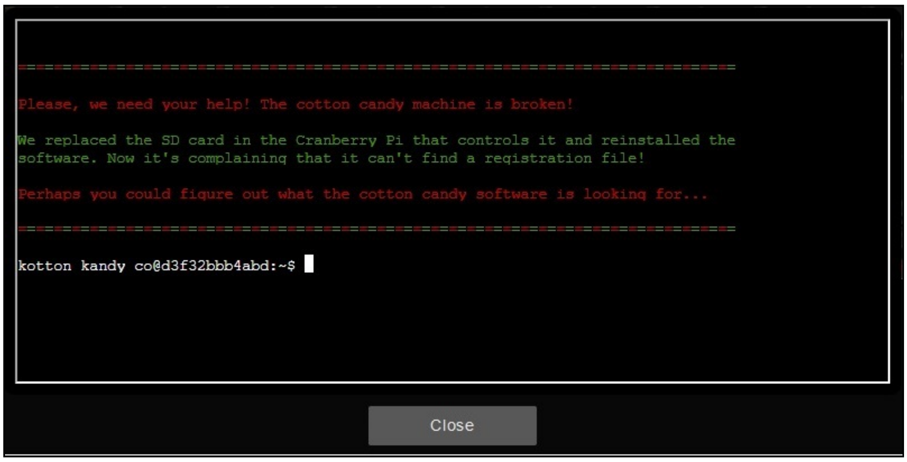
KringleCon 4: Calling Birds!

Kitchen

1. Click to talk to `Tinsel Upatree`

```
Hiya hiya, I'm Tinsel Upatree!

Say, do you know what's going on next door?

I'm a bit worried about the whole FrostFest event.

It feels a bit... ill-conceived, somehow. Nasty even.

Well, regardless – and more to the point, what do you know about tracing processes in Linux?

We rebuilt this here Cranberry Pi that runs the cotton candy machine, but we seem to be missing a file.

Do you think you can use strace or ltrace to help us rebuild the missing config?

We'd like to help some of our favorite children enjoy the sweet spun goodness again!

And, if you help me with this, I’ll give you some hints about using Wireshark filters to look for unusual options that might help you achieve Objectives here at the North Pole.
```

2. Click `Strace Ltrace Retrace` Cranberry Pi terminal



```
================================================================================

Please, we need your help! The cotton candy machine is broken!

We replaced the SD card in the Cranberry Pi that controls it and reinstalled the
software. Now it's complaining that it can't find a registration file!

Perhaps you could figure out what the cotton candy software is looking for...

================================================================================

kotton kandy co@d3f32bbb4abd:~$
```

3. Type `ls`

```
make_the_candy*
```

4. Type `ltrace ./make_the_candy`

```
fopen("registration.json", "r")                           = 0
puts("Unable to open configuration fil"...Unable to open configuration file.
)               = 35
+++ exited (status 1) +++
```

5. Type `echo "{}" > registration.json`

6. Type `ltrace ./make_the_candy`

```
fopen("registration.json", "r")                           = 0x55af40de6260
getline(0x7ffe2a5708a0, 0x7ffe2a5708a8, 0x55af40de6260, 0x7ffe2a5708a8) = 3
strstr("{}\n", "Registration")                            = nil
getline(0x7ffe2a5708a0, 0x7ffe2a5708a8, 0x55af40de6260, 0x7ffe2a5708a8) = -1
puts("Unregistered - Exiting."Unregistered - Exiting.
)                           = 24
+++ exited (status 1) +++
```

7. Type `echo "{ Registration }" > registration.json`

8. Type `ltrace ./make_the_candy`

```
fopen("registration.json", "r")                           = 0x55d160a41260
getline(0x7ffda9f6eec0, 0x7ffda9f6eec8, 0x55d160a41260, 0x7ffda9f6eec8) = 17
strstr("{ Registration }\n", "Registration")              = "Registration }\n"
strchr("Registration }\n", ':')                           = nil
getline(0x7ffda9f6eec0, 0x7ffda9f6eec8, 0x55d160a41260, 0x7ffda9f6eec8) = -1
puts("Unregistered - Exiting."Unregistered - Exiting.
)                           = 24
+++ exited (status 1) +++
```

9. Type `echo "{ Registration : }" > registration.json`

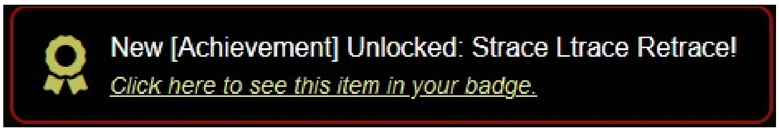
10. Type `ltrace ./make_the_candy`

```
fopen("registration.json", "r")                           = 0x55d4c6440260
getline(0x7ffd2c653670, 0x7ffd2c653678, 0x55d4c6440260, 0x7ffd2c653678) = 20
strstr("{ Registration :  }\n", "Registration")           = "Registration :  }\n"
strchr("Registration :  }\n", ':')                        = ":  }\n"
strstr(":  }\n", "True")                                  = nil
getline(0x7ffd2c653670, 0x7ffd2c653678, 0x55d4c6440260, 0x7ffd2c653678) = -1
puts("Unregistered - Exiting."Unregistered - Exiting.
)                           = 24
+++ exited (status 1) +++
```

11. Type `echo "{ Registration : True }" > registration.json`

12. Type `./make_the_candy`

```
Launching...

     *                              *
.
.
.
          *            *
           *          *
            *        *
             *      *
              *    *
               *  *
                **
     Candy making in progress
```

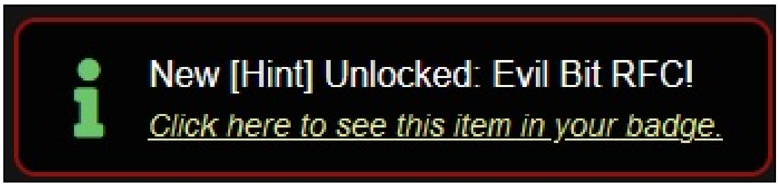


New [Achievement] Unlocked: Strace Ltrace Retrace!
Click here to see this item in your badge.

13. Click the `Close` button

14. Click to talk to `Tinsel Upatree`

```
Great! Thanks so much for your help!

I'm sure I can put those skills I just learned from you to good use.

Are you familiar with RFC3514?
```

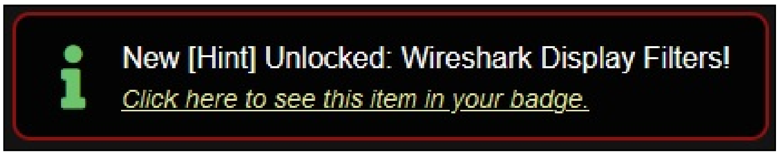


New [Hint] Unlocked: Evil Bit RFC!

Click here to see this item in your badge.

15. Click to talk to `Tinsel Upatree`

```
Wireshark uses a different name for the Evil Bit: ip.flags.rb.

HTTP responses are often gzip compressed. Fortunately, Wireshark decompresses them for us
automatically.

You can search for strings in Wireshark fields using display filters with the contains keyword.
```

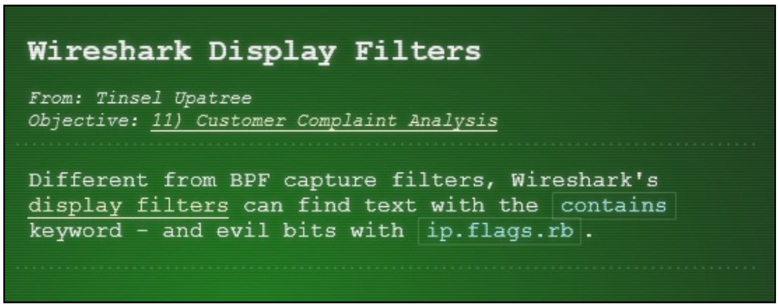


New [Hint] Unlocked: Wireshark Display Filters!

Click here to see this item in your badge.

16. Click the `i` (Hints) icon

17. Click `Wireshark Display Filters`

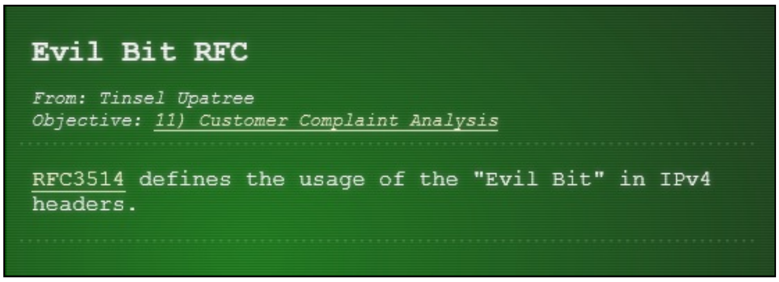


Wireshark Display Filters

From: Tinsel Upatree
Objective: 11) Customer Complaint Analysis

Different from BPF capture filters, Wireshark's display filters can find text with the `contains` keyword - and evil bits with `ip.flags.rb`.

18. Click `Evil Bit RFC`



Evil Bit RFC

From: Tinsel Upatree
Objective: 11) Customer Complaint Analysis

RFC3514 defines the usage of the "Evil Bit" in IPv4 headers.

19. Click `[Exit]`

20. Move `Right` to enter the Great Room

21. Move `Down` and then `Left` to enter Entry

22. Click `Destinations` (map icon) and then click `Talks Lobby`