

メーリングリスト (ML)の作成とログ取得

K05 元会計・サーバ主任 森久 (marine)

みなさん、こんにちは。森久です。

今回はメーリングリスト(ML)の作成と Perl によるログの取得について紹介していきたいと思います。

さて、私はシス研においてサーバ管理者をしています。さまざまな種類のサーバソフトが稼動していますが、その中でも私が一番時間と手間をかけたものがメールサーバです。ここでサーバの紹介をしておきます。OS は FreeBSD6.2 メールサービスソフトは qmail を使用しています。

警告

以下の内容はシス研のサーバ内で有効に動いているものを紹介しています。

もしこの内容を別の環境で実行する場合、予期せぬ動作によりサービスの停止やサーバのダウンにつながる恐れがあります。

導入する際には、十分注意してテストを行ってからにしてください。

システム工学会、及び、私 (森久) は発生したいかなるトラブルに対しても責任を負いません。

1. メーリングリストについて

メーリングリストとはある特定のメールアドレスに対しメールを送信すると、あらかじめ登録されているメールアドレス (複数可) に同じ内容のメールを送信するというものです。CCではなくBCCと似ていますが、同じ内容のメールを送信していることはわかるが他人のメールアドレスはわからない。送信した人のアドレスも基本的にはわからない。という特徴があります。使われる状況としては、グループのメンバーに同じ情報を伝えたいとき、ニュースを配信するときなどです。このように情報を共有する際に大変有効な手段となります。

2. qmail について

qmail はメールサービスを提供するサーバソフトウェアです。qmail 以外では sendmail や Postfix などが多く使われています。qmail の特徴としては一連の動作プログラムが細かく分かれており、カスタマイズがしやすいということがあげられます。今回のメーリングリスト作成についてはその他のメールサービスソフトでも実現できると思いますが、詳細はググってください。

3. メーリングリストの基本的な考え方

メールサーバとはそもそもデータ (メール) を転送することが主な仕事です。ここに別

のメールサーバから転送されてきたメールがあるとしします。そのあて先が自分のメールサーバ内で管理しているメールアドレスならばローカルのメールボックスに保存します。もし違っていた場合はまた別のメールサーバに転送します。

ML ではメールを転送する機能を用いて実現します。具体的には ML のあて先メールアドレスを自分の管理するメールアドレスに設定します。また ML のメンバーのメールアドレスを 1 つのファイルに一覧として保存しておき、ML のアドレスにきたメールの内容をそのままそのファイル内のメールアドレスに送信するように設定します。

おおまかな流れ

ML のアドレスを作りユーザに通知

ユーザが ML のアドレス宛にメールを送信

メールサーバが ML のアドレス宛のメールを受信

設定ファイルに書かれているメールアドレスに同じ内容のメールを転送

4 . メーリングリストを作成

では ML の仕組みがわかったところで作成をしていきます。前提条件としてメールサーバが稼動しておりインターネット上からメールを受信でき、また、別のメールサーバにメールが転送できているものとします。

qmail のインストール方法によって環境が違っていると思うので以下では

.../qmail/

がルートディレクトリとします。.../は環境によって変わるので適宜読み替えてください。

まず、qmail でメールアドレスを作成する方法は 2 つあります。1 つ目はサーバにログインできるメンバーのホームディレクトリに「.qmail」ファイルと「Maildir」ディレクトリを作成する方法です。この場合アドレスは アカウント名@ドメイン名 となります。ドメイン名についてはメールサーバが稼動しているならすでに知っている単語だと思われるのでスルーします。

2 つ目の方法はルートディレクトリの中の alias ディレクトリに「.qmail-hogehoge」というファイルを作成する方法です。この場合アドレスは hogehoge@ドメイン名となります。

このどちらの方法でも ML の作成はできます。今回は導入が手軽な 1 つ目の方法をとりたいと思いますが、実は 2 つ目の方法を利用して、各個人のアカウントにメールアドレスを増やすことができます。まずホームディレクトリ直下に「.qmail-ML にしたいアドレス」というファイルを作成します。これにより新しく アカウント名-ML にしたいアドレス@ドメイン名 というメールアドレスが作成されます。次にこのファイルの中にメーリング

リストに参加するメールアドレスを追加していきます。

ここで注意してほしいのですが、メールサーバの設定で特定のドメインからのメールしか受け付けないという設定になっている場合、ファイルに追加したメールアドレスからメールの送受信ができない恐れがあります。もしメールが届かない場合そちらを確認してみてください。

メールアドレスを記述する場合は先頭に&を書き続けてメールアドレスを書いてください。また 1 行に 1 つのメールアドレスを記述することが基本です。

このファイルを作成したら終了です。実際に アカウント名・ML にしたいアドレス@ドメイン名 にメールを送ってみてください。ファイル内のメールアドレスすべてにメールが届いたら無事に ML が作成できた証となります。もし届かない場合、

- ・特定のメールアドレスに届かない -> ファイルのメールアドレス記述ミスや受信ドメインの設定ミス

- ・すべてのメールアドレスにメールが届かない -> ML のメールが間違っている、設定ファイルの保存場所が間違っている

などが考えられます。

以上でメーリングリストの作成は終了です。

この方法はメールアドレスがの数が少ない場合は非常に手軽に作成でき、また、管理も楽なのでおすすめです。しかし、せいぜい 2・30 個のメールアドレスを記述するとメールアドレスの書き換えが面倒になり、またメールサーバへの負担も増加します。そのため登録メールアドレスが多いとわかっている場合には、素直にメーリングリスト作成支援ソフトを使用しましょう。

5 . Perl によるログ取得

さて ML が作成できて、運用していくとこんな意見が出てきます。それは「過去のメールを見たい」というものです。途中から ML に参加した人は当然として、あのときのメールに書いてあったんだけど・・・、うんぬんかんぬん、ということもありやはりログはほしいものです。また Web ブラウザで確認できればいちいちメールクライアントソフトを立ち上げる必要もありませんからね。

そこでこれからは Perl 言語を用いて ML に流したメールをログとして保存するプログラムを紹介していきたいと思います。プログラム自体は付録の mail.pl を参照してください。

あらかじめ、Perl を実行できるようにしておき、mail.pl をメールサーバがアクセスできるようにしておく（実行権限を与えておく）ことをしておいてください。

警告

付録の mail.pl はシス研のサーバ内で動かしているプログラムです。

内容を改変して使用することは問題ありませんが、動作の保障はしません。
また、いかなるトラブルが起こってもシステム工学研究会、及び、私（森久）
は責任を負いません。
メールサーバのログファイルにプログラムの動きが保存されるので標準出力は控え
めにしておいてください。

ブラウザから確認できるようにするために、インターネットに公開できるディレクトリ
にログを保存していきます。mail.pl 中では

```
.../public_html/mail/
```

というディレクトリに保存しています。もし、第3者に見られたくない場合は.htaccess フ
ァイルなどで閲覧制限を行ってください。

上記では.../public_html が公開されているとします。

まずは先ほどの ML 用のアカウント作成と同じように.qmail ファイルを作成しましょう。
ここではホームディレクトリ直下に「.qmail-log」というファイルを作成したとして話を進
めます。「.qmail-log」ファイルには以下のコマンドを記述しておいてください。

```
| cat | /usr/local/bin/nkf -e | mail.pl
```

 ファイルまでの絶対パス

意味は、メールのデータを標準出力し、それを nkf コマンドによって文字エンコードを
EUC コードにデコード。最後に mail.pl に標準入力で渡すというコマンドです。nkf コマ
ンドが使えない場合は削除しても OK ですが、その場合文字化けが起こる可能性がありま
す。またその他のオプションをつけて指定された文字エンコードに変更しても（たぶん）
問題ありません。これは使用する環境に合わせてください。

メールアドレスの作成ができればログを保存したい ML の.qmail ファイルに「&アカウ
ント名-log@ドメイン名」のメールアドレスを追加しておいてください。

さて、プログラムの動きの説明の前に、簡単にですがメールのデータフォーマットにつ
いて紹介しておきます。

ここで紹介するものはテキストメールのフォーマットです。そのため HTML メールであ
ったり、添付ファイルがあったりすると multipart メールになるので、フォーマットがちょ
っと違ってきますので注意してください。

メールクライアントによってはテキストメールと HTML の 2 つを送信するものがありま
す。mail.pl で対応することも可能ですが、今回は必要最低限のプログラムのみなのでプロ
グラミングの腕に自信がある方は実装してみてください。

まずメールデータにはヘッダと本文があります。その境目になるものは空行です。空行
の前の行まではヘッダ、後の行から末尾までは本文となります。またメールデータの文字

エンコードは特に指定されていないので、日本語のメール内容をプログラムで処理するときには特に気をつけましょう。そのメールがどの文字エンコードになっているかはヘッダの Content-Type: の charset を見れば書いてあります。

ヘッダ部分についてですが、各行は必ず

データ識別名:データ

というフォーマットになっています。たとえば、From: はメールの送り主のメールアドレス、Subject: はメールのタイトル、Date: は日付といった種類があります。ヘッダの処理にはこのフォーマットを利用するので覚えておいてください。

それでは本題の mail.pl について解説していきます。

・1行目は Perl のプログラムであることを表すので、各環境に合わせて perl コマンドまでの絶対パスを書いてください。

警告でも書いたように、このプログラム中で標準出力を行うと、メールサーバのログファイルに書き出されます。そのため、多量に標準出力を行うとログファイルが膨大になってしまうのでやめましょう。特にメールの内容を表示するなどはもってのほかです。

・3行目では Perl のプログラムを実行したという履歴を残す為に標準出力をしています。

GETTIME という関数は引数に変数を1つとり、その変数へ処理した時刻を代入します。フォーマットは年月日時分秒です。1桁の場合は0を先頭につけています。

ファイル名ですが、同名のファイルが作られると不完全なログになってしまうので、時刻を用いファイル名がかぶらないようにしています。保存はルートディレクトリの中にファイル名.html とファイル名.txt の2つを保存します。ファイル名.html はブラウザで見やすいように装飾するために使い、ファイル名.txt はテキストだけの情報を残すようにしているためです。もし.txt 版が必要ないならば該当部分をすべて削除してもらってかまいません。

・15行目ではあらかじめ用意しておいた HTML ファイルのヘッダ部分をファイル名.html に追加しています。head.txt のパスは絶対パスで書いておいてください。

・17-39行目は実際にメールデータをファイルに適切に書き出す部分です。基本的に、改行コードのみの行までが入力されるまではヘッダ部分の処理(elseの中)、改行のみが出てきたらそれ以後は本文を処理(ifの中)します。

プログラム順に説明すると、本文の処理はタグの無効化と改行文字の変換を行っています。RMTAG 関数は引数に文字列の変数を取ります。その変数の中に '<文字があれば <' に置き換える処理をします。これによりブラウザからの見た目は変わりませんが、タグを無効化することができます。また、改行文字である '\n' はブラウザで改行されないため
 タグに置き換えています。その後それぞれのファイルへ書き出しています。

次にヘッダ部分の処理ですが最初に空行かどうかを判定します。もし空行であればヘッダ部分が終了なので <div> タグを出力し次のループへ進みます。

空行でなければその行の最初のコロン(:)までを判定し各フォーマットにあったデータを取り出します。この処理は DS 関数によって行います。(名前は何でもかまいませんが、ここでは Data Select の頭文字をとりました)

tag1 と tag2 変数はスタイルシートで装飾しやすいように書いてあります。

if-elseif では index 関数を用いそれぞれ、Date: From: Subject: で始まっていないか確認し、もし始まっているならそれぞれを装飾します。

日付ならば、Date: を"日付"という文字列に変換します。

メールアドレスならば、From: を"メールアドレス"という文字列に変換し、@以下を(a)に変えています。

タイトルならば、Subject: を"タイトル"という文字列に変換し、RMTAG 関数でタグを無効化しています。

もしこれらのどのフォーマットでもない場合は変数を空にして DS 関数を終了します。

DS 関数で引数进行处理した後、空でなければファイルに書き出しています。

あとは今までのループを標準入力が終わるまで(メールの最後の行まで)行います。

・41 行目でファイル名.html にフッタ部分の情報を追加しています。

・43・44 行目でファイルをクローズして終わりです。それ以下は関数を記述してあります。

mail.pl の解説は以上です。動きを実際に確認するためには ML のメールアドレスにテストメールを送信してください。正常に動作すれば.../public_html/mail/ディレクトリの中に時刻.html と時刻.txt の 2 つのファイルが作成されているはずです。もし作成されていない場合は

- 1 . . .qmail ファイルに誤りはないか？
- 2 . . ./public_html/mail ディレクトリが存在するか？
- 3 head.txt や tail.txt が存在するか、またプログラム中に絶対パスで記述されているか？
- 4 . プログラムが実行できたか？ →メールサーバのログに"_Perl_時刻"という記述があるか？

などを確認してください。文字化けする場合は適宜文字エンコードを変えて試してみてください。

再度書いておきますが、第 3 者に見られたくない ML ならば.htaccessなどで閲覧制限をしておいてください。

以上で、ML の作成とログ取得についての紹介を終わります。閲覧していただきありがとうございました。

Mar/1/2008

森久 (marine)