How I cross-compiled Kurento Media Server for Windows

Kyrylo Polezhaiev

A lot of modules and dependencies

- kms-core
- kurento-media-server
- kurento-filters
- kurento-elements
- kms-jsonrpc
- jsoncpp

- gstreamer-1.5
- gst-plugins-base-1.5
- gst-plugins-good-1.5
- gst-plugins-bad-1.5
- gst-plugins-ugly-1.5
- openwebrtc-gst-plugins-1.5

- libiconv
- boost
- libffi
- Glib, Glibmm
- gettext
- gcc
- Libxml2
- usrsctp
- libevent

- libnice
- SQLite
- libsigc++
- libopus
- OpenCV
- libsoup
- OpenSSL
- winpthreads
- libvpx

Ideas

- To cross-compile KMS from Fedora using MinGW compiler
- Build prototype using handmade Makefiles then use CMake

Pros. of cross-compiling using Fedora

- We have Linux environment and package manager
- Fedora has a lot of ready-to-use packages prebuilt with MinGW

```
sudo dnf install autoconf mingw32-filesystem cmake mingw32-gcc-c++ maven mingw32-boost gettext-devel bison flex mingw32-glib2 mingw32-orc mingw32-libtheora mingw32-libvorbis mingw32-opus mingw32-libsigc++20 mingw32-glibmm24 yasm mingw32-openssl indent astyle
```

A lot of modules and dependencies

- kms-core
- kurento-media-server
- kurento-filters
- kurento-elements
- kms-jsonrpc
- jsoncpp

- gstreamer-1.5
- gst-plugins-base-1.5
- gst-plugins-good-1.5
- gst-plugins-bad-1.5
- gst-plugins-ugly-1.5
- openwebrtc-gst-plugins-1.5

- libiconv
- boost
- libffi
- Glib, Glibmm
- gettext
- gcc
- Libxml2
- usrsctp
- libevent

- libnice
- SQLite
- libsigc++
- libopus
- OpenCV
- libsoup
- OpenSSL
- winpthreads
- libvpx

A lot of modules and dependencies

- kms-core
- kurento-media-server
- kurento-filters
- kurento-elements
- kms-jsonrpc
- jsoncpp

- gstreamer-1.5
- gst-plugins-base-1.5
- gst-plugins-good-1.5
- gst-plugins-bad-1.5
- gst-plugins-ugly-1.5
- openwebrtc-gst-plugins-1.5

- ~~libiconv~~
- ~~boost~~
- ~~libffi~~
- ~~Glib, Glibmm~~
- ~~gettext~~
- ~~gcc~~
- ~~Libxml2~~
- usrsctp
- libevent

- libnice
- ~~SQLite~~
- ~~libsigc++~~
- ~~libopus~~
- OpenCV
- ~~libsoup~~
- ~~OpenSSL~~
- ~~winpthreads~~
- libvpx

Build using handmade Makefiles

- I wrote huge Makefiles manually
- Used vanilla GStreamer

Tweaks I had to do:

- libuuid (uuid_generate, uuid_unparse) → rpcrt4 (CoCreateGuid, UuidToStringA, RpcStringFree)
- ushort, uint → unsigned short, unsigned or guint etc.
- <sys/socket.h>, <arpa/inet.h> → <ws2tcpip.h>, <winsock2.h>
- Link against -lws2_32 -liphlpapi
- Use –no-undefined flag, don’t use –dl and -fPIC
- %ld, %zd, %hhx → G_GUINT64_FORMAT, G_GSIZE_FORMAT
- Define -DWIN32_LEAN_AND_MEAN=1

Standard CMake + Autotools way

- For Autotool-based cases, simple **autogen.sh + mingw32-configure + make + make** install scenario works
- For libvpx this magic works:

```
eval `rpm --eval %{mingw32_env}`
export AS=yasm
./configure \
--target=x86-win32-gcc \
--prefix=/usr/i686-w64-mingw32/sys-root/mingw/
```

Standard CMake + Autotools way

- For CMake-controlled KMS modules:

```
mingw32-cmake -DCMAKE_BUILD_TYPE=Release \
 -DCMAKE_MODULE_PATH=\
/usr/i686-w64-mingw32/sys-root/mingw/share/cmake-3.5/Modules/ \
 -DCMAKE_INSTALL_PREFIX=\
/usr/i686-w64-mingw32/sys-root/mingw \
 -DKURENTO_MODULES_DIR=\
/usr/i686-w64-mingw32/sys-root/mingw/share/kurento/modules/ \
<path>
```

Tweaks to CMakeFiles

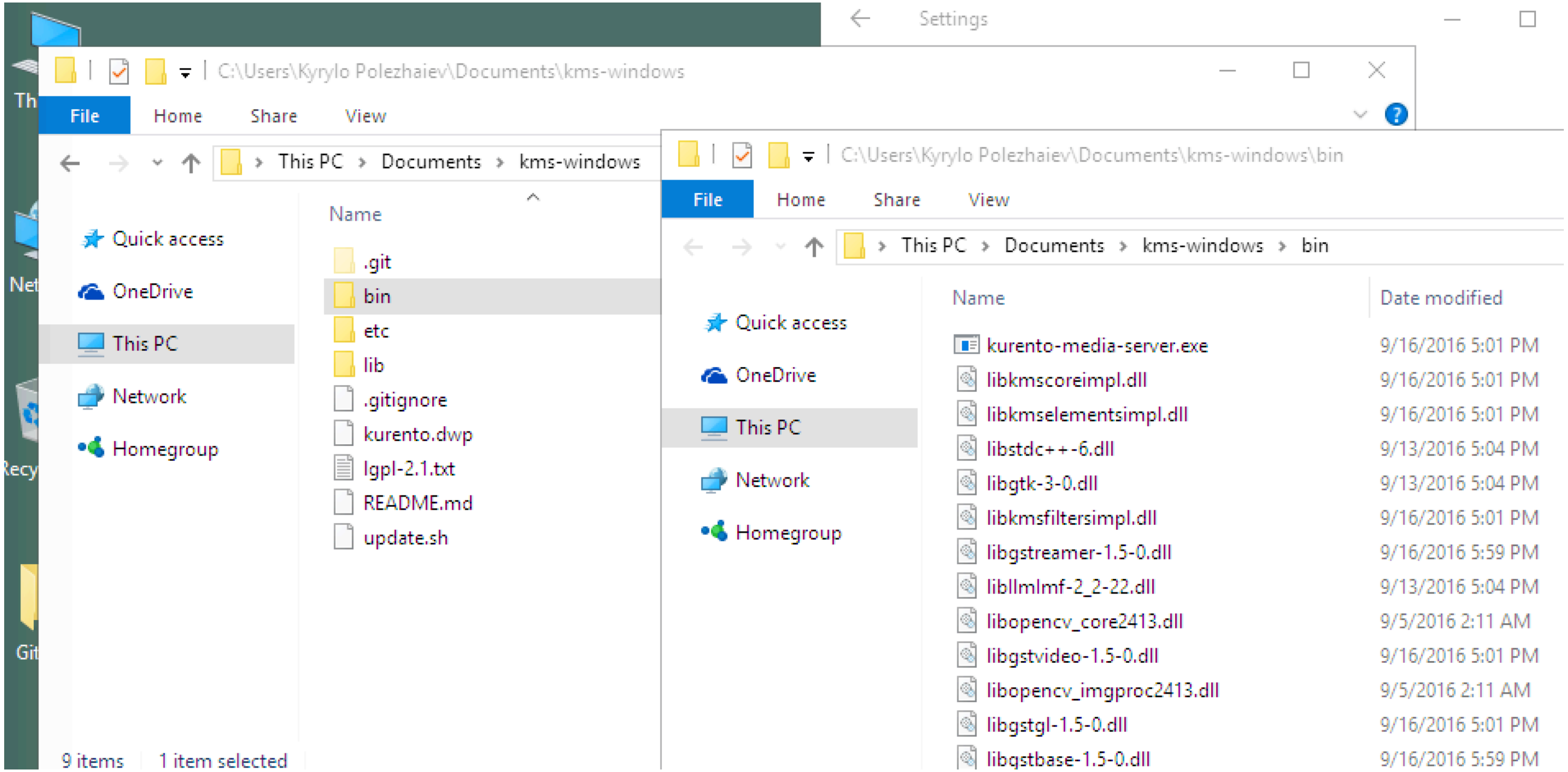
- Explicitly defined some libraries STATIC
- Explicitly link *endpoints with -lkmssdpagent -lgstsdp-1.5.dll
- Replaced linking order for some libs

Prepare Distro

- After everything is build, it’s installed in

```
/usr/i686-w64-mingw32/sys-root/mingw/
```

- Take kurento-media-server.exe and some of .dlls to bin/
- Put needed GStreamer plugins into lib/gstreamer-1.5/
- Put Kurento moduless into lib/kurento/modules/
- Put configuration files into etc/kurento/ and etc/kurento/modules/kurento



C:\Users\Kyrylo Polezhaiev\Documents\kms-windows

Name
.git
bin
etc
lib
.gitignore
kurento.dwp
lgpl-2.1.txt
README.md
update.sh

9 items 1 item selected

C:\Users\Kyrylo Polezhaiev\Documents\kms-windows\bin

Name	Date modified
kurento-media-server.exe	9/16/2016 5:01 PM
libkmscoreimpl.dll	9/16/2016 5:01 PM
libkmselementsimpl.dll	9/16/2016 5:01 PM
libstdc++-6.dll	9/13/2016 5:04 PM
libgtk-3-0.dll	9/13/2016 5:04 PM
libkmsfiltersimpl.dll	9/16/2016 5:01 PM
libgstreamer-1.5-0.dll	9/16/2016 5:59 PM
libIlmImf-2_2-22.dll	9/13/2016 5:04 PM
libopencv_core2413.dll	9/5/2016 2:11 AM
libgstvideo-1.5-0.dll	9/16/2016 5:01 PM
libopencv_imgproc2413.dll	9/5/2016 2:11 AM
libgstgl-1.5-0.dll	9/16/2016 5:01 PM
libgstbase-1.5-0.dll	9/16/2016 5:59 PM

Debug Problems

- Can not test Winsock 2 on Wine
- Difficult to use & navigate sources on Windows

Debug

- Take gdbserver.exe provided by Fedora MinGW package and put into Windows machine near kurento-media-server.exe
- Make sure Windows firewall allows gdbserver and KMS listen ports.
- Run gdbserver.exe like that:

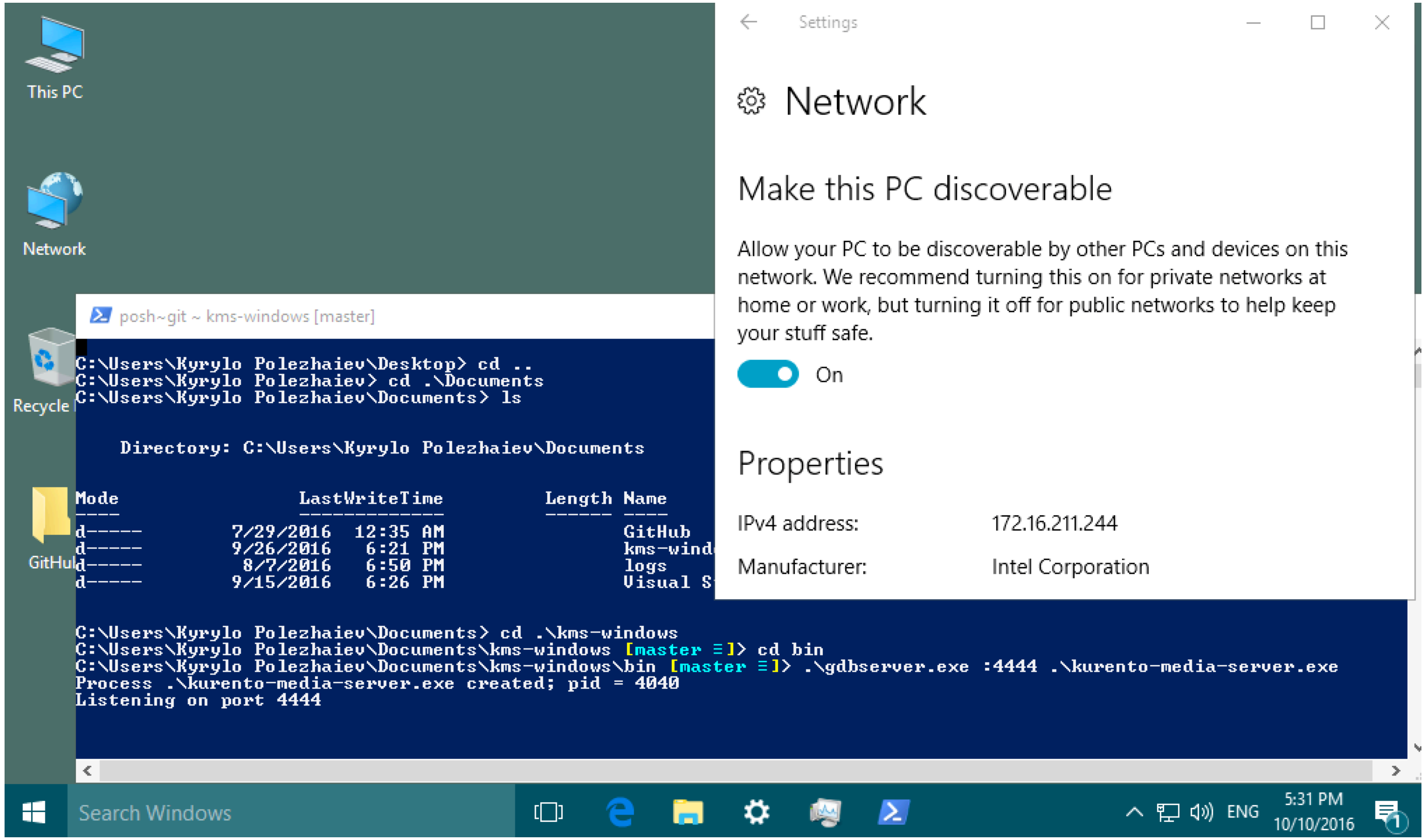
```
gdbserver.exe :4444 kurento-media-server.exe
```

- Connect to it via gdb, specifying something like:

```
 set target-file-system-kind dos-based

target remote 172.16.211.244:4444
```

Don’t select text in Console Window! ☺



posh~git ~ kms-windows [master]

```
C:\Users\Kyrylo Polezhaiev\Desktop> cd ..
C:\Users\Kyrylo Polezhaiev> cd .\Documents
C:\Users\Kyrylo Polezhaiev\Documents> ls


    Directory: C:\Users\Kyrylo Polezhaiev\Documents


Mode                LastWriteTime         Length Name
----                -------------         ------ ----
d-----        7/29/2016  12:35 AM                GitHub
d-----        9/26/2016   6:21 PM                kms-wind
d-----         8/7/2016   6:50 PM                logs
d-----        9/15/2016   6:26 PM                Visual S


C:\Users\Kyrylo Polezhaiev\Documents> cd .\kms-windows
C:\Users\Kyrylo Polezhaiev\Documents\kms-windows [master ≡]> cd bin
C:\Users\Kyrylo Polezhaiev\Documents\kms-windows\bin [master ≡]> .\gdbserver.exe :4444 .\kurento-media-server.exe
Process .\kurento-media-server.exe created; pid = 4040
Listening on port 4444
```

Settings

Network

Make this PC discoverable

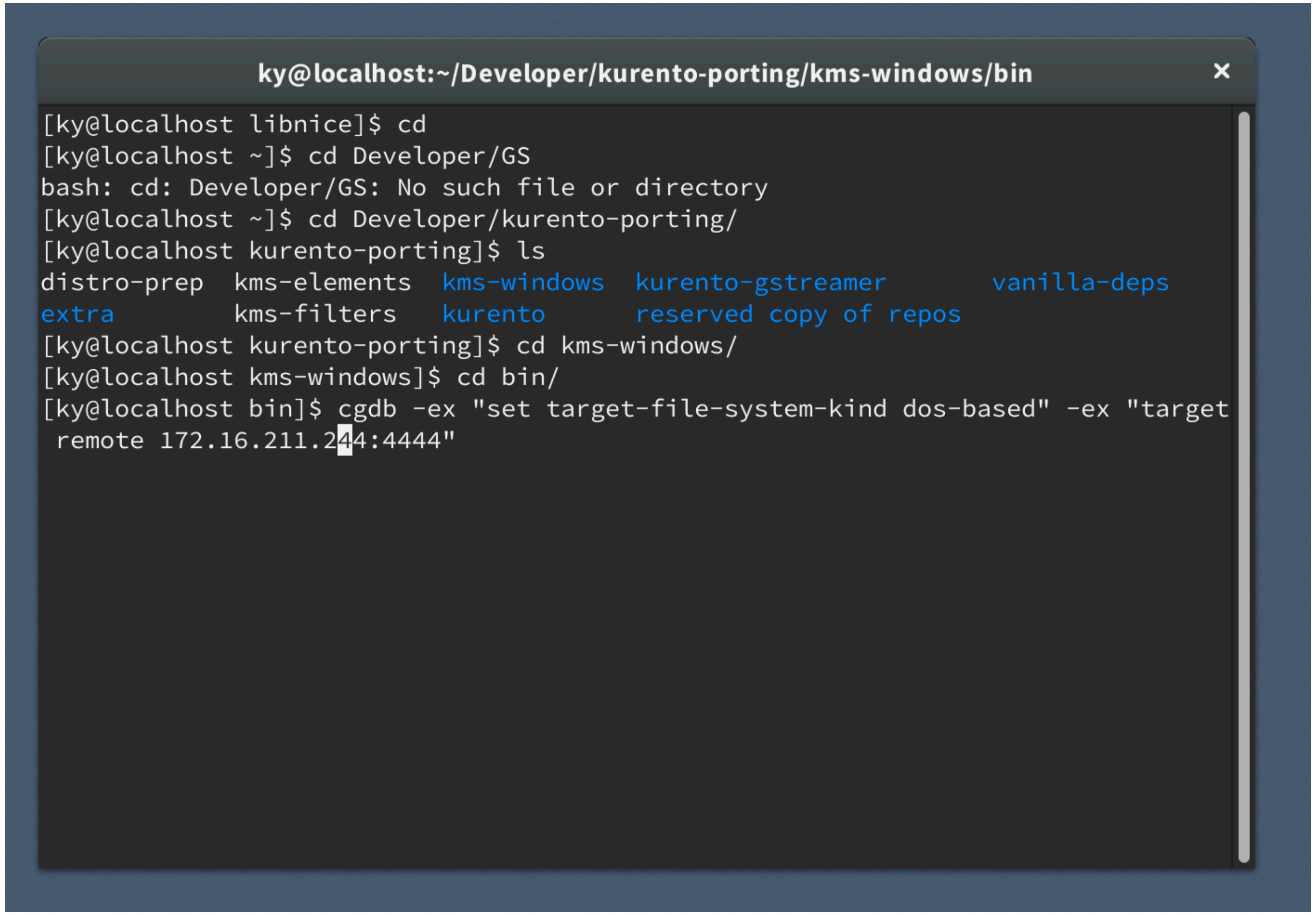
Allow your PC to be discoverable by other PCs and devices on this network. We recommend turning this on for private networks at home or work, but turning it off for public networks to help keep your stuff safe.

● On

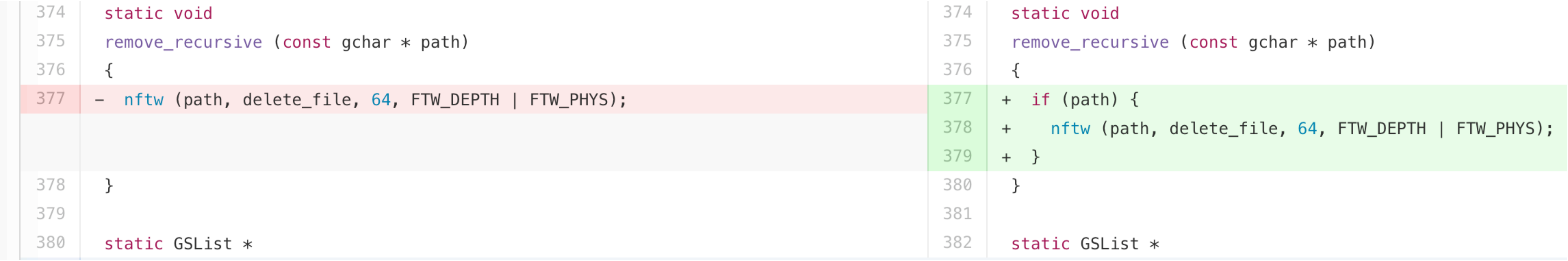
Properties

IPv4 address: 172.16.211.244

Manufacturer: Intel Corporation



```
[ky@localhost libnice]$ cd
[ky@localhost ~]$ cd Developer/GS
bash: cd: Developer/GS: No such file or directory
[ky@localhost ~]$ cd Developer/kurento-porting/
[ky@localhost kurento-porting]$ ls
distro-prep  kms-elements  kms-windows  kurento-gstreamer       vanilla-deps
extra        kms-filters   kurento      reserved copy of repos
[ky@localhost kurento-porting]$ cd kms-windows/
[ky@localhost kms-windows]$ cd bin/
[ky@localhost bin]$ cgdb -ex "set target-file-system-kind dos-based" -ex "target
 remote 172.16.211.244:4444"
```



```
 static void                                               static void
 remove_recursive (const gchar * path)                     remove_recursive (const gchar * path)
 {                                                         {
-  nftw (path, delete_file, 64, FTW_DEPTH | FTW_PHYS);    +  if (path) {
                                                          +    nftw (path, delete_file, 64, FTW_DEPTH | FTW_PHYS);
                                                          +  }
 }                                                         }

 static GSList *                                           static GSList *
```

To do:

- Make KMS work as standard Windows Service
- Fix random sequences generation
- Fix paths-related problems

Thank you!

You can come closer and take a look how it works