

1. In this challenge we are given ppt file.

```
none@alpha:~/hck$ ls
chall.pptx
none@alpha:~/hck$
```

2. After opening it we will see an qr code, if we scan and we will be rick rolled, so don't do it.

3. Every PPT is a type of zip file, single ppt contains many other type of files inside it. Let's check if we find something interesting.

```
none@alpha:~/hck$ unzip chall.pptx
Archive:  chall.pptx
  extracting: [Content_Types].xml
   inflating: _rels/.rels
   inflating: docProps/app.xml
   inflating: docProps/core.xml
   inflating: ppt/_rels/presentation.xml.rels
   inflating: ppt/media/image1.png
   inflating: ppt/presProps.xml
   inflating: ppt/presentation.xml
   inflating: ppt/slideLayouts/_rels/slideLayout1.xml.rels
   inflating: ppt/slideLayouts/_rels/slideLayout10.xml.rels
   inflating: ppt/slideLayouts/_rels/slideLayout11.xml.rels
   inflating: ppt/slideLayouts/_rels/slideLayout2.xml.rels
   inflating: ppt/slideLayouts/_rels/slideLayout3.xml.rels
   inflating: ppt/slideLayouts/_rels/slideLayout4.xml.rels
   inflating: ppt/slideLayouts/_rels/slideLayout5.xml.rels
   inflating: ppt/slideLayouts/_rels/slideLayout6.xml.rels
   inflating: ppt/slideLayouts/_rels/slideLayout7.xml.rels
   inflating: ppt/slideLayouts/_rels/slideLayout8.xml.rels
   inflating: ppt/slideLayouts/_rels/slideLayout9.xml.rels
   inflating: ppt/slideLayouts/slideLayout1.xml
   inflating: ppt/slideLayouts/slideLayout10.xml
   inflating: ppt/slideLayouts/slideLayout11.xml
   inflating: ppt/slideLayouts/slideLayout2.xml
   inflating: ppt/slideLayouts/slideLayout3.xml
   inflating: ppt/slideLayouts/slideLayout4.xml
```

After unzip we got this

```
none@alpha:~/hck$ ls
chall.pptx  '[Content_Types].xml'  docProps  ppt  _rels
none@alpha:~/hck$
```

4. Now there are too many folders and file, going inside of every folder and then searching will take some time.
5. To make the process a little bit easier, we will mix-match a Linux command.

ls -R | less

```
.:
chall.pptx
[Content_Types].xml
docProps
ppt
_rels

./docProps:
app.xml
core.xml

./ppt:
media
presentation.xml
presProps.xml
_rels
slideLayouts
slideMasters
slides
tableStyles.xml
theme
viewProps.xml

./ppt/media:
image1.png

./ppt/_rels:
presentation.xml.rels
:█
```

6. Here we found something

```
none@alpha:~/hck$ cat ppt/theme/_rels/something
RmxhR2hvc3R7MTRiX05ldmVyX2cwMDBOTk4zX2dpdmVfWTAxX3VwXzEzeX0=

none@alpha:~/hck$
```

7. This looks like base64

8. After decoding it, we will get our flag

Decode from Base64 format


Simply enter your data then push the decode button.

RmxhR2hvc3R7MTRiX05ldmVyX2cwMDBOTk4zX2dpdmVfWTAxX3VwXzEzeX0=

 For encoded binaries (like images, documents, etc.) use the file upload form a little further down on this page.

UTF-8  Source character set.

☐ Decode each line separately (useful for when you have multiple entries).

 Live mode OFF Decodes in real-time as you type or paste (supports only the UTF-8 character set).

 **DECODE**  Decodes your data into the area below.

FlaGhost{14b_Never_g000NNN3_give_Y01_up_13y}