

# 株式会社 ITS MORE

2020年4月設立

ITS more

2020年6月30日 投稿者: SATOXITS

## 蜜壺アリ

開発：多数のVMを作って全員で互いのディスクを共有させようという作戦ですが。

基盤：同じリージョン内のライトセールVM間では50MB/sでsshできることが、引越しのファイル転送でわかっています。これはこの2.4GHzというCPUで乱数列を作るのと同様な速度です。確認しますと、まずCPUはコレ。

```
awsLS% dmesg | grep CPU0
[    0.280073] smpboot: CPU0: Intel(R) Xeon(R)
CPU E5-2676 v3 @ 2.40GHz (family: 0x6, model:
0x3f, stepping: 0x2)
```

基盤：乱数データの生成と転送性能の測定。

```
awsLS% time openssl rand 100000000 > 100MB
real    0m2.230s
user    0m2.080s
sys     0m0.064s
awsLS% time scp -p 100MB localhost:100MB-l
100MB          100%  95MB  95.4MB/s   00:01
real    0m1.355s
user    0m0.456s
sys     0m0.156s
awsLS% time scp -p 100MB pachimon.shop:100MB-r
100MB          100%  95MB  95.4MB/s   00:01
real    0m2.010s
user    0m0.460s
sys     0m0.208s
```

開発：100MB/s出てますね。隣のホストとでも。

基盤：SSH接続のオーバーヘッドだけの違いくらいしか。

```
awsLS% cat /dev/null > 0B
awsLS% time scp -p 0B localhost:0B-l
0B          100%    0      0.0KB/s   00:00

real    0m0.114s
user    0m0.008s
sys     0m0.000s
awsLS% time scp -p 0B pachimon.shop:0B-r
0B          100%    0      0.0KB/s   00:00

real    0m0.585s
user    0m0.008s
sys     0m0.000s
```

開発：まあSSHの接続に1秒近くかかるのはよくあることですね。

基盤：あるいは、gzip で圧縮するスピードより速い。まあ乱数列には圧縮は効かないですから、最悪の場合ですけど。

```
awsLS% time gzip < 100MB > /dev/null

real    0m3.542s
user    0m3.512s
sys     0m0.020s
awsLS%
```

開発：そういうことか。zlibで圧縮転送すると、逆にそのために最悪30MB/sに律速されるかも知れないと。

社長：100MB/sっていうと、ローカルHDDと同じ性能が出るということですね。

開発：それじゃファイル共有してみまじょうか。とりあえずNFSですかね。っていってもすっかりやり方を忘れてしまいました。

社長：ちょっとコーヒブレークを。 … しまった！コーヒをばら撒いてしまった！

開発：挽いた豆ってアロマティックというか、大した量でなくてもすごい存在感という  
 か有り難みがありますよね。お湯を通して出しガラになったらゴミですが。

基盤：それで、本当に働きアリだけの集団で良いのか、一つミツツボアリを作るのか、  
 という事は検討課題かなと思います。ちょっとお勧めプランを… あ、なんかポップア  
 ップした。



開発：どうやって気づいたんでしょうね。しかも今頃。3日前に作った時に一瞬だけ動  
 かしたただけなんですけど。

基盤：やっぱり、ストレージに関してはライトセールもお得感が無いですね。

### インスタンスプランの選択 ?

**新規!** 新しい 16 GB および 32 GB RAM のバンドルをご確認ください!

並べ替え: 月次料金 メモリ 処理中 ストレージ 転送

	\$5 USD	\$10 USD	\$20 USD	\$40 USD	\$80 USD	
	\$5 USD	\$10 USD	\$20 USD	\$40 USD	\$80 USD	月次料金
	1 GB	2 GB	4 GB	8 GB	16 GB	メモリ
	1 vCPU	1 vCPU	2 vCPU	2 vCPU	4 vCPU	処理中
	40 GB のSSD	60 GB のSSD	80 GB のSSD	160 GB のSSD	320 GB のSSD	ストレージ
	2 TB	3 TB	4 TB	5 TB	6 TB	転送

開発：やはりクラウドの安いドライブをマウントして使うのかな。あるいは1台でrsyncして、それを仲間にマウントさせるとか…

社長：使い方にあった美味しい使い方ができるの良いですね。

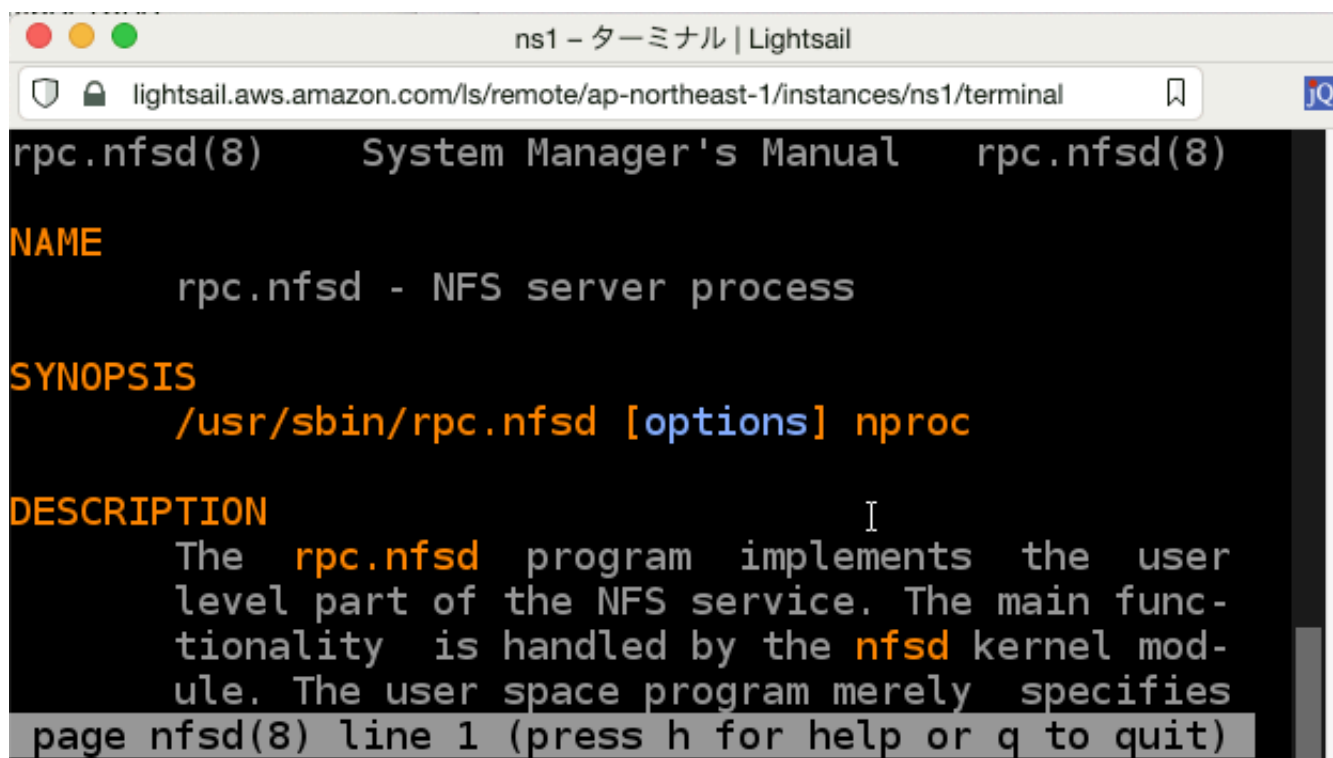
開発：とにかくNFSから始めましょう。man nfsd … あ、無いの。nfsd …

```
awsLS% nfsd
No command 'nfsd' found, did you mean:
  Command 'nsd' from package 'nsd' (universe)
  Command 'afsd' from package 'openafs-client' (universe)
  Command 'nnsd' from package 'tcclib' (universe)
  Command 'nasd' from package 'nas' (universe)
  Command 'gfsd' from package 'gfsd' (universe)
nfsd: command not found
```

開発：apt search nfs … なんだか本体が無いような。

基盤：そういえば試しに作ったきりにしてあるAmazon Linuxってどうなんですかね。

開発：ブラウザでターミナルできるというこのお手軽さ… あれ、nfsd あるじゃん。



```
rpc.nfsd(8)      System Manager's Manual      rpc.nfsd(8)
NAME
  rpc.nfsd - NFS server process
SYNOPSIS
  /usr/sbin/rpc.nfsd [options] nproc
DESCRIPTION
  The rpc.nfsd program implements the user level part of the NFS service. The main functionality is handled by the nfsd kernel module. The user space program merely specifies
page nfsd(8) line 1 (press h for help or q to quit)
```

基盤：こういうところ、一事が万事だったりして。

社長：うちは別にUbuntuもBitnamiも要らないんで、Amazon Linux に身を委ねるのでも良いです。

基盤：ちょっと調査。cc … ないですね。apt search cc。おっと、apt がない。man -k package … RPM と出ました。回転毎分ですかね。

開発：RPMってRedHatだよな。Amazon Linux ってそっち系だったのか。man rpm。ふーん。rpm -q -a | grep cc。無いですよ。これ、apt get 的に探して落とすにはどうするんですかね。

基盤：この、yum っていうのがそうじゃないですか？man yumによると「」うわ、Vivaludiが固まった。ウェブ端末からコピペしようとしただけなんです。… あ、収まりました。この8GB RAMのMacMini、もう終わりだね君が遠くに見える。

経理：iMac到着して4日間。箱に入ったままです。

基盤：man yumによると「yum is similar to other high level package managers like apt-get and smart.」だそうです。

社長：確かに yum というのは昔、聞いた事も使ったこともあるような気がします。

開発：aptと同じですか。じゃ sudo yum install gcc … 「Complete!」。で cc。OK。似たようなものですね。ついでに sudo yum install golang … おー、やってるやってる。Complete!。ではおもむろにHello World…

```
AmaLin% cat hello.go
package main
import "fmt"
func main(){
    fmt.Println("Hello World!!");
}
AmaLin% time go run hello.go
Hello World!!

real    0m0.212s
user    0m0.171s
sys     0m0.027s
AmaLin% time go build hello.go

real    0m0.060s
user    0m0.047s
sys     0m0.012s
AmaLin% ls -l hello
-rwxrwxr-x 1 ec2-user ec2-user 2008640 Jun 30 11:35 hello
AmaLin% time ./hello
Hello World!!

real    0m0.001s
user    0m0.001s
sys     0m0.000s
```

開発：なんですかねこれ、Hello World が 1msで。なんでそんなに軽いのかよ〜♪って。Ubuntu VM でも試しましょう。

```
Ubu16% cat hello.go
package main
import "fmt"
func main(){
    fmt.Println("Hello World!!");
}
Ubu16% time go run hello.go
Hello World!!

real    0m0.453s
user    0m0.304s
sys     0m0.048s
Ubu16% time go build hello.go

real    0m0.395s
user    0m0.360s
sys     0m0.028s
Ubu16% ls -l hello
-rwxrwxr-x 1 ysato ysato 2288320 Jun 30 20:43 hello
Ubu16% time ./hello
Hello World!!

real    0m0.001s
user    0m0.000s
sys     0m0.000s
```

社長：うーむ、2倍速？ユーザ時間からして2倍近く違う。本当に同じハードですよ？

```
AmaLin% uname -o -r -v
4.14.181-108.257.amzn1.x86_64 #1 SMP Wed May 27
02:43:03 UTC 2020 GNU/Linux
AmaLin% dmesg | grep CPU0
[    0.216080] smpboot: CPU0: Intel(R) Xeon(R)
CPU E5-2676 v3 @ 2.40GHz (family: 0x6, model: 0
x3f, stepping: 0x2)
AmaLin% top -n 1 | grep ^Mem
Mem: 1009136k total, 774308k used, 23482
```

```
Ubu16% uname -o -r -v
4.4.0-1109-aws #120-Ubuntu SMP Fri Jun 5 01:26:
57 UTC 2020 GNU/Linux
Ubu16% dmesg | grep CPU0
[    0.280073] smpboot: CPU0: Intel(R) Xeon(R)
CPU E5-2676 v3 @ 2.40GHz (family: 0x6, model: 0
x3f, stepping: 0x2)
Ubu16% top -n 1 | grep Mem
KiB Mem : 1014416 total, 162512 free, 183
```

開発：まあ Ubuntu 16版は運用に入ってますから、サーバ動いててメモリは使ってます。でも十分空いてますよね…。CPUは全く同じ。なにしろ利用料金は同じ500円/月ポッキリです。

社長：一応確認のため、まっさらのUbuntu版を作って、性能を確認して、こういう性

能差であるなら、Amazon Linux にしましょう。

基盤：それがもう、バカみたいに簡単なんですよねw。ライトセールコンソールから、インスタンスの作成。OSのみでUbuntu選択。\$5プランを選択。そして作成をぷちっと。この保留中にでる「利用不可」という表現はなんだか不吉なイメージなのでやめてほしいですね。あ、できました、実行中。1分ちょっとでしたか？で、このコンソールアイコンをクリックすると端末が開く。以上です。

開発：では、sudo apt install golang … あれ？これ Unknown とかエラーが。というか Ubuntu 18 じゃないですかこれ。apt update して、再度 apt install golang。やってるやってる。なんかせっかく apt にも馴染んだのにもうお別れですか。

社長：いや、Ubuntuもまだ使うでしょ。

開発：ではお手並み拝見。

```
Ubu18% uname -o -r -v
4.15.0-1021-aws #21-Ubuntu SMP Tue Aug 28 10:2
3:07 UTC 2018 GNU/Linux
Ubu18% dmesg | grep CPU0
[    0.120087] smpboot: CPU0: Intel(R) Xeon(R)
CPU E5-2676 v3 @ 2.40GHz (family: 0x6, model:
0x3f, stepping: 0x2)
Ubu18% top -n 1 | grep Mem
KiB Mem : 1007540 total, 334648 free, 9
```

```
Ubu18% cat hello.go
package main
import "fmt"
func main(){
    fmt.Println("Hello World!!");
}
Ubu18% time go run hello.go
Hello World!!

real    0m0.212s
user    0m0.194s
sys     0m0.018s
Ubu18% time go build hello.go

real    0m0.063s
user    0m0.051s
sys     0m0.012s
Ubu18% ls -l hello
-rwxrwxr-x 1 ubuntu ubuntu 2011612 Jun 30 12:2
2 hello
Ubu18% time ./hello
Hello World!!

real    0m0.001s
user    0m0.000s
sys     0m0.001s
```

基盤：Amazon Linux と同等のようですね。

開発：うーむ、こんだけメモリが小さくてシングルコアという世界は最近足を踏み入れてなかったの、見当が狂います。

基盤：では一応、まっさらの Ubuntu 16 でも確認しましょう。…。はい、できました。

開発：では apt update して apt install golang で hello.go。あれ？やっぱり遅い。あれ？マウスが動かなくなった…

社長：ああ、それもう2ヶ月前から電池切れになっては手で温めて復活させてましたからね。

開発：ボイジャーですか。

社長：電池交換して、一服しましょう。



\* \* \*

基盤：ていうか、せっかくだから Ubuntu 20.04 にしたいですね。

開発：まあ、いずれライトセールからも出るでしょう。

社長：実家から送ってきた地酒ですグビ。

開発：それでは気を取り直しまして再試験。

```
Ubu16% uname -o -r -v
4.4.0-1052-aws #61-Ubuntu SMP Mon Feb 12 23:05:58 UTC 2018 GNU/Linux
Ubu16% dmesg | grep CPU0
[    0.268082] smpboot: CPU0: Intel(R) Xeon(R) CPU E5-2676 v3 @ 2.40GHz (fam
ily: 0x6, model: 0x3f, stepping: 0x2)
[    1.626281] microcode: CPU0 sig=0x306f2, pf=0x1, revision=0x43
Ubu16% top -n 1 | grep Mem
KiB Mem : 1014540 total, 205636 free, 55432 used, 753472 buff/cache
KiB Swap: 0 total, 0 free, 0 used, 782128 avail Mem
Ubu16% cat hello.go
package main
import "fmt"
func main(){
    fmt.Println("Hello World!!");
}
Ubu16% time go run hello.go
Hello World!!

real    0m0.327s
user    0m0.280s
sys     0m0.040s
Ubu16% time go run hello.go
Hello World!!

real    0m0.334s
user    0m0.308s
sys     0m0.020s
Ubu16% time go build hello.go

real    0m0.398s
user    0m0.352s
sys     0m0.036s
Ubu16% ls -l hello
-rwxrwxr-x 1 ubuntu ubuntu 2288312 Jun 30 12:50 hello
Ubu16% time ./hello
Hello World!!

real    0m0.001s
user    0m0.000s
sys     0m0.000s
```

基盤：やっぱりすっからかん状態でも50%ばかり遅いですね Ubuntu 16。Goの固有の問題？

開発：さて、何を基準に。

社長：うちのには25年間、コンパイルの速度を重要なベンチマークにして来ました。それまでUnixワークステーションでやってたコンパイルをWindows PCのVC5だかでコンパイル出来た時には感動したものです。当時はそもそも各社のコンパイラを通るとい

う事自体がとても大変なことでした…

開発：gccも警告のレベルが変わったりすると速度が色々みたいですが… とりあえずDeleGateのコンパイルで。

```
real 1m40.285s
user 1m27.472s
sys 0m5.472s
Ubu16%
```

```
real 1m48.868s
user 1m38.127s
sys 0m9.352s
Ubu18%
```

開発：あれ、Amazon Linux C++が無い… yum search gcc … これですかね sudo yum install gcc64-c++.x86\_64。さて make…

```
real 1m42.834s
user 1m33.446s
sys 0m8.052s
AmaLin%
```

開発：コンパイルについては有意差が無い、ということですかね。

基盤：乱数生成も同等。これ、以前一瞬、Ubuntu16のVMが4倍くらい速いことがあったんですが、あれは何だったんでしょうね？

```
Ubu16% time openssl rand 1000000000 > /dev/null
real    0m20.263s
user    0m20.224s
sys     0m0.028s
```

```
Ubu18% time openssl rand 1000000000 > /dev/null
real    0m22.471s
user    0m22.273s
sys     0m0.188s
```

```
AmaLin% time openssl rand 1000000000 > /dev/null
real    0m21.368s
user    0m21.321s
sys     0m0.040s
```

基盤：ディスクI/Oも大差無しです。

```
AmaLin% time openssl rand 1000000000 > 1GB
real    0m22.465s
user    0m21.522s
sys     0m0.591s
AmaLin% time cat 1GB 1GB 1GB 1GB > 4GB
real    1m1.671s
user    0m0.021s
sys     0m3.424s
AmaLin% time cat 4GB > /dev/null
real    1m1.503s
user    0m0.000s
sys     0m1.302s
```

```
Ubu16% time openssl rand 1000000000 > 1GB
real    0m21.294s
user    0m20.348s
sys     0m0.656s
Ubu16% time cat 1GB 1GB 1GB 1GB > 4GB
real    1m1.510s
user    0m0.008s
sys     0m3.660s
Ubu16% time cat 4GB > /dev/null
real    1m0.035s
user    0m0.000s
sys     0m1.724s
```

```
Ubu18% time openssl rand 1000000000 > 1GB
real    0m24.156s
user    0m22.952s
sys     0m0.864s
Ubu18% time cat 1GB 1GB 1GB 1GB > 4GB
real    1m1.560s
user    0m0.010s
sys     0m3.974s
Ubu18% time cat 4GB > /dev/null
real    1m1.522s
user    0m0.000s
sys     0m1.501s
```

開発：4GBを60秒ってことは、おおよそ70MB/秒ですね。ディスクとも親しい中だと。

基盤：一方、退役する旧 [delegate.org](http://delegate.org) のスペックです。Ubuntu 18.04。

```
AzuUbu% uname -a
Linux itsv20az2 5.3.0-1031-azure #32~18.04.1-Ubuntu SMP Mon Jun
 22 15:27:23 UTC 2020 x86_64 x86_64 x86_64 GNU/Linux
AzuUbu% dmesg | grep CPU0
[ 0.704532] smpboot: CPU0: Intel(R) Xeon(R) CPU E5-2673 v4 @
 2.30GHz (family: 0x6, model: 0x4f, stepping: 0x1)
AzuUbu% top -n 1 | grep Mem
KiB Mem : 910992 total, 59988 free, 412612 used, 4383
```

開発：クロックが2.3GHzとか、メモリも微妙にケチですね。

基盤：でディスクI/Oですが。アクセス量で従量課金されたりしないか一抹の不安を覚えつつベンチ…

```
AzuUbu% time openssl rand 1000000000 > 1GB
real    0m56.643s
user    0m3.617s
sys     0m1.104s
AzuUbu% time cat 1GB 1GB 1GB 1GB > 4GB
cat: write error: No space left on device
real    1m41.051s
user    0m0.020s
sys     0m2.344s
AzuUbu% time cat 4GB > /dev/null
real    0m52.669s
user    0m0.017s
sys     0m0.964s
```

開発：なんか No space left 出てますが（笑）

基盤：あれ確かに… Azure VM、30GBしか無いですからねえ。まあこのVM、もうすぐ廃棄だから何を切ってもいいっちゃいいんですが…

社長：いやその30GBに、8年間苦しんできたのです私は。しかも月1万円払ってですよ。これら飲まずにいられるかつ一の。

基盤：ちょっと刻んで行きます。まず 1GB の乱数発生。

```
AzuUbu% time openssl rand 1000000000 > /dev/null  
  
real    0m3.775s  
user    0m3.644s  
sys     0m0.061s
```

開発：なるほど。前々からこのベンチはとてもデリケートで面白いですね。6倍の性能差がサクッと出る。

基盤：というか、ローカルなメモリリッチマシンではこんな感じなわけですよ。さらにドンと6倍。合わせて36倍ですよお客さん！

```
MacMini% time openssl rand 1000000000 > /dev/null  
  
real    0m0.594s  
user    0m0.586s  
sys     0m0.006s
```

経理：私はあれをジャパネット銀行としか読めないんです（笑）。最初はぶっきらぼうな奴と思っていましたが、最近は頼りにしています。

社長：私は今、バイっていう字が倍という字体だということに、今とても違和感を感じています。本当にこれで正しいのでしょうか？あと、英語の by との関係が気になります。

開発：そうこうするうちに西暦2020年6月もあと30分で終わりですね。

経理：7月10日までにあれを提出しないと、懲役6ヶ月もしくは罰金50万円だそうです。

社長：ちょっと小腹もすきました。今日は目先を変えて緑のたぬき、これに生卵をトッピングで。

基盤：私はあの加薬だけ、特にフリーズドライのネギ。あと、レトルトのもやしを単独で売ってたら、買いますね。で、結果です。

```
AzuUbu% time openssl rand 200000000 > 200MB
real    0m5.309s
user    0m0.669s
sys     0m0.222s
AzuUbu% time cat 200MB 200MB 200MB 200MB 200MB > 1GB
real    1m3.840s
user    0m0.005s
sys     0m1.427s
AzuUbu% time cat 1GB > /dev/null
real    0m36.018s
user    0m0.000s
sys     0m0.684s
```

開発：あー、1GB書くのに1分超って、15MB/秒程度ってところですね。あん、びりーばー！

社長：いずれにしてももう、我々はもう出Azureの旅に出たのです。

\* \* \*

経理：もうお湯をそそいで6分くらい経ったような。

社長：いや、歳をとるとやわいのが好きになるんですよね。やらかいですかー、ダァ！ずずっ。あれ？

基盤：粉末スープ入ってないですね。いや、粉末の液体って変だなと思いますが。

社長：あーこれこれ。一瞬、最近味がしなくなるというアレかと思いましたよ。ズズッ。

開発：これ、となりのそば屋より美味しいですよ。ずずっ。

社長：いや、何事においてもレギュラーとインスタントは別のジャンルですからね。比較は難しいです。ズズッ。

経理：生卵を落としたからにはスープまで完食しないと。ずずっ。

基盤：この天ぷらもどきをガリガリ感で食したいという意味がわからないですね。天かすは出し汁吸ってふによふによに限ります。ズズッ。

開発：そういえば昔、1杯のかけそばといショートストーリーが流行りました。当時は興味なかったんで読まなかったのですが。しばらく前のあの騒ぎの時に、ネタかも知れませんが、LCCでやってきてそば屋か何かで分け合って食べてるという話を読んで、あれを思い出しました。ホロッと来てしまいましたね、ズズ。

基盤：お笑いのよしもとが不祥事で謹慎状態の中、巧まざるお笑い提供。で私はあの騒ぎで、自分のお気に入りのボールペンがジェットストリームというちょっと恥ずかしい名前のものであったって知ったんです。ズッ。

社長：ズー。スープまで完食いたしました。

基盤：日本人的にはインスタント味噌汁が理想かなと思います。具たくさんでケンチン的で150円くらいな。里芋と油揚げとこんにゃくとネギ、フリーズドライ+レトルト的にとてもマッチした組み合わせのように思います。

開発：それって要するに豚汁？

\* \* \*

開発：それで本日の当社の結論としましては、コスパ的にAzureはもう無い、アマゾンライトセールならどれでも良い。RedHat系Amazon Linuxも面白いかも知れない。そういうことですね。

社長：そうですね。

— 2020-0630 SatoxITS