ネットワークトラフィック情報の自動収集&統計機構

北橋 雅子*

*情報処理振興事業協会 (IPA)
富士ゼロックス情報システム(株)より出向中
Email: kita@stc.ipa.go.jp

ネットワークの利用状況を把握することは、ネットワークを運用管理する上で是非必要なことである。しかし、トラフィック情報を収集し、統計処理をしてそれを評価するのは、手間がかかり、また、長期間の情報を得るためには、ディスク容量も消費する。そこで、なるべく自動的にトラフィック情報を収集し、一定時間ごとに統計処理を行なって、統計結果だけを残すようにすれば、長期間に渡ってトラフィックに関する統計情報を得ることができ、ディスクの消費も少なくてすむであろう。そのような機構を実現するために、3つのデーモンを利用する方法を提案する。3つのデーモンとは、モニターデーモン(NMD)、ロギングデーモン(LOGD)、スタットデーモン(STATD)である。これらは複数のマシン上に分散して存在することができる。これらのデーモンの仕様について述べ、さらに、現在、これを実験運用しているので、それについて報告する。

1. はじめに

通信メディアを流れるパケットをモニターして、そのパケット数やバイト数をカウントする機器やツールは、いろいろある。それらを使って集められた情報は、そのネットワークの運用方針に従って、ある項目ごとに統計が取られる。この統計処理は、その方針が決まれば、一定期間、定期的に行われなければならない。しかし、情報収集機構とリンクしていないために、これにはかなりの工数がかかる場合がほとんどである。そこで、統計処理を先送りしようとすると、収集した情報をストアしておくために莫大なディスク要領を必要とすることになる。

この問題を解決するために、情報収集機構と統計機構をリンクさせ、ある程度、自動的に統計処理を行うシステムを提案する。このシステムは、3つのデーモンによって構成される。

2. 3つのデーモン

3つのデーモンとは、ネットワークモニタリングデーモン(NMD)、ロギングデーモン(LOGD)、スタットデーモン(STATD)である。図1に、これらの関係を示す。

NMDとLOGDは、アーキテクチャ上は別のホスト上にあってもよいが、現在の実装では、必ず同一ホスト上になければならない。NMDとLOGDは、モニター情報の必要な各セグメントごとにひとつずつインストールされる必要がある。STATDは、LOGDからモニター情報を得て、統計ファイルを作成する。

図中、上位ユーザとは、STATDを起動し、作成された統計ファイルを読み込んで表やグラフを作成するプログラムのことである。

NMDとLOGDは、inetdに登録されなければならない。STATDの起動によって、LOGD

A network trafic information auto.collecting & statistics system
by Masako Kitahashi*,
*Information - technology Promotion Agency, JAPAN
loan from Fuji Xerox Information Systems CO., LTD

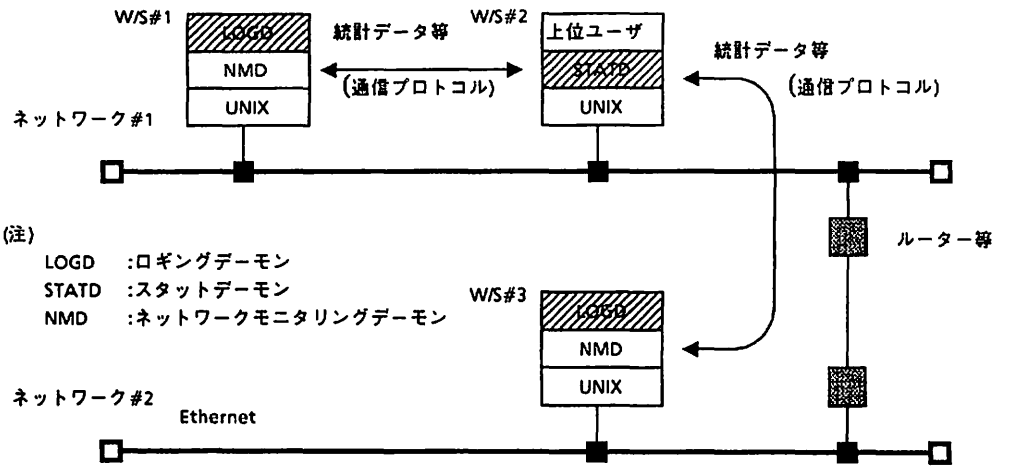

図1　3つのデーモンの関係

が起動され、さらに、LOGDは、NMDを起動する。ひとつのLOGDは、複数のSTATDを相手にすることができる。

● ネットワークモニタリングデーモン(NMD)

nitデバイスを用いて、イーサネットを流れるパケットをモニターする。

● ロギングデーモン(LOGD)

ネットワークモニタリングデーモンからモニタリングデータを取得して1次レベルの統計処理を行い、その結果をスタットデーモンに渡す。1次レベルの統計処理とは、指定された単位時間分のモニタリングデータを統計することである。

● スタットデーモン(STATD)

ロギングデーモンから、1次レベルの統計情報を受取り、それをファイルに落とす。さらに、2次レベル以上の統計処理を行う。これは、単位時間分の統計ファイルが、ある時間分に達したら、その時間分の統計ファイルを作成するものである。1次レベルの統計ファイル、あるいは、低次レベルの統計ファイルを読み込んで、より大きな時間分の統計ファイルを作成する。

3. 統計処理パラメータ

時間パラメータは、STATD起動時のコマンド引数として与えられ、それ以外のパラメータはコンフィグファイルによって与えられる。

3.1 時間パラメータ

スタットタイム
　統計情報を生成する単位時間(秒)

スタットピリオド
　統計処理の開始から、終了までの時間(秒)（スタットタイムより大きな値)

レベルアップ処理を指定した場合、スタットタイムの値は、その値より大きなレベルアップ周期の約数でなければならない。ただし、最大のレベルアップ周期(24時間)以上の値の場合には任意の値が指定できる。レベルアップ処理を指定しない場合には、スタットタイムは任意の値を指定できる。

3.2 プロトコルパラメータ

統計処理の対象とするレイヤおよびプロトコルを指定する。

3.3 アドレスパラメータ

アドレスタイプおよびアドレスを指定する。ここでアドレスを指定すると、そのアドレスを送受信先とするパケットのみが統計処

理の対象となる。また、この時、アドレスファイルには、ここで指定された順番にアドレスが書き込まれる。アドレスの指定がなければ、モニターした全てのアドレスが統計の対象となり、それらのアドレスは、出現順にアドレスファイルに書き込まれる。

3.4 レベルアップ処理パラメータ

レベルアップするかしないかを、パラメータで指定することができる。レベルアップとは、低次レベルの統計ファイルを読み込んで、より大きな時間分の統計ファイルを作成することである。

4. 統計情報のファイル生成

STATDは、リモートまたはローカルホスト上で動作するLOGDから、統計情報を取得して統計ファイルを生成する。統計ファイルは、スタットタイムで指定された時間ごとに生成される。

この統計ファイルとは別に、アドレスファイルが生成される。アドレスファイルは、STATD 1個につき、ひとつ生成される。コンフィグファイルのアドレスパラメータでアドレスが指定されていない限り、モニターしたアドレスが出現順に書き込まれる。

4.1 アドレスファイル名

生成されるファイルは、起動時に指定された文字列に以下の文字列を付け足したものである。

ファイル拡張子: sadr

例えば、ユーザが指定した文字列が"log"である場合、log.sadrとなる。

4.2 統計ファイル名

生成されるファイルは、起動時に指定された文字列に以下の文字列を付け足したものである。

YYMMDD_HHMMSS_u or _l1から_l4
これは、西暦年月日、時分秒、統計レベルを示している。
ファイル拡張子: sdat

例えば、ユーザが指定した文字列が"log"である場合、スタットタイムごとに生成される統計データファイル名は、
log940102_034500_u.sdatのようになる。

5. 統計レベルアップ処理

STATDは、レベルアップ周期ごとに、複数の統計ファイルをまとめて、上位の統計ファイルを生成する。統計レベルアップ処理の対象となるファイルは、共通のアドレスファイルを持つ統計ファイルでなければならない。

上位の統計ファイルは、下位の統計ファイルの各項目を合計した合計ファイルと、下位の統計ファイルの各項目をくし刺しにして最大値をまとめた最大ファイルの二通りである。合計ファイルは、単位時間ごとの統計ファイルと同じデータ構造である。

5.1 レベルアップ周期

原則として以下の周期でレベルアップを行なう。

1分
10分
1時間
1日

5.2 ファイル名

合計ファイルの拡張子は、統計データファイルと同様".sdat"である。

最大ファイルの拡張子は、".smax"である。

6. コンフィグ・ファイル

統計処理に関する各種のパラメータを格納するテキストファイルである。ファイルの拡張子は".conf"である。各行はキーワード、等号、数値または文字列(予約語)からなる。各々はスペースまたはタブで区切られていなければならない。なお、空行または行の最初にシャープ記号のある行は無視する。

表1　キーワード一覧

keyword	説明	備考
level-up	レベルアップ処理の実行	実行をするかしないかを指定(YES/NO)
filter	プロトコル･フィルター	レイヤ、プロトコルを指定 (DE_ETHER/NW_IP/NW_XNS)
upper-filter	上位フィルター	上位プロトコル(値)
option-filter	オプション･フィルター	IPでのでのみ使用(ポート番号) (値)
address-type	アドレス･タイプ	収集するアドレスのタイプ (ETHERNET/INTERNET/NETWORK)
address	アドレス･フィルター	プロトコルに準じたアドレス値(値)
netmask	ネットワーク･マスク	IPでのみ使用(値)

表2　データリンク層

項目	値	備考
プロトコル･フィルター	DL_ETHER	必須
上位フィルター	タイプ･フィールド値	1個以上を指定
アドレス･タイプ	ETHERNET	必須
アドレス･フィルター	物理アドレス	0または複数個の指定(16進指定)

表3　ネットワーク層

項目	値	備考
プロトコル･フィルター	NW_IP/NW_XNS	必須
上位プロトコル･フィルター	トランスポート･プロトコル値	複数個の指定が可能
アドレス･タイプ	INTERNET/NETWORK	必須
アドレス･フィルター	論理アドレス	0または複数個の指定

以下に、コンフィグファイルの例を示す。

```
#
#        -- Configuration File --
#
#level-up        = YES / NO
#filter =   DL_ETHER
            NW_XNS
            NW_IP
#
#        << for ethernet >>
#upper-filter  =   0x0600 / 0x0800
                   0x0806 / 0xffff
#        << for xns    >>
#upper-filter  = 1 / 4 / 5
#        << for ip     >>
#upper-filter  = 1 / 6 / 17
#option-filter  = 20 / 23 / 25
#
#address-type  =  ETHERNET
                 INTERNET
                 NETWORK
#
#        << for ethernet >>
#address = 0x0800200ad2a1
#        << for xns    >>
#address =    0x00000080/0x08003700dcb1
#        -- network
#address = 0x00000082
#        << for ip     >>
#address = 131.221.64.80
#        -- network
#address = 131.221.64.0
#netmask = 255.255.64.0
#   ----------------------------------------
level-up     = YES
filter       = NW_IP

#    ICMP
upper-filter   = 1

#     TCP
upper-filter   = 6
option-filter  = 23
option-filter  = 20
option-filter  = 21
option-filter  = 25
option-filter  = 43
option-filter  = 53
option-filter  = 119
option-filter  = 123

#    UDP
upper-filter   = 17
option-filter  = 53
option-filter  = 69
option-filter  = 517
option-filter  = 2049          #nfs
option-filter  = 520

address-type   = INTERNET
```

この例では、IPパケットだけを収集し、IP上位プロトコルICMP, TCP, UDPごとに統計することになっている。さらに、TCP, UDPの場合は、ポート番号ごとに統計する。アドレスフィルターは指定されていないので、検出したホストアドレスを順次アドレスファイルに書き込み、それらのアドレスのマトリックス上で、upper-filter、option-filterごとの統計値が統計ファイルに記録される。すなわち、アドレスからアドレスへどのようなパケットがどのくらい流れたか、が記録される。

level-up = YESとなっているので、一定時間ごとにレベルアップが行われる。

7. 実験運用

現在、このシステムは、WIDE東京NOC、および、IPAで実験運用されている。図2-5に、7月のIPAの統計結果の一部を示す。これは、スタットデーモンの上位ユーザとして、IPANeMa(1)を利用して作成したものである。

毎日の1秒あたり平均バイト数を、IP上位プロトコルごとに色分けして表示したグラフを図2に示す。数字はIP上位プロトコル番号である。6、17以外はほとんど存在しない。すなわち、TCP, UDPが、IPAのIPトラフィックのほとんどである。1日のトラフィックが最大なのは、7月13日である。この日の1秒あたりのトラフィック平均は、

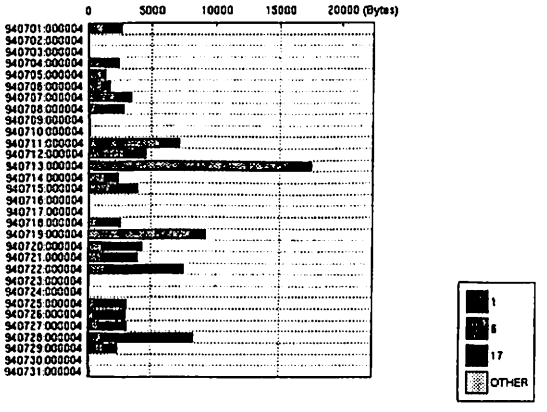

図2　7月 - 1日ごとの平均トラフィック

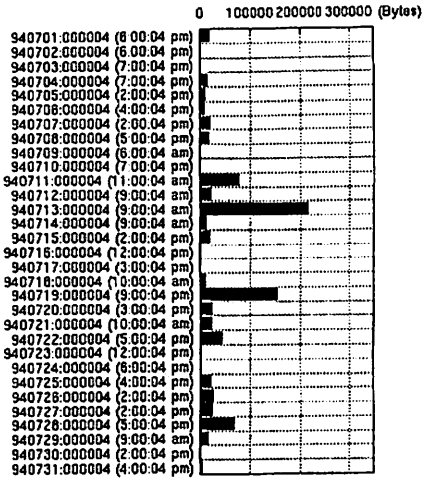

図3　7月 - 1日ごとの最大平均

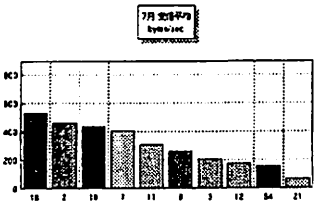

図4　7月 - 1か月受信平均

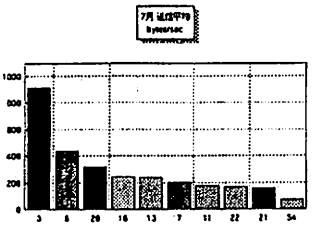

図5　7月 - 1か月送信平均

17533.42 bytes/sec すなわち、140267.36 bps である。つまり、1.4% の負荷である。Ethernet は、10% 以下の負荷で利用するのが望ましいとされている(2)が、これは、利用率が非常に低い、と言える。また、週末の利用率が極端に低いことがわかる。

図3に、どの時間帯に最大トラフィックが発生したかをグラフで示す。7月1日は、6:00 - 7:00 pm のトラフィックが1日の最大となっている。7月4日は、7:00 - 8:00 pm、7月5日は、2:00 - 3:00 pm、というように、最大を記録する時間帯に規則性は見られない。7月13日の最大は、9:00 - 10:00 pm で、216928.81 bytes/sec すなわち、1735430.48 bps で、約17% の負荷である。

どのホストからの送信が多いか、また、どのホストの受信が多いか、それぞれ、ホストごとに統計して、その TOP10 をグラフで示したのが、図4、図5である。下の数字はホストの識別番号である。7月の全受信トラフィック1秒あたり平均 3170.34 bytes/sec のうち、ほとんどを TOP10 のホストで占めている。このセグメントには、約25台のホストがつながっている。

7月の送信トラフィックが最も多いのは、3番のホストである。これは、このホストが何らかのサービスを提供しているためと考えられる。

8. おわりに

実験運用の結果、さまざまの問題点が明らかになった。まず、データの信頼性の問題

がある。1時間分の統計ファイルを作成するときに、だいたい、3000から8000パケットくらいを落としている。トラフィックが集中すれば、それだけ、パケットを落とす確率も高くなる。落としたパケット数は、レベルアップした時に、残らないので、1時間分の統計ファイルを消してしまうと、情報の信頼性がどれくらいかわからなくなる。しかし、1秒あたりの平均値は、それなりに信頼できると思われる。なぜなら、8000パケット落としていたとしても、1秒あたりにすると、2パケットとなるからである。それにしても、いつ、どのくらいのパケットを落としたかは、重要な情報なので、レベルアップファイルにもこれを残すようにしなければならない。

レベルアップしたときに、いかに重要な情報だけを残すか、という問題がある。現在のやり方だと、かなり、不要な情報が残ってしまう。もっと減らす余地がある。

現在は、パケット長の平均や最大をとっていないが、この情報が必要であると思われる。高負荷の場合はパケット長が長いほうが効率が良いとされている(2)。実際に流れているパケット長と、負荷の関係を見て、傾向を把握する必要があるだろう。

ある期間、統計を取って、そのネットワークの普通の状態を把握し、普通でない状態になった時にアラームを上げるようなサービスを追加したいと考えている。普通の状態を把握するために必要な情報は、ネットワークの性質によって異なる。それらをどのようにシステムに反映させるか、今後、実験運用を続けながら模索する。

このシステムは、IPANeMa(1)の一部として、FreeSoftとして配布している。(3) ftp.ipa.mgt.go.jp の /pub/IPANeMa から、anonymous ftp で get できる。

謝辞

本研究を進めるに当たって、日本大学 野口正一 教授、日本アイ・ビー・エム(株)小林善和 氏、日本電気(株)勅使河原 可海 氏、富士通(株)高橋 修 氏、(株)SRA佐原 伸 氏、広島市立大学 村山 優子 氏、富士ゼロックス情報システム(株)龍田 直紀 氏、藤野 晃延 氏、羽生田 栄一 氏、また、INTAP NM WG、およびWIDE Projectの方々、そして、情報処理振興事業協会 棟上 昭男 理事はじめ、その他の関係者の皆様に感謝の意を表します。

参考文献

(1) 北橋 著：ネットワーク管理システムIPANeMa、第13回IPA技術発表会論文集、1994

(2) 上谷 晃弘 編著：ローカルエリアネットワーク -イーサネット概説-、PP.135-150、丸善(株)、1985

(3) 情報処理振興事業協会：IPANeMa MonitorService User's Guide 、1994