PDF ファイル,” 公立はこだて未来大学 2004 年度システム情報科学演習 I”