

Hello World! My name is Panda!

And I am a PDF generator

I handle multiple line text ok, once you have set a loading

A line in **ca-Bold**

INFRONTINFRONTINFRONT

clear
clear SAL
delete BPRAN
delete everything

A second textbox on the page

The background image on this page is Copyright 2000 Andrew Cagney and is distributed under the terms of the GPL...

Breakpoints - conditions

Also watchpoints and catchpoints.

break SAL if EXPRESSION
condition BPNUM [EXPRESSION]

Yes, any expression:

```
(gdb) list print
3 void
4 print (const char *s)
5 {
6   printf ("%s", s);
7 }
(gdb) break print if strcmp (s, "22\n") == 0
Breakpoint 2 at 0x804819b: file lines.c, line 6.
(gdb) c
Continuing.
21
Breakpoint 2, print (s=0x807ba88 "22\n") at lines.c:6
6   printf ("%s", s);
(gdb)
```

condition BPNUM - remove expr

Breakpoint scrip

```
(gdb) break EXPRESSION
(gdb) commands $bpnum
silent
print s
end
(gdb)
```

IDBEHIND

Examining values

print/FMT EXPRESSION

heX, Decimal, Unsigned, Octal, Two, Address, Character, Float

```
(gdb) print i++
$2 = 4
```

Controlling Breakpoints et al.

```
ignore BPNUM COUNT
enable BPRANGE
disable BPRANGE
```

Deleting

```
clear everything
clear SAL
delete BPRANGE
delete everything
```

+ Watchpoint
+ Catchpoint
\$bpnum is the next recent b/p set.

Breakpo

Also watchpoints a

break SAL if EXPRESSION
condition BPNUM [EXPRESSION]

Yes, any expression

```
(gdb) list print
3 void
4 print (const char *s)
5 {
6   printf ("%s", s);
7 }
(gdb) break print if s
Breakpoint 2 at 0x804819b: file lines.c, line 6.
(gdb) c
Continuing.
21
Breakpoint 2, print (s=0x807ba88 "22\n") at lines.c:6
6   printf ("%s", s);
(gdb)
```



condition BPNUM - remove expr

Breakpoint s

```
(gdb) break EXPRESSION
(gdb) commands $bpnum
silent
print s
end
(gdb)
```

- Use s to keep GDB quiet
- Use s to keep GDB

Examining values

print/FMT

heX, Decimal, Unsigned, Octal, Two, Address, Character, Float

```
(gdb) print i++
$2 = 4
(gdb) print &i
$3 = (int *) 0xbffffa34
(gdb) print/x i
$4 = 0x4
```

set output-radix 16

Hello World! My name is Panda!
And I am a PDF generator
I handle multiple line text ok .once you have set a leading.
A line INFRONT OF INFRONT

Breakpoint scrip
(gdb) break EXPRESSION
(gdb) commands \$bpnum
silent
print s
end
(gdb)

A second textbox on the page
• Use continue to resume GDB

The background image on this page is Copyright 2000 Andrew Cagney
and is distributed under the terms of the GPL...

Breakpoints - conditions
Also watchpoints and catchpoints.
break SAL if EXPRESSION
condition BPNUM [EXPRESSION]
Yes, any expression:
(gdb) list print
3 void
4 print (const char *s)
5 {
6 printf ("%s", s);
7 }
(gdb) break print if strcmp (s, "22\n") == 0
Breakpoint 2 at 0x804819b: file lines.c, line 6.
(gdb) c
Continuing.
21
Breakpoint 2, print (s=0x807ba88 "22\n") at lines.c:6
6 printf ("%s", s);
(gdb)

Examining values
print/FMT EXPRESSION
heX, Decimal, Unsigned, Octal, Two, Address, Character, Float
(gdb) print i++
\$2 = 4

Breakpo
Also watchpoints a
break SAL if EXPRESSION
condition BPNUM [EXPR
Yes, any expression
(gdb) list print
3 void
4 print (const char *s
5 {
6 printf ("%s", s);
7 }
(gdb) break print if s
Breakpoint 2 at 0x804819b: file lines.c, line 6.
(gdb) c
Continuing.
21
Breakpoint 2, print (s=0x807ba88 "22\n") at lines.c:6
6 print
(gdb)

ing values
print/FMT
heX, Decim Unsigned, Octal, Two, Address, Character, Float
(gdb) print i++
\$2 = 4
(gdb) print &i
\$3 = (int *) 0xbffffa34
(gdb) print/x i
\$4 = 0x4
set output-radix 16

Controlling Breakpoints et al.

Hello World! My name is Panda!

And I am a PDF generator

I handle line text ok once you have set a leading.

A line Helvetica-Bold

everything

45

Breakpoints - conditions

Also watchpoints and catchpoints.

```
break SAL if EXPRESSION
condition BPNUM [ EXPRESSION ]
```

Yes, any expression:

```
(gdb) list print
3 void
4 print (const char *s)
5 {
6     printf ("%s", s);
7 }
(gdb) break print if strcmp (s, "22\n") == 0
Breakpoint 2 at 0x804819b: file lines.c, line 6.
(gdb) c
Continuing.
21
Breakpoint 2, print (s=0x807ba88 "22\n") at lines.c:6
6     printf ("%s", s);
(gdb)
```

condition BPNUM - remove expr

46

Breakpoint script

```
(gdb) break EXPRESSION
(gdb) commands $bpnum
silent
print s
end
(gdb)
```

A second textbox on the page

- Use continue to resume GDB

47

BEHIND

Examining values

```
print/FMT EXPRESSION
```

heX, Decimal, Unsigned, Octal, Two, Address, Character, Float

```
(gdb) print i++
$2 = 4
```

48

Controlling Breakpoints et al.

ignore BPNUM COUNT

enable BPRANGE
disable BPRANGE

Deleting

clear everything

clear SAL
delete BPRANGE
delete everything

Watchpoint
Catchpoint
\$bpnum is the next recent b/p set.

Breakpoints

Also watchpoints and catchpoints.

```
break SAL if EXPRESSION
condition BPNUM [ EXPRESSION ]
```

Yes, any expression:

```
(gdb) list print
3 void
4 print (const char *s)
5 {
6     printf ("%s", s);
7 }
(gdb) break print if strcmp (s, "22\n") == 0
Breakpoint 2 at 0x804819b: file lines.c, line 6.
(gdb) c
Continuing.
21
Breakpoint 2, print (s=0x807ba88 "22\n") at lines.c:6
6     printf ("%s", s);
(gdb)
```

condition BPNUM - remove expr

46

Breakpoint script

```
(gdb) break EXPRESSION
(gdb) commands $bpnum
silent
print s
end
(gdb)
```

- Use silent to keep GDB quiet

- Use continue to resume GDB

47

Examining values

```
print/FMT EXPRESSION
```

heX, Decimal, Unsigned, Octal, Two, Address, Character, Float

```
(gdb) print i++
$2 = 4
(gdb) print &i
$3 = (int *) 0xbffffa34
(gdb) print/x i
$4 = 0x4
```

```
set output-radix 16
```

48

The background image on this page is Copyright 2000 Andrew and is distributed under the terms of the GPL...

Controlling Breakpoints et al.

ignore BPNUM COUNT

enable BPRANGE
disable BPRANGE

Deleting

clear
clear SAL
delete BPRAN
delete



everything

+ Watchpoint
+ Catchpoints
\$bpnum is the next
recent b/p set.

45

Breakpoints - conditions

Also watchpoints and catchpoints.

break SAL if EXPRESSION
condition BPNUM [EXPRESSION]

Yes, any expression:

```
(gdb) list print
3 void
4 print (const char *s)
5 {
6     printf ("%s", s);
7 }
(gdb) break print if strcmp (s, "22\n") == 0
Breakpoint 2 at 0x804819b: file lines.c, line 6.
(gdb) c
Continuing.
21
Breakpoint 2, print (s=0x807ba88 "22\n") at lines.c:6
6     printf ("%s", s);
(gdb)
```

condition BPNUM - remove
expr

46

Breakpoint script

```
(gdb) break EXPRESSION
(gdb) commands $bpnum
silent
print s
end
(gdb)
```

- Use silent to keep GDB quiet
- Use continue to resume GDB

47

Examining values

print/FMT EXPRESSION

heX, Decimal, Unsigned, Octal, Two,
Address, Character, Float

```
(gdb) print i++
$2 = 4
```

48

Controlling Breakpoints et al.

ignore BPNUM COUNT

enable BPRANGE
disable BPRANGE

Deleting

clear
clear SAL
delete BPRANGE
delete

everything

+ Watchpoint
+ Catchpoints
\$bpnum is the next
recent b/p set.

Breakpoints

Also watchpoints and catchpoints.

break SAL if EXPRESSION
condition BPNUM [EXPRESSION]

Yes, any expression:

```
(gdb) list print
3 void
4 print (const char *s)
5 {
6     printf ("%s", s);
7 }
(gdb) break print if strcmp (s, "22\n") == 0
Breakpoint 2 at 0x804819b: file lines.c, line 6.
(gdb) c
Continuing.
21
Breakpoint 2, print (s=0x807ba88 "22\n") at lines.c:6
6     printf ("%s", s);
(gdb)
```

condition BPNUM - remove
expr

46

Breakpoint script

```
(gdb) break EXPRESSION
(gdb) commands $bpnum
silent
print s
end
(gdb)
```

- Use silent to keep GDB quiet
- Use continue to resume GDB

Examining values

print/FMT

heX, Decimal, Unsigned, Octal, Two,
Address, Character, Float

```
(gdb) print i++
$2 = 4
(gdb) print &i
$3 = (int *) 0xbffffa34
(gdb) print/x i
$4 = 0x4
```

set output-radix 16

47

48

Controlling Breakpoints et al.

ignore BPNUM COUNT

enable BPRANGE

disable BPRANGE

Deleting

clear

clear SAL

delete BPRANGE

delete

\$bpnum is the next recent b/p set.

Hello World! My name is Panda!
And I am a PDF generator
I handle multiple line text ok
A line in Helvetica-Bold

Breakpoint scripts

(gdb) break EXPRESSION

(gdb) commands \$bpnum

silent

print s

end

(gdb)

- Use silent to keep GDB quiet
- Use continue to resume GDB

47

Controlling Breakpoints et al.

ignore BPNUM COUNT

enable BPRANGE

disable BPRANGE

Deleting

clear

clear SAL

delete BPRANGE

delete

\$bpnum is the next recent b/p set.

Breakpoint scripts

(gdb) break EXPRESSION

(gdb) commands \$bpnum

silent

print s

end

(gdb)

- Use silent to keep GDB quiet
- Use continue to resume GDB

47

Breakpoints - conditions

Also watchpoints and catchpoints.

break SAL if EXPRESSION
condition BPNUM [EXPRESSION]

Yes, any expression:

(gdb) list print

3 void

4 print (const char *s)

5 {

6 printf ("%s", s);

7 }

(gdb) break print if strcmp (s, "22\n") == 0

Breakpoint 2 at 0x804819b: file lines.c, line 6.

(gdb) c

Continuing.

21

Breakpoint 2, print (s=0x807ba88 "22\n") at lines.c:6

6 printf ("%s", s);

(gdb)

condition BPNUM - remove expr

46

Examining values

print/FMT EXPRESSION

hex, Decimal, Unsigned, Octal, Two, Address, Character, Float

(gdb) print i++

\$2 = 4

(gdb) print &i

\$3 = (int *) 0xbffffa34

(gdb) print/x i

\$4 = 0x4

set output-radix 16

48

Breakpoints - conditions

Also watchpoints and catchpoints.

break SAL if EXPRESSION
condition BPNUM [EXPRESSION]

Yes, any expression:

(gdb) list print

3 void

4 print (const char *s)

5 {

6 printf ("%s", s);

7 }

(gdb) break print if strcmp (s, "22\n") == 0

Breakpoint 2 at 0x804819b: file lines.c, line 6.

(gdb) c

Continuing.

21

Breakpoint 2, print (s=0x807ba88 "22\n") at lines.c:6

6 printf ("%s", s);

(gdb)

condition BPNUM - remove expr

46

Examining values

print/FMT EXPRESSION

hex, Decimal, Unsigned, Octal, Two, Address, Character, Float

(gdb) print i++

\$2 = 4

(gdb) print &i

\$3 = (int *) 0xbffffa34

(gdb) print/x i

\$4 = 0x4

set output-radix 16

48

Controlling Breakpoints et al.

ignore BPNUM COUNT

enable BPRANGE

disable BPRANGE

Deleting

clear *everything*

clear *World!*

delete BPRANGE

delete

everything

am

PDF

generator

line

text

ok

once you have set a

condition BPNUM - remove expr

A line in Helvetica-Bold

*+ Watchpoint
+ Catchpoint
\$bpnum is the next
recent b/p set.*

Breakpoint scripts

(gdb) break EXPRESSION

(gdb) commands \$bpnum

silent

print s

end

(gdb)

• Use silent to keep GDB quiet

• Use continue to resume GDB

47

Breakpoints - conditions

Also watchpoints and catchpoints.

break SAL if EXPRESSION
condition BPNUM [EXPRESSION]

Yes, any expression:

(gdb) list print

3 void

4 print (const char *s)

5 {

6 printf ("%s", s);

7 }

(gdb) break print if strcmp (s, "22\n") == 0

Breakpoint 2 at 0x804819b: file lines.c, line 6.

(gdb) c

Continuing.

21

Breakpoint 2, print (s=0x807ba88 "22\n") at lines.c:6

6 printf ("%s", s);

(gdb)

condition BPNUM - remove expr

once you have set a

condition BPNUM - remove expr

once you have set a

condition BPNUM - remove expr

Examining values

print/FMT EXPRESSION

hex, Decimal, Unsigned, Octal, Two,
Address, Character, Float

(gdb) print i++

\$2 = 4

(gdb) print &i

\$3 = (int *) 0xbffffa34

(gdb) print/x i

\$4 = 0x4

set output-radix 16

48

Controlling Breakpoints et al.

ignore BPNUM COUNT

enable BPRANGE

disable BPRANGE

Deleting

clear *everything*

clear SAL

delete BPRANGE

delete

everything

*+ Watchpoint
+ Catchpoint
\$bpnum is the next
recent b/p set.*

Breakpoint scripts

(gdb) break EXPRESSION

(gdb) commands \$bpnum

silent

print s

end

(gdb)

• Use silent to keep GDB quiet

• Use continue to resume GDB

47

Breakpoints - conditions

Also watchpoints and catchpoints.

break SAL if EXPRESSION
condition BPNUM [EXPRESSION]

Yes, any expression:

(gdb) list print

3 void

4 print (const char *s)

5 {

6 printf ("%s", s);

7 }

(gdb) break print if strcmp (s, "22\n") == 0

Breakpoint 2 at 0x804819b: file lines.c, line 6.

(gdb) c

Continuing.

21

Breakpoint 2, print (s=0x807ba88 "22\n") at lines.c:6

6 printf ("%s", s);

(gdb)

condition BPNUM - remove expr

once you have set a

condition BPNUM - remove expr

once you have set a

condition BPNUM - remove expr

Examining values

print/FMT EXPRESSION

hex, Decimal, Unsigned, Octal, Two,
Address, Character, Float

(gdb) print i++

\$2 = 4

(gdb) print &i

\$3 = (int *) 0xbffffa34

(gdb) print/x i

\$4 = 0x4

set output-radix 16

48

Controlling Breakpoints et al.

ignore BPNUM COUNT

enable BPRANGE
disable BPRANGE

Deleting

clear *everything*
clear SAL
delete BPRANGE
delete *everything*

*+ Watchpoint
+ Catchpoint*
\$bpnum is the next
recent b/p set.

Hello World! *My* name is Panda!
And I am a PDF generator
I handle multiple line text ok
A line in Helvetica-Bold

Breakpoint scripts

```
(gdb) break EXPRESSION
(gdb) commands $bpnum
silent
print s
end
(gdb)
```

- Use silent to keep GDB quiet
- Use continue to resume GDB

47

Breakpoints - conditions

Also watchpoints and catchpoints.

break SAL if EXPRESSION
condition BPNUM [EXPRESSION]

Yes, any expression:

```
(gdb) list print
3 void
4 print (const char *s)
5 {
6     printf ("%s", s);
7 }
(gdb) break print if strcmp (s, "22\n") == 0
Breakpoint 2 at 0x804819b: file lines.c, line 6.
(gdb) c
Continuing.
21
Breakpoint 2, print (s=0x807ba88 "22\n") at lines.c:6
6     printf ("%s", s);
(gdb)
```

condition BPNUM - remove
once you have set a leading.

Examining values

print/FMT EXPRESSION

hex, Decimal, Unsigned, Octal, Two, Address, Character, Float

```
(gdb) print i++
$2 = 4
(gdb) print &i
$3 = (int *) 0xbffffa34
(gdb) print/x i
$4 = 0x4
```

set output-radix 16

48

Controlling Breakpoints et al.

ignore BPNUM COUNT

enable BPRANGE
disable BPRANGE

Deleting

clear *everything*
clear SAL
delete BPRANGE
delete *everything*

*+ Watchpoint
+ Catchpoint*
\$bpnum is the next
recent b/p set.

Breakpoint scripts

```
(gdb) break EXPRESSION
(gdb) commands $bpnum
silent
print s
end
(gdb)
```

- Use silent to keep GDB quiet
- Use continue to resume GDB

47

Breakpoints - conditions

Also watchpoints and catchpoints.

break SAL if EXPRESSION
condition BPNUM [EXPRESSION]

Yes, any expression:

```
(gdb) list print
3 void
4 print (const char *s)
5 {
6     printf ("%s", s);
7 }
(gdb) break print if strcmp (s, "22\n") == 0
Breakpoint 2 at 0x804819b: file lines.c, line 6.
(gdb) c
Continuing.
21
Breakpoint 2, print (s=0x807ba88 "22\n") at lines.c:6
6     printf ("%s", s);
(gdb)
```

condition BPNUM - remove
once you have set a leading.

Examining values

print/FMT EXPRESSION

hex, Decimal, Unsigned, Octal, Two, Address, Character, Float

```
(gdb) print i++
$2 = 4
(gdb) print &i
$3 = (int *) 0xbffffa34
(gdb) print/x i
$4 = 0x4
```

set output-radix 16

48

Controlling Breakpoints et al.

ignore BPNUM COUNT

enable BPRANGE
disable BPRANGE

Deleting

clear *everything*
clear SAL
delete BPRANGE
delete *everything*

*+ Watchpoint
+ Catchpoint*
\$bpnum is the next
recent b/p set.

45

Breakpoint scripts

```
(gdb) break EXPRESSION
(gdb) commands $bpnum
silent
print s
end
(gdb)
```

- Use silent to keep GDB quiet
- Use continue to resume GDB

47

Breakpoints - conditions

Also watchpoints and catchpoints.

break SAL if EXPRESSION
condition BPNUM [EXPRESSION]

Yes, any expression:

```
(gdb) list print
3 void
4 print (const char *s)
5 {
6     printf ("%s", s);
7 }
(gdb) break print if strcmp (s, "22\n") == 0
Breakpoint 2 at 0x804819b: file lines.c, line 6.
(gdb) c
Continuing.
21
Breakpoint 2, print (s=0x807ba88 "22\n") at lines.c:6
6     printf ("%s", s);
(gdb)
```

condition BPNUM - remove expr

46

Examining values

print/FMT EXPRESSION

hex, Decimal, Unsigned, Octal, Two, Address, Character, Float

```
(gdb) print i++
$2 = 4
(gdb) print &i
$3 = (int *) 0xbffffa34
(gdb) print/x i
$4 = 0x4
```

set output-radix 16

48

Controlling Breakpoints et al.

ignore BPNUM COUNT

enable BPRANGE
disable BPRANGE

Deleting

clear *everything*
clear SAL
delete BPRANGE
delete *everything*

*+ Watchpoint
+ Catchpoint*
\$bpnum is the next
recent b/p set.

45

Breakpoint scripts

```
(gdb) break EXPRESSION
(gdb) commands $bpnum
silent
print s
end
(gdb)
```

- Use silent to keep GDB quiet
- Use continue to resume GDB

47

Breakpoints - conditions

Also watchpoints and catchpoints.

break SAL if EXPRESSION
condition BPNUM [EXPRESSION]

Yes, any expression:

```
(gdb) list print
3 void
4 print (const char *s)
5 {
6     printf ("%s", s);
7 }
(gdb) break print if strcmp (s, "22\n") == 0
Breakpoint 2 at 0x804819b: file lines.c, line 6.
(gdb) c
Continuing.
21
Breakpoint 2, print (s=0x807ba88 "22\n") at lines.c:6
6     printf ("%s", s);
(gdb)
```

condition BPNUM - remove expr

46

Examining values

print/FMT EXPRESSION

hex, Decimal, Unsigned, Octal, Two, Address, Character, Float

```
(gdb) print i++
$2 = 4
(gdb) print &i
$3 = (int *) 0xbffffa34
(gdb) print/x i
$4 = 0x4
```

set output-radix 16

48