Hacking

having fun with
sh, perl, and python scripts

budi rahardjo @rahard

hacking

Hacking is the gaining of access (wanted or unwanted) to a computer and viewing, copying, or creating data (leaving a trace) without the intention of destroying data or maliciously harming the computer.

This represents the Good Guys most of the time for they are the ones who search for these exploits to prevent crackers use a method called cracking(opposite of hacking).

Hacking and hackers are commonly mistaken to be the bad guys most of the time. Crackers are the ones who screw things over as far as creating virus, cracks, spyware, and destroying data.

hack·er (hăkʹər)
n. informal

1. a. **One who is proficient at using or programming a computer; a computer buff**
   b. One who uses programming skills to gain illegal access to a computer network or file.
2. Onewho demonstrates poor or mediocre ability, especially in a sport: a weekend tennis hacker.

Word History: Computer programmers started using the word hacker in the 1960s as a positive term for a person of skillful programming ability. The usage probably derives from hack meaning "to chop," or from hacker, "an amateurish player, as at golf." As time went on, hacker became less positive, however. Already in the 1960s, engineering students at such universities as Cal Tech used the related noun hack to mean "an ingenious prank." Among the pranks that some computer programmers would engage in, of course, were break-ins into other computer systems. As such break-ins attracted national attention, the media seized upon the word hacker as the label for the perpetrators—a usage that many programmers object to because they know it used to be a term of praise.

sh, bash, …

SHELL SCRIPT

the history of **sh**

- What is “shell”?
- The history of UNIX
  - UNIX vs. *nix-like (*BSD, Minix, Linux, Hurd,…)
  - sh, csh, ksh, bash, tcsh, zsh, …
- Shell programming
  - Orchestrating UNIX tools/utilities (which are good at doing their stuff)

Web-site/HTML (pre)processor

- How to create a static web site (for the paranoids)?
  - Lots of pages/files
  - Need consistent look and feel/UI
  - Modify “theme” when needed
  - (Offline CMS)

Disecting a web page

HEAD (before title)

TITLE

HEAD (after title)

BODY (navigation bar)

Content

BODY (end)

Core Javascript

process.sh

```
#! /bin/bash

cwd="/Users/Budi/Public/budi-insan2/src"
template="/Users/Budi/Public/budi-insan2/src"
output="/Users/Budi/Public/budi-insan2/out"

filename=$1
outfile=$output/$filename.html

echo "$outfile"
cat $template/1-head-before-title.html > $outfile
cat $cwd/$filename.title >> $outfile
cat $template/2-head-after-title.html >> $outfile
cat $template/3-body-nav-bar.html >> $outfile
cat $cwd/$filename.txt >> $outfile
cat $template/4-body-end.html >> $outfile
cat $template/5-core-javascript.html >> $outfile
```

Source directory

```
$ ls

1-head-before-title.html
2-head-after-title.html
3-body-nav-bar.html
4-body-end.html
5-core-javascript.html
about.title
about.txt
academics.title
academics.txt
articles.title
articles.txt
index.title
index.txt
makalah-security.title
makalah-security.txt
new.title
new.txt
personal.title
personal.txt
struktur.txt
students.title
students.txt
template.html
```

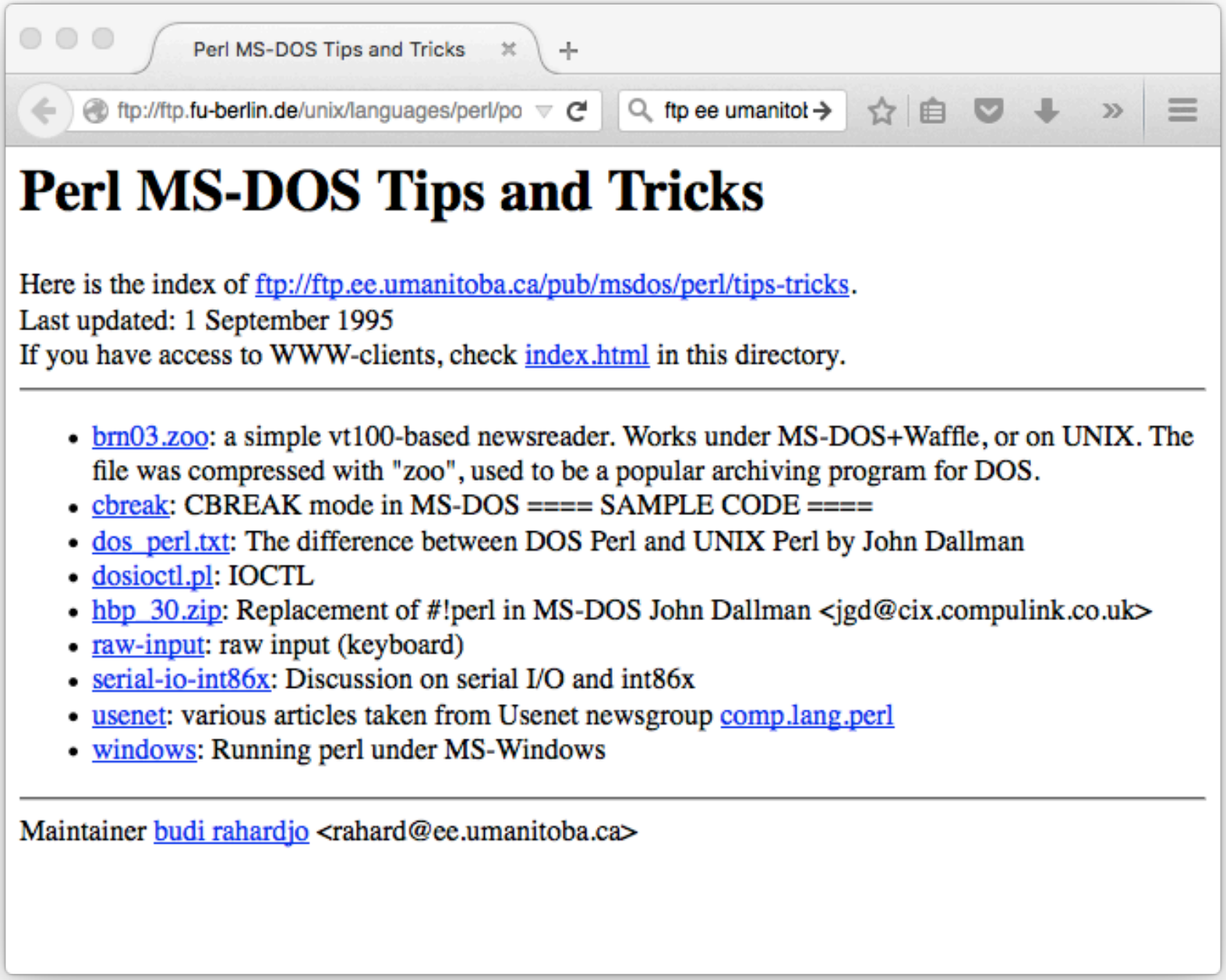
```
$ cat run.sh
#! /bin/bash

./process.sh index
./process.sh about
./process.sh academics
./process.sh articles
./process.sh new
./process.sh personal
./process.sh students
./process.sh makalah-security
```

PERL


Unmatched Power for
Text Processing and Scripting
4th Edition
Covers Version 5.14
Programming
Perl
Tom Christiansen,
brian d foy & Larry Wall
with Jon Orwant
O'REILLY®

What is perl?

- Combination of sh, unix utilities, basic, C, … (kitchen sink)
- Available for all kinds of platforms
- I was the maintainer for DOS port of perl
  - I had to use perl at that time because I had multiple platforms (various UNIXes; SunOS, AIX, HPUX, [Linux was not even born], and DOS at home)
  - Only “C” and “Perl” were available

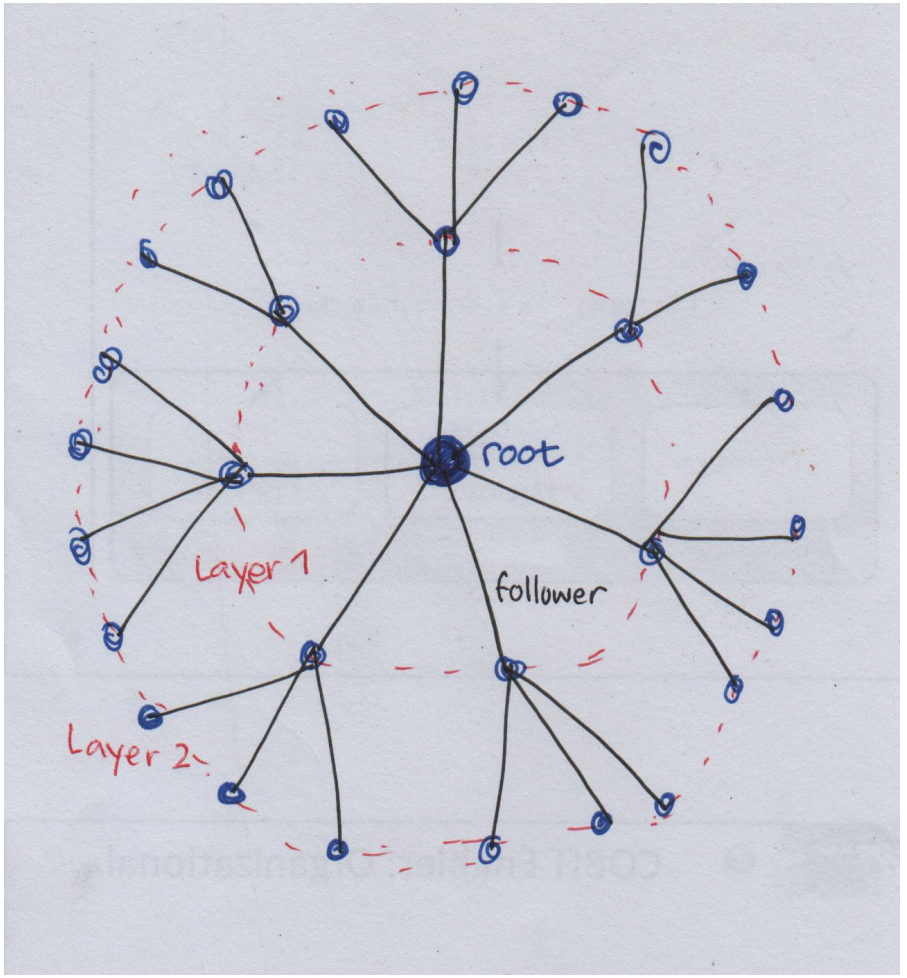

Perl MS-DOS Tips and Tricks
ftp://ftp.fu-berlin.de/unix/languages/perl/po
ftp ee umanitol
Perl MS-DOS Tips and Tricks
Here is the index of ftp://ftp.ee.umanitoba.ca/pub/msdos/perl/tips-tricks.
Last updated: 1 September 1995
If you have access to WWW-clients, check index.html in this directory.
brn03.zoo: a simple vt100-based newsreader. Works under MS-DOS+Waffle, or on UNIX. The file was compressed with "zoo", used to be a popular archiving program for DOS.
cbreak: CBREAK mode in MS-DOS ==== SAMPLE CODE ====
dos_perl.txt: The difference between DOS Perl and UNIX Perl by John Dallman
dosioctl.pl: IOCTL
hbp_30.zip: Replacement of #!perl in MS-DOS John Dallman <jgd@cix.compulink.co.uk>
raw-input: raw input (keyboard)
serial-io-int86x: Discussion on serial I/O and int86x
usenet: various articles taken from Usenet newsgroup comp.lang.perl
windows: Running perl under MS-Windows
Maintainer budi rahardjo <rahard@ee.umanitoba.ca>

topic-generator.pl

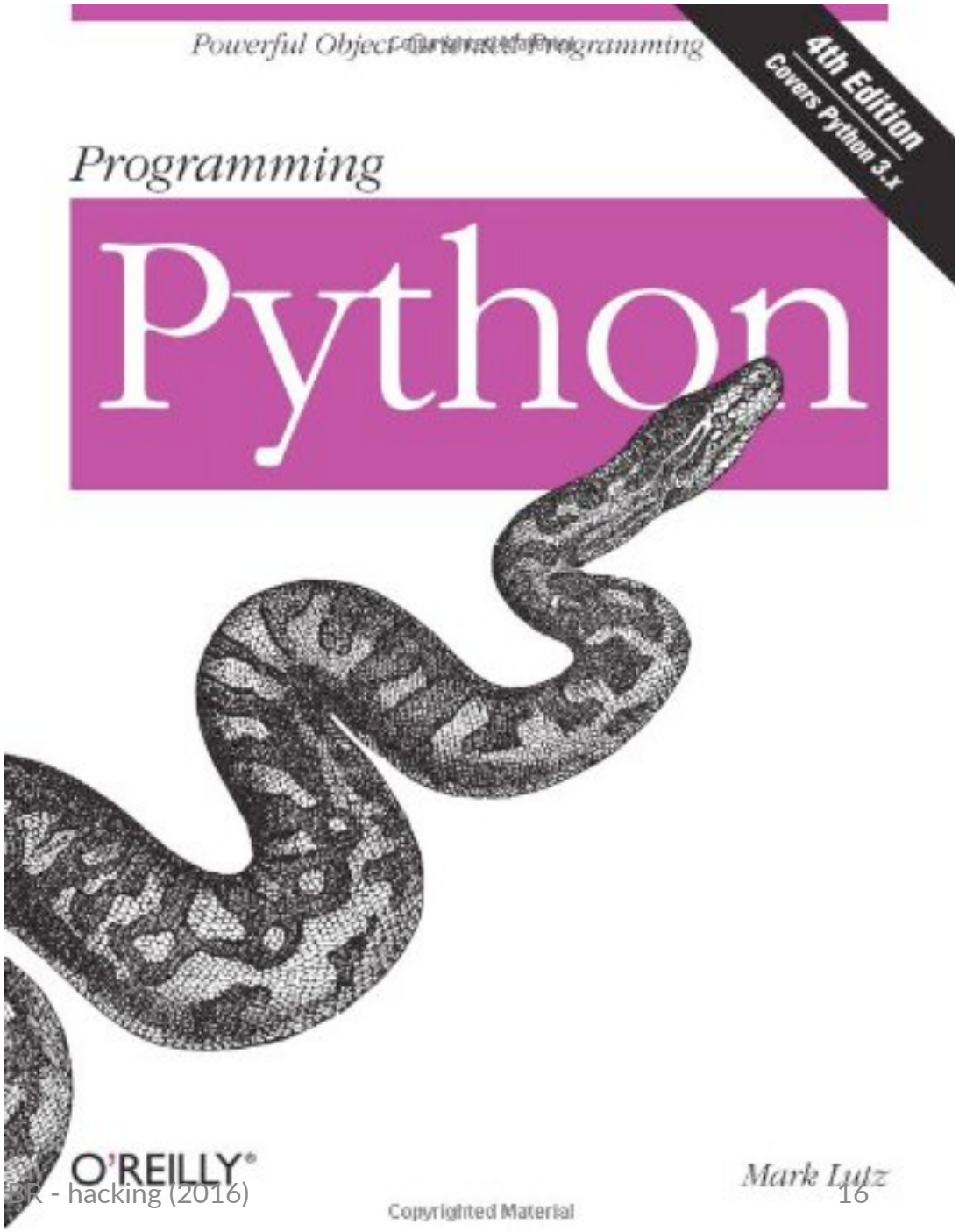
```
#! /usr/bin/perl

$topicDB='topics.txt';
open(my $DB, '<', $topicDB) or die $!;
$count=0;
while (<$DB>) { if ($_ =~ "^--") { $count++; }
else { $topic{$count} = $topic{$count} . $_;} }
close($DB);

$luckyone =  int(rand($count-1));
print $topic{$luckyone};
exit;
```

More perl scripts

- Twitter crawler
- Twitter analysis
- Session hog (create multiple connections to internet services, DoS attack)
- Mailbomb
- Perl poetry
- … [too many]



PYTHON


Powerful Object-Oriented Programming
4th Edition
Covers Python 3.x
Programming
Python
O'REILLY®
Mark Lutz
Copyrighted Material

(Hologram) Client-Server App

- An application to determine user's character (by asking questions) and display the imaginary character of the person
- "Character" is displayed as hologram (video)
- **Server**:
  - wait for command to display a particular video
- **Client**:
  - ask user (questionaire), send result to server

Server

- [see source code. A bit too long if pasted here.]
- Basically, it's a telnet-based server
- Wait for a (telnet) connection in a certain port and respond to the commands
- Commands
  - "play videoname.mp4" (play that video)
  - "kill" (kill the previous play)
  - "quit" (exit the telnet sever)

Client

- [see source code – on screen. Too long if typed here.]
- Ask user based on questions in “questions.txt” file
- Display the questions using Tkinter
- Based on user selection, calculate the point.
- Based on this point, select the appropriate video
- Telnet to the server with command “play videoname.mp4”
- When done, send “quit” command
- Back into the loop

Future Applications

- Interactive video (advertising) display
- Interactive games
- Server can run in a cheap single board computer (attached to a TV screen)
- Too many …

Concluding Remarks

Hacking is fun. (No cracking please)