**Compilation des librairies de libsodium à zyre pour Windows 32Bits avec MSCV2015**

Outils :

* GIT Bash
* CMD
* Visual Studio 2015

Avant-propos : Lance la console GIT Bash  et crée un nouveau répertoire example :

cd D:

mkdir test

cd test

1. Compiler Libsodium (dernière version stable):

Cloner le projet dans le répertoire de travail et rentrer dans le répertoire build du répertoire projet libsodium correspondant au build MSVC.

git clone --depth 1 -b stable <https://github.com/jedisct1/libsodium.git>

cd libsodium\builds\msvc\build

Lancer le cmd depuis le GIT Bash et exécuter le fichier buildall.bat :

cmd

buildall.bat

exit

cd ..\..\..\..

1. Compiler Libzmq (4.2.2) :

Cloner le projet dans le répertoire de travail et rentrer dans le répertoire build du répertoire projet libzmq pour MSVC.

git clone git://github.com/zeromq/libzmq.git

cd libzmq\builds\msvc

Lancer le cmd depuis le GIT Bash et exécuter le fichier avant de compiler la lib avec buildall.bat.

cmd

configure.bat

cd build

buildall.bat

exit

cd ..\..\..\..

1. Compiler Libczmq (4.0.2) :

Cloner le projet dans le répertoire de travail et rentrer dans le répertoire build du répertoire projet libczmq pour MSVC.

git clone git://github.com/zeromq/czmq.git

cd czmq\builds\msvc

Lancer le cmd depuis le GIT Bash et exécuter le fichier configure.bat.

cmd

configure.bat

Le fichier czmq\_selftest.vcxproj est incomplet et provoque une erreur de compilation, il vous faudra pour remédier au problème éditer le fichier et rajouter après la ligne 132 :

<ClCompile Include="..\..\..\..\src\czmq\_selftest.c" />

les deux lignes suivantes:

<ClCompile Include="..\..\..\..\src\czmq\_private\_selftest.c" />

<ClCompile Include="..\..\..\..\src\zgossip\_msg.c" />

Et vous pouvez executer le fichier buildall.bat

cd msvc2015

build.bat

exit

cd ..\..\..\..

1. Compiler zyre (2.0.0):

Cloner le projet dans le répertoire de travail et rentrer dans le répertoire build du répertoire projet zyre pour MSVC.

git clone git://github.com/zeromq/zyre.git

cd zyre\builds\msvc

Corriger dans les sources, le fichier zyre\_node.c en modifiant la ligne 330 par :

zsys\_info (" - uuid=%s name=%s endpoint=%s connected=%s ready=%s sent\_seq=%s PRIu16 want\_seq=%s PRIu16",

Exécuter le fichier configuration.bat et ouvrer le fichier projet zyre.sln dans le répertoire vs2015.

1. Résultat des compilations :

Les sorties de compilation en debug et release sont trouvables dans le répertoire de compilation dans chacun des répertoires projets :

* bin/Win32/Debug/v140/dynamic/….
* bin/Win32/Release/v140/dynamic/….

1. Compilation de yajl :

Tout d’abord, la compilation de la lib nécessite CMAKE : <https://cmake.org/download/>

Cloner le projet dans le répertoire de travail :

git clone https://github.com/lloyd/yajl.git

Lancer l’invite de commandes développeur pour VS2015 (effectuer une recherche dans la barre de recherche rapide d’application de Windows.) et accéder au répertoire dans lequel vous avez cloné le projet yajl.

Créer un répertoire build et accéder au répertoire. Finalement compiler le projet en utilisant cmake and nmake.

mkdir buildDebug

cd buildDebug

cmake -G"NMake Makefiles" -DCMAKE\_BUILD\_TYPE=Debug ..

nmake

Puis faite la même chose pour compiler la version en release.

mkdir buildRelease

cd buildRelease

cmake -G"NMake Makefiles" -DCMAKE\_BUILD\_TYPE=Release ..

nmake