

[Full-disclosure] Mtr – remote and local stack overflow – uncoment situation in libresolv.

Source: <http://www.derkeiler.com/Mailing-Lists/Full-Disclosure/2008-05/msg00494.html>

- *From:* pi3@xxxxxxxx (Adam Zabrocki)
 - *Date:* Mon, 19 May 2008 23:13:28 +0200
-

Name: Mtr – network diagnostic tool.

Author: Adam Zabrocki <pi3@xxxxxxxx> or <pi3ki31ny@xxxxxxxx>

Date: February 28, 2008

Issue:

Mtr allows local and remote attackers to overflow buffer on stack.

Description:

Mtr combines the functionality of the traceroute and ping programs in a single network diagnostic tool. For more detail please read manual page.

Details:

It is possible to overflow buffor on stack in suid program – mtr. Remote attack is possible too. Bug is in function which print result of runing program with parametr 'split' (-p). Victim must use DNS which we can control or we can try exploit this vulnerability by spoofing technique. In remote exploiting this vulnerability we must know which IP user gave to program – or he must simply run program and argument must be IP adres which we can controle in DNS server.

Look for this code:

```
"split.c"  
#define MAX_LINE_SIZE 256  
  
void split_redraw(void)  
{  
int max;  
int at;  
ip_t *addr;  
char *name;  
char newLine[MAX_LINE_SIZE];
```

```
int i;

...

for(at = 0; at < max; at++) {
addr = net_addr(at);

if( addrcmp( (void *) addr, (void *) &unspec_addr, af ) != 0 ) {
name = dns_lookup(addr); [1]
if(name != NULL) {
/* May be we should test name's length */ [!!]
sprintf(newLine, "%s %d %d %d %d %d %d", name, [2]
net_loss(at),
net_returned(at), net_xmit(at),
net_best(at)/1000, net_avg(at)/1000,
net_worst(at)/1000);
} else {
...
sprintf(newLine, "???");
}

...
}
}
```

As we can see in [2] there is unsecure call for function `sprintf()`. Argument 'name' is RevDNS for IP address. In details exploiting this situation will be later because normal we can't do that!

Now let's look what call this function:

```
"display.c"
void display_redraw(void)
{
switch(DisplayMode) {

...
case DisplaySplit: /* BL */
split_redraw();
break;
...
}
}
```

Call for function `display_redraw()` is here:

```
"select.c"
void select_loop(void) {

...
}
```

```
while(1) {
```

```
...<
```

That's all. I test it on version 0.72 and 0.69. Probably all versions are vulnerability. Thanks and Best regards Adam Zabrocki (pi3 / pi3ki31ny).

Ps2. For program autors. You wrote in [!]:

```
/* May be we should test name's length */
```

the answer is:

```
"Yes you should ;-)"
```

```
--
```

pi3 (pi3ki31ny) – pi3 (at) itsec pl

<http://pi3.hack.pl>

<http://pi3.phrack.pl>

<http://pi3.shellcode.pl>

<http://pi3.itsec.pl>

Full-Disclosure – We believe in it.

Charter: <http://lists.grok.org.uk/full-disclosure-charter.html>

Hosted and sponsored by Secunia – <http://secunia.com/>