New post for new botan upp project.

Now Compiles with VC9 and MingW.

Before changing compiler you must run the configure.pl script, I tried to automate this or make some adjustments to the project with no success.

So, for Microsoft Visual Studio compiler run

perl configure.pl -cc msvc

For Mingw GCC (in my case I have to disable TR1 so)

perl configure.pl -cc gcc --with-tr1=none

Either MSVC and Mingw compilation gives me a lot of warnings

I'm using Ultimate++ build 1393   witch adds some interesting options like internal includes.
So I changed my project file, no more /I compiler options.

Some test code


```
#include <botan/botan.h>
#include <botan/pbkdf2.h>
#include <botan/hmac.h>
#include <botan/sha160.h>

...

 void DoBotan()
 {
  using namespace Botan;
  try{
   LibraryInitializer init;

       AutoSeeded_RNG rng;

       std::auto_ptr<S2K> s2k(get_s2k("PBKDF2(SHA-1)"));
       s2k->set_iterations(8192);
       s2k->new_random_salt(rng, 8);

       SymmetricKey key = s2k->derive_key(16, "TEST");
```

```
    std::string alg = "AES/CBC/PKCS7";
    Pipe enc(get_cipher(alg, key, ENCRYPTION), new Hex_Encoder);
    Pipe dec(new Hex_Decoder, get_cipher(alg, key, DECRYPTION));
    String secret = ToUtf8(txtIn.GetText());
    enc.process_msg(secret);
    String cipher = enc.read_all_as_string();
    dec.process_msg(cipher);
    String bubu = dec.read_all_as_string();

    PromptOK(bubu);
  }
  catch (std::exception se)
  {
   String err;
   err << "Error \n" << se.what();
   PromptOK(err);
  }
 }
```

Assist++ is still a bit broken try to list the enc variable methods for example, many are missing.

Created a poll to spice discussion.

Next is to create a simple Botan Wrapper


File Attachments
1) Botan.upp, downloaded 818 times

Botan VS Crypto++(total votes: 3)

Botan of course  1/(33%)
Crypto++ Rules  2/(67%)