

NexTube Setup Guide [Linux]

Step 1 - Installing Dependencies

This guide is assuming you're using either Ubuntu or Fedora 40. I tried providing Ubuntu commands so they hopefully should work, but I used Fedora 40.

Commands (Don't include the - Distro)

Git

```
sudo dnf install git - Fedora
```

```
sudo apt-get install git - Ubuntu
```

Node.Js

<https://linuxconfig.org/how-to-install-node-js-on-ubuntu-24-04>

```
sudo dnf install nodejs - Fedora
```

FFMPEG

<https://www.liberiangeek.net/2024/04/install-ffmpeg-ubuntu-24-04/>

<https://computingforgeeks.com/how-to-install-ffmpeg-on-fedora/>

LIBVPX

`sudo dnf install libvpx - Fedora`

`sudo apt-get install libvpx-dev`

JAVA

<https://docs.fedoraproject.org/en-US/quick-docs/installing-java/>
(Java 8 is what I used iirc)

<https://docs.datastax.com/en/jdk-install/doc/jdk-install/installOpenJdkDeb.html> (Java 8 is what I used iirc)

Wine

<https://phoenixnap.com/kb/how-to-install-wine-on-ubuntu>

<https://wine.htmlvalidator.com/install-wine-on-fedora-40.html>

Wine Tricks

`sudo apt install winetricks - Ubuntu`

`sudo dnf install winetricks - Fedora`

ImageMagick

<https://itsfoss.com/install-imagemagick-ubuntu/>
<https://linuxcapable.com/how-to-install-imagemagick-on-fedora-linux/>

Step 2 - Gathering Files

Grabbing Programs

JPEXS

I use 15.1.1 since I had issues before with newer versions on YouTube stuff, you can try a newer version, but it may or may not work. Also, you may need to put paths in advance settings, but it should tell you what to do and the files it needs in it.

https://github.com/jindrapetrik/jpexs-decompiler/releases/download/version15.1.1/ffdec_15.1.1.zip

NexTube

git clone <https://github.com/erievs/yt2009-wii.git>

Step 3 - Setting Up YT2009 (NexTube)

Commands

Open the folder yt2009-wii into your terminal (most file explorers let you, at worst you can use the cd command to get into it).

This should be the same process as setting up any YT2009 instance,
As NexTube is a simple fork of YT2009.

We first run (this will install dependencies needed by NexTube)

```
npm install
```

Next we will run (we need our local network ip address, you can share it as it is standard across most networks and such, this not a public ip and won't give info on where you live)

```
hostname -I | awk '{print $1}'
```

Mine is 192.168.1.150 (REMEMBER THIS!!!)

After that, we will go through setup.

Run

```
node yt2009setup.js
```

```
MATE Terminal
File Edit View Search Terminal Help

sprintf-js.
npm WARN deprecated minimatch@0.3.0: Please update to minimatch 3.0.2 or higher
to avoid a RegExp DoS issue
npm WARN deprecated glob@3.2.11: Glob versions prior to v9 are no longer support
ed

added 131 packages, and audited 132 packages in 6s

20 packages are looking for funding
  run `npm fund` for details

6 vulnerabilities (1 low, 5 high)

To address all issues (including breaking changes), run:
  npm audit fix --force

Run `npm audit` for details.
ksportalcraft@fedora ~/t/yt2009-wii (rewrite-august-2024)> node yt2009setup.js

[yt2009] yt2009 setup

what port should yt2009 run on? 
```

Use Port 80 (DX, the creator of YT2009 suggests it
iirc)

```
what environment should be used? (dev/prod)
dev:
- logs usage in terminal
- doesn't require access tokens
- no possibility of creating an ssl version
prod:
- doesn't log normal activity
- if enabled, will create a bunch of random tokens users will have to enter to use
- will be possible to create an ssl version on a separate port
environment (dev/prod):
```

Type in dev

```
what environment should be used? (dev/prod)
dev:
- logs usage in terminal
- doesn't require access tokens
- no possibility of creating an ssl version
prod:
- doesn't log normal activity
- if enabled, will create a bunch of random tokens users will have to enter to use
- will be possible to create an ssl version on a separate port
environment (dev/prod): dev

what IP should be used where it needs to be coded in?
(eg. the flash player related)
preferably use your IP you will use to connect to yt2009. localhost/127.0.0.1
isn't recommended because it may break as soon as you use a different pc.
IP address:
```

Use the ip address we got earlier (remember
hostname -I | awk '{print \$1}')

Now you can just click enter for the rest of the fields and use the default values.

Now run the `post_config_setup.js`

Step 4 - Patching The Files

Process

Now this is where the fun begins, this is where it gets pretty hard

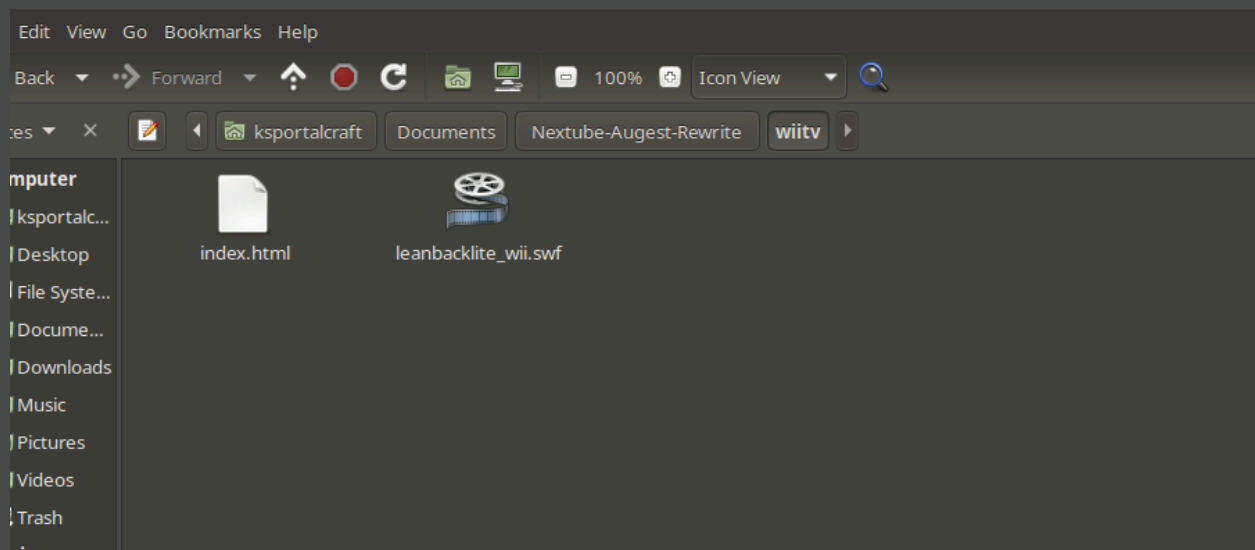
If your local ip address is 192.168.1.150, you can skip!

Okay, so let's first go to the assets folder and go into site-assets and open `leanback_ajax.json` in a text editor or something. And change every instance of 192.168.1.150 to your ip address (you can find and replace). This file is what shows up on the homepage of the channel.

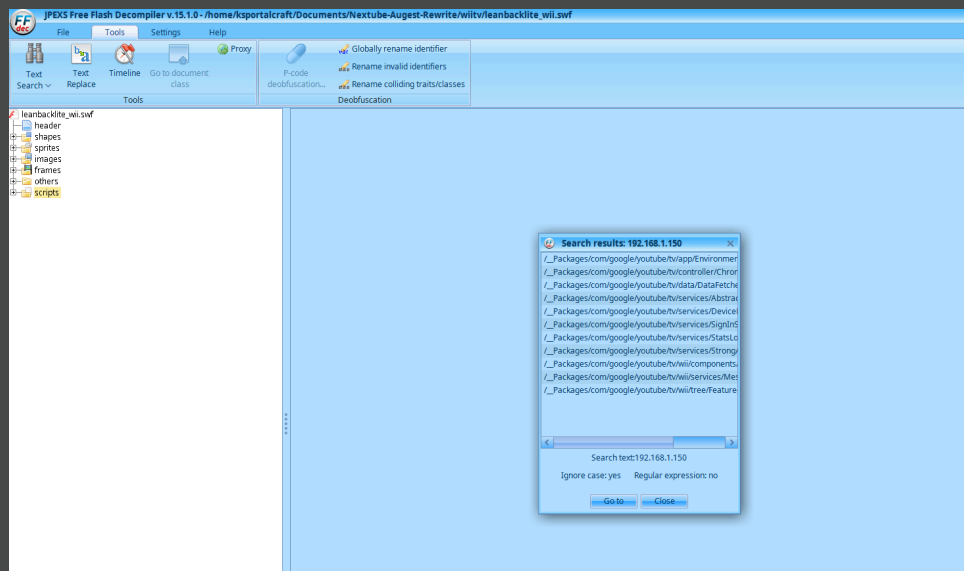
```
1  id
2  "sets": [
3    {
4      "title": "Most Popular",
5      "video_count": 50,
6      "gdata_list_id": "5tmost_popular",
7      "icon": "popular",
8      "gdata_url": "http://192.168.1.150/feeds/api/playlists/PLEnWor732-BHxpp_PmS_VieP68F9s14-",
9      "list_id": "lbpopp.5t.en_US",
10     "tab": "featured",
11     "thumbnails": "http://192.168.1.150/feeds/api/thumbnails/PLEnWor732-BHxpp_PmS_VieP68F9s14-/playlists"
12   },
13   {
14     "title": "Music",
15     "video_count": 30,
16     "gdata_list_id": "5tmost_popular_Music",
17     "icon": "music",
18     "gdata_url": "http://192.168.1.150/feeds/api/standardfeeds/US/most_popular_Music?time=today",
19     "list_id": "lbpopp.10.t.en_US",
20     "tab": "featured",
21     "thumbnails": "http://192.168.1.150/feeds/api/thumbnails/most_popular_Music?time=today"
22   },
23   {
24     "title": "Gaming",
25     "video_count": 30,
26     "gdata_list_id": "5tmost_popular_Games",
27     "icon": "gaming",
28     "gdata_url": "http://192.168.1.150/feeds/api/videos?q=gaming",
29     "list_id": "lbpopp.20.t.en_US",
30     "tab": "featured",
31     "thumbnails": "http://192.168.1.150/feeds/api/thumbnails/gaming/results"
32   },
33   {
34     "title": "Sports",
35     "video_count": 30,
36     "gdata_list_id": "5tmost_popular_Sports",
37     "icon": "sports",
38     "gdata_url": "http://192.168.1.150/feeds/api/videos?q=sports",
39     "list_id": "lbpopp.17.t.en_US",
40     "tab": "featured",
41     "thumbnails": "http://192.168.1.150/feeds/api/thumbnails/sports/results"
42   },
43   {
44     "title": "Film & Animation",
45     "video_count": 30,
46     "gdata_list_id": "5tmost_popular_Film",
47     "icon": "film",
48     "gdata_url": "http://192.168.1.150/feeds/api/videos?q=film&animation",
49     "list_id": "lbpopp.1.t.en_US",
50     "tab": "featured",
51     "thumbnails": "http://192.168.1.150/feeds/api/thumbnails/film&animation/results"
52   }
53 ]
```

We Need To Launch JEPXS

Open `leanback_lite.swf` from the `wiitv` folder in the root.



Go to tools and use text search (you may need to do path stuff in advance settings, but it'll tell you what to do and such)



Click on one of the search results and click the edit button below You will have to change every single 192.168.1.150 to your IP address. I am not going to show a screenshot of all of em since that would make

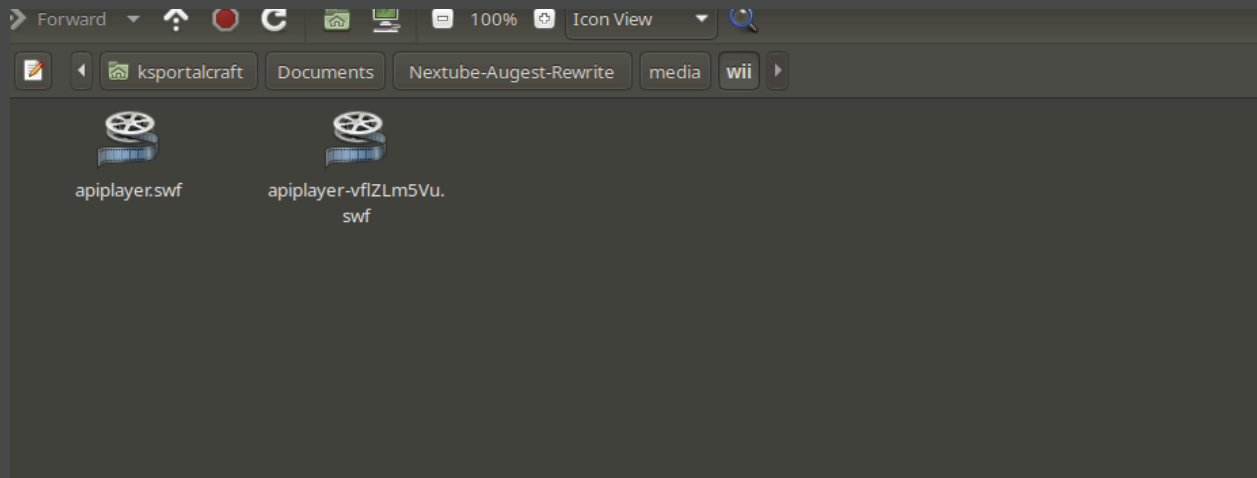
this guide so long. And click save. When you're done with every page under file, click save or save as to make double sure.

```

1 class com.google.youtube.tv.app.Environment
2 {
3     static var ABR_PARAM = "abr";
4     static var LAUNCH_PARAM = "launch";
5     static var PLAYLIST_PARAM = "p";
6     static var QUERY_PARAM = "q";
7     static var VIDEO_PARAM = "v";
8     static var VIDEO_PLUS_RELATED_PARAM = "vv";
9     static var SECRET_PARAM = "ea";
10    static var TEST_ENVIRONMENT = "testenv";
11    static var LOADER_PARAM = "loader";
12    static var PACKAGE_PARAM = "package";
13    static var TEST_ENVIRONMENT_REGEX = new com.as2regexp.RegExp("^[-a-z]+$");
14    static var OVERRIDEN_FLASHVARS_URL = "http://192.168.1.150:80/leanbacklite?action_get_flashvars&" + "country=${country}&vondor=${vondor}";
15    static var TEST_HOSTS = ["web-release-qa.youtube.com", "web-release-olddtfe-qa.youtube.com", "web-trunk-qa.youtube.com", "web-trunk-olddtfe-qa.youtube.com"];
16    static var OVERRIDEN_FLASHVARS = {};
17    static var loading = true;
18    function Environment()
19    {
20    }
21    static function init(swfUrl, opt_secure)
22    {
23        var _loc3_ = new com.google.utils.Url(swfUrl);
24        var _loc4_ = _loc3_.getHost();
25        var _loc5_ = "http://";
26        var _loc6_ = com.google.youtube.tv.app.Environment.getTestHost();
27        if(_loc6_)
28        {
29            com.google.youtube.tv.app.Environment.SERVLET_HOST = _loc6_;
30        }
31        else if(com.google.youtube.tv.app.Environment.isTestHost(_loc4_))
32        {
33            com.google.youtube.tv.app.Environment.SERVLET_HOST = _loc4_;
34        }
35        else
36        {
37            com.google.youtube.tv.app.Environment.SERVLET_HOST = "www.youtube.com";
38        }
39        com.google.youtube.tv.app.Environment.BASE_URL = _loc5_ + com.google.youtube.tv.app.Environment.SERVLET_HOST;
40        com.google.youtube.tv.app.Environment.GDATA_URL = "http://192.168.1.150:80";
41        com.google.youtube.tv.app.Environment.LOCAL_HOST = "http://192.168.1.150";
42        com.google.youtube.tv.util.StringUtil.addCommonPlaceholder("baseUrl", com.google.youtube.tv.app.Environment.BASE_URL);
43        com.google.youtube.tv.util.StringUtil.addCommonPlaceholder("localhost", com.google.youtube.tv.app.Environment.LOCAL_HOST);
44        com.google.youtube.tv.util.StringUtil.addCommonPlaceholder("gdataUrl", com.google.youtube.tv.app.Environment.GDATA_URL);
45        com.google.youtube.tv.util.StringUtil.addCommonPlaceholder("servletHost", com.google.youtube.tv.app.Environment.SERVLET_HOST);
46        com.google.youtube.tv.util.StringUtil.addCommonPlaceholder("country", com.google.youtube.tv.platform.Box.getCountry());
47        com.google.youtube.tv.util.StringUtil.addCommonPlaceholder("lang", com.google.youtube.tv.platform.Box.getLang());
48        com.google.youtube.tv.util.StringUtil.addCommonPlaceholder("model", com.google.youtube.tv.util.StringUtil.encodeUriComponent(com.google.youtube.tv.app.Environment.model));
49        com.google.youtube.tv.util.StringUtil.addCommonPlaceholder("vondor", com.google.youtube.tv.util.StringUtil.encodeUriComponent(com.google.youtube.tv.app.Environment.vondor));
50    }
51    static function loadOverridenFlashVars()
52    {
53    }
54 }

```

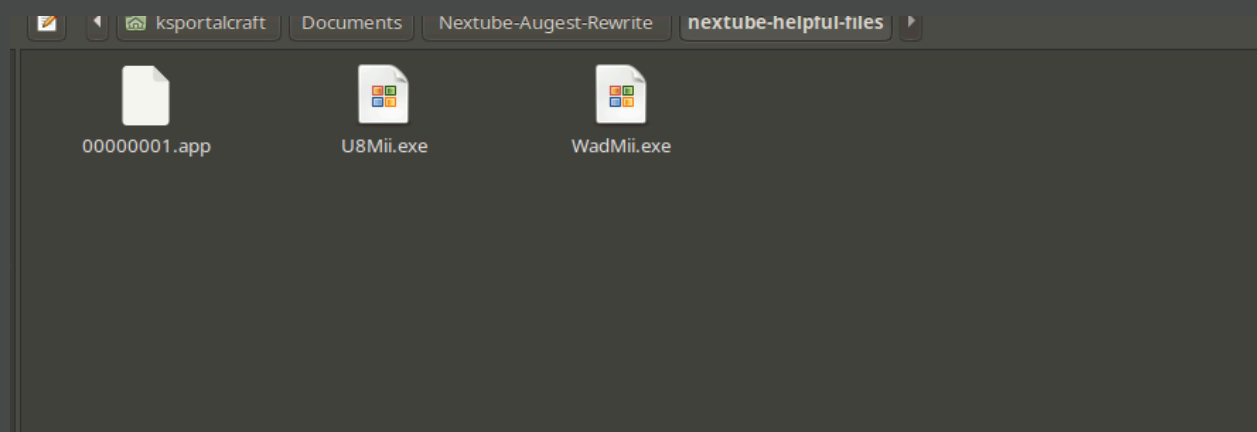
Do the same for apiplayer.swf and apiplayer-vf1ZLm5Vu.swf (it'll be the same process)



Step 5 - Patching The Wad

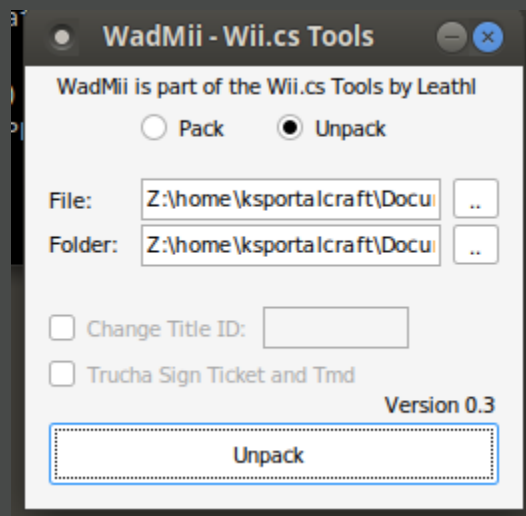
Process (big credit to Wii.cs Tools)

Okay there should be a folder called nextube-helpful-files with some files U8Mii.exe and 00000001.app and WADMii. This is what we will use. You will need a WAD of the YouTube channel (I'd suggest finding the last US version).



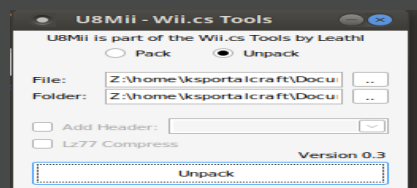
Try winetricks dotnet20 , if that fails try installing <https://www.microsoft.com/en-us/download/details.aspx?id=6041> to wine, so run wine in your terminal and drag and drop the installer.

Run WADMii by opening a terminal and running wine WADMii.exe. And unpack your wad. You may need to go into My Computer , Z: then home and then your username and where ever the wad is.



Copy 00000001.app and go into the newly extracted folder and paste and replace. This is to bypass wii wfc by using wiimmfi, as the patcher has issues on Linux with wine.

Now run U8Mii.exe and unpack 00000002.app



In that new folder, go into config and open common.pcf with a text editor. And replace <https://web-trunk-qa.youtube.com/wiitv> with your ip address. Like this <https://192.168.1.150/wiitv> (this may not be needed, but I do it just in case) and save

```
##### MediaStream #####

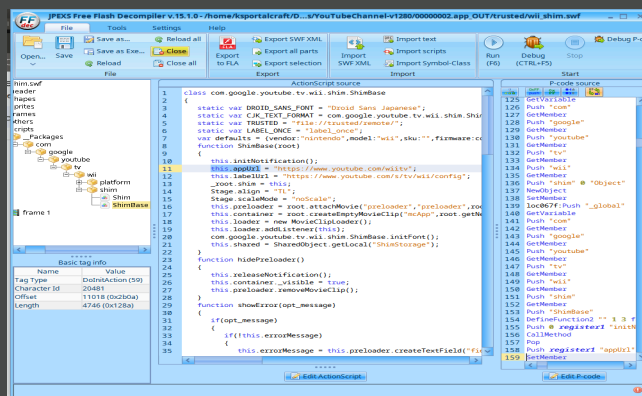
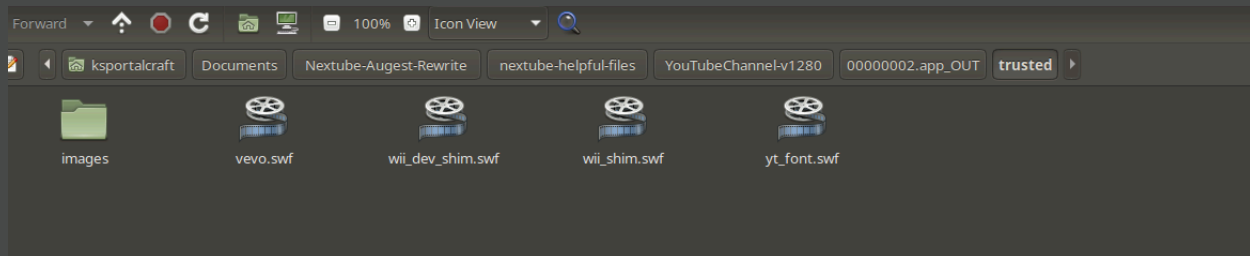
content_domain    file:///trusted/           #Local Data
|
debug_content_url  file:///trusted/wii_dev_shim.swf

# Debug settings
# load from web-trunk-qa:
debug_flash_vars  dev=1&app=file:///trusted/remote/http://192.168.1/wiitv/leanbacklite_wii.swf
# load from web-release-qa:
debug_flash_vars  dev=1&app=file:///trusted/remote/https://web-release-qa.youtube.com/wiitv
# load from horcrux (no-auto-build):
debug_flash_vars  dev=1&relax=1&app=file:///trusted/remote/http://horcrux.sbo.corp.google.com/wii/leanbacklite_wii.swf&urlmap=s.yting.com/yts/swfbin/apiplayer30http://horcrux.sbo.corp.google.com/wii/vast_ads_module.swf
# load from prod:
debug_flash_vars  dev=1
# load from prod/charles-ready:
debug_flash_vars  dev=1&relax=6

final_content_url  file:///trusted/wii_shim.swf

# Final settings (can't be empty)
# load from prod
```

Now go in trusted folder and open JPEXs and patch wii_shim and wii-dev_shim like you did in Step 4 but instead of looking for 192.168.1.150 look for youtube.com, I think all you need to do is this.appUrl, but you can do all youtube.com to be safe. And save



Okay now what you need to do is open up U8MII and pack 00000002.app_OUT, replace the 00000002.app with the patched one. And then pack the wad (make sure to delete folders) and you should be done. Just look at the images in the earlier steps in Step 5.

Install the wad like any normal wad and enjoy!

Troubleshooting

Make sure EVERY SINGLE URL is using your ip address

Make sure it is running

Make sure your wii is connected to the same network

Make sure you have the dependencies installed

My discord is pnj3.0