

# Optimize the script

---

*Source:* <http://coding.derkeiler.com/Archive/Perl/perl.beginners/2008-07/msg00169.html>

---

- *From:* [speedjet5@xxxxxxxxxxxxxx](mailto:speedjet5@xxxxxxxxxxxxxx) (Jet Speed)
  - *Date:* Fri, 4 Jul 2008 12:08:16 +0100
- 

Hi All,

I put together a piece of code with some help, to capture the output as below from a file "cxout1" as below. I am sure this can be written in few lines or even differnt way of getting the same out put with use of hash etc. Kindly help me to optimize the code. Any help would be much appericated.

output from the script

-----  
The Clarion Array Serial Number is CK200061101100

LUN INFO POLICY-TYPE OWNER PSEUDO DEVICE  
LUN 415 policy=CLAROpt current=SP A emcpower3a  
LUN 815 policy=CLAROpt current=SP B emcpower4a

file contents cxout1

-----  
Pseudo name=emcpower3a  
CLARiiON ID=CK200061101100 [JAZZ]  
Logical device ID=600601601496160012288D48703EDB11 [LUN 415]  
state=alive; policy=CLAROpt; priority=0; queued-IOS=0  
Owner: default=SP A, current=SP A

=====

Host	Stor	I/O Path	Stats
------	------	----------	-------

----

### HW Path I/O Paths Interf. Mode State Q-IOS

Errors

=====

2310 pci@1c/pci@1/lpfc@4 c3t20d0s0 SP A0	active	alive	0
1			
2310 pci@1c/pci@1/lpfc@4 c3t21d0s0 SP A1	active	alive	0
1			

Pseudo name=emcpower4a  
CLARiiON ID=CK200061101100 [JAZZ]  
Logical device ID=6006016014961600625987643E38DB11 [LUN 815]

## Optimize the script

```
state=alive; policy=CLAROpt; priority=0; queued-IOs=0
Owner: default=SP B, current=SP B
```

```
=====
----- Host ----- - Stor - -- I/O Path - -- Stats
```

```
### HW Path I/O Paths Interf. Mode State Q-IOs
```

```
Errors
```

```
=====
2310 pci@1c/pci@1/lpfc@4 c3t20d1s0 SP A0 active alive 0
1
2310 pci@1c/pci@1/lpfc@4 c3t21d1s0 SP A1 active alive 0
1
```

my code

```
#!/usr/bin/perl -w
```

```
use strict;
use warnings;
use List::MoreUtils qw(:all);
no warnings qw/syntax/;
```

```
#PRINT SERIAL NUMBER
```

```
my @sl;
my $filename;
$filename = "cxout1" ;
open (FILE, "<$filename") or die "Could not open $filename: $!";
while (<FILE>) {
next unless $_ =~ /CK/;
$_ = ~ (/b(ID)\=(\w+)\s+/i) ;
push(@sl,$2);
}
```

```
# PRINT UNIQUE SERIAL NUMBER
```

```
my %seen = ();
my @uniq = ();
my $item;
foreach $item (@sl) {
unless ($seen{$item}) {
# if we get here, we have not seen it before
$seen{$item} = 1;
push(@uniq, $item);
}
}
```

Optimize the script

## Optimize the script

```
print "\nThe Clarion Array Serial Number is @uniq\n\n";
```

### # PRINT LUN INFORMATION

```
my @luns;  
#my $filename;  
$filename = "cxout1" ;  
open (FILE, "< $filename" ) or die "Could not open $filename: $!";  
while (<FILE>) {  
  chomp;  
  # next unless $_ =~ /LUN/;  
  # $_ =~ /\[(LUN\s+\d+)\]/;  
  next unless /\[(LUN\s+\d+)\]/;  
  push(@luns,$1);  
  
}
```

### # PRINT POLICY INFORMATION

```
my @pol;  
#my $filename;  
$filename = "cxout1" ;  
open (FILE, "< $filename" ) or die "Could not open $filename: $!";  
while (<FILE>) {  
  chomp;  
  next unless $_ =~ /policy/;  
  $_ = unpack ("x13 A14", $_);  
  push(@pol,$_);  
}
```

### # PRINT OWNER INFORMATION

```
my @own;  
#my $filename;  
$filename = "cxout1" ;  
open (FILE, "< $filename" ) or die "Could not open $filename: $!";  
while (<FILE>) {  
  chomp;  
  next unless $_ =~ (/^b(current)\=\w+/);  
  $_ = unpack ("x21 A14",$_);  
  push(@own, $_);  
}
```

### #PRINT PSEUDO INFORMATION

```
my @pseudo;  
open (FILE, "< $filename" ) or die "Could not open $filename: $!";  
while (<FILE>) {
```

## Optimize the script

```
chomp;  
next unless $_ =~ (/b(Pseudo)/);  
$_ = unpack ("x12 A18", $_);  
push (@pseudo, $_);  
}
```

```
#CONSOLIDATE & PRINT
```

```
my $result = each_array(@luns, @pol, @own, @pseudo);
```

```
print "LUN INFO\t POLICY-TYPE\t OWNER\t PSEUDO DEVICE\t\t\n";
```

```
while ( my ($a, $b, $c, $d) = $result->() )  
{  
  
print "$a\t\t $b\t\t $c\t\t $d\t\t\n";  
}
```

Thanks in Advance

Js