

Re: converting line input into columns

Source: <http://coding.derkeiler.com/Archive/Perl/comp.lang.perl.misc/2006-07/msg00528.html>

- *From:* David Squire <David.Squire@xxxxxxxxxxxxxxxxxxxxxxxx>
 - *Date:* Mon, 10 Jul 2006 10:54:33 +0100
-

vanagas99@xxxxxxxxx wrote:

(Note: I *do* address your code problem at the bottom)

A. Sinan Unur wrote:

vanagas99@xxxxxxxxx wrote in
news:1152496902.511701.167670@xx:

Mr. McClellan, Thanks for the counseling session.
Appreciate it.
Great addition to my perl learning experience.....

Sarcasm is not going to help you.

[sig snipped]

Sinan,

I understand that. There is no reason for Tad to give a speech, either.
I hate when people think they own the world. I didn't even know what
"posting on top" means...

Yet by the time it came to this you had been directed to the posting guidelines for this group several times, and were clearly not following them. The explain what top-posting is and the reason it is not liked here. They also advise not quoting sigs, which you did this time.

plus, blabing about the sincerity of my "sorry"???

It doesn't sound very sincere when you apologize for whacking someone whilst continuing to whack them. That's effectively what you're post was like.

Re: converting line input into columns

"Abuse us?? what kind of statement is that?"

Perfectly fair. Your failure to give an accurate problem description wastes the time of everyone who tries to help you. That fits the definition of abuse (of their time and patience).

As you have no doubt guessed already, this is a tough group. Folks here are happy to give new posters a couple of chances, politely directing them to the clpm posting guidelines. If these continue to be ignored, the new poster soon becomes invisible. It seems that you have already been kill-filed by two of the most knowledgeable and helpful people here.

The code you quote below does not "work" because it was designed for the record specification you gave in your first posts, not the data you are actually feeding it. See comments below.

```
use strict;
use warnings;

use Data::Dumper;
use English qw{ -no_match_vars };

$INPUT_RECORD_SEPARATOR = ";

RECORD:
while (my $record = <DATA>) {
my (%record) = $record =~ m{\A\s*
(Severity)
:(.+?)
```

So, this expected a record to start with a 'Severity' field, whereas your data below shows records starting with a single line that seems to specify a class of record (e.g. "Physical Security). You're going to need some thing to capture that, e.g.:

```
my ($record_class, %record) = $record =~ m{\A\s*
(.+?)
(Severity)\s*:(.+?)

(Status)
:(.+?)
(PDI . ID)
:(.+?)
(Finding . Details)
(.+?)
(Vulnerability . Discussion)
```

Re: converting line input into columns

Your records below seem to have a "Category" field in this position. Is that supposed to be captured separately? At present it will be swallowed as part of 'Vulnerability Discussion'.

```
(.+?)  
(Manual . Fix . recommendations)
```

Below you have "Manual Fix Procedures"

```
(.+?)  
(References and additional information) (.+?)
```

You need to put in '.'s (as in the lines above) to match the spaces between the words in "References and additional information". The 'x' switch at the end of the match causes whitespace to be ignored (and permits comments), which is what allows the regex to be laid out nicely for readability like this.

```
\z}xms;  
if (not %record) {  
  warn "Malformed record";  
  next RECORD;  
}  
else {  
  # fix up spacing  
  for my $entry ( values %record ) {  
    $entry =~ s/^\s+//gm;  
    $entry =~ s/\s+$//gm;  
    $entry =~ s/\n/ /g;  
  }  
  print Dumper \%record;  
}  
}
```

__DATA__

```
Physical Security  
Severity : Cat II / Important  
Status : Unknown  
PDI ID : 1836
```

These variable numbers of spaces before the colons are going to cause the patterns above to fail. There was no space there in your original post. You will need to allow for this by adding '\s*' in the pattern before the colons.

Re: converting line input into columns

Finding Details

This vulnerability could not be checked by the program, it must be checked manually.

Vulnerability Discussion

Category: II Inadequate physical protection can undermine all other security precautions utilized to protect the system. This can jeopardize the confidentiality, availability, and integrity of the system. Physical security of the individual machine is the first line protection of any system.

Manual Fix Procedures

Ensure the computer equipment is located in a protected controlled access area.

References and additional information

FSO Checklist: 3.1 CJCSM 6510.01, C-D.3

Administrator Documentation

Severity : Cat II / Important

Status : Unknown

PDI ID : 1788

Finding Details

This vulnerability could not be checked by the program, it must be checked manually.

Vulnerability Discussion

Category: II Using a privileged account to perform routine functions makes the computer vulnerable to attack by any virus or Trojan Horse inadvertently introduced during a session that has been granted full privileges. The rule of least privilege should always be enforced.

Manual Fix Procedures

Ensure administrative personnel have two accounts assigned, a standard user account and an account with membership in the Administrators group. Personnel should be instructed to use the less privileged account for day-to-day use. Each System Administrator will have a unique userid dedicated for administering the system. Each System Administrator will have a separate account for normal user tasks. The built-in Administrator account will not be used to administer the system. Administrators will be properly trained. The IAO will maintain a list of users belonging to the Administrators group.

References and additional information

FSO Checklist: 3.2 DODI 8200.2, E3.4.7 CJCSM 6510.01, A-A.3