



よしだともこの Linux 事始めの書

第9回 UNIXコマンド使いへの道

その7 カレントディレクトリにパスを通すとなぜ危険か

めっきり春らしくなりましたね。My Gardenのチューリップの芽が、着実に伸びているのを見るたびに、美しい花をつける日を想像してワクワクしています。

よしだともこ <http://www.tomo.gr.jp/>

My “Happy ハッカー好き Life”

このところのマスコミによる「ハッカー」という言葉の誤用について、みなさんはどう思われます？ 私は「クラッカー」という言葉が用いられるべき文脈で「ハッカー」が悪者を表す言葉として用いられているのを見るたびに、悲しい気分になっています。ま、これまでもUNIX使いとして、「うーん、これってちょっと悲しい……」と思うことは、多々あったんですけどね。

例えば、ネットサーフィンすることが「インターネットする」と表現されることとか、HTML形式で本文を送るようなメールを使ったメールが、ときにはLinux関係のメーリングリストに流れてしまうこととか……。あと、最近、携帯電話のiモードで自分のWebページを見たら文字化けしていたのでiモードのマニュアルを調べたら、「iモード対応のページは、日本語はソフトJISで書くこと」という意味の説明が書いてあったのも、ちょっと悲しい……。でも、こういうものも、「ハッカー」の誤用に比べれば、かわいいものに思えてきます。

私は真のハッカーを非常に尊敬してるんですよ。よく、少女マンガなんか、

『 に夢中になっている 君って すてき 』

みたいなセリフがあるじゃないですか。真のハッカーを見ると、なんかそういう気持ちになるんで……。それで、お近づきになりたい一心で、共通の話ができるぐらいの知識を身に付けようと、無理して技術雑誌を読んだり、取材記事を書いてたりしてる私なのです。

ここで、真のハッカーをきちんと定義しておきたいなあ……。と、こちょこちょ調べていたら、すばらしいWebページ

「hacker is *not* cracker!」がありました(記事末RESOURCE [1 参照])。必要なところには、ばっちりリンクが張られています。また、ハッカーについてとことん扱った本、「ハッカーズ大辞典」もあります[2]。

この本の「ハッカー」の用語説明は、次の通りです。

ハッカー(hacker)ももとは、斧で家具を製作する人]

プログラム可能なシステムの細かい部分を探ったり、その機能を拡張する方法を探究したりするのに喜びを感じる人。必要最低限のことしか勉強しながら大半のユーザーとは対照的。

熱中して(さらには取りつかれたように)プログラミングする人、またはプログラミングを単に理論化するのではなく、プログラミングを楽しむ人。

ハック価値(hack value)を評価できる人。

手早くプログラミングするのが得意な人。

ある特定のプログラムのエキスパート、または頻繁にそれを使って仕事をする人。たとえば「a UNIX hacker」(語義 ~ は相互に関連しており、これらにあてはまる人は集団をなす)。

あらゆる種類のエキスパートまたは熱狂的なファン。たとえば天文ハッカーなどと言う。

創意工夫を発揮して制約を打破したり回避したりするという知的な難問を楽しむ人。

[誤用] あちこち調べ回って機密情報を探り出そうとする悪意の詮索好き。このことからパスワードハッカー(password hacker)、ネットワークハッカー(network hacker)などと言われる。正しい用語はクラッカー(cracker)。

つまり、マスコミが「ハッカー」と言っている意味は、 [誤用] 以外の何物でもありません。

この本は、ハッカーについてより深く、そして正しく知る

ために役立つ情報や、ハッカーの知り合いがいると妙に納得する説明も多いのです。例えば、「ハッカー倫理」のところには「インターネットに代表される大規模共同ネットワークが、中央制御なしに運用できているのは、ハッカー王国のもっとも貴重な無形財産と言える連帯感に依存すると同時に、それを強化する役割を果たしている」というふうに書いてありました。思想的にとっても正しいことがさらりと書いてあるあたり、拍手ものです。

さらに、RFC(Request For Comment: コメント要求)についての説明も、GNU(“GNU's Not UNIX!”の頭文字で、Richard Stallman氏が率いるFree Software FoundationによるUNIX互換ツールの開発プロジェクト)についての説明も、非常に的を射てました。編者のEric S. Raymond氏は、ただ者ではないですね。これは、今、知ったことでもないけど(笑)、とにかく、私のお気に入りの1冊です。

ただ、残念ながらあまり本屋に並んでないんですよね。品薄なのかな。アスキーさんには、これを機に、ぜひ、増刷して本屋にバンバン積んでいただきたいものです。

でね、ハッカーってやっぱり男性が多いじゃないですか。技術屋さんに男性の割合が多いから仕方がないのかな……とか思いつつ、最近では女性のインターネット利用者も多いし、Linux利用者の女性もこんなに増えているのだから、そのうち、女性のハッカーの数は飛躍的に伸びるのだろうか？と考えてみたのです。でも、ある程度までは増えたとしても、ハッカーと言われるほどに極める人は、やっぱり男性が多いような気がしてならない。うーん、なぜなんだあ……と考えていたときに、本屋で興味深い本「やわらかな脳のつくり方」を見つけました[3]。

この本の「男の保守性、女性の向上心」という單元には、男性の脳と女性の脳は構造が違っているため(女性の脳は、右脳と左脳をつなぐ連絡橋の部分が大きい)、「何かに凝る(入れ上げる、のめり込む)」というのは男性の脳の特徴といえると書いてありました。

性差による脳の違いは、人類の歴史の中で「男性は野に出て獲物を追いかけて、女性は集落に残って子供の世話をする」という役割分担の時期が長かったことに起因していると考えられるらしい。獲物を追いかけるとき、自分をとりまく環境は予想不可能な事柄で埋まっているので、ある場所である手法がうまくいった場合、男性はかたくなにその手法を守り続ける習性ができた。逆に、女性の方は、ほぼ毎日同じ仕事を受け取っていたので、日々、改善を心がけるようになり「あっちの方法の方がいいのかも……」と思いがちなそうです。

インターネットの女性ユーザーや、Linuxの女性ユーザーが

増える理由も、これで説明がついたりするんですよね。「電話よりも電子メールの方が便利なのかも」とか「WindowsよりもLinuxの方がいいのかも」と思う女性が多いことも、「Linuxの新しいディストリビューションが使ってみたいな」と、好奇心でワクワクする女性がいることも、それは脳の構造によるものだったのか……。

もちろん、女性らしい脳を持った男性や、男性らしい脳を持った女性もいくらでもいるそうなので、いちがいに言えませんけどね。ただ、私がハッカーが好きな理由が、なんとなく説明がつくような気がしました。自分自身は、ふらふらと「あっちの方法の方がいいのかも……」と思いがちなので、それとは反対に、信念を持って、あることにとことん集中している一途な人に魅せられるわけです。自ら応援団として、旗なんかを振らせてもらいたくもなるわけです。

ディレクトリにパスを通すということ

さて、今回は「シェルスクリプトの紹介」のつもりだったのですが、ちょっと予定を変更して、シェルスクリプトの置き場所について説明しておきます。

というのは、先日、尊敬するハッカーな友人の1人から、私のところに、以下のようなメールが届いたのです。

Linux Japan誌2000年3月号「Linux事始めの書」で発見。
よしださん、カレントディレクトリにパスを通していることを前提に書いていますね。危ないなあ。

最初のメールの内容はこの3行だけでした。私がハッカーな友人が好きな理由の1つに、メールの文章が簡潔なことがあります。とにかく無駄がない。そして、自分に知識がないと理解できないこともある分、理解できたときは嬉しいわけです。

この文章を理解するには、まず、「カレントディレクトリにパスを通す」という意味が分かる必要がありますね。

カレントディレクトリとは、自分が今いるディレクトリのことです。ログイン時のディレクトリは、ホームディレクトリです。そして、cdコマンドでどこかに移動したら、移動先がカレントディレクトリです。例えば、ホームディレクトリの下にある、Mailというディレクトリに移動したとします。

```
tomolx login:
Last login: Mon Feb 28 18:17:00 on tty1.
No mail.
$ cd Mail
$ pwd
/home/tomo/Mail
```

この状態は、「現在のカレントディレクトリは、/home/tomo/Mailである」と表現できます。この下に、本誌2000年3月号で紹介したmlistというシェルスクリプトが存在していたとします。

```
$ pwd
/home/tomo/Mail
$ ls -lF mlist
-rwxr-xr-x  1 tomo  tomogrp 3048 Feb 27 18:26 mlist*
```

このとき、「カレントディレクトリにパスが通っている」なら、

```
$ mlist
```

というふうに入力すれば、mlistを実行することができます。逆に「カレントディレクトリにパスが通っていない」のなら、

```
$ mlist
mlist: Command not found.
```

と言われて、実行できません。「実際にmlistというシェルスクリプトはここに存在するのに、なんで Command not found (そんなコマンドはないよーん)なんて意地悪なことを言うの〜?」と思うかもしれませんが、カレントディレクトリにパスが通っていないなら、そのディレクトリにあっても、見つけられないのです。

ちなみに、カレントディレクトリにパスが通っているかどうかは、インストール時のデフォルトの状態が決まっています。調べ方は、bashに代表されるBシェル系を使っているなら、

```
$ echo $PATH
/usr/local/bin/:/bin/:/usr/bin:/usr/X11/bin/..
```

となりますし、tcshに代表される、Cシェル系を使っているなら、“echo \$path”というコマンドを使うことになります。

いずれの場合も、ただらだらと表示されるディレクトリ名がパスが通っているディレクトリたちです。カレントディレクトリは、“.”(ピリオド)で表現されますので、ディレクトリの一覧の中に、“.”があれば「カレントディレクトリにパスが通っている」状態で、“.”がなければ、「カレントディレクトリにパスが通っていない」ことが分かります。

そして、「mlistを実行してね」と命令されたシェルは、パスが通っている複数のディレクトリを、前から順に見ていき、mlistが見つかったら、そのmlistを実行します。そのため、いろんなディレクトリにたくさんのmlistがあった場合は、パスがより前に書いてあるディレクトリの下にあるmlistが実行されます。

逆に、それらのパスのいずれにも見つからなかったときの

エラーメッセージが、先ほどの No such file or directory” でした。「カレントディレクトリにパスが通っていない」ために、実際はあるのに見つけれなかったわけですね。そんなときは、「カレントディレクトリにあるこのコマンドを、あえて実行してちょうだい」というふうに指示すればよいのです。それが“./mlist”です。

```
$ ls -lF mlist
-rwxr-xr-x  1 tomo  tomogrp 3048 Feb 27 18:26 mlist*
$ mlist
mlist: Command not found.
$ ./mlist
Hideo XXXXX <XXXXX@YYYYY.ac.jp>
Hidesuke XXXXX <XXXXX@246.ne.jp>
Mariko Katagiri <XXXXX@notredame.ac.jp>
Susumu XXXXX <XXXXX@yo.rim.or.jp>
Tomoko Yoshida <tomo@tomo.gr.jp>
XXXXX Katsumi <XXXXX@ops.dti.ne.jp>
:
```

で、やっとこさ次が「カレントディレクトリにパスが通っていると危ない理由」なのですが、これについては、数多くの説明文が存在します。そのうちの1つが、1999年の3月に流れた「linux-users ML」の48929番です。「危険な理由について、参考になる文献があれば教えてください」と、ハッカーな方に聞いたときの返事メールは、次の3行となるでしょう。

```
理由は下の文献をご覧ください。
[linux-users:48929] Re: PATHとセキュリティー
http://his.ktarn.or.jp/ML/linux-users.4/msg08775.html
```

なんて簡潔な。で、これを読めと言われたので、読んでいくわけです。ここには、楠根氏により、カレントディレクトリにパスを通しておくことによって、なぜ危険かが詳しく紹介されていました[4]。重要な部分をそのまま掲載させていただきます(リスト1)。

この説明で分かった人は、めでたし、めでたし。

残念ながら、「ちんぷんかんぷん」な人もいると思いますので、私の言葉で説明します。今、カレントディレクトリが、/home/taro/(taroのホームディレクトリ)だとします。ここに、taroさんが“ls”という名前のシェルスクリプトを作っています。そして、なんと！ その記述内容は、なんらかの理由で、大胆にも“rm \$HOME/*”だったとします。つまり、ホームディレクトリのファイルをすべて消してしまうコ

マンドなのです。

taroさんは、lsという名前のシェルスクリプトを作って、自分のホームディレクトリに置いた本人なので、自分が作ったlsがほんとうは“rm”であるということを知っているからいいのですが、tomoさんは知りません。

tomoさんは、たまたま/home/taro/に移動して、ファイルをリストアップしようとして、lsと入力するかもしれません。そのとき、カレントディレクトリにパスが通っていて、しかも通常のlsコマンドのある、/bin/の下というパスよりもより前にカレントディレクトリのパスが記述されているなら、taroさんの作った「実際はrmなのに、シェルスクリプト名はlsという、非常に危険なファイル」が実行されてしまうというわけなのです。

```
$ cd ../taro
$ pwd
/home/taro/
$ ls -lF ls
-rwxr-xr-x  1 taro  tomogrp 48 Feb 29 14:33 ls*
$ ls
$ cat ls
#!/bin/sh
rm $HOME/*
$ echo $PATH
./usr/local/bin/:/bin/:/usr/bin:/usr/X11/bin/
$
```

確かにこういうケースが起こるなら、カレントディレクトリにパスが通っていると、危ない、危ない。ま、自分しか使わないようなローカルなシステムなら、こういう状況には、なかなかならないんですけどね。

と、ここまでが、上記の [linux-users:48929] の説明を、私なりに解説したものでした。いま一度、[linux-users:48929] を読むと、「なるほど〜」って納得できるはずですよ。

✂ シェルスクリプトの置場所

では最後に、自分が作ったシェルスクリプトは、どこに置けば良いのでしょうか。

これは、自分のホームディレクトリの下に、binというディレクトリの下にコピーし、そのディレクトリ(/home/tomo/binつまり\$HOME/bin)に、パスを通しておくことが推奨されています。

```
$ cd ; mkdir bin
$ cp /home/tomo/Mail/mlist bin
$ echo $PATH
/usr/local/bin/:/bin/:/usr/bin:/usr/X11/bin/
$ PATH=$HOME/bin:/usr/local/bin/:/bin/:/usr/bin:/usr/X11/bin/
$ echo $PATH
/home/tomo/bin:/usr/local/bin/:/bin/:/usr/bin:/usr/X11/bin/
```

リスト1

[linux-users:48929] Re: PATHとセキュリティー

⋮

1. 氏素性のわからない command を起動するのは危険である
2. PATH に入っているものは自動的に選択されて実行されてしまう
3. current directory が必ず secure とは限らない

といったかんじでしょうか。

例えば、freeware の source archive の中に make という名前の実行 file が入っていたとして、何も考えずに展開して make と入力すると、それが実行されてしまう可能性がありますよね。

そんなの展開したらまず ls してみりゃわかることですが、じゃあ ls という名前のものも含まれてたらどうします？

というように、意図に反した危険性が高くなるので、current directory やあまり secure でない directory、あまり access することのない directory は PATH に含めずに、必要に応じて pathname で指定して実行する方が安全であると言えます。

⋮

ただし、このように画面で追加したパスは、その場では有効になりますが、保存されないので、“~/profile”あるいは、“~/bashrc”の記述に、“\$HOME/bin”を追加しておくのがベターです。1度、エディタ(Muleかviなど)で、これらのファイルを開いてみましょう。

また、tcshに代表される、Cシェル系を使っているなら、“~/login”あるいは“~/cshrc”の中でパスを記述している部分に、“~/bin”を追加しておきます。つまり個人のパスは、これらの初期設定ファイルによって設定されているので、ここを手直しすれば良いのです。

では、また。

R E S O U R C E

- [1] hacker is *not* cracker!
<http://www.vacia.is.tohoku.ac.jp/~s-yamane/articles/hacker.html>
- [2] 「ハッカーズ大辞典」
 アスキー出版、Eric S. Raymond編、福崎 俊博訳、ISBN4-7561-0374-X、3200円1995年7月1日 初版発行(1993年に発行された“The New Hacker's Dictionary - 2nd Edition”の邦訳)
- [3] 「やわらかな脳づくり方」
 新潮社、吉成真由美著、ISBN4-10-433001-9、1600円、1999年12月15日発行
- [4] [linux-users:48929 Re: PATHとセキュリティ
<http://his.ktarn.or.jp/ML/linux-users.4/msg08775.html>

COLUMN

オープンソースOSの良さを分かってもらうには？

先日、「大阪府高等学校情報教育研究会」主催の研究会で、「先生のためのネットワークリテラシー講座」というタイトルの講演をしました(詳しくは、<http://www.edu-c.pref.osaka.jp/~mjyoho/>参照)。このときの、Q&Aタイムの最初の質問が、非常に興味深いものだったので紹介します。

Q：私自身は数年前からLinuxを使っており、オープンソースの良さを十分に分かっているつもりです。それで、高校で教育用に使う環境はクライアントまでオープンソースのものが良いと思っているのですが、まわりの人には、オープンソースの良さをなかなか理解してもらえません。「そんないいものなら、もっと前から注目されたはずだ」と言われたこともあります。どういう説明をすれば、オープンソースの良さを分かってもらえると思われませんか？

この質問を聞いた私は、何をどう説明するかも決めないまま、条件反射的に、黒板にスタンドアロンのパソコンの絵を書き、こういう説明を始めました。

オープンソースの環境が、もっと前から注目されなかった理由は、インターネット環境が日常的になる前は、パソコンというのは、他の家電製品と同じだったから。古くなって使えなくなるまで、その環境を使い続ける。例えば、学校で買ったパソコンは5年も10年も、付属のOS、付属のアプリケーションを使い続けていたはず。そういう時代は、付属のOSやアプリケーションがオープンソースでなくても、誰も何も

困らなかった。

ここまで話した私は、だんだんと盛り上がってきました。

でも、インターネットが発達して、外部とデータをやりとりするようになると、10年前のOS、5年前のアプリケーションでは、外部から入ってくるデータが見られないようになってしまうことがある。商用のOSや商用のソフトウェアは、新しいものを買ってもらわないと商売にならないために、新しいバージョンを出してくるから。学校現場の多くは、それに追従するお金がないから、取り残される。一方、Linuxが元にしたUNIXの世界では、お金がないから取り残されることは少ない。やりとりされるのはテキストデータが基本だし、優秀なソフトウェアがオープンソースだから。

ますます熱くなった私は、ここで「オープンソースと手作りぬいぐるみ」の話だとか、メキシコの教育機関ではクライアントにまでLinuxが導入されてるといった話を始めようとしたのですが、司会者に止められてしまいました。

「時間の限られた貴重なQ&Aタイムですから、本日の本題からそれたUNIXやLinuxの説明は、このあたりでおしまいしてください」

がーん。確かにその通り。ということで、話足りないまま、しょんぼり席についた私なのでした……。

(よしだともこ)