

Report SNAFU - Abenteuer Vielzeiler

„The three most dangerous things are a programmer with a soldering
iron, a manager who codes, and a user who gets ideas.“

Florian Westphal - Hagen Paul Pfeifer

8. Juli 2005

Themenübersicht

1. /dev/fun
2. Error-Handling für Profis
3. Behandlung externe Eingaben
4. Race Conditions

Code Stereogramm

nach ca. 2 Minuten denkt man, man wäre auf psychoaktiven Drogen ...

tcp.c bei Solaris

```
tcp->tcp_conn.tcp_eager_conn_ind = NULL;
ASSERT(listener->tcp_conn_req_cnt_q0 > 0);
listener->tcp_conn_req_cnt_q0--;
listener->tcp_conn_req_cnt_q++;
tcp->tcp_eager_next_q0->tcp_eager_prev_q0=
tcp->tcp_eager_prev_q0;
tcp->tcp_eager_prev_q0->tcp_eager_next_q0=
tcp->tcp_eager_next_q0;
tcp->tcp_eager_prev_q0 = NULL;
tcp->tcp_eager_next_q0 = NULL;
tcp->tcp_conn_def_q0 = B_FALSE;
```

Funktionen verstehen

```
fprintf(stderr, "Usage:      elfdump <file> <output data>\n");
fprintf(stderr, "\nValid options for <output data> are:\n");
fprintf(stderr, "  --basic      Displays just the ELF version, file "
    "type and target architecture\n");
fprintf(stderr, "  --depend     Displays Shared Object dependency"
    " information\n");
fprintf(stderr, "  --dynamic    Displays information regarding dynamic"
    " linking requirements\n");
fprintf(stderr, "  --dynsym     Displays information regarding dynamic"
    " linking symbols\n");
fprintf(stderr, "  --exports    Displays exported symbol information\n");
fprintf(stderr, "  --files      Displays file information for the"
    " binary\n");
fprintf(stderr, "  --funcrep    Displays information about functions"
    " in a column format\n");
fprintf(stderr, "  --got        Displays the values contained within the"
    " Global Offset Table\n");
fprintf(stderr, "  --interp     Displays the name of the executable"
    " that interprets this file\n");
fprintf(stderr, "  --reloc      Displays relocation information"
    " (no addends)\n");
fprintf(stderr, "  --secrep     Displays section information in a"
```

```
    " column format\n");
fprintf(stderr, "  --sections    Displays section information\n");
fprintf(stderr, "  --segments    Displays segment information\n");
fprintf(stderr, "  --segrep      Displays segment information in a"
    " column format\n");
fprintf(stderr, "  --symbols     Displays symbol information\n");
fprintf(stderr, "  --symrep     Displays symbol information in a"
    " column format\n");
fprintf(stderr, "\nNote that the following options only work for"
    " non-stripped binaries:\n");
fprintf(stderr, "  --symbols, --symrep, --exports, --funcrep, --files\n");
```

Verwenden sie sprechende Namen!

```
private void replaceAttributeRequiredInXMLTaskNodeWithAttributeRequiredInHTMLNode(  
    Node aTargetTaskNode, Node aSourceTaskNode) {  
    [...]  
}  
  
[...]  
  
private void replaceAttributeRequiredInAttributeNodeWithValueFoundInNodeVector(  
    Node aTargetAttributeNode, Vector aSourceAttributeVector) {  
    [...]  
}
```

switch für Experten

```
int dl_tmap_num_unique(short dl_tmap_num) {
  switch (dl_tmap_num) {
    case 0: case 2: case 4: case 5: case 6: case 7: case 9:
    case 10: case 11: case 12: case 17: case 18:
    case 20: case 21: case 25: case 28:
    case 38: case 39: case 41: case 44: case 49:
    case 50: case 55: case 57: case 88:
    case 132: case 141: case 147:
    case 154: case 155: case 158: case 159:
    case 160: case 161: case 167: case 168: case 169:
    case 170: case 171: case 174: case 175: case 185:
    case 193: case 194: case 195: case 198: case 199:
    case 200: case 202: case 210: case 211:
    case 220: case 226: case 227: case 228: case 229: case 230:
    case 240: case 241: case 242: case 243: case 246:
    case 250: case 251: case 252: case 253: case 257: case 258: case 259:
    case 260: case 263: case 266: case 283: case 298:
    case 315: case 317: case 319: case 320: case 321:
    case 330: case 331: case 332: case 333: case 349:
    case 351: case 352: case 353: case 354:
    case 355: case 357: case 358: case 359:
    case 362: case 370: return 1;
    default: return 0;
  }
}
```

Hä?

```
if (dl_tmap_num >= 29 && dl_tmap_num <= 37)
    return dl_tmap_num - 16;
if (dl_tmap_num >= 58 && dl_tmap_num <= 87)
    return dl_tmap_num - 24;
if (dl_tmap_num >= 89 && dl_tmap_num <= 131)
    return dl_tmap_num - 25;
if (dl_tmap_num >= 133 && dl_tmap_num <= 140)
    return dl_tmap_num - 26;
if (dl_tmap_num >= 176 && dl_tmap_num <= 184)
    return dl_tmap_num + 33;
if (dl_tmap_num >= 186 && dl_tmap_num <= 192)
    return dl_tmap_num + 32;
if (dl_tmap_num >= 203 && dl_tmap_num <= 209)
    return dl_tmap_num + 25;
if (dl_tmap_num >= 212 && dl_tmap_num <= 219)
    return dl_tmap_num + 23;
if (dl_tmap_num >= 231 && dl_tmap_num <= 239)
    return dl_tmap_num + 18;
if (dl_tmap_num >= 267 && dl_tmap_num <= 281)
    return dl_tmap_num + 10;
if (dl_tmap_num >= 287 && dl_tmap_num <= 297)
    return dl_tmap_num + 13;
if (dl_tmap_num >= 299 && dl_tmap_num <= 314)
    return dl_tmap_num + 12;
```


Kreativer Umgang mit *\$#! Hardware ...

```
* IOC3 is fucked fucked beyond believe ... Don't even give the
* IOC3 is fucked fucked beyond believe ... Don't even give the
* IOC3 is fucked fucked beyond believe ... Don't try to access
* IOC3 is fucked fucked beyond believe ... Don't even give the
* IOC3 is fucked fucked beyond believe ... Don't even give the
* IOC3 is fucked fucked beyond believe ... Don't try to access
* IOC3 is fucked fucked beyond believe ... Don't even give the
* IOC3 is fucked fucked beyond believe ... Don't even give the
* IOC3 is fucked fucked beyond believe ... Don't try to access
* IOC3 is fucked fucked beyond believe ... Don't even give the
* IOC3 is fucked fucked beyond believe ... Don't even give the
* IOC3 is fucked fucked beyond believe ... Don't try to access
```

Bye the way

```
$ egrep -rin ".*fuck.*" /usr/src/linux-2.6.12-rc6-mm1-kocotpus | wc -l  
57  
$ egrep -rn ".*FIXME.*" /usr/src/linux-2.6.12-rc6-mm1-kocotpus | wc -l  
3097
```

Ratespiel

```
if ((in[0] == 0x00) && (in[1] == 0x00) &&
    (in[2] == 0x00) && (in[3] == 0x3C))
    return(XML_CHAR_ENCODING_UCS4BE);
if ((in[0] == 0x3C) && (in[1] == 0x00) &&
    (in[2] == 0x00) && (in[3] == 0x00))
    return(XML_CHAR_ENCODING_UCS4LE);
if ((in[0] == 0x00) && (in[1] == 0x00) &&
    (in[2] == 0x3C) && (in[3] == 0x00))
    return(XML_CHAR_ENCODING_UCS4_2143);
if ((in[0] == 0x00) && (in[1] == 0x3C) &&
    (in[2] == 0x00) && (in[3] == 0x00))
    return(XML_CHAR_ENCODING_UCS4_3412);
if ((in[0] == 0xFE) && (in[1] == 0xFF))
    return(XML_CHAR_ENCODING_UTF16BE);
if ((in[0] == 0xFF) && (in[1] == 0xFE))
    return(XML_CHAR_ENCODING_UTF16LE);
```

... oder doch massiv dokumentieren ...

```
libpcap-2005.06.18/pcap-linux.c
```

```
calc $(egrep -rn ".*[/*] .*" pcap-linux.c|wc -l) \  
      / ($(cat pcap-linux.c|wc -l)/100)  
30.2903
```

... oder einfach nur unwartbar?!

```
#ifdef SUPPORT_PROXY
static int add_proxy(const char* c) {
struct cgi_proxy* x=malloc(sizeof(struct cgi_proxy));
int i;
if (!x) return -1;
byte_zero(x,sizeof(struct cgi_proxy));
i=scan_ip6if(c,x->ip,&x->scope_id);
if (c[i]!='/') { nixgut: free(x); return -1; }
c+=i+1;
i=scan_ushort(c,&x->port);
if (c[i]!='/') goto nixgut;
c+=i+1;
if (regcomp(&x->r,c,REG_EXTENDED|REG_NOSUB))
goto nixgut;
if (!last)
    cgis=last=x;
else
```

fw - hgn

8. Juli 2005

```
    last->next=x; last=x;  
    return 0;  
}
```

Systemcalls liefern nie Fehler ...

```
fd = socket (AF_UNIX, SOCK_STREAM, 0);  
[...]  
if (bind (fd, (struct sockaddr*) &cli_adr, length) < 0) {  
    perror ("bind");  
    exit (EXIT_FAILURE);  
}
```

Pufferlängenberechnung

```
p = (char *)fs_get((strlen(nick ? nick : "Alternate Role") +
    strlen(comment ? comment : "") +
    strlen(to_pat ? to_pat : "") +
    strlen(from_pat ? from_pat : "") +
    strlen(sender_pat ? sender_pat : "") +
    strlen(cc_pat ? cc_pat : "") +
[.. 22 weitere solcher Zeilen entfernt .. ]
    strlen(fcc_act ? fcc_act : "") +
    strlen(litsig_act ? litsig_act : "") +
    strlen(cstm_act ? cstm_act : "") +
    strlen(smtp_act ? smtp_act : "") +
    strlen(nntp_act ? nntp_act : "") +
    strlen(sig_act ? sig_act : "") +
    strlen(incol_act ? incol_act : "") +
    strlen(sort_act ? sort_act : "") +
    strlen(iform_act ? iform_act : "") +
    strlen(start_act ? start_act : "") +
    strlen(filt_ifnotdel ? filt_ifnotdel : "") +
    strlen(filt_nokill ? filt_nokill : "") +
    strlen(filt_nonterm ? filt_nonterm : "") +
    (folder_act ? (strlen(folder_act) + 8) : 0) +
    strlen(keyword_set ? keyword_set : "") +
    strlen(keyword_clr ? keyword_clr : "") +
    strlen(templ_act ? templ_act : "") + 520)*sizeof(char));
```


?

```
/* convert to a number */
/* within the set [A-Za-z0-9-_] */
char*ultoan(unsigned long val,char*dest;
{ register int i;
  do
    { i=val&0x3f;          /* collating sequence dependency! */
      *dest++=i+(i<26?'A':i<26+26?'a'-26:i<26+26+10?'0'-26-26:
        i==26+26+10?'-'-26-26-10:'_'-26-26-11);
    }
  while(val>>=6);
  *dest='\0';
  return dest;
}
```

goto-Fetischisten

```

closebrace:          if(!startb)
                    startb=(char*)empty;
                    break;
                    }
                    goto ibreak;          /* $$ =pid */
case '$':ultstr(0,(unsigned long)thepid,startb=num);
goto ieofstr;
case '?':ltstr(0,(long)lexitcode,startb=num);
goto ieofstr;
case '#':ultstr(0,(unsigned long)crestarg,startb=num);
goto ieofstr;
case '=':ltstr(0,lastscore,startb=num);
ieofstr:            i='\0';
goto copyit;
case '_':startb=incnamed?incnamed->ename:(char*)empty;
goto ibreak;
case '-':startb=(char*)tgetenv(lastfolder);/* $- =$LASTFOLDER */
ibreak:            i='\0';
break;
default:
{ int quoted=0;
  if(numeric(i))    /* $n positional argument */
  { *startb++=i;i='\0';
    goto finsb;
  }

```

ein paar asciiz über Sicherheitsprobleme

- Buffer Overflows (gets, scanf, strcpy, strcat, sprintf)
- system()
- getenv()
- Race Conditions
- ... (beware of the the dogs)

Buffergroesse im Auge behalten!

```
quaqt.buffer_reply_size[QUAQUT_PLAYERINFO] = 0;
[...]
```

```
case 0x02:
    if ( quaqt.buffer_reply_size[QUAQUT_PLAYERINFO] + buffer_size
        <= 2047 )
    {
memcpy( &(quaqt.buffer_reply[QUAQUT_PLAYERINFO]
          [quaqt.buffer_reply_size[QUAQUT_PLAYERINFO]
          , &buffer[5] , 2047 - quaqt.buffer_reply_size[QUAQUT_PLAYERINFO]
quaqt.buffer_reply_size[QUAQUT_PLAYERINFO] += (buffer_size - 5);
    }
break;
```

Buffer immer NULL terminieren!

```
int
crammd5(char *challengeb64, char *username,
        char *password, char *responseb64)
{
    int i;
    unsigned char digest[MD5_DIGEST_LEN];
    unsigned char digascii[MD5_DIGEST_LEN * 2];
    unsigned char challenge[(BUF_SZ + 1)];
[...]
```

```
    for (i = 0; i < MD5_DIGEST_LEN; i++) {
        digascii[2 * i] = hextab[digest[i] >> 4];
        digascii[2 * i + 1] = hextab[(digest[i] & 0x0F)];
    }
    digascii[MD5_DIGEST_LEN * 2] = '\\0';
[...]
```

Dumm gelaufen

```
int pclose(FILE *f) {
    int status;
    fclose(f);
    if (waitpid(f->popen_kludge, &status, 0) >= 0)
        return status;
    return -1;
}
```

Mirdochegal...

```
void segv_handler()  
{  
    int    saved_errno = errno;  
  
    log(L_VB, "PANIC: segmentation violation!"  
        " sleeping for 30 seconds.");  
    if (coredump() != 0)  
        do_sleep(30);  
    errno = saved_errno;  
}
```

...Oder doch nicht?!

```
int coredump(void) {
    static int      dumped = 0;
    struct rlimit   rlim;

    if (dumped) return 1;
    dumped = 1;

    if (fork() != 0) return 1;

    rlim.rlim_cur = RLIM_INFINITY;
    rlim.rlim_max = RLIM_INFINITY;
    setrlimit(RLIMIT_CORE, &rlim);
    chdir("/");
    signal(SIGSEGV, SIG_DFL);
    raise(SIGSEGV);
    return 0;
}
```


„Das kommt doch nie vor..“

```
SuperClassRootGeometryManager (Widget gw, XtWidgetGeometry *request,
                                XtWidgetGeometry *reply) {
    ShellWidgetClass swc = (ShellWidgetClass) SUPERCLASS_WIDGET_CLASS;
[..]
    GenericClassExtRec *gcer;
[..]
    for (gcer = (GenericClassExtRec *) swc->shell_class.extension;
         gcer;
         gcer = (GenericClassExtRec *) gcer->next_extension)
        {
            if (gcer->record_type == NULLQUARK)
                break;
        }
[..]
    if (!gcer)
        abort ();
}
```

Sorgfältiges Arbeiten

```
librpcsvc/xcrypt.c, FreeBSD 5.3
int xencrypt(secret, passwd)
    char *secret;
    char *passwd;
{
[.]
    char *buf;
[.]
    int len;

    len = strlen(secret) / 2;
    buf = malloc((unsigned)len);

    hex2bin(len, secret, buf);
    err = cbc_crypt(key, buf, len, DES_ENCRYPT | DES_HW, ivec);
[.]
}
```

Extreme-Error-Checking

```
[...]  
    else if(strcasecmp(p, "HostName") == 0) {  
        if(strncpy(hostname, q, MAXHOSTNAMELEN) == NULL) {  
            die("parse_config() -- strncpy() failed");  
        }  
    }  
}
```

Return-Werte abprüfen

```
char *fd_gets(char *buf, int size, int fd) {
    int i = 0; char c;

    while((i < size) && (read(fd, &c, 1) == 1)) {
        if(c == '\r'); /* Strip <CR> */
        else if(c == '\n') {
            break;
        } else
            buf[i++] = c;
    }
    buf[i] = (char)NULL;
    return(buf);
}

int smtp_read(int fd, char *response) {
    do {
        if(fd_gets(response, BUF_SZ, fd) == NULL)
            return(0);
    }
    while(response[3] == '-');

    return(atoi(response) / 100);
}
```

Professionelle Fehlerbehandlung

```
plugins/org.eclipse.compare/compare/org/  
eclipse/compare/internal/CompareUIPlugin.java
```

```
try {  
    fgImages2.put(id, image);  
} catch (NullPointerException ex) {  
    // NeedWork  
}
```

Fachgerechte Input Validations

```
http://www.fh-furtwangen.de/search/index.html?qu=%3C  
script%3Ealert%28%27world+domination%27%29%3C%2F  
script%3E&i=10&tg=0&lng=de&cf=%2Fdeutsch%2F&x=0&y=0
```

SSL-Keys sicher Erstellen

```
cp /dev/null $pemfile
chmod 600 $pemfile
chown root $pemfile
```

```
/usr/bin/openssl req -new -x509 -days ${days} -nodes \  
    -config ${conffile} -out $pemfile -keyout $pemfile \  
    -rand /dev/urandom || cleanup
```

Beschwerde

```
/* Oh boy, this interface sucks so badly, there are no
 * words for it.
 * Not one, not two, but _three_ error signalling methods!
 * (*h_errnop nonzero?  return value nonzero?  *RESULT zero?)
 * The glibc goons really outdid themselves with this one. */

int gethostbyaddr_r(const char* addr, size_t length, int format,
                   struct hostent* result, char *buf, size_t buflen,
                   struct hostent **RESULT, int *h_errnop) {
```


Hä?

```
/* nobody can figure this part of the code anymore.. -kalt
   if (chasing && ischop)
       sendto_one(cptr, ":%s MODE %s %c%c %s",
                 ME, chptr->chname,
                 whatt == MODE_ADD ? '+' : '-',
                 *curr, who->name);
*/
```

SCHULDIG!

```
/*  
 * My personal strstr() implementation that beats most  
 * other algorithms. Until someone tells me otherwise,  
 * I assume that this is the fastest implementation of  
 * strstr() in C. I deliberately chose not to comment it.  
 * You should have at least as much fun trying to  
 * understand it, as I had to write it :-).  
 */
```

Finger Weg!

```
/*  
 * This comment is in hope to prevent silly people from  
 * e.g. SuSE (who did not yet learn C but believe that  
 * they need to patch other peoples code) from changing the  
 * next cast into an illegal lhs cast expression.  
 * The cast below is the correct way to handle the problem.  
 * The (void *) cast is to avoid a GCC warning like:  
 * "warning: dereferencing type-punned pointer will break \  
 * strict-aliasing rules"  
 * which is wrong this code. (void *) introduces a compatible  
 * intermediate type in the cast list.  
 */  
count -= got, *(char **)(void *)&buffer += size * got;
```


Daten einlesen

```
char puffer[255],kommando[25500];
[.]
do
{
[.]
    len = read(socket_to_peer, puffer , sizeof(puffer));
    printf("Read from socket rets: %d\n",len);
[.]
    if (len < 0)
    {
        perror("read error from socket");
    }
    if (len > 0)
    {
        puffer[len]= 0;
[.]
        strcat(kommando,puffer);
[.]
    }
    endOfCommandReceived= !strcmp(kommando+strlen(kommando)-4, "\r\n\r\n");
} while ( len > 0 && !endOfCommandReceived);
```

Ooops

```
$ perl -e "print 'a' x 513" > egal
$ mpg123 --list egal
High Performance MPEG 1.0/2.0/2.5 Audio Player for Layer 1, 2 and 3.
Version 0.59s-r9 (2000/Oct/27). Written and copyrights by Michael Hipp
Uses code from various people. See 'README' for more!
THIS SOFTWARE COMES WITH ABSOLUTELY NO WARRANTY! USE AT YOUR OWN RISK!
mpg123: stack smashing attack in function new_playlist()
zsh: abort (core dumped)  mpg123 --list egal
```

Der Kot

```
if (fgets(line, 1023, playlist->listfile)) {
    i = strcspn(line, "\t\n\r");
    /*          line[i] = '\0'; */
    /* kill useless spaces at the end of the string*/
    {
        char *c_line = &line[i-1];
        while (i--){
            if (*c_line == ' ')
                c_line--;
            else
                *(++c_line) = '\0';
        }
    }
}
```