

f (<https://www.facebook.com/OpenSourceForU/>) **t** (<https://twitter.com/opensourceforu>)

► (https://www.youtube.com/channel/UCJbnMYV_yigckub3RjtXD1Q)

Submit Tips (<http://opensourceforu.com/submit-your-tips-tricks/>)

Subscribe to Print Edition (<http://subscribe.efyindia.com/electronicsforu/subscription/newsusbc2scheme.asp>)

Magazine Feedback (https://docs.google.com/forms/d/1NnBouU0ZUmu1yUICTAj1aEFiDDWvS-1LIS85-qXorrM/viewform?usp=send_form)

Latest in Open Source (<http://opensourceforu.com/category/news/>)

Write For Us (<http://opensourceforu.com/write-for-open-source-for-you/>)

Contact Us (<http://opensourceforu.com/contact-us/>)



(<http://opensourceforu.com>)

[Home \(http://opensourceforu.com/\)](http://opensourceforu.com/) › [Developers](#)

(<http://opensourceforu.com/category/developers/>) › Understanding a Kernel Oops!

(<http://opensourceforu.com/2011/01/understanding-a-kernel-oops/>)

Understanding a Kernel Oops!

DEVELOPERS ([HTTP://OPENSOURCEFORU.COM/CATEGORY/DEVELOPERS/](http://opensourceforu.com/category/developers/))

SHARE  (<http://www.facebook.com/sharer.php?u=http://opensourceforu.com/2011/01/understanding-a-kernel-oops/&t=Understanding%20a%20Kernel%20Oops!>)  (<https://twitter.com/home?status=Understanding%20a%20Kernel%20Oops!-http://opensourceforu.com/2011/01/understanding-a-kernel-oops/>)  (<https://plus.google.com/share?url=http://opensourceforu.com/2011/01/understanding-a-kernel-oops/>)  (<http://pinterest.com/pin/create/button/?url=http://opensourceforu.com/2011/01/understanding-a-kernel-oops/&media=http://opensourceforu.com/wp-content/uploads/2011/01/kernel-oops.jpg&description=Understanding%20a%20Kernel%20Oops!>)  (https://www.tumblr.com/widgets/share/tool?shareSource=legacy&canonicalUrl=&url=http%3A%2F%2Fopensourceforu.com%2F2011%2F01%2Funderstanding-a-kernel-oops%2F&posttype=link&title=Understanding+a+Kernel+Oops%21&content=))



Understanding a kernel panic and doing the forensics to trace the bug is considered a hacker's job. This is a complex task that requires sound knowledge of both the architecture you are working on, and the internals of the Linux kernel. Depending on type of error detected by the kernel, panics in the Linux kernel are classified as hard panics (Aieee!) and soft panics (Oops!). This article explains the workings of a Linux kernel 'Oops', helps to create a simple version, and then debug it. It is mainly intended for beginners getting into Linux kernel development, who need to debug the kernel. Knowledge of the Linux kernel, and C programming, is assumed.

An “Oops” is what the kernel throws at us when it finds something faulty, or an exception, in the kernel code. It’s somewhat like the segfaults of user-space. An Oops dumps its message on the console; it contains the processor status and the CPU registers of when the fault occurred. The offending process that triggered this Oops gets killed without releasing locks or cleaning up structures. The system may not even resume its normal operations sometimes; this is called an unstable state. Once an Oops has occurred, the system cannot be trusted any further.

Let's try to generate an Oops message with sample code, and try to understand the dump.

FREE NEWSLETTER

Want Daily Updates in Your
Inbox?

(https://feedburner.google.com/fb/a/mailverify?uri=LinuxForYou&loc=en_US)

CASE STUDIES



(<http://opensourceforu.com/2016/03/we-have-not-faced-any-glitches-as-we-do-the-security-hardening%c2%94/>)

We have not faced any glitches, as we do the security hardening (<http://open-source-foru.com/2016/03/we-have-not-faced-any-glitches-as-we-do-the-security-hardening%0c2%094/>)

DIKSHA P GUPTA , ...



(<http://opensourceforu.com/2016/01/%c2%93from-a-personal-preference-perspective-i-would-choose-an-open-source-solution-any-day/>)

Setting up the machine to capture an Oops

The running kernel should be compiled with `CONFIG_DEBUG_INFO` , and `syslogd` should be running. To generate and understand an Oops message, Let's write a sample kernel module, `oops.c` :

```

1 #include <linux/kernel.h>
2 #include <linux/module.h>
3 #include <linux/init.h>
4
5 static void create_oops() {
6     *(int *)0 = 0;
7 }
8
9 static int __init my_oops_init(void) {
10     printk("oops from the module\n");
11     create_oops();
12     return (0);
13 }
14 static void __exit my_oops_exit(void) {
15     printk("Goodbye world\n");
16 }
17
18 module_init(my_oops_init);
19 module_exit(my_oops_exit);

```

The associated `Makefile` for this module is as follows:

```
obj-m      := oops.o
KDIR       := /lib/modules/$(shell uname -r)/build
PWD        := $(shell pwd)
SYM=$(PWD)

all:
    $(MAKE) -C $(KDIR) SUBDIRS=$(PWD) modules
```

Once executed, the module generates the following Oops:

```

BUG: unable to handle kernel NULL pointer dereference at (null)
IP: [] my_oops_init+0x12/0x21 [oops]
PGD 7a719067 PUD 7b2b3067 PMD 0
Oops: 0002 [#1] SMP
last sysfs file: /sys/devices/virtual/misc/kvm/uevent
CPU 1
Pid: 2248, comm: insmod Tainted: P                2.6.33.3-85.fc13.x86_64
RIP: 0010:[] [] my_oops_init+0x12/
RSP: 0018:ffff88007ad4bf08 EFLAGS: 00010292
RAX: 0000000000000018 RBX: fffffffffa03e1000 RCX: 00000000000013b7
RDX: 0000000000000000 RSI: 0000000000000046 RDI: 0000000000000246
RBP: ffff88007ad4bf08 R08: ffff88007af1cba0 R09: 0000000000000004
R10: 0000000000000000 R11: ffff88007ad4bdc8 R12: 0000000000000000
R13: 00000000016b0030 R14: 0000000000019db9 R15: 00000000016b0010
FS:  00007fb79dadf700(0000) GS:ffff880001e80000(0000) knlGS:000000000000
CS:  0010 DS: 0000 ES: 0000 CR0: 000000008005003b
CR2: 0000000000000000 CR3: 000000007a0f1000 CR4: 000000000000006e0
DR0: 0000000000000000 DR1: 0000000000000000 DR2: 0000000000000000
DR3: 0000000000000000 DR6: 00000000ffff0fff DR7: 0000000000000040
Process insmod (pid: 2248, threadinfo ffff88007ad4a000, task ffff88007a
Stack:
ffff88007ad4bf38 ffffffff8100205f fffffffffa03de060 ffffffff803de060
 0000000000000000 00000000016b0030 ffff88007ad4bf78 ffffffff8107aac9
 ffff88007ad4bf78 00007fff69f3e814 0000000000019db9 0000000000020000
Call Trace:
[] do_one_initcall+0x59/0x154
[] sys_init_module+0xd1/0x230
[] system_call_fastpath+0x16/0x1b
Code: <c7> 04 25 00 00 00 00 00 00 00 00 31 c0 c9 c3 00 00 00 00 00 00
RIP [] my_oops_init+0x12/0x21 [oops]
RSP <ffff88007ad4bf08>
CR2: 0000000000000000

```

Understanding the Oops dump

Let's have a closer look at the above dump, to understand some of the important bits of information.

```
BUG: unable to handle kernel NULL pointer dereference at (null)
```

The first line indicates a pointer with a NULL value.

```
IP: [] my oops init+0x12/0x21 [oops]
```

IP is the instruction pointer.

Ops: 0002 [#1] SMP

This is the error code value in hex. Each bit has a significance of its own:

- bit 0 == 0 means no page found, 1 means a protection fault

From a personal preference perspective, I would choose an open source solution, any day (<http://opensourceforu.com/2016/01/%c2%93from-a-personal-preference-perspective-i-would-choose-an-open-source-solution-any-day/>)

DIKSHA P GUPTA , ...



(<http://opensourceforu.com/2015/11/open-source-technology-first-love-of-the-leaders-at-askmebazaar-com/>)

Open Source Technology: First Love of the Leaders at AskMeBazaar.com!
(<http://opensourceforu.com/2015/11/open-source-technology-first-love-of-the-leaders-at-askmebazaar-com/>)

DIKSHA P GUPTA , ...



(<http://opensourceforu.com/2015/08/open-source-and-problem-solving-attitude-is-the-way-to-work-at-zopper/>)

“Open source and problem solving attitude is the way to work at Zopper”
(<http://opensourceforu.com/2015/08/open-source-and-problem-solving-attitude-is-the-way-to-work-at-zopper/>)

- bit 1 == 0 means read, 1 means write
 - bit 2 == 0 means kernel, 1 means user-mode
 - [#1] — this value is the number of times the Oops occurred.
- Multiple Oops can be triggered as a cascading effect of the first one.



HOME (HTTP://WWW.OPENSOURCEFORU.COM/)

The Tainted flag points to P here. Each flag has its own meaning. A few other flags are:

- P — Proprietary module has been loaded.
- F — Module has been forcibly loaded.
- S — SMP with a CPU not designed for SMP.
- M — System experienced a machine check exception.
- B — System has hit bad page.
- U — Userspace-defined naughtiness.
- A — ACPI table overridden.
- W — Taint on warning.

RIP: 0010:[<ffffffffffa03e1012>] [<ffffffffffa03e1012>] my_oops_init+0x12/0x21

RIP is the CPU register containing the address of the instruction that is getting executed. 0010 comes from the code segment register. my_oops_init+0x12/0x21 is the <symbol> + the offset/length.

RSP: 0018:ffff88007ad4bf08 EFLAGS: 00010292
RAX: 0000000000000018 RBX: fffffffffa03e1000 RCX: 00000000000013b7
RDX: 0000000000000000 RSI: 0000000000000046 RDI: 0000000000000246
RBP: ffff88007ad4bf08 R08: ffff88007af1cba0 R09: 0000000000000004
R10: 0000000000000000 R11: ffff88007ad4bd68 R12: 0000000000000000
R13: 00000000016b0030 R14: 0000000000019db9 R15: 00000000016b0010

This is a dump of the contents of some of the CPU registers.

Stack:
ffff88007ad4bf38 ffffffff8100205f ffffffff8100205f ffffffff8100205f
0000000000000000 00000000016b0030 ffff88007ad4bf78 ffffffff8107aac9
ffff88007ad4bf78 00007fff69f3e814 0000000000019db9 0000000000020000

The above is the stack trace.

Call Trace:
[<ffffffffff8100205f>] do_one_initcall+0x59/0x154
[<ffffffffff8107aac9>] sys_init_module+0xd1/0x230
[<ffffffffff81009b02>] system_call_fastpath+0x16/0x1b

The above is the call trace — the list of functions being called just before the Oops occurred.

Code: <c7> 04 25 00 00 00 00 00 00 00 00 31 c0 c9 c3 00 00 00 00 00 00

The Code is a hex-dump of the section of machine code that was being run at the time the Oops occurred.

Debugging an Oops dump

The first step is to load the offending module into the GDB debugger, as follows:

```
[root@DELL-RnD-India oops]# gdb oops.ko
GNU gdb (GDB) Fedora (7.1-18.fc13)
Reading symbols from /code/oops/oops.ko...done.
(gdb) add-symbol-file oops.o 0xfffffffffa03e1000
add symbol table from file "oops.o" at
       .text_addr = 0xfffffffffa03e1000
```

DIKSHA P GUPTA , ...



(http://opensourceforu.com/2015/07/india-rides-on-meru-meru-rides-on-open-source/)

India rides on Meru, Meru rides on open source!
(http://opensourceforu.com/2015/07/india-rides-on-meru-meru-rides-on-open-source/)

DIKSHA P GUPTA , ...



CONNECT WITH US

(https://www.youtube.com/channel/UC66bce55vzU)

| | | |
|--------|-----------|-------------|
| 618861 | 21893 | 4870 |
| Likes | Followers | Subscribers |

(http://opensourceforu.com)

3645
Comments

INTERVIEWS



(http://opensourceforu.com/2016/07/organisations-reap-benefits-open-source-controlling-security-associated/)

We help organisations reap the benefits of open source, while controlling security and associated risks

Next, add the symbol file to the debugger. The `add-symbol-file` command's first argument is `oops.o` and the second argument is the address of the text section of the module. You can obtain this address from `/sys/module/oops/sections/.init.text` (where `oops` is the module name):

```
OpenSource (http://opensourceforu.com)
The complete portal on open source For U.com
x(http://opensourceforu.com)
ps.o" at
e1000
(y or n) y
Reading symbols from /code/oops/oops.o...done.
HOME (HTTP://WWW.OPENSOURCEFORU.COM/)
From the RIP instruction line, we can get the name of the offending
function, and disassemble it
IT ADMIN (HTTP://OPENSOURCEFORU.COM/CATEGORY/ADMIN/)
(gdb) disassemble my_oops_init
Dump of assembler code for function my_oops_init:
0x0000000000000038 <+0>: mov $0x0,%rdi
0x0000000000000039 <+1>: mov $0x0,%rdi
0x0000000000000040 <+8>: xor %eax,%eax
0x0000000000000042 <+10>: mov %rsp,%rbp
0x0000000000000045 <+13>: callq 0x4a<my_oops_init+18>
0x0000000000000047 <+15>: movl $0x0,%eax
0x0000000000000055 <+29>: xor %eax,%eax
0x0000000000000057 <+31>: leaveq
0x0000000000000058 <+32>: retq
End of assembler dump.
HOW-TO'S (HTTP://OPENSOURCEFORU.COM/CATEGORY/HOW-TOS/)
```

Now, to pin point the actual line of offending code, we add the starting address of the instruction line. In our case, we are adding

```
0x0000000000000038 + 0x012 = 0x000000000000004a. This points to
the movl instruction.
BUZZ (HTTP://OPENSOURCEFORU.COM/CATEGORY/NEWS/)
(gdb) list *0x000000000000004a
0x4a is in my_oops_init (/code/oops/oops.c:6).
1  #include <linux/kernel.h>
2  #include <linux/module.h>
3  #include <linux/init.h>
4
5  static void create_oops() {
6      *(int *)0 = 0;
7  }
```

This gives the code of the offending function.

References

The [kerneloops.org](http://www.kerneloops.org/) (<http://www.kerneloops.org/>) website can be used to pick up a lot of Oops messages to debug. The Linux kernel documentation directory has information about Oops — [kernel/Documentation/oops-tracing.txt](#) . This, and numerous other online resources, were used while creating this article.

Related Posts:



Share this:

Google+ (http://opensourceforu.com/2011/01/understanding-a-kernel-oops/?share=google-plus-1&nb=1)

Facebook (http://opensourceforu.com/2011/01/understanding-a-kernel-oops/?share=facebook&nb=1)

Twitter (http://opensourceforu.com/2011/01/understanding-a-kernel-oops/?share=twitter&nb=1)

More

TAGS: C ([HTTP://OPENSOURCEFORU.COM/TAG/C/](http://opensourceforu.com/tag/c/)), DEBUGGING ([HTTP://OPENSOURCEFORU.COM/TAG/DEBUGGING/](http://opensourceforu.com/tag/debugging/)), FEDORA ([HTTP://OPENSOURCEFORU.COM/TAG/FEDORA/](http://opensourceforu.com/tag/fedora/)), GDB ([HTTP://OPENSOURCEFORU.COM/TAG/GDB/](http://opensourceforu.com/tag/gdb/)), KERNEL AIEE ([HTTP://OPENSOURCEFORU.COM/TAG/KERNEL-AIEE/](http://opensourceforu.com/tag/kernel-aiee/)), KERNEL CODE ([HTTP://OPENSOURCEFORU.COM/TAG/KERNEL-CODE/](http://opensourceforu.com/tag/kernel-code/)), KERNEL DEVELOPMENT ([HTTP://OPENSOURCEFORU.COM/TAG/KERNEL-DEVELOPMENT/](http://opensourceforu.com/tag/kernel-development/)), KERNEL OOPS ([HTTP://OPENSOURCEFORU.COM/TAG/KERNEL-OOPS/](http://opensourceforu.com/tag/kernel-oops/)), KERNEL PANIC ([HTTP://OPENSOURCEFORU.COM/TAG/KERNEL-PANIC/](http://opensourceforu.com/tag/kernel-panic/)), KERNELLOOPS.ORG ([HTTP://OPENSOURCEFORU.COM/TAG/KERNELLOOPS-ORG/](http://opensourceforu.com/tag/kerneloops-org/)), LFY JANUARY 2011

(<http://opensourceforu.com/2016/07/organizations-reap-benefits-open-source-controlling-security-associated/>)

JAGMEET SINGH , J...



(<http://opensourceforu.com/2016/05/cio-should-have-digitisation-on-his/>)

A CIO should have digitisation on his agenda (<http://opensourceforu.com/2016/05/cio-should-have-digitisation-on-his/>)

RAHUL CHOPRA , ...



(<http://opensourceforu.com/2016/05/our-goal-is-to-give-the-world-niche-cost-effective-technology/>)

Our goal is to give the world niche, cost-effective technology solutions (<http://opensourceforu.com/2016/05/our-goal-is-to-give-the-world-niche-cost-effective-technology/>)

NIRAJ SAHAY , MAY...



(<http://opensourceforu.com/2016/04/what-we-like-about-open-source-is-that-we-can-tweak-it-the-way-we-want/>)

What we like about open source is that we can tweak it the way we want to (<http://opensourceforu.com/2016/04/what-we-like-about-open-source-is-that-we-can-tweak-it-the-way-we-want/>)

we-like-about-open-source-is-that-we-can-tweak-it-the-way-we-want/)
RAHUL CHOPRA , A.



(<http://opensourceforu.com/2016/02/iot-is-not-an-it-initiative-but-a-business-function/>)

It is not an IT initiative, but a Business Function

(http://opensourceforu.com/2016/02/06-not-an-it-initiative-but-a-business-function/)

DIKSHA P GUPTA

HOW-TOS

(<http://opensourceforu.com/2016/08/develop-android-apps-using-mit-app-inventor/>)

Develop
Android Apps
Using MIT App
Inventor
(<http://open-source-eon/2016/08/develop-p-android-apps-using-mit-app-inventor/>)
MEGHRAJ SINGH B...

Url=&url=http%3A
Monitor Logs in
Real-time with
Swatch
Understanding a+
(http://omni-
sourceford.com/2
016/08/23262/
)

SWAYAM PRAKASH..

(<http://opensourceforu.com/2016/08/configuring->

<

Previous Article

Do You Know the Trick to Track Your Code's DNA?

<http://opensourceforu.com/2011/01/better-queries-with-mysql-part-1/>


Next Article

>

pfSense Dual WAN Failover Mode


<http://opensourceforu.com/2016/08/configuring-pfsense-dual-wan-failover-mode/>

RAJESH DEODHAR ,...

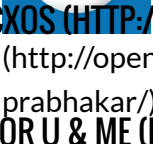


The complete portal on open source For U

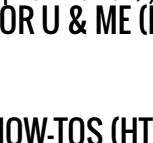
HOME (<http://www.opensourceforu.com/>)



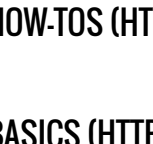
IT ADMIN (<http://opensourceforu.com/category/admin/>)



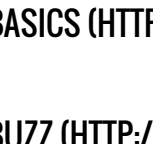
CXO (<http://opensourceforu.com/category/cxo/>)




FOR U & ME (<http://opensourceforu.com/category/everyone/>)




HOW-TOS (<http://opensourceforu.com/category/how-tos/>)



BASICS (<http://opensourceforu.com/category/basics/>)




BUZZ (<http://opensourceforu.com/category/news/>)



Surya Prabhakar

<http://opensourceforu.com/author/surya-prabhakar/>

The author is an engineering advisor in the Product Group at Dell India R&D Centre, Bengaluru, and has eight years of experience in Linux. He spends most of his time hacking and playing around with Linux.




<http://opensourceforu.com/2016/07/4-best-ways-to-perform-hassle-free-mailchimp-integration-wordpress/>

4 best ways to perform hassle-free MailChimp integration into WordPress


<http://opensourceforu.com/2016/07/4-best-ways-to-perform-hassle-free-mailchimp-integration-wordpress/>

NOLA ARNEY , JULY...

RELATED ARTICLES




NARENDRA VAGHELA (...)




<http://opensourceforu.com/2015/08/cakephp-how-to-create-a-component/>

CakePHP: How to Create a Component

<http://opensourceforu.com/2015/08/cakephp-how-to-create-a-component/>




VIVEK SHANGARI (HT...




<http://opensourceforu.com/2011/11/lisp-tears-of-joy-part-6/>

Lisp: Tears of Joy, Part 6

<http://opensourceforu.com/2011/11/lisp-tears-of-joy-part-6/>




PAVITHRA RAMAN (HT...



<http://opensourceforu.com/2015/07/working-with-data-and-cache-storage-in-drupal/>

Working with Data and Cache Storage in Drupal

<http://opensourceforu.com/2015/07/working-with-data-and-cache-storage-in-drupal/>



<http://opensourceforu.com/2016/07/publishing-app-inventor-app-google-play-store/>

Publishing an App Inventor app to Google Play Store

<http://opensourceforu.com/2016/07/publishing-app-inventor-app-google-play-store/>

MEGHRAJ SINGH B...

http://opensourceforu.com/2011/01/understanding-a-kernel-oops/

6/9

Comments

Community

1 Login

Recommend

Share

Sort by Newest

Join the discussion...

OpenSource

(http://opensourceforu.com)

The complete portal on open source

I'm curious how you get the address of the text section of the module? If you load this module, then the kernel oops. How you read the file /sys/module/oops/sections/.init.text?

IT ADMIN (HTTP://OPENSOURCEFORU.COM/CATEGORY/ADMIN/)

Deepak • 2 years ago

CXOS (HTTP://OPENSOURCEFORU.COM/CATEGORY/CXO/)

I want to know the meaning of "(OF)" or "(OF+)" in kernel trace. Eg: Here is one of the kernel trace i got.

FOR U & ME (HTTP://OPENSOURCEFORU.COM/CATEGORY/EVERYONE/)

Modules linked in: cxgb4(OF+) toecore(OF) ip6table_filter ip6_tables ebtable_nat ebtables nf_conntrack_ipv4 nf_defrag_ipv4 xt_state nf_conntrack

HOW-TOS (HTTP://OPENSOURCEFORU.COM/CATEGORY/HOW-TOS/)

BUZZ (HTTP://OPENSOURCEFORU.COM/CATEGORY/NEWS/)

Thanks a lot

^ | v • Reply • Share ›

ALSO ON OPEN SOURCE FOR YOU

Build your first extension for Google Chrome browser in

1 comment • 2 months ago•

Karan — This is awesome, easy to follow tutorial.

5 tools to easily develop an online shopping website

1 comment • a month ago•

Urban Mischief — Why no mention.on Woocommerce, which is one of the most

An introduction to OpenShift

1 comment • 2 months ago•

Steve Speicher — Thanks for the article, it would be great if this were updated using the

Common mistakes online entrepreneurs make while

1 comment • 3 months ago•

Curtis Haug — Creative piece ! I learned a lot from the insight ! Does someone know

TODAY

WEEK

MONTH

TAG CLOUD

LFY April 2012
(http://opensourceforu.com/tag/lfy-april-2012/) FOSS
(http://opensourceforu.com/tag/foss/) Microsoft
(http://opensourceforu.com/tag/microsoft/) ubuntu
(http://opensourceforu.com/tag/ubuntu/) lets try
(http://opensourceforu.com/tag/lets-try/) web applications
(http://opensourceforu.com/tag/web-

RECENT STORIES

(http://opensourceforu.com/2016/08/blackarch-linux-iso-now-comes-1500-hacking-tools/) **BlackArch Linux ISO now comes with over 1,500 hacking tools**
(http://opensourceforu.com/2016/08/blackarch-linux-

http://opensourceforu.com/2011/01/understanding-a-kernel-oops/

7/9



OpenSource
The complete portal on open source **For U.com**

unix (<http://opensourceforu.com/tag/unix/>)

Developers

(<http://opensourceforu.com/tag/developers/>)

PHP

(<http://opensourceforu.com/tag/php/>)

python

(<http://opensourceforu.com/tag/python/>)

RAM (<http://opensourceforu.com/tag/ram/>)

python

(<http://opensourceforu.com/tag/python/>)

Oracle (<http://opensourceforu.com/tag/oracle/>)

Apache

(<http://opensourceforu.com/tag/apache/>)

HOME ([HTTP://WWW.OPENSOURCEFORU.COM/](http://www.opensourceforu.com/))

IT ADMIN ([HTTP://OPENSOURCEFORU.COM/CATEGORY/ADMIN/](http://opensourceforu.com/category/admin/))





Creative Commons - Attribution + Noncommercial © Copyright
EFY ([HTTP://OPENSOURCEFORU.COM/CATEGORY/CXO/](http://opensourceforu.com/category/cxo/))
2016. EFY Enterprise Pvt. Ltd.

- FOR U & ME ([HTTP://OPENSOURCEFORU.COM/CATEGORY/EVERYONE/](http://opensourceforu.com/category/everyone/))

HOW-TOS ([HTTP://OPENSOURCEFORU.COM/CATEGORY/HOW-TOS/](http://opensourceforu.com/category/how-tos/))

BASICS ([HTTP://OPENSOURCEFORU.COM/CATEGORY/BASICS/](http://opensourceforu.com/category/basics/))

BUZZ ([HTTP://OPENSOURCEFORU.COM/CATEGORY/NEWS/](http://opensourceforu.com/category/news/))