

# OpenSolaris 3分クッキング

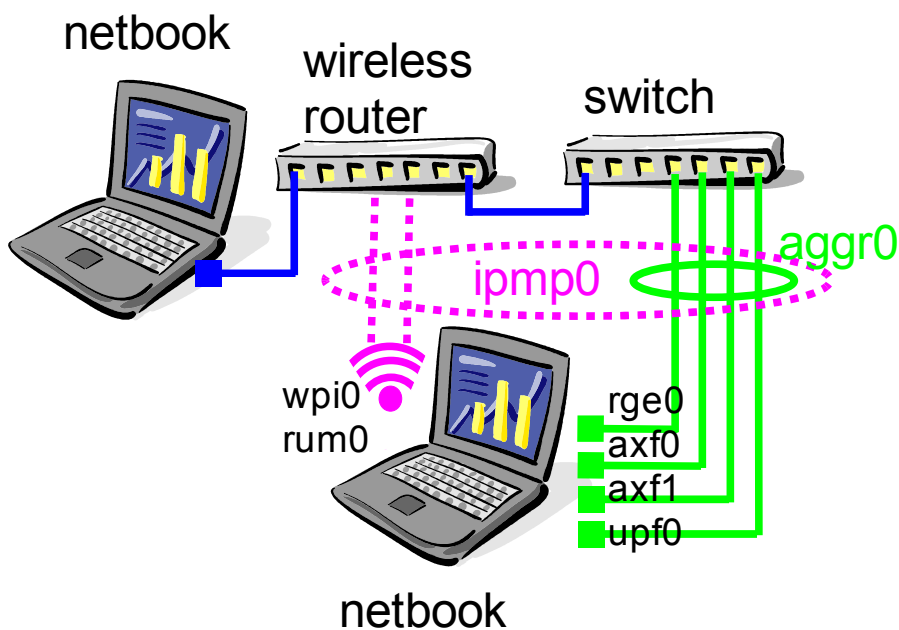
## ☆レシピ 第14巻☆

### ネットブックで Crossbow/IPMP 「切っても切っても切れない ネットブック偏」

#### Solaris 3分クッキング: ネットブックで Crossbow と IPMP のデモ

今夜の3分クッキングは、OpenSolaris 2009.06 から導入された CrossBow や新しくなった IPMP の機能を使って、ネットワークの冗長化のデモを紹介いたします。詳しい設定方法などはこちらのレシピを参照ください。

#### デモのネットワークの概略図



#### デモ環境の説明



OpenSolaris 2009.06 版から導入された Crossbow(仮想ネットワーク)や進化した Ip マルチパスの機能は、仮想環境や複雑なネットワーク環境では、とても便利な機能ですが、手軽なネット PC や簡単なネットワーク環境でも、これらの機能を気軽に利用できなかと、少し変わった実験にチャレンジしてみました。一般にネットブックと呼ばれる小型のノート PC では、ワイヤードとワイヤレス(WiFi)の2つのネットワークインターフェースが標準で装備されていますが、この二つを一本のネットワークのフェイル・オーバーとして使えないか？、さらに、これに上の写真のような USB 接続で使用可能な、市販のネットワークアダプタを追加で接続し、Crossbow や IPMP の機能をつかってネットワーク接続を多重化させてみました。

接続ネットワーク機器の詳細:

RealTek RTL8101E/8102E PCI Express Fast Ether (本体内蔵) - rge

Intel PRO/Wireless 3945ABG (本体内蔵) - wpi

Logtec LAN-TX/U2H3B (USB2.0 Ether)- axf

Planex UE-200TX-G(USB2.0 Ether) - axf

IO Data ETX2-US2(USB2.0 Ether) - upf

Planex GW-US54Min2 (USB Wifi) - rum

GLDv3 対応の3本の USB 2.0 Ether アダプターと本体内蔵の Ether の4本を Crossbow の Link-aggregation 機能を使って aggr0 という名前で1本に束ねます。さらに、この aggr0 と2本の WiFi アダプタを Active-Standby 形式で IP マルチパス機能を使って ipmp0 という名前で、一本に束ねます。つまり、このデモ環境では、ネットブック本体は、ipmp0 という一つの論理ネットワークインターフェースで通信を行いますが、6つの物理ネットワークアダプタの全部の接続が切れない限り、ネットワーク接続は保持されます。デモでは2つのネットブックを LAN 接続して、ネットワークケーブルを順番に抜いていっても接続が保持されることを紹介いたします。

### 各設定の手順

1) デフォルトの Network Auto Magic (nwm) サービスをオフにして、マニュアル設定に切り替えます。

```
user@opensolaris:~$ pfexec svcadm disable nwam
user@opensolaris:~$ pfexec svcadm enable svc:/network/physical:default
```

## 2) Link-aggregation の作成

```
user@opensolaris:~$ dladm show-link
LINK          CLASS      MTU      STATE    OVER
rge0         phys      1500    up      --
wpi0         phys      1500    up      --
rum0         phys      1500    up      --
axf0         phys      1500    up      --
axf1         phys      1500    up      --
upf0         phys      1500    up      --
user@opensolaris:~$
user@opensolaris:~$ pfexec dladm create-aggr -l rge0 -l axf0 -l axf1 -l upf0
aggr0
user@opensolaris:~$ dladm show-link
LINK          CLASS      MTU      STATE    OVER
...
aggr0         aggr      1500    up      rge0 axf0 axf1 upf0
user@opensolaris:~$ pfexec ifconfig aggr0 plumb
user@opensolaris:~$ pfexec ifconfig aggr0 192.168.11.101/24 up
```

ここまでの設定で、新しく設定した `aggr0` を4つのネットワークインターフェースの束として使用可能になります。

## 3) Ip マルチパス `ipmp0` の設定

```
< WiFi の接続 >
user@opensolaris:~$ pfexec ifconfig wpi0 plumb
user@opensolaris:~$ pfexec dladm connect-wifi -e test_ssid wpi0
user@opensolaris:~$
user@opensolaris:~$ pfexec ifconfig rum0 plumb
user@opensolaris:~$ pfexec dladm connect-wifi -e test_ssid rum0
user@opensolaris:~$
< IPMP の作成 >
user@opensolaris:~$ pfexec ifconfig ipmp0 ipmp
user@opensolaris:~$ pfexec ifconfig ipmp0 192.168.11.100/24 up

< aggr0 をフェイルオーバーで再設定 >
user@opensolaris:~$ pfexec ifconfig aggr0 group ipmp0 -failover
192.168.11.101/24 up

user@opensolaris:~$ ipmpstat -g
GROUP          GROUPNAME     STATE      FDT          INTERFACES
ipmp0          ipmp0         ok         --           aggr0

< wpi (WiFi) を Active/Standby フェイルオーバーで設定 >
user@opensolaris:~$ pfexec ifconfig wpi0 group ipmp0 -failover
192.168.11.102/24 standby up
user@opensolaris:~$ ipmpstat -g
GROUP          GROUPNAME     STATE      FDT          INTERFACES
ipmp0          ipmp0         ok         10.00s      aggr0 (wpi0)
user@opensolaris:~$
< rum (WiFi) を Active/Standby フェイルオーバーで設定 >
user@opensolaris:~$ pfexec ifconfig rum0 group ipmp0 -failover
192.168.11.103/24 standby up
user@opensolaris:~$ ipmpstat -g
GROUP          GROUPNAME     STATE      FDT          INTERFACES
```

```

ipmp0      ipmp0      ok          10.00s     aggr0 (rum0 wpi0)
user@opensolaris:~$ ipmpstat -i
INTERFACE  ACTIVE  GROUP      FLAGS      LINK      PROBE      STATE
aggr0      yes     ipmp0      --mb---   up        ok         ok
rum0       no      ipmp0      is-----  up        ok         ok
wpi0       no      ipmp0      is-----  up        ok         ok
user@opensolaris:~$

```

これで ipmp0 が使用可能となります。ipmp0 は、aggr0 の東に加え、2つの WiFi インターフェースを IPMP で束にしたものです、aggr0 の4本のネットワークがフェイルすると、スタンバイ状態の WiFi インターフェースがアクティブになります。つまり最終的に6つの全部のネットワークインターフェースが同時に切れない限り、IP 接続は保持できます。また、それぞれのインターフェースの接続が復旧すると、元の状態に回復します。

## 接続状況の確認

### 1) Link-aggregation の状況確認

```

user@opensolaris:~$ dladm show-aggr -x
LINK      PORT          SPEED DUPLEX  STATE   ADDRESS          PORTSTATE
aggr0     --            100Mb full   up      0:1d:92:59:c:cb  --
          rge0         100Mb full   up      0:1d:92:59:c:cb  attached
          axf0         100Mb unknown down    0:1d:92:59:c:cb  standby
          axf1         100Mb full   up      0:1d:92:59:c:cb  attached
          upf0         100Mb unknown down    0:a0:b0:a2:61:38  standby

```

### 2) IPMP の状況確認

```

user@opensolaris:~$ ipmpstat -i
INTERFACE  ACTIVE  GROUP      FLAGS      LINK      PROBE      STATE
wpi0       no      ipmp0      is-----  up        ok         ok
rum0       no      ipmp0      is-----  up        ok         ok
aggr0      yes     ipmp0      --mb---   up        ok         ok
user@opensolaris:~$

```

### Ifconfig -a 出力

```

user@opensolaris:~$ ifconfig -a
lo0: flags=2001000849<UP,LOOPBACK,RUNNING,MULTICAST,IPv4,VIRTUAL> mtu 8232 index 1
    inet 127.0.0.1 netmask ff000000
ipmp0: flags=8001000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4,IPMP> mtu 1500 index 2
    inet 192.168.11.100 netmask ffffffff00 broadcast 192.168.11.255
    groupname ipmp0
aggr0: flags=9040843<UP,BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4,NOFAILOVER> mtu 1500 index 3
    inet 192.168.11.101 netmask ffffffff00 broadcast 192.168.11.255
    groupname ipmp0
rum0: flags=69040843<UP,BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4,NOFAILOVER,STANDBY,INACTIVE> mtu 1500 index 4
    inet 192.168.11.103 netmask ffffffff00 broadcast 192.168.11.255
    groupname ipmp0
wpi0: flags=69040843<UP,BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4,NOFAILOVER,STANDBY,INACTIVE> mtu 1500 index 5
    inet 192.168.11.102 netmask ffffffff00 broadcast 192.168.11.255
    groupname ipmp0
lo0: flags=2002000849<UP,LOOPBACK,RUNNING,MULTICAST,IPv6,VIRTUAL> mtu 8252 index 1
    inet6 ::1/128

```

## 参考情報

USB Ether アダプター用のドライバーの入手方法

今回のデモで紹介した USB Ether アダプターのドライバーは、Murayama さんの、Free NIC drivers for Solaris

<http://homepage1.nifty.com/mrym3/taiyodo/eng/>

から入手可能です。 Link-aggregation 機能を使うためには GLDv3 対応である必要があるため、GLDv3 用に用意された Makefile を使って再コンパイルします。

#### IPMP と Link Aggregation の特徴の比較

	IPMP	Link Aggregation
テクノロジータイプ	レイヤー3 (Network レイヤー)	レイヤー2 (データリンクレイヤー)
構成のためのツール	Ifconfig(1M)	d1am(1M)
リンクベースのフェイルの検知	可能	可能
プローブベースのフェイルの検知	ICMP ベース	LACP ベース
スタンバイ 設定	設定可	なし
必要なリンクレイヤー	ブロードキャスト有効(WiFi も可)	Ethernet 特定
必要なドライバーのフレームワーク	なし	GLDv3 フレームワーク

#### 参考ドキュメント

<http://www.opensolaris.com/learn/features/networking/>

<http://docs.sun.com/app/docs/doc/819-6990?!=en>

<作成: 原口 >