# 「Tera Term」を使用した メールトラブルシューティング

2016年7月1日

株式会社グローバルネットコア
秋山祐介
<yusuke.akiyama@global-netcore.jp>

# ことの発端はサーバーのリプレイス

- ある顧客の専用サーバーをリプレイスしました。

  - OSがCentOS5系からCentOS6系に
  - 搭載しているMTAの構成は変更無し netqmail-1.06を使用
  - 切替え前後でのメール送受信テストで問題無し

# リプレイス後にトラブル発生！

ところが、切替え後にお客様から連絡が。

「メールマガジンのメールが
送信できないんだけど・・・」

# トラブル調査を開始したけれど・・

- 通常のメール送信では問題が無い
- メルマガのメール送信では、エラーコードが返る
- 出力されたログから資料を調べたが原因の特定できず

# トラブル調査を開始したけれど・・

/var/log/messages

---

kernel: qmail-smtpd[27367]: segfault at 10000000b ip 000000000041fad0 sp 00007fff379a5d48 error 6 in qmail-smtpd[400000+2b000]

---

qmail-smtpdのログ

---

tcpserver: pid 10960 from xxx.xxx.xxx.xxx
～中略～
tcpserver: end 10960 status 11

---

# 最後の手段 パケットキャプチャ

- お客様にご協力いただき、新旧メールサーバーに対して問題のメールマガジンを送信してもらいました。
- サーバー上でパケットキャプチャを実施

```
tcpdump -s0 -w /tmp/capture.dat -i eth0
```

※オプションの意味
-s0 ・・・・サイズ指定。ゼロは全て取得
-w ・・・・パケットをファイルに書き出す
-i ・・・・インターフェース指定

# データをWire Sharkで見てみる

## 旧サーバー（メルマガが正常に送れていた方）

| No. | Protocol | Length | src port | dst port | Info |
|---|---|---|---|---|---|
| 82 | SMTP | 81 | 587 | 27853 | S: 235 ok, go ahead (#2.0.0) |
| 83 | SMTP | 128 | 27853 | 587 | C: RSET \| MAIL FROM:[redacted] \| RCPT TO:[redacted] \| DATA |
| 84 | SMTP | 97 | 587 | 27853 | S: 250 flushed \| 250 ok \| 250 ok \| 354 go ahead |
| 85 | SMTP | 606 | 27853 | 587 | C: DATA fragment, 552 bytes |
| 86 | SMTP | 1294 | 27853 | 587 | C: DATA fragment, 1240 bytes |
| 87 | TCP | 54 | 587 | 27853 | submission→27853 [ACK] Seq=257 Ack=1954 Win=9472 Len=0 |
| 88 | SMTP | 68 | 27853 | 587 | C: DATA fragment, 14 bytes |
| 89 | TCP | 54 | 587 | 27853 | submission→27853 [ACK] Seq=257 Ack=1968 Win=9472 Len=0 |
| 90 | IMF | 433 | 27853 | 587 | from: =[redacted], subje |
| 91 | TCP | 54 | 587 | 27853 | submission→27853 [ACK] Seq=257 Ack=2347 Win=11904 Len=0 |
| 92 | SMTP | 82 | 587 | 27853 | S: 250 ok 1438736253 qp 29389 |

## 新サーバー（メルマガが送信できなくなった方）

| No. | Protocol | Length | src port | dst port | Info |
|---|---|---|---|---|---|
| 208 | SMTP | 81 | 587 | 31937 | S: 235 ok, go ahead (#2.0.0) |
| 209 | SMTP | 128 | 31937 | 587 | C: RSET \| MAIL FROM:<[redacted]> \| RCPT TO:<[redacted]> \| DATA |
| 210 | SMTP | 97 | 587 | 31937 | S: 250 flushed \| 250 ok \| 250 ok \| 354 go ahead |
| 213 | SMTP | 630 | 31937 | 587 | C: DATA fragment, 576 bytes |
| 214 | TCP | 54 | 587 | 31937 | submission→31937 [FIN, ACK] Seq=271 Ack=738 Win=15808 Len=0 |
| 215 | SMTP | 1294 | 31937 | 587 | C: DATA fragment, 1240 bytes |
| 216 | TCP | 54 | 587 | 31937 | submission→31937 [RST] Seq=271 Win |
| 217 | SMTP | 68 | 31937 | 587 | C: DATA fragment, 14 bytes |
| 218 | TCP | 54 | 587 | 31937 | submission→31937 [RST] Seq=271 Win= |
| 221 | IMF | 433 | 31937 | 587 | from: =[redacted], subj |
| 222 | TCP | 54 | 587 | 31937 | submission→31937 [RST] Seq=272 Win=0 Len=0 |
| 223 | TCP | 60 | 31937 | 587 | 31937→submission [FIN, ACK] Seq=2371 Ack=272 Win=16896 Len=0 |
| 224 | TCP | 54 | 587 | 31937 | submission→31937 [RST] Seq=272 Win=0 Len=0 |

本来[ACK]を返すところが
[FIN,ACK]を返している

# 原因は改行コード？

- [FIN,ACK]を送る直前のデータを調査
  - 本文の一部に、CRLFではなくLFのみの改行コードが含まれていることを発見

```
0110  74 65 6e 74 2d 54 72 61  6e 73 66 65 72 2d 45 6e   tent-Tra nsfer-En
0120  63 6f 64 69 6e 67 3a 20  37 62 69 74 0d 0a 43 6f   coding:  7bit..Co
0130  6e 74 65 6e 74 2d 54 79  70 65 3a 20 74 65 78 74   ntent-Ty pe: text
0140  2f 70 6c 61 69 6e 3b 0a  20 63 68 61 72 73 65 74   /plain;.  charset
0150  3d 22 69 73 6f 2d 32 30  32 32 2d 6a 70 22 0d 0a   ="iso-20 22-jp"..
0160  44 61 74 65 3a 20 57 65  64 2c 20 35 20 41 75 67   Date: We d, 5 Aug
0170  20 32 30 31 35 20 30 39  3a 35 37 3a 33 32 20 2b    2015 09 :57:32 +
0180  30 39 30 30 0d 0a 4d 49  4d 45 2d 56 65 72 73 69   0900..MI ME-Versi
```

# SMTPにおける改行コード

「RFC 5321 Simple Mail Transfer Protocol 」の規定

2.3.8. Lines

Lines consist of zero or more data characters terminated by the sequence ASCII character "CR" (hex value 0D) followed immediately by ASCII character "LF" (hex value 0A). This termination sequence is denoted as <CRLF> in this document. Conforming implementations MUST NOT recognize or generate any other character or character sequence as a line terminator. Limits MAY be imposed on line lengths by servers (see Section 4).

In addition, the appearance of "bare" "CR" or "LF" characters in text (i.e., either without the other) has a long history of causing problems in mail implementations and applications that use the mail system as a tool. SMTP client implementations MUST NOT transmit these characters except when they are intended as line terminators and then MUST, as indicated above, transmit them only as a <CRLF> sequence.

# SMTPにおける改行コード

「RFC 5321 」の規定（超意訳バージョン）

2.3.8. 行

行はゼロ文字か一文字以上のデータ文字で構成されるよ。
行はASCII文字のCR（16進で0x0d）の直後にASCII文字のLF（16進で0x0a）のふた文字で終了するよ。

～中略～

単独のCRやLFは、メールの実装でいろんな問題を起こしてきたなが～い歴史があるから、行の終わりは、CRLFのセットじゃなきゃいけないよ。

# qmailにおける改行コード

- qmail-smtpdにはLF(0x0a)単独の改行コードにエラーを返す仕様があるらしい

  - 以下qmail referenceより抜粋

「*qmail-smtpd は、SMTP の改行(CR LF)を、UNIX の改行(LF)に変換します。LF がCRの後ではなく単独で現れた場合、一時エラーとし、接続を切ります。*」

どうやら改行コードが
原因で間違いないようだ・・・

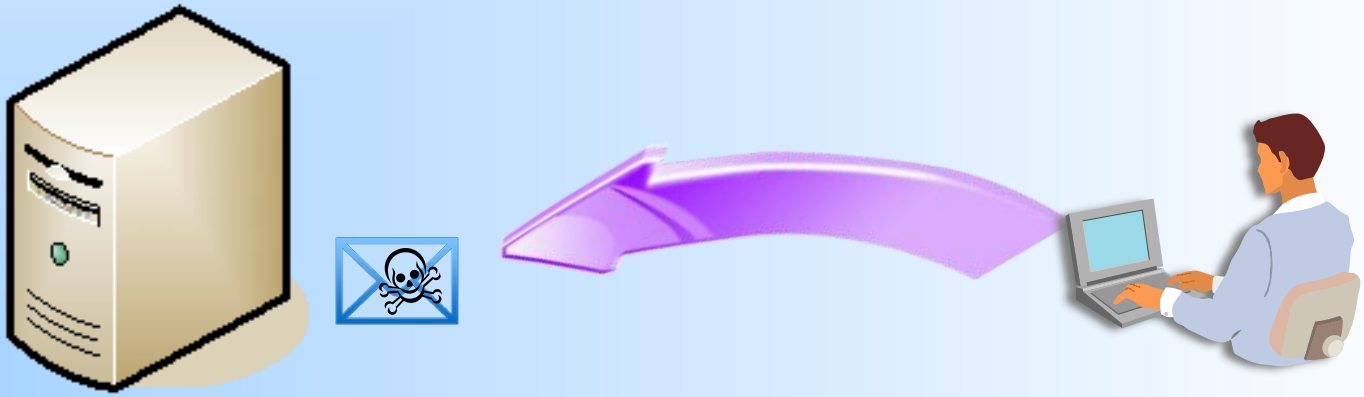
# 手がかりはつかんだけれども・・

- 対応を行うには現象を再現させなければいけない
  - メルマガ送信にはMailMagicPro 4というソフトを使用しているが、バージョンが古い模様
  - ダウンロードできる最新版では現象が再現しない

- ターミナルからtelnetコマンドでメール送信するのでは不正な改行コードを送信することが出来ない

# 現象を再現させるには？

不正な改行コードを含んだバイナリデータをSMTPで送信する必要がある！



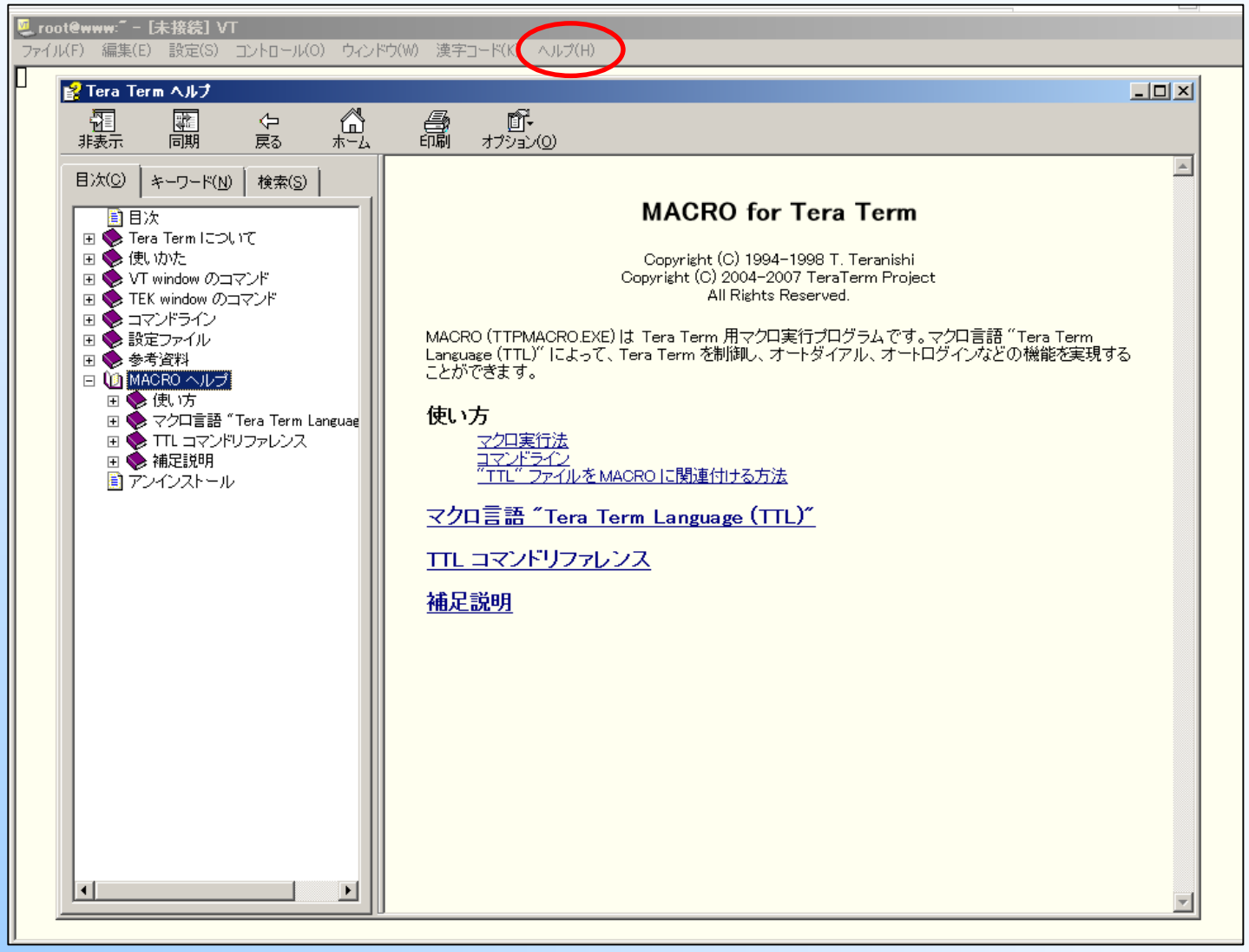
# 「Tera Term」のマクロ機能を使用してみた

- Tera Term には、マクロ機能というものがあり、マクロを使用してメールを送信できる

*「MACRO (TTPMACRO.EXE) は Tera Term 用マクロ実行プログラムです。マクロ言語 "Tera Term Language (TTL)" によって、Tera Term を制御し、オートダイアル、オートログインなどの機能を実現することができます。」*

→マクロ機能を使用してバイナリデータを直接送信することができた！

# 「Tera Term」のマクロ機能



# 「Tera Term」のマクロ機能でメール送信

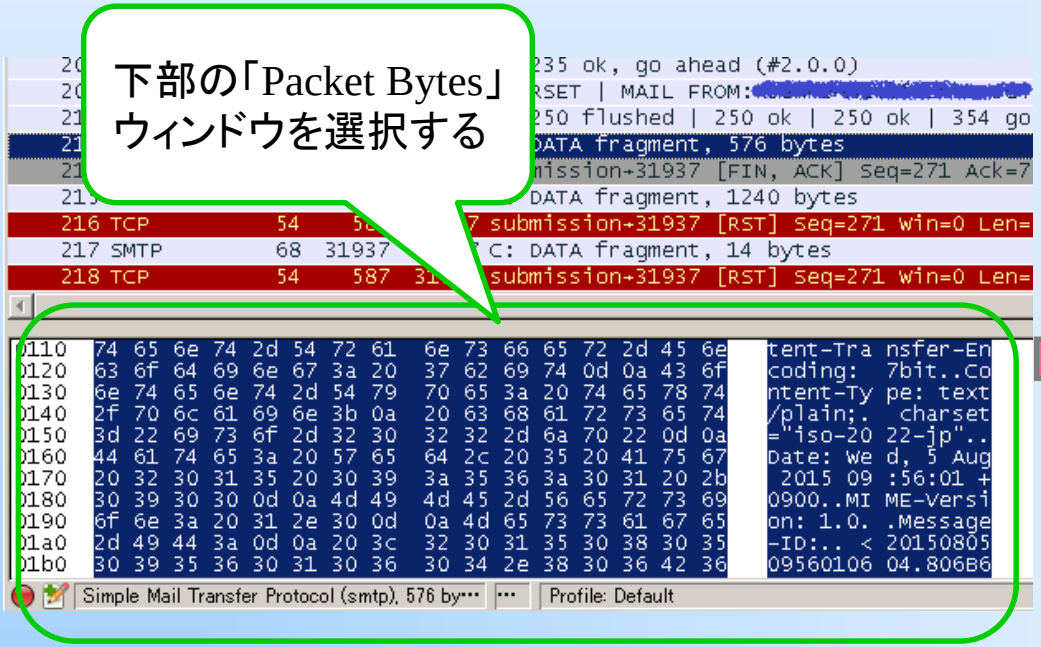
- 使用したマクロのコマンド
  - connect <command line parameters>
    - 宛先サーバにtelnet接続する
    - （実行例）*connect ‘xxx.xxx.xxx.xxx:25‘*
  - sendln <data1> <data2>....
    - データと改行を送信する
    - （実行例）*sendln ‘abc’*
  - sendfile <filename> <binary flag>
    - ファイルを送信する
    - <binary flag>が0 のとき、ファイルの中に含まれる漢字、改行文字を変換して送信する。
    - （実行例）*sendfile 'data.dat' 1*

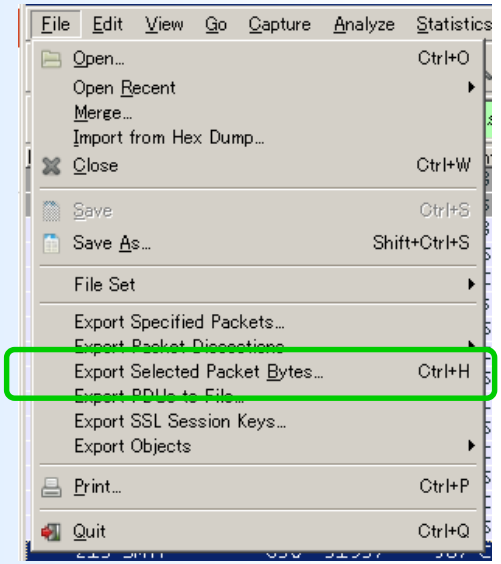
Flag 1は、バイナリをそのまま送信

# マクロ機能を使用したメール送信実行例

まずはWire Sharkでメール本文のバイナリデータを抽出

- 「Packet Bytes」ペインのデータを選択
- File > Export Selected Packet Bytesで保存



# マクロ機能を使用したメール送信実行例

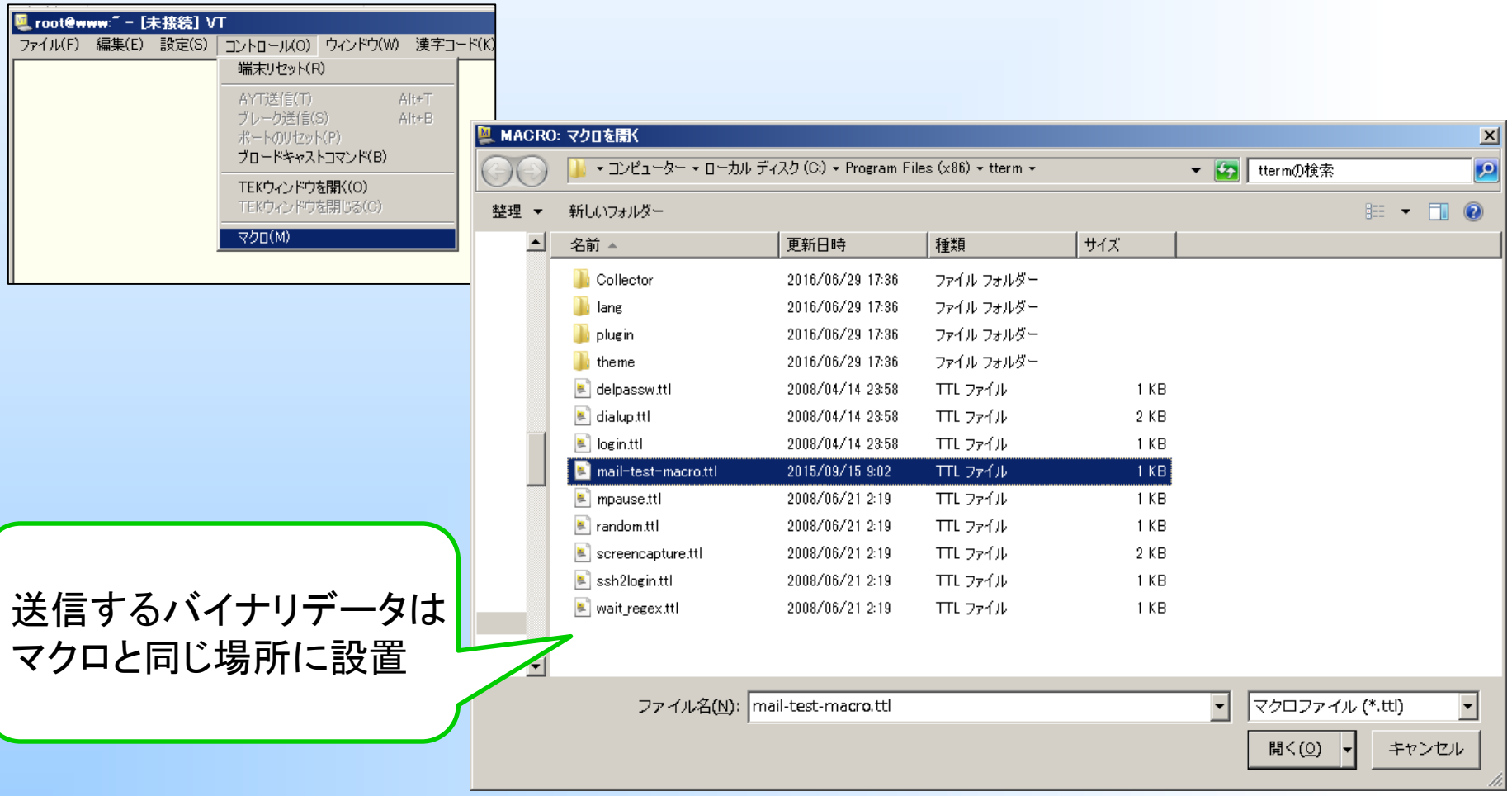
- メール送信を行うTera Termのマクロファイルを作成

```
;SMTPサーバーにコネクト
connect ‘aaa.bbb.ccc.ddd:25'
;EHLOコマンドを送信
sendln 'EHLO example.jp'
;送信元メールアドレス
sendln 'MAIL FROM: <testmail@example.jp>'
;送信先メールアドレス
sendln 'RCPT TO: <postmaster@example.jp>'
;送信データ
sendln 'DATA'
;バイナリファイルを直接送る
sendfile 'data.dat' 1
sendln '.'
sendln 'QUIT'
```

テキストエディタで
作成して、
拡張子「.TTL」で保存

# マクロ機能を使用したメール送信実行例

- 保存したマクロファイルをTera Termで実行




送信するバイナリデータは
マクロと同じ場所に設置

# トラブル対応の顛末

- 現象が再現できたことで、メールの送信試験が可能に。
- net-qmailのソース改修で対応を完了
  - 旧サーバではLFのみの改行コードでもエラーを返さないようにパッチを充てていた
    ※qmail-smtpd-newline.patch
  - 新サーバでもパッチは適用していたが、CentOS6系になったことでパッチが正常に機能していなかった
  - ソースの該当部分を改修してqmailの再インストールを実施

(ソースの修正内容)

```
- substdio_seek(ssin,-1);
+ substdio_seek(&ssin,-1);
```

# まとめ

- Tera Term のマクロを使うと、特殊なバイナリデータを送信することが可能になります。
  - メール以外にもtelnet接続できるプロトコルで利用可能
  - 脆弱性の再現テストとかにも利用できるかも

- マクロ機能は他にはこんな使い方も
  - 本来の使い方は、自動ログインです。
  - SSH多段接続の自動ログインも可能。
  - SCPを使用したファイル転送も自動化できます。

是非いろいろ試してみてください。

# ご清聴ありがとうございました