NetFlow(softflowd)の使用方法について

2020 年 4 月　初版
2020 年 11 月　第 2 版　F70/F71 を追加
2021 年 3 月　第 3 版
システムコンテナの Alpine 化にともない変更

NetFlow 方式によるトラフィック監視方法についてご説明します。FITELnet F70/F71/F220/F221（以下、本装置）のコンテナでは、フロー情報を NetFlow パケットとしてコレクタへ送信することが可能です（コレクタは別途ご準備いただくことが必要です）。

本装置で NetFlow パケット生成とコレクタへの送信を行うために、必要な手順をご説明します。

※文中で引用している資料は下記「FITELnet LXC アプリケーション」のページに掲載しております。

https://www.furukawa.co.jp/fitelnet/product/container/lxc/index.html

下記、図 1 のネットワーク構成のように本装置の設定を行った状態で、Giga1/4 ポートをモニタリングして、コンテナにモニタリング結果を出力して、コンテナからコレクタにトラフィック情報（NetFlow パケット）を送信するための手順をご説明します。

図 1 の構成において、コンテナのインタフェース eth11 は本装置内部の L2 スイッチを介して(*1)サブインタフェース Giga1/1.1 ポートに接続しています。コンテナは Giga1/4 ポートのモニタリングによりフローデータをキャッシュして NetFlow パケットを生成します。NetFlow パケットはコンテナから出力された後、Giga1/1.1 を経由して Giga1/4 から WAN 側回線に出力されて、コレクタに送信されます。また、Giga1/4 では NAT 変換を行い、パケットの送信元アドレス（LAN のアドレス）を Giga1/4 のインタフェースのアドレスに変換しています。この設定によってモニタリングされるパケットの送信元アドレスは Giga1/4 につけられた global アドレス(100.1.1.2)になります。

(*1) 機能説明書「2.19.3 ブリッジグループの装置内部構成」をご参照ください。

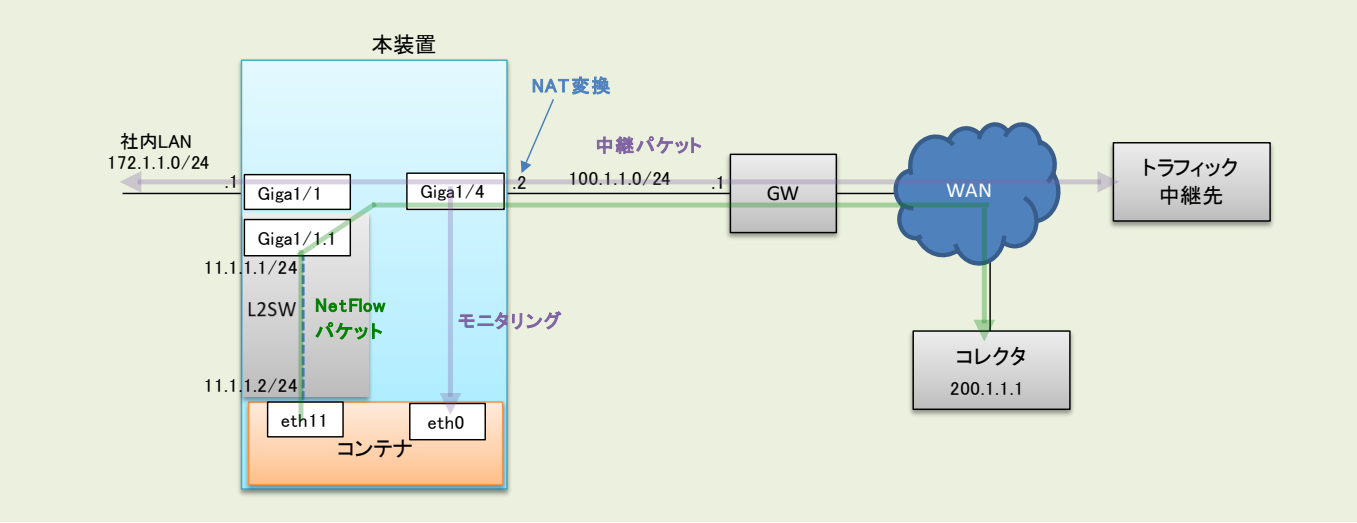

図 1. ネットワーク構成例

1．ルータ OS のインタフェース・コンテナ設定

ルータ OS で、container enable や container-use のコンフィグ設定を行います。上記ネットワーク構成に合わせた本装置のコンフィグを以下に示します。

```
!
container enable                          ←コンテナのサービスを有効にする設定
!
ip route 0.0.0.0 0.0.0.0 100.1.1.1        ←デフォルトルートをGWに設定
ip nat list 1 172.1.1.0 0.0.0.255         ←172.1.1.0/24 を list 1 と定義
ip nat list 2 11.1.1.2 0.0.0.255          ←11.1.1.0/24 を list 2 と定義
!
interface GigaEthernet 1/1
 vlan-id 1
 bridge-group 1
 channel-group 1
exit
!
interface Port-channel 1
 ip address 172.1.1.1 255.255.255.0
exit
!
interface GigaEthernet 1/1.1
 vlan-id 11
 bridge-group 11
 channel-group 11
 container-use                            ←コンテナに論理インタフェースを設定
exit
!
interface Port-channel 11
 ip address 11.1.1.1 255.255.255.0
exit
!
interface GigaEthernet 1/4
 vlan-id 8
 bridge-group 8
 channel-group 8
exit
!
interface Port-channel 8
 ip address 100.1.1.2 255.255.255.0
 ip nat inside source list 1 interface    ←list 1、list 2 に含まれる送信元アドレスを
 ip nat inside source list 2 interface      インタフェースのアドレスにNAT変換する
exit
!
!
!
end
```

コンテナ環境では、”eth + bridge-group 番号”が名前となるインタフェースが生成されます（上記の例では、”eth11”が生成されます。

⇒詳細は、『コンテナ型仮想環境の使用方法』の『1.2.ルータ OS の設定(CLI) 』の章をご参照ください。

２．コンテナ環境のネットワーク設定

コンテナ環境で使用するネットワークの設定を行います。

2-1) コンテナ環境の起動

ルータ OS の CLI 上で『container start』コマンドを実行し、コンテナ環境を起動させた後、『container attach』コマンドでコンテナ環境内のシェルを起動します。

2-2) ネットワーク設定ファイルの編集

コンテナ環境で/etc/network フォルダ配下のネットワーク設定ファイル（interfaces）を編集し、ルータ OS 側の設定で生成された”eth11”にアドレス設定を行います。以下に設定例を示します。

```
auto eth0
iface eth0 inet dhcp
auto eth11                              ←ルータOS側の設定で生成された論理IF
iface eth11 inet static
      address 11.1.1.2                  ←IP addressを11.1.1.2に設定
      netmask 255.255.255.0
      gateway 11.1.1.1                  ←gatewayをルータOS(11.1.1.1)に設定
```

2-3) 設定反映

設定後、『/etc/init.d/networking restart』コマンドで設定を反映させてください。

⇒コンテナ環境の起動方法の詳細は『コンテナ型仮想環境の使用方法』の『1.4.コンテナ環境の起動』と『1.5.1. シェル起動』を、設定方法の詳細は『1.6.コンテナ環境のネットワーク設定』をご参照ください。

３．ルータ OS の port monitor container 設定

ルータ OS で port monitor の設定を行い、モニタリングポートを container に設定します。

今回の設定例では Giga1/4 をミラーしますので、『port-monitor mirrored gigaethernet 1/4』でミラーポートの設定を行い、『port-monitor monitor container』でモニタリングポートを container に設定します。

設定した情報は『show port-monitor』で確認できます。

```
#port-monitor mirrored gigaethernet 1/4

#port-monitor monitor container

#
#show port-monitor
[Current state]
  mirrored port(ingress): 1/4
  mirrored port(egress) : 1/4
  monitor port: container
```

⇒詳細は、『コンテナ型仮想環境の使用方法』の『付録 C ポートモニタリング機能の使用』を参照ください。

４．softflowd スタート

各種パラメータを指定して softflowd を起動します。例えば、NetFlow のバージョンを 9、サンプリング対象のインタフェースを eth0、コレクタの IP アドレスを 200.1.1.1、ポート番号を 5141 に設定して動作させる場合、次の手順を行ってください。

4-1) /etc/init.d/ ファルダに移動してください。

4-2) -v、-i、-n のオプションをつけて softflowd を起動してください。

softflowd -v 9 -i eth0 -n 200.1.1.1:5141

起動後、コンテナから Giga1/4 ポートを経由して NetFlow パケットの配信が開始されます。コレクタで受信できているかご確認下さい。

５．softflowd 情報確認

/etc/init.d フォルダ配下で下記のコマンドを実行して、情報確認することができます。

softflowctl statistics……統計情報を確認できます。

softflowctl dump-flows……expire 待ちのフローの情報を確認できます。各フロー情報の末尾に表示される”EXPIRY EVENT for flow xx in xxxx seconds”が expire までの残り時間です。

softflowd はフローの受信～NetFlow パケットとしてコレクタへ配信するまで、下記①②③の順に処理を行います。上記コマンドの表示はそれぞれ下記のように変化します。

softflowd 処理	コマンドの表示
① フローをモニタリングポートより受信	・softflowctl statistics”の”Number of active flows”を加算 ・softflowctl dump-flows にフロー情報を表示
② フローデータをキャッシュ（expire 待ち）	・softflowctl statistics”の”Number of active flows”を保持 ・softflowctl dump-flows にフロー情報を表示（継続）
③ フローデータが expire したら、NetFlow パケットとしてコレクタへ配信	・softflowctl statistics”の”Number of active flows”を減算 ・softflowctl statistics”の”Flows expired”を加算 ・softflowctl statistics”の”Flows exported”を加算 ・softflowctl dump-flows からフロー情報を削除

```
/etc/init.d # softflowctl statistics
softflowd[954]: Accumulated statistics since 2020-02-27T08:47:34 UTC:
Number of active flows: 6
Packets processed: 26986255
Fragments: 0
Ignored packets: 16 (16 non-IP, 0 too short)
Flows expired: 10 (0 forced)
Flows exported: 10 (10 records) in 6 packets (0 failures)
Packets received by libpcap: 26986327
Packets dropped by libpcap: 0
Packets dropped by interface: 0

Expired flow statistics:  minimum       average       maximum
  Flow bytes:                 272     858996850    2147492100
  Flow packets:                 1       1738861       4347150
  Duration:                 0.00s        399.74s       1187.06s

Expired flow reasons:
       tcp =         0   tcp.rst =         0   tcp.fin =         0
       udp =         6      icmp =         0   general =         0
   maxlife =         0
over 2 GiB =         4
  maxflows =         0
   flushed =         0

Per-protocol statistics:     Octets      Packets   Avg Life    Max Life
        Unknown (6):     8589966424     17388596     741.91s    1187.06s
       Unknown (17):           2080           10     171.62s     260.66s
```

```
/etc/init.d # softflowctl dump-flows
softflowd[954]: Dumping flow data:
ACTIVE seq:15 [20.1.1.10]:34449 <> [158.202.233.239]:5141 proto:17 octets>:164 packets>:1
octets<:0 packets<:0 start:2020-02-27T09:23:01.002 finish:2020-02-27T09:23:01.002 tcp>:00
tcp<:00 flowlabel>:00000000 flowlabel<:00000000 vlan>:1 vlan<:0 ether:00:80:bd:f0:5a:c2
EXPIRY EVENT for flow 15 in 210 seconds

ACTIVE seq:16 [15.1.1.1]:34449 <> [158.202.233.239]:5141 proto:17 octets>:164 packets>:1
octets<:0 packets<:0 start:2020-02-27T09:23:01.002 finish:2020-02-27T09:23:01.002 tcp>:00
tcp<:00 flowlabel>:00000000 flowlabel<:00000000 vlan>:0 vlan<:0 ether:00:80:bd:f0:5a:b8
EXPIRY EVENT for flow 16 in 210 seconds

ACTIVE seq:4 [0.0.0.0]:68 <> [255.255.255.255]:67 proto:17 octets>:10846 packets>:34
octets<:0 packets<:0 start:2020-02-27T08:48:23.187 finish:2020-02-27T09:23:34.478 tcp>:00
tcp<:00 flowlabel>:00000000 flowlabel<:00000000 vlan>:0 vlan<:0 ether:ff:ff:ff:ff:ff:ff
EXPIRY EVENT for flow 4 in 243 seconds

ACTIVE seq:2 [15.1.1.1]:1 <> [30.1.1.3]:1 proto:6 octets>:1336592582 packets>:2705653
octets<:0 packets<:0 start:2020-02-27T08:47:34.610 finish:2020-02-27T09:24:31.070 tcp>:12
tcp<:00 flowlabel>:00000000 flowlabel<:00000000 vlan>:0 vlan<:0 ether:00:80:bd:f0:5a:b8
EXPIRY EVENT for flow 2 in 3600 seconds

ACTIVE seq:11 [15.1.1.1]:1 <> [30.1.1.2]:1 proto:6 octets>:1862284658 packets>:3769807
octets<:0 packets<:0 start:2020-02-27T09:07:21.666 finish:2020-02-27T09:24:31.070 tcp>:12
tcp<:00 flowlabel>:00000000 flowlabel<:00000000 vlan>:0 vlan<:0 ether:00:80:bd:f0:5a:b8
EXPIRY EVENT for flow 11 in 3600 seconds

ACTIVE seq:14 [15.1.1.1]:1 <> [30.1.1.1]:1 proto:6 octets>:1577078204 packets>:3192466
octets<:0 packets<:0 start:2020-02-27T09:17:15.194 finish:2020-02-27T09:24:31.070 tcp>:12
tcp<:00 flowlabel>:00000000 flowlabel<:00000000 vlan>:0 vlan<:0 ether:00:80:bd:f0:5a:b8
EXPIRY EVENT for flow 14 in 3600 seconds
```

６．expire 時間の変更

フローを受信してから expire するまでの時間はパケット種別毎にデフォルトで下記のように設定されています。これらの値は-t オプションで設定変更することができます。

```
/etc/init.d # softflowctl timeouts
softflowd[954]: Printing timeouts:
             TCP timeout:  3600s
    TCP post-RST timeout:  120s
    TCP post-FIN timeout:  300s
             UDP timeout:  300s
            ICMP timeout:  300s
         General timeout:  3600s
        Maximum lifetime:  604800s
         Expiry interval:  60s
```

参考までに、Maximum lifetime を 60 秒に変更する場合は次の手順を行ってください。

6-1) /etc/init.d/ ファルダに移動してください。

6-2) softflowd shutdown コマンドで softflowd を停止してください（「７．softflowd 停止」の項をご参照ください)。

6-3) softflowd -v 9 -i eth0 -n 200.1.1.1:5141 -t maxlife=60 を実行して、softflowd を起動してください。

７．softflowd 停止

/etc/init.d/フォルダ配下で『softflowctl shutdown』コマンドを実行することで softflowd を停止することができます。

※softflowd を起動した後、別の条件で再度 softflowd コマンドを実行する場合は、再実行前に softflowd を停止して下さい。停止せずに実行すると softflowd が正常動作しなくなる可能性があります。

参考：softflowd（https://manpages.debian.org/unstable/softflowd/index.html）

以上