

# 株式会社 ITS MORE

2020年4月設立

ITS more

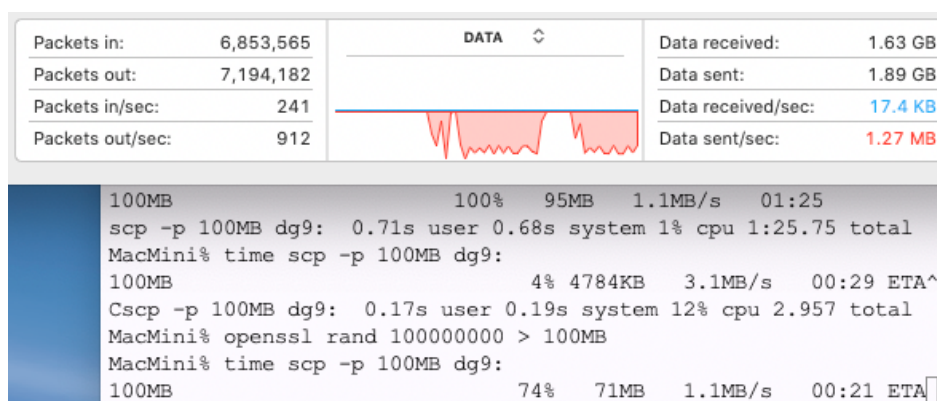
2020年8月4日 投稿者: SATOXITS

## macOS ssh upload の、呪い...

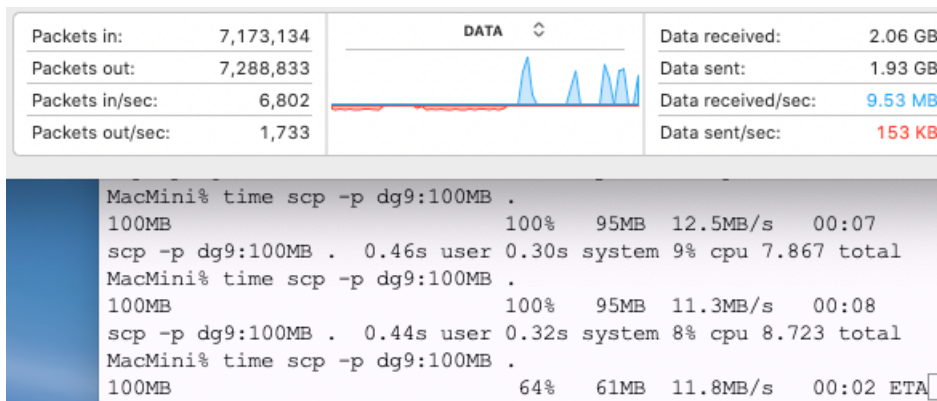
基盤：クラウドのサーバにscpでファイルをアップロードするのが異常に遅いのが気になっていました。

開発：なんか1MB/sくらいしか出ないとか。

基盤：で、どうやらそれは macOS 固有の問題なのでは無いかということがわかりました。まず、上りは1.1MB/s。



基盤：一方、下りは11MB/s。



基盤：これは、途中の回線が非対称なのか、サーバ側でチョークされてるのかと思っていたのですが、実はFTPなら両方とも11MB/s以上でることがわかりました。FTPサーバはDeleGate、クライアントはncftpです。

基盤：それでもしやと思い、Parallels上のUbuntuからも試してみました。

```

u18% time scp -p 100MB dg9:
100MB          100%  95MB 13.3MB/s  00:07

real    0m7.450s
user    0m0.522s
sys     0m0.378s
u18% time scp -p dg9:100MB .
100MB          100%  95MB 14.3MB/s  00:06

real    0m6.974s
user    0m0.405s
sys     0m0.330s

```

開発：対称ですね。

基盤：いったい何やってんですかね？iMacでも同様です。

```

iMac% time scp -p 100MB dg9:
100MB          100%  95MB  1.1MB/s  01:25
scp -p 100MB dg9: 0.70s user 0.58s system 1% cpu 1:25.81 total
iMac% time scp -p dg9:100MB .
100MB          100%  95MB 15.8MB/s  00:06
scp -p dg9:100MB . 0.41s user 0.28s system 10% cpu 6.291 total
iMac%

```

開発：まあ、tcpdump してみればわかるでしょうけど。

基盤：めんどくさいんですね。

社長：統計情報だけ見ればよいのですけどね。

開発：それもこれも、macOSのアクティビティモニタがおもちゃのタコだからいけない。

社長：それをどうにかしようというのが4月の予定でしたが...

基盤：/etc/ssh/ssh\_config とか ~/.ssh/ssh\_config とか特になにも設定してないようです。

基盤：ローカルなホストとでは、上り下り対称です。

```
MacMini% scp -p 100MB u18a:
100MB                               100%   95MB 109.1MB/s   00:00
MacMini% scp -p u18a:100MB .
100MB                               100%   95MB 111.4MB/s   00:00
MacMini% █
```

開発：1Gbps使い切ってますね。

社長：ひょっとしてsocketのバッファが小さいとかじゃないですかね。送受信でサイズが非対称なのはよくあることです。昔は数十KBの世界でしたが、最近では1MBなんていう設定も見ます。

開発：そういうシンプルな話ですかね... とりあえずDeleGateのtcprelayを通してみますか。まずDeleGateのソケットバッファって...

```

MacMini% idg9 -Fver
--
DeleGate/9.9.13+its01 (July 7, 2020)
Copyright (c) 2020 ITS more Co., Ltd.
++
AIST-Product-ID: 2000-ETL-198715-01, H14PRO-049, H15PRO-165, H18PRO-443
Copyright (c) 1994-2000 Yutaka Sato and ETL,AIST,MITI
Copyright (c) 2001-2014 National Institute of Advanced Industrial Science and Technology (AIST)
--
FTP: <URL:ftp://ftp.delegate.org/pub/DeleGate/>
WWW: <URL:http://www.delegate.org/>
Mail:<URL:mailto:feedback@delegate.org>
    <URL:http://www.delegate.org/feedback/>
_execPath=/Users/ysato/bin/idg9
Config: Darwin/19.6.0; FileSize-Bits=64/64,64/32,64,64; socket=131072/131072; sockpair=8192/8192,1016++U; char=signed; fcF=A; thread=PThread/pthread,88/128; zlib=112,308024; pam=2; stty=tcsetattr/sgtty; regex=regex; addr=A/10CE9CEA0/7FFEE2D92AE8; umem=3954/2048/0M
ADMIN=Author@DeleGate.ORG
DGROOT=/Users/ysato/delegate
Loaded: LibreSSL 2.2.7
Loaded: Zlib 1.2.11
HostId: 2020/05/02/13/12-6F (Darwin/19.6.0) its02-macmini.local
Usage: add -vv option to trace the library search
--

```

社長：I/Oともに131072バイト、つまり 0x20000バイトにしていますね。10年前では常識的なサイズだったようにも思います。

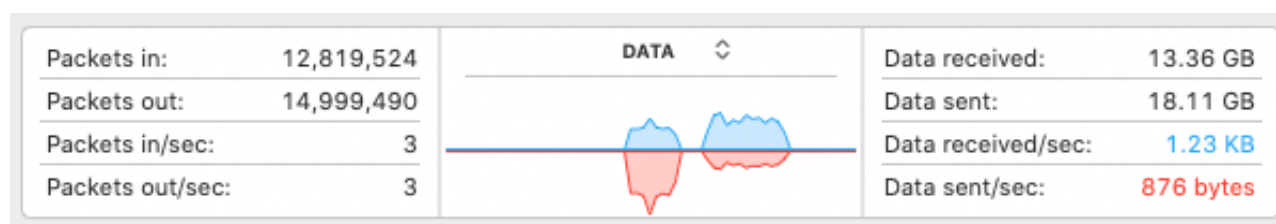
開発：ではこれで、idg9 -fv -P9022 SERVER=teprelay://jp1:22とかして、ローカルの9022ポートをdg9サーバのSSH/22につなぐ。でもって scp …

```

MacMini% scp -P 9022 1GB localhost:
1GB          100% 954MB 25.8MB/s 00:36
MacMini% scp -P 9022 localhost:1GB .
1GB          100% 954MB 15.8MB/s 01:00

```

基盤：驚愕の25倍速！



開発：なぜか送信だけ速くなりましたね。もういっちょトライ。



社長：送信パケット自体は60MB/s出てるようですね。いや、これってマシンローカルのパケットもカウントしてるのかな。だとすると値が2倍に出ますが。

開発：DeleGateでソケットバッファサイズを指定するのってどうするんですたっけ。

社長：SOCKOPTとかなんとか… マニュアルを見て下さい。

開発：… これですね。

```

buffsize:size[kiosc][:connMap]
  set the size of socket buffers. The size value can be followed with
  following specifiers:
    k -- size in kilo bytes
    i -- input buffer (SO_RCVBUF)
    o -- output buffer (SO_SNDBUF)
    s -- socket to server
    c -- socket from client
  For example, SOCKOPT="buffsize:2kis" set the size of the socket buffer to
  receive data from a server to 2048 bytes. Multiple buffer size
  specifications can be concatenated with "+". For example,
  SOCKOPT="buffsize:2k" is equivalent to
  SOCKOPT="buffsize:2kis+2kos+2kic+2koc:*:*:*".

```

社長：ああ、キオスクのだじゃれだったのを覚えています。

開発：-Fver には反映されないような。まあ、1MBにしてみますか。-vd  
SOCKOPT=buffsize:1024kic+1024kos を追加… DeleGateのログは…

```
08/04 10:09:15.17 [21637] 2+3: ConnectToServer: DFLT=tcprelay://jpl:22 REAL=://  
/:0  
08/04 10:09:15.17 [21637] 2+3: ConnectToServer connect tcprelay://jpl:22  
08/04 10:09:15.17 [21637] 2+3: connect[22] ready=1, err=0  
08/04 10:09:15.17 [21637] 2+3: ConnectToServer connected [22] {18.181.38.61:22  
<- 192.168.10.101:60336} [0.009s]  
08/04 10:09:15.17 [21637] 2+3: KeepAlive[22] = 8  
08/04 10:09:15.17 [21637] 2+3: setsockopt[22] in:0->0 out:132352->1048576  
08/04 10:09:15.17 [21637] 2+3: BUFSIZE-SV[22] 132352/132352 -> 132352/1048576  
(0/1048576) <= service.c:2748
```

基盤：拡張されてますね。

```
MacMini% scp -P 9022 1GB localhost:  
1GB          100% 954MB 27.6MB/s 00:34
```

基盤：劇的には変わらないですね。

開発：いや立ち上がりは劇的なんで、どっかで自動的に抑制するアルゴリズムが働いてるんじゃないですかね。250MBではどうか。

```
MacMini% scp -P 9022 250MB localhost:  
250MB          100% 238MB 32.0MB/s 00:07
```

社長：まあ、2割程度の変動はあると。

開発：でたとえばバッファサイズを8KBにすると、1MB/sになります。

```
MacMini% idg9 -P9022 SERVER=tcprelay://jpl:22 -fv -vd SOCKOPT=bufsize:8kos+8kic
```

```
MacMini% scp -P 9022 100MB localhost:  
100MB          100% 95MB 1.0MB/s 01:33
```

基盤：SSH自体が何か動的な制御をしている可能性もありますよね。

社長：しかし、TCPは立ち上がり時にバッファサイズを自動調整するのは知ってますが、全速巡航の時に何かするとは思えないんです。そこはまた何かのチョーク的なものが働いているのかも。

開発：この場合はDeleGateというユーザプロセスがバッファ機能を果たしていて、マ

ルチコアだからパイプラインが効くということかと思いますが、そもそもscpというかsshのソケットバッファを大きくすればよいのでは無いかと思いますね。オプションであるんですかね？

社長：ですが、この問題はssh/scpに限らないわけです。ソケットバッファのサイズを前時代のまま放置しているOSとかアプリとかライブラリっていくらでもあるのかなと。というか、ソケットバッファだけでなく、アプリのI/Oバッファが小さい場合も、システムコールは増えるし。

開発：対処法はこういうところですかね。

- カーネル内のバッファサイズ設定
- アプリ（ユーザプログラム）のバッファサイズ設定
- TCP中継サーバ（バッファリング・圧縮、明示的 / 透明）

社長：アプリでやる場合には、send / recv あたりを動的ライブラリでラップするのが良さそうです。

基盤：これはひょっとして、Macのシステム環境設定にあるのでは… Network > Advanced … それっぽいのは無いですね。

開発：あまり全域に影響が及ぶ設定だと、リソースが無駄になるとか、予測不能な副作用もあるんでしょうね。だから「遠隔との大規模ファイル転送」にTCPを使う、という場合にだけ適用できるようにする、自動判別もしくは明示指定、というのが良いのかなと思います。

社長：うーん、タイムスリッパでやったように動的ライブラリではできそうだけど、トランスペアレントプロキシ方式が面白そうです。これだと、既存のプログラムには何の影響も与えない。パイプライン動作とか、対向で使えば圧縮とかもできそう。この線を試してみたいですね。

経理：先程の固定電話は、鹿児島から黒酢ドリンクのセールスでした。今あちらでは気温が35度に達しているそうで、大変ですね。種子島のとっぴーの話とかで盛り上がりちゃいました。

基盤：随分長電話してましたが、長距離電話って今は安いんですかね。ネット電話？

社長：昼食行ってきます。

\* \* \*

社長：メロンソーダを飲みながら帰社であります。

開発：夏休みですね。

社長：いやー、外はまさに真夏。でも社内は涼しいですね。

基盤：ただいまの室温、31.6度。

経理：心頭滅却すれば火もまた涼し。

基盤：気休めにベランダに打ち水しておきます。

社長：それで今日は初めてのラーメンやさんに入りました。真鯛のスープ+えびすの小瓶。あれはハマリでしたね。

社長：それで食べながら考えたんですが、やはりトランスペアレントプロキシ方式はsuにならないと使えないのが嫌だなと。実現性がOSにも依存するし。それで、accept と connect を動的ライブラリでフックして、socket pairで中継用のスレッドかプロセスを挿入するのが良いかなと。

基盤：気化熱作戦、効かないですね。室温0.1度上昇。

開発：水道の水をラジエータに通すってというのはどうですかね。室温よりだいぶ低いでしょう。

基盤：東京都の水道水だと8月でも27~28度くらいのようです。

社長：井戸水があれば強力なのに。あれなら、15度くらいですよ。

\* \* \*



開発：しかしとりあえずscpが高速化できればいいや的なら、DeleGateみたいな中継器を立てて、そこにsshすれば良いので、今でも十分なわけです。ただちょっと使い勝手が悪いのは、複数のサーバを相手にしている場合に、ホスト名とかアドレスを変えるたびにいちいちknown\_hostsが一、って言うてくるところです。何なんですかね？

基盤：ユーザの鍵で認証しているのに、その上ホストの認証とか必要なんですかね？ユーザの鍵を盗まれた時点で詰んでるように思うんですが。

社長：鍵認証じゃない場合に必要なんじゃないですかね？あとはログイン後にsuとかでパスワードを入れる場合とか。

開発：現在のセッションのアカウントは認証されてるわけだから、その大元との間でワнтаイムパスワードみたいのを発行して、それを中間で入れられる、で良いように思いますね。

社長：まあ、ひとつのテキスト端末で全部やりとりしてるって想定でしょうからね。telnetならout-of-band的に、端末には表示されないメッセージをエンドツーエンドで交換して、クライアント側のウィンドウなりをポップアップするとかできそうですが。SSHはそういうのできるんですかね？

基盤：パスワードをユーザの鍵で暗号化してloginに渡すのでも良い気がします。

開発：そもそも「ホストにログインする」というホストって何？ログインて何？とも思うのです。まあたしかにsshはコマンドインタプリタとしてホストというものにログインする形態のソフトだから、そういう概念はあるんだと思いますが。リモートにコマンドを送るだけとか、とにかく端末からコマンドshellとか使わない場合も多いわけです。scpとかトンネルとか。

\* \* \*

開発：いやあ、opensshのビルドでつまづいて懲りました。が、わかったことは、scpでのアップロードが遅くなったのは、openssh-7.8からだということです。

```
MacMini% openssh-7.7p1/scp 100MB dg9:
100MB          100%   95MB  14.2MB/s   00:06
MacMini% openssh-7.8p1/scp 100MB dg9:
100MB          100%   95MB   1.1MB/s   01:25
```

開発：Catalina 10.15.6 の ssh は 8.1p1 です。

```
MacMini% uname -a
Darwin its02-macmini.local 19.6.0 Darwin Kernel Version 19.6.0: Sun Jul  5
 00:43:10 PDT 2020; root:xnu-6153.141.1~9/RELEASE_X86_64 x86_64
MacMini% ssh -V
OpenSSH_8.1p1, LibreSSL 2.7.3
```

開発：一方 Ubuntu18.04 の ssh は 7.6 だったのです。

```
u18% uname -a
Linux u18a 4.15.0-112-generic #113-Ubuntu SMP Thu Jul 9 23:41:39 UTC 2020
x86_64 x86_64 x86_64 GNU/Linux
u18% ssh -V
OpenSSH_7.6p1 Ubuntu-4ubuntu0.3, OpenSSL 1.0.2n  7 Dec 2017
```

社長：OpenSSLの版の影響は無いですか？

開発：同じ OpenSSL 1.0.11 での比較です。

```
MacMini% openssh-7.7p1/ssh -V
OpenSSH_7.7p1, OpenSSL 1.0.11 15 Jan 2015
MacMini% openssh-7.8p1/ssh -V
OpenSSH_7.8p1, OpenSSL 1.0.11 15 Jan 2015
MacMini%
```

開発：Linuxでも状況は同じでした。

```
u18% uname -a
Linux u18a 4.15.0-112-generic #113-Ubuntu SMP Thu Jul 9 23:41:39
  UTC 2020 x86_64 x86_64 x86_64 GNU/Linux
u18% openssh-7.7p1/ssh -V
OpenSSH_7.7p1, OpenSSL 1.0.2n  7 Dec 2017
u18% openssh-7.7p1/scp ~/100MB dg9:
100MB          100%   95MB  11.3MB/s   00:08
u18%
u18% openssh-7.8p1/ssh -V
OpenSSH_7.8p1, OpenSSL 1.0.2n  7 Dec 2017
u18% openssh-7.8p1/scp ~/100MB dg9:
100MB          100%   95MB   1.1MB/s   01:25
```

開発：ただよくわからないのは、openssh のコードを見ても、ソケットバッファのサ

イズをいじってないことです。となると、バッファがうまる前にソケットをflushというかpushして細切れパケットを発生させているのかも知れません。送信の間にコンテキストスイッチみたいな遅延が入ってるとか。

社長：うーむ、面白いですね。diff を見るのが楽しみです。ちょっとひとやすみしましょう。飲んできます。

— 2020-0804 SatoxITS