

Linux-Kernel: [PATCH 2/3] kernel-doc: print arrays in declarations correctly.

[PATCH 2/3] kernel-doc: print arrays in declarations correctly.

Source: <http://linux.derkeiler.com/Mailing-Lists/Kernel/2004-10/10363.html>

From: Alexey Dobriyan (adobriyan_at_mail.ru)

Date: 10/31/04

To: sam@ravnborg.org

Date: Mon, 01 Nov 2004 00:37:57 +0300

Do not convert arrays into pointers while generating documentation for them.

I.e, print

```
struct sk_buff {
    char cb[40];
};
```

as "char cb[40]", not "char * cb".

Signed-off-by: Alexey Dobriyan <adobriyan@mail.ru>

```
--- a/scripts/kernel-doc 2004-11-01 05:57:59.000000000 +0000
+++ b/scripts/kernel-doc 2004-10-31 20:25:16.714503272 +0000
@@ -452,7 +452,10 @@
    print "<h2>".$args{'type'}." ".$args{'struct'}."</h2>\n";
    print "<b>".$args{'type'}." ".$args{'struct'}."</b> {<br>\n";
    foreach $parameter (@{$args{'parameterlist'}}) {
- ($args{'parameterdescs'}{$parameter} ne $undescribed) || next;
+ my $parameter_name = $parameter;
+ $parameter_name =~ s/\.*/;
+
+ ($args{'parameterdescs'}{$parameter_name} ne $undescribed) || next;
    $stype = $args{'parametertypes'}{$parameter};
    if ($stype =~ m/([^\(]*\(\*)s*)\s*\([^\)]*\)/) {
        # pointer-to-function
@@ -468,10 +471,13 @@
    print "<h3>Members</h3>\n";
    print "<dl>\n";
    foreach $parameter (@{$args{'parameterlist'}}) {
- ($args{'parameterdescs'}{$parameter} ne $undescribed) || next;
+ my $parameter_name = $parameter;
+ $parameter_name =~ s/\.*/;
+
+ ($args{'parameterdescs'}{$parameter_name} ne $undescribed) || next;
```

[PATCH 2/3] kernel-doc: print arrays in declarations correctly.

Linux-Kernel: [PATCH 2/3] kernel-doc: print arrays in declarations correctly.

```

    print "<dt><b>".$parameter."</b>\n";
    print "<dd>";
- output_highlight($args{'parameterdescs'}{$parameter});
+ output_highlight($args{'parameterdescs'}{$parameter_name});
    }
    print "</dl>\n";
    output_section_html(@_);
@@ -507,10 +513,13 @@
    print "<h3>Arguments</h3>\n";
    print "<dl>\n";
    foreach $parameter (@{$args{'parameterlist'}}) {
- ($args{'parameterdescs'}{$parameter} ne $undescribed) || next;
+ my $parameter_name = $parameter;
+ $parameter_name =~ s/\[.*/;
+
+ ($args{'parameterdescs'}{$parameter_name} ne $undescribed) || next;
    print "<dt><b>".$parameter."</b>\n";
    print "<dd>";
- output_highlight($args{'parameterdescs'}{$parameter});
+ output_highlight($args{'parameterdescs'}{$parameter_name});
    }
    print "</dl>\n";
    output_section_html(@_);
@@ -602,10 +611,13 @@
    if ($#{ $args{'parameterlist'} } >= 0) {
        print "<variablelist>\n";
        foreach $parameter (@{$args{'parameterlist'}}) {
+ my $parameter_name = $parameter;
+ $parameter_name =~ s/\[.*/;
+
        print "<varlistentry>\n <term><parameter>$parameter</parameter></term>\n";
        print "<listitem>\n <para>\n";
        $lineprefix=" ";
- output_highlight($args{'parameterdescs'}{$parameter});
+ output_highlight($args{'parameterdescs'}{$parameter_name});
        print "</para>\n </listitem>\n </varlistentry>\n";
    }
    print "</variablelist>\n";
@@ -644,8 +656,11 @@
    print "<programlisting>\n";
    print $args{'type'}." ".$args{'struct'}." {\n";
    foreach $parameter (@{$args{'parameterlist'}}) {
- defined($args{'parameterdescs'}{$parameter}) || next;
- ($args{'parameterdescs'}{$parameter} ne $undescribed) || next;
+ my $parameter_name = $parameter;
+ $parameter_name =~ s/\[.*/;
+
+ defined($args{'parameterdescs'}{$parameter_name}) || next;
+ ($args{'parameterdescs'}{$parameter_name} ne $undescribed) || next;
        $stype = $args{'parametertypes'}{$parameter};
        if ($stype =~ m/([^\(]*\(\*)s*\)\s*\(([\^\\]*)*)/) {

```

Linux-Kernel: [PATCH 2/3] kernel-doc: print arrays in declarations correctly.

```

# pointer-to-function
@@ -665,12 +680,15 @@
    print " <variablelist>\n";
    foreach $parameter (@{$args{'parameterlist'}}) {
- defined($args{'parameterdescs'}{$parameter}) || next;
- ($args{'parameterdescs'}{$parameter} ne $undescribed) || next;
+ my $parameter_name = $parameter;
+ $parameter_name =~ s/\[.*/;
+
+ defined($args{'parameterdescs'}{$parameter_name}) || next;
+ ($args{'parameterdescs'}{$parameter_name} ne $undescribed) || next;
    print " <varlistentry>";
    print " <term>$parameter</term>\n";
    print " <listitem><para>\n";
- output_highlight($args{'parameterdescs'}{$parameter});
+ output_highlight($args{'parameterdescs'}{$parameter_name});
    print " </para></listitem>\n";
    print " </varlistentry>\n";
    }
@@ -725,10 +743,13 @@
    print " <title>Constants</title>\n";
    print " <variablelist>\n";
    foreach $parameter (@{$args{'parameterlist'}}) {
+ my $parameter_name = $parameter;
+ $parameter_name =~ s/\[.*/;
+
    print " <varlistentry>";
    print " <term>$parameter</term>\n";
    print " <listitem><para>\n";
- output_highlight($args{'parameterdescs'}{$parameter});
+ output_highlight($args{'parameterdescs'}{$parameter_name});
    print " </para></listitem>\n";
    print " </varlistentry>\n";
    }
@@ -839,10 +860,13 @@
    print "<colspec colwidth=\"8*\">\n";
    print "<tbody>\n";
    foreach $parameter (@{$args{'parameterlist'}}) {
+ my $parameter_name = $parameter;
+ $parameter_name =~ s/\[.*/;
+
    print " <row><entry align=\"right\"><parameter>$parameter</parameter></entry>\n";
    print " <entry>\n";
    $lineprefix=" ";
- output_highlight($args{'parameterdescs'}{$parameter});
+ output_highlight($args{'parameterdescs'}{$parameter_name});
    print " </entry></row>\n";
    }
    print " </tbody></tgroup></informaltable>\n";
@@ -906,8 +930,11 @@

```

```

    print ".SH ARGUMENTS\n";
    foreach $parameter (@{$args{'parameterlist'}}) {
+ my $parameter_name = $parameter;
+ $parameter_name =~ s/[\.*/];
+
    print ".IP \"\".\"$parameter.\"\" 12\n";
    - output_highlight($args{'parameterdescs'}{$parameter});
+ output_highlight($args{'parameterdescs'}{$parameter_name});
    }
    foreach $section (@{$args{'sectionlist'}}) {
        print ".SH \"\", uc $section, \"\"\\n\"";
@@ -944,8 +971,11 @@

    print ".SH Constants\n";
    foreach $parameter (@{$args{'parameterlist'}}) {
+ my $parameter_name = $parameter;
+ $parameter_name =~ s/[\.*/];
+
    print ".IP \"\".\"$parameter.\"\" 12\n";
    - output_highlight($args{'parameterdescs'}{$parameter});
+ output_highlight($args{'parameterdescs'}{$parameter_name});
    }
    foreach $section (@{$args{'sectionlist'}}) {
        print ".SH \"$section\"\\n\"";
@@ -968,7 +998,10 @@
        print $args{'type'}." \"$args{'struct'}.\" {\\n\"";

    foreach my $parameter (@{$args{'parameterlist'}}) {
    - ($args{'parameterdescs'}{$parameter} ne $undescribed) || next;
+ my $parameter_name = $parameter;
+ $parameter_name =~ s/[\.*/];
+
+ ($args{'parameterdescs'}{$parameter_name} ne $undescribed) || next;
        print "\\n.br\\n\"";
        $type = $args{'parametertypes'}{$parameter};
        if ($type =~ m/([^\(\)*\(\)*\s*\)\s*\(([\^\)]*)\)/) {
@@ -986,9 +1019,12 @@

    print ".SH Arguments\n";
    foreach $parameter (@{$args{'parameterlist'}}) {
    - ($args{'parameterdescs'}{$parameter} ne $undescribed) || next;
+ my $parameter_name = $parameter;
+ $parameter_name =~ s/[\.*/];
+
+ ($args{'parameterdescs'}{$parameter_name} ne $undescribed) || next;
        print ".IP \"\".\"$parameter.\"\" 12\n";
    - output_highlight($args{'parameterdescs'}{$parameter});
+ output_highlight($args{'parameterdescs'}{$parameter_name});
    }
    foreach $section (@{$args{'sectionlist'}}) {

```

Linux–Kernel: [PATCH 2/3] kernel–doc: print arrays in declarations correctly.

```

    print ".SH \"$section\"\n";
@@ -1055,7 +1091,10 @@

    print "Arguments:\n\n";
    foreach $parameter (@{$args{'parameterlist'}}) {
- print $parameter."\n\t".$args{'parameterdescs'}{$parameter}."\n";
+ my $parameter_name = $parameter;
+ $parameter_name =~ s/^[.*/];
+
+ print $parameter."\n\t".$args{'parameterdescs'}{$parameter_name}."\n";
    }
    output_section_text(@_);
}
@@ -1120,7 +1159,10 @@
    print $args{'type'}." ".$args{'struct'}.":\n\n";
    print $args{'type'}." ".$args{'struct'}." {\n";
    foreach $parameter (@{$args{'parameterlist'}}) {
- ($args{'parameterdescs'}{$parameter} ne $undescribed) || next;
+ my $parameter_name = $parameter;
+ $parameter_name =~ s/^[.*/];
+
+ ($args{'parameterdescs'}{$parameter_name} ne $undescribed) || next;
        $type = $args{'parametertypes'}{$parameter};
        if ($type =~ m/([^\(\)*\s*\)\s*\(([\^\\]*)\)/) {
            # pointer-to-function
@@ -1135,9 +1177,12 @@

    print "Members:\n\n";
    foreach $parameter (@{$args{'parameterlist'}}) {
- ($args{'parameterdescs'}{$parameter} ne $undescribed) || next;
+ my $parameter_name = $parameter;
+ $parameter_name =~ s/^[.*/];
+
+ ($args{'parameterdescs'}{$parameter_name} ne $undescribed) || next;
        print "$parameter\n\t";
- print $args{'parameterdescs'}{$parameter}."\n";
+ print $args{'parameterdescs'}{$parameter_name}."\n";
    }
    print "\n";
    output_section_text(@_);
@@ -1317,10 +1362,8 @@
        $type =~ s/([^\(\)*\s*\)\s*\(([\^\\]*)\)$param/$1/;
        push_parameter($param, $type, $file);
    } else {
- # evil magic to get fixed array parameters to work
- $arg =~ s/(.+s+)(.+)\.*/$1* $2/;
-
        $arg =~ s/s*\s*/:/g;
+ $arg =~ s/s*\[/[/g;

        my @args = split("\s*,\s*", $arg);

```

Linux-Kernel: [PATCH 2/3] kernel-doc: print arrays in declarations correctly.

```
    if ($args[0] =~ m/^*/) {
@@ -1350,6 +1393,9 @@
    my $type = shift;
    my $file = shift;

+ my $param_name = $param;
+ $param_name =~ s/^[.*/];
+
    if ($type eq "" && $param eq "...")
    {
        $type="...";
@@ -1362,8 +1408,8 @@
        $param="void";
        $parameterdescs{void} = "no arguments";
    }
- if (defined $type && $type && !defined $parameterdescs{$param}) {
- $parameterdescs{$param} = $undescribed;
+ if (defined $type && $type && !defined $parameterdescs{$param_name}) {
+ $parameterdescs{$param_name} = $undescribed;

    if (($type eq 'function') || ($type eq 'enum')) {
        print STDERR "Warning(${file}): Function parameter ".

```

–
To unsubscribe from this list: send the line "unsubscribe linux-kernel" in
the body of a message to majordomo@vger.kernel.org
More majordomo info at <http://vger.kernel.org/majordomo-info.html>
Please read the FAQ at <http://www.tux.org/lkml/>