さまざまなアクセス手段を備えたルータ運用支援機構

辻元 孝博† 大野 浩之†
† 東京工業大学大学院 情報理工学研究科

概要

さまざまな機器が今日のコンピュータネットワークを支えている。コンピュータネットワークの普及に伴い管理作業も増加し、ネットワーク管理者に大きな負担を与えている。その要因として、構成機器の多様化と管理手段の不足があげられる。本論文では、さまざまなアクセス手段からネットワークの運用ポリシを記述することで、これらの要因を改善し、管理作業の負荷を軽減するシステムを提案する。運用例としてルータの重要な機能の一つであるパケットフィルタリングに対して、このシステムが有効であることを示す。

Supporting router management using various access methods

Takahiro TUJIMOTO† Hiroyuki OHNO†
† Graduateschool of Information Science and Engineering, Tokyo Institute of Technology.

abstract

Nowadays, network administrators use various machinery and tools to support computer networks. As computer networks grow, network administrators have been load with management tasks because of the variety of machines and the lack of management methods. In this paper, we propose the system that reduce the loads of the management tasks by inputs of the management policy using various interfaces. for example, we show that our system is effective to the packet filtering which is the important function of routers.

1 はじめに

コンピュータネットワークの普及が急速に進み、ネットワークに接続される機器が増えた。それにともない管理作業も増加し、ネットワーク管理者に大きな負担を与えている。

ネットワークの管理が管理者に与える負担の主な要因は、次の2点である。1つは、ネットワーク構成機器が多様化し、管理対象が増えたため、ネットワーク管理に幅広い知識が求められること、もう1つはネットワーク管理手段がネットワークに依存することである。

そこで著者らは、この2つの問題点を改善するためのシステムを提案し、NIIS(Network Infomation Integrated Support) システムと名付けた。本論文では、NIIS システムの設計と実装、およびその利点について述べる。

2 ネットワーク管理の負担

計算機ネットワークの発展に伴い、さまざまな機器がネットワークに接続され、ネットワーク構成が複雑になっている。このような状況の変化にともない、ネットワーク管理者にかかる管理作業の負担が増加している。

本章では、ネットワーク管理者にとってネットワーク管理の負荷を増加させる要因について述べ、この問題を解決するためのネットワーク管理システムを提案する。

2.1 ネットワーク管理の問題点

1. 管理作業の複雑化

ネットワークに接続されるコンピュータが増加し、ネットワークを構成する機器や、使用される管理ツールが多種多様になった。それらの操作法、設定法が機種ごとに違うために、管理者には豊富な知識と経験が要求される。

企業の研究機関や大学の研究室、学科などの組織では、ネットワーク管理は本業ではなく、システムに詳しい人が副業として行うことが多い。しかし、ルータのパケットフィルタリングに関する設定ひとつにしても、ルータの機種ごとに設定方法が違う。設定ファイルが数字の羅列などで記述されるものが多く、設定意図も理解しにくい。そのため新規にルールを記述することはもとより、既存の設定変更作業すらも困難である。これはネットワーク管理のさまざまな局面で発生

しており、副業として管理作業を行なうことは、以前よりも難しくなっている。

2. ネットワーク管理手段の不足

現在のネットワーク管理手段の多くが、「ネットワーク管理をネットワーク経由」で行うことを前提としている。そのため、ネットワークが正常に機能していて、管理者や管理プログラムが管理対象の計算機に到達できるときには期待通りの能力を発揮するが、何らかの原因によりネットワークに障害が発生した場合、目的の計算機に到達できないため管理できない。

2.2 解決法

第1の問題点を解決するためには、管理作業をある程度自動化することが必要である。そのためには「システムをどのように運用していくか」という運用ポリシを明確にする必要がある。

また、運用ポリシから具体的な管理作業への変換機構が必要である。機種ごとの設定方法の違いがこの機構に吸収されるため、設定や変更に要する作業の負担を大幅に軽減できる。異なる機器ごとの設定方法の違いが管理者に対して隠蔽されるため、管理者は設定方法の違いを意識しなくてよい。

第2の問題点を解決するためには、ネットワーク構成機器やネットワーク管理支援ツールを、ネットワークに依存しない手段で設定できる必要がある。

3 NIISシステムの設計と実装

本章では、NIIS システムの設計と実装について述べる。NIIS システムは前章で述べた問題点を解決することを目的として設計、実装されたネットワーク管理支援システムである。

3.1 システムの概要

NIIS システムは NIIS サーバと NIIS クライアントから構成される (図 1)。NIIS サーバは、運用ポリシをもとに、設定対象に適した設定ファイルを生成し、機器の設定を更新する。運用ポリシとは「システムの運用方針を明確に記述したもの」であり、その記述法が定められている。運用ポリシを記述するためのさまざまな手段を提供するものが NIIS クライアントである。サーバ-クライアント間の通信プロトコルとして、NIIS プロトコルを定めた。NIIS プロトコルに従うことで、新たなクライアントを容易に開発できる。

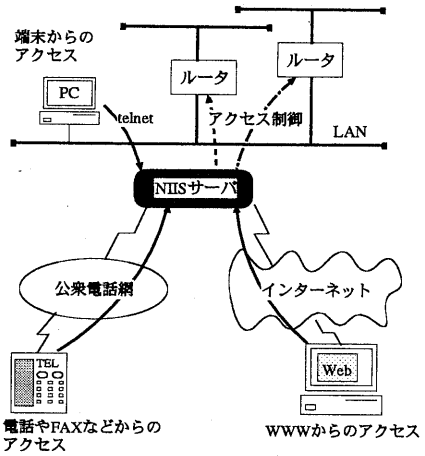

図 1: システム構成

3.2 NIIS サーバ

ネットワークに接続された複数の機器は NIIS サーバにより一元管理される。NIIS サーバは、NIIS クライアントから受け取った運用ポリシを内部の設定ファイル作成部で処理する (図 2)。設定ファイル作成部は、受け取った運用ポリシをもとに、管理している機器や管理ツールの設定ファイルを生成する。運用ポリシは、管理対象の機器に対応した形式の expect[?] スクリプトに変換される。

生成された expect スクリプトを実行することにより、設定の変更や管理情報の入手を行なう。expect は対話的に行なう設定を自動的に処理できるので、さまざまな機器の設定変更に対処できる。

3.3 NIIS クライアント

管理者は運用ポリシ記述法に従い、機器の運用ポリシを記述する。NIIS クライアントは、この運用ポリシを NIIS サーバに送信する。ここでは、WWW、端末インターフェイス、電話、その他のインターフェイスを用いた NIIS クライアントについて述べる。

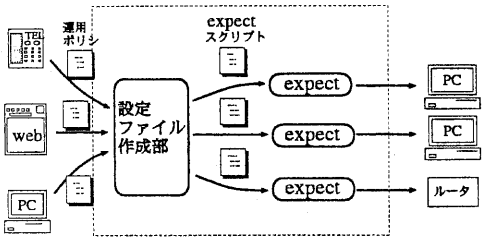

図 2: NIIS サーバ

3.3.1 WWW クライアント

WWW クライアントは CGI[1] を利用している。フォームにデータを入力することにより、運用ポリシを記述し、NIIS サーバにデータを送信する。WWW は機器の設定手段としては標準的なものである。

3.3.2 端末クライアント

telnet を使用して、NIIS サーバを利用できる。この場合、端末から NIIS プロトコルを直接入力する。端末クライアントを使うことにより、NIIS サーバとの実際の通信を参照できるため、NIIS クライアントを開発する際に役立つ。また、telnet 以外に shell や emacs 上から起動できる文字型クライアントがある。

3.3.3 電話クライアント

遠隔地から NIIS サーバにアクセスするための手段として、電話を用いて運用ポリシを入力する NIIS クライアントを作成した。これを電話クライアントと呼ぶ。電話クライアントを用いることで、ネットワークに依存せずに公衆電話網を介して、NIIS サーバにアクセスできる。これにより、ネットワークが分断され遠隔地からネットワーク経由でのアクセスができない状態でも有効な通信手段を確保できる。

電話クライアントは、ユーザからの入力に DTMF 信号[2] を用い、電話クライアントからユーザへの指示や問い合わせに音声ガイダンスを用いる。電話クライアントは内部で、ユーザからの DTMF 信号による入力を NIIS プロトコルに変換し NIIS サーバへ送信する。また NIIS サーバからの応答を音声ガイダンスに変換してユーザに送る。電話クライアントの実装には WIDE/PhoneShell[?] を利用した。

[1] Common Gateway Interface
[2] Dual Tone Multi Frequency

3.3.4 その他のインターフェイス

上記以外の NIIS クライアントとして、FAX やページャ、電子メールの利用したものが考えられる。例えば、本間らのシステム [?] を用いることで、運用ポリシを紙に記述し、FAX を利用して NIIS サーバに送信できる。これらのさまざまなアクセス手段により、管理作業の負担の軽減が期待できる。

4 NIIS システムを用いたルータの運用事例

NIIS システム運用事例として、ファイアウォールの設定を取り上げる。今回は、特にファイアウォールの重要な機能のひとつであるパケットフィルタリングの運用支援を行なった。本章では、NIIS システムが実際に行なったファイアウォールの設定について事例を述べる。

4.1 フィルタ運用ポリシ

パケットフィルタリングの方法の 1 つとして、始点と終点のアドレス、ポート番号などを指定してパケットの通過を制限する方法がある。今回、NIIS システムでフィルタリングの運用ポリシを記述するために図 3 で示される記述事項を定め、これをフィルタ運用ポリシと呼ぶ。*印を付した 6 項目は必須項目である。

*	始点 IP アドレス
*	終点 IP アドレス
*	プロトコル
*	始点ポート番号
*	終点ポート番号
*	許可または拒否
	始点サブネットマスクアドレス
	終点サブネットマスクアドレス
	フィルタ設定に関する注釈

図 3: フィルタ運用ポリシ

NIIS クライアントは、入力されたフィルタ運用ポリシを NIIS プロトコルに変換して、NIIS サーバに送る。図 4 は始点 IP アドレスの指定を NIIS プロトコ

ルに変換したものである。SAはアドレスが始点IPであることを表す識別子である。

```
SA  <address>
<address> ::= <num> "." <num> "." <num>
              "."<num> "." | <*>
<num> ::= <d> | <d><d> | <d><d><d>
<d> ::= any one of the ten digits 0 through 9
<*> ::= the asterisk character (ASCII code 42)
```

図 4: NIISプロトコルの例 (一部)

NIISサーバは、NIISクライアントから送信されるフィルタ運用ポリシをもとに、ルータの種類に応じた設定ファイルを生成する。この生成された設定ファイルによりフィルタ運用ポリシがネットワークに反映される。図5はソフトウェアフィルタリングを行なうscreendの設定ファイルの一例である。

```
#!/usr/contrib/bin/expect -f

spawn ftp $router
while 1 {
   expect {
        "Name*" {send "$user\r"}
        "Password:*" {send "$password\r"}
        "230*" {send "cd /etc\r"}
        "250*" {send "put $screend.conf\r"}
        "*226*" {send "bye\r";break}
        }
}
spawn rlogin $router
while 1 {
   expect {
        "Password:" {send "$password\r"}
        ">"         {send "ps\r";break}
        }
}

send "/tmp/find_screend.pl\r"
expect  -re "/etc/screend.conf\r\r\n(.*)>" {
        set process_id $expect_out(1,string)
        send "screend -L /var/log/screend"
        send "kill -9 $process_id\r"
        }
expect ">"
send "exit\r"
```

図 5: screend対応の設定ファイル

次に、各NIISクライアントがどのようにフィルタ運用ポリシを記述するかを示す。

4.2 WWWクライアント

図6にWWWクライアントを示す。CGIのフォームにしたがって、フィルタ運用ポリシを入力する。入力が終了した後、左下の"Submit Registration"と書かれたボタンを押すと、WWWクライアントからNIISサーバにフィルタ運用ポリシが送信される。

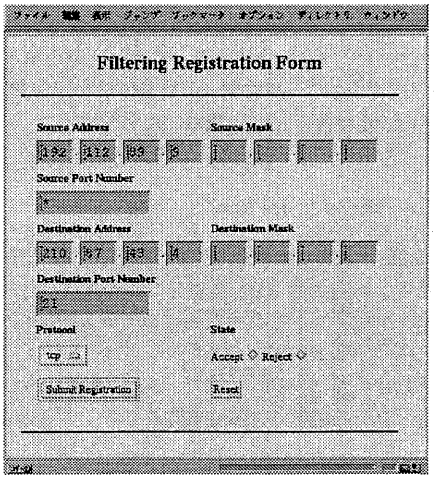

図 6: Webクライアント

4.3 電話クライアント

図7に電話クライアントを用いる際のシステムの動作を示す。ユーザが電話をかけるとWIDE/PhoneShellが電話クライアントを起動する。電話クライアントがNIISサーバに接続し、サービスが開始される。ユーザが電話クライアントからの音声ガイダンス指示に従ってDTMF信号を入力すると、WIDE/PhoneShellはDTMF信号が入力されたことを電話クライアントに通知する。電話クライアントは入力された信号列をNIISプロトコルに変換してNIISサーバに送信する。NIISサーバは電話クライアントからの要求を処理し、電話クライアントに応答する。電話クライアントは、NIISサーバからの応答を音声に変換してユーザに送

る。以上の繰り返しにより、ユーザは対話的にサービスを受けフィルタ運用ポリシを記述していく。

ユーザが電話を切ると、WIDE/PhoneShell は電話クライアントに対して、終了の指示をする。指示を受けた電話クライアントは、NIIS サーバとの接続を切断したのちに処理を終了する。

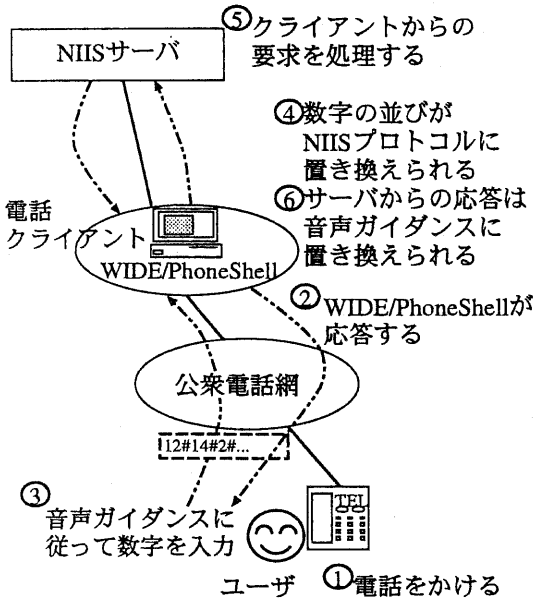

図 7: 電話クライアント概要図

電話クライアントにはさまざまな利用法がある。例えば、ユーザがファイアウォールの外部からメールを読む際に、pop3 のポートをアクセス可能にする必要がある。その場合、電話クライアントを通してフィルタ運用ポリシを記述すれば、フィルタの設定を外部からでも容易に変更できる。従来のようにシステムにログインし、システム管理者になってから変更する方式や、残留管理者に頼んで変更してもらう方式と比べて短時間でフィルタの変更作業を実施できる。

5 考察

さまざまなアクセス手段により、運用ポリシを記述することができるため、以下のような利点が生まれた。

WWW クライアントによる利点

- 設定項目を把握しやすい。
- 視覚的に設定できるので、設定間違いが少なく、不慣れな管理者でも容易に設定できる。

端末クライアントによる利点

- 対話的に設定変更していくので、わかりやすい。
- NIIS プロトコルを直接入力するので、NIIS サーバとの通信の詳細がわかる。

電話クライアントによる利点

- 電話は広く普及しているので、利用が容易である。
- コンピュータ、PDA 等の特別な機材を必要としないのでこれらの機器を持ち歩かなくてもよい。
- 携帯電話があれば、外出先などからも管理作業ができる。仮に携帯電話がない場合でも公衆電話等から利用できる。

電話クライアントは、DTMF 信号と音声を用いて運用ポリシを記述するため扱える情報量が少ないが、上述の利点があるため有用であるといえる。

また、全ての NIIS クライアントに共通する利点として、入力時に設定エラーをチェックするために設定間違いが減少することが挙げられる。ユーザはこれらのアクセス手段を状況に応じて選択できるため、それぞれの NIIS クライアントが持つ利点をいかせる。

NIIS システム全体では以下のような利点がある。

- 異なる種類のルータ間の設定方法の違いが隠蔽されるため、管理者は運用ポリシを記述するだけで、設定方法の違いを気にしなくてよい。
- 管理機能が NIIS サーバに一元化されているので、システムの保守が容易になる。
- expect を使用することで、多様な機器の設定に柔軟に対応でき、拡張性が高い。

NIIS サーバにルータ制御に関する新たな機能が追加された場合には、NIIS クライアントは NIIS プロトコルに従った実装を行なうことで、追加された機能を利用できる。

6 今後の課題

NIIS システムの実装、運用を進める上で、以下の課題がある。

NIISクライアントの追加

今回は、電話クライアント、端末クライアント、WWWクライアントを実装したが、今後さまざまなユーザインターフェイスに対応したNIISクライアントを提供する必要がある。アクセス手段を複数用意することで、ネットワークが分断された状態でも必要最低限のネットワーク管理ができる。今後は、すでにインフラストラクチャの整っているページャ、FAXなどを用いたNIISクライアントの開発を行い、システムの可用性を高める。

NIISシステムの運用

現段階は実験段階であり、管理者の手間、単純な入力間違い、管理作業に要する処理時間等どの程度減少するか十分な統計調査を行なっていない。実際に運用を開始するためには、十分に実験を繰り返して安定した評価を得る必要がある。

運用ポリシの抽象化

今後の展開を考えると、運用ポリシをあるレベルで抽象化することが重要である。そのために運用ポリシを記述する言語を用意する必要がある。この運用ポリシ記述言語を構成するためには、管理のために必要な事項を体系化しなければならない。

7 関連研究

ポリシの記述により、ルータのアクセス制御の設定ファイルを生成するシステムとして、今泉らによるシステム [?] がある。このシステムではネットワークのトポロジとサービスに関するポリシを記述することで、 複数のルータの設定ファイルを生成する。NIISシステムと比較すると、NIISシステムはポリシを記述する手段に特に重点を置いているのに対し、今泉らによるシステムでは、ポリシの記述言語、解析法に重点が置かれている。

RPSL(Routing Policy Specification Language)[?]は、AS(Autonomous System)から個々のルータを制御するレベルまで、記述することができ拡張性も高いため、今後運用ポリシの記述法として、検討する必要がある。

8 まとめ

本論文では、電話やWWWなど、さまざまなアクセス手段を利用してネットワーク管理を支援するシステムを提案した。NIISシステムにより、種々の構成機器や管理ツールを十分な設定知識がなくても、容易に運用でき管理作業を軽減できる。

また、管理手法の幅が広がったことで、ネットワークの外からでもシステム管理が可能になり、小人数の管理者によるシステム運用が可能になった。特に、電話などネットワークに依存しないアクセス手段は非常に有効である。管理者は、時間と場所にかかわらず、最も適したアクセス手段を選択できる。

現段階では、NIISシステムはscreendのみに対応しているが、現在YAMAHAやCiscoのルータなどへの対応を進めている。また、今後さまざまなNIISクライアントの開発を行なう予定である。