Rafał Dadura

Juliusz Kuzyka

Bezpieczeństwo danych (BDAN)

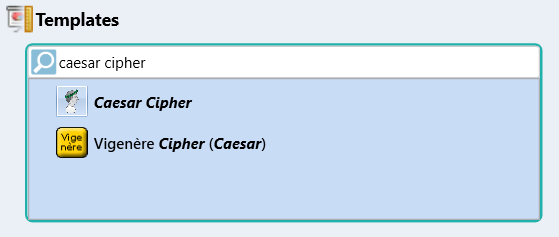
***Projekt 1***

Zadania do Tematu 1: Cryptool

Zadanie 1 – Caesar

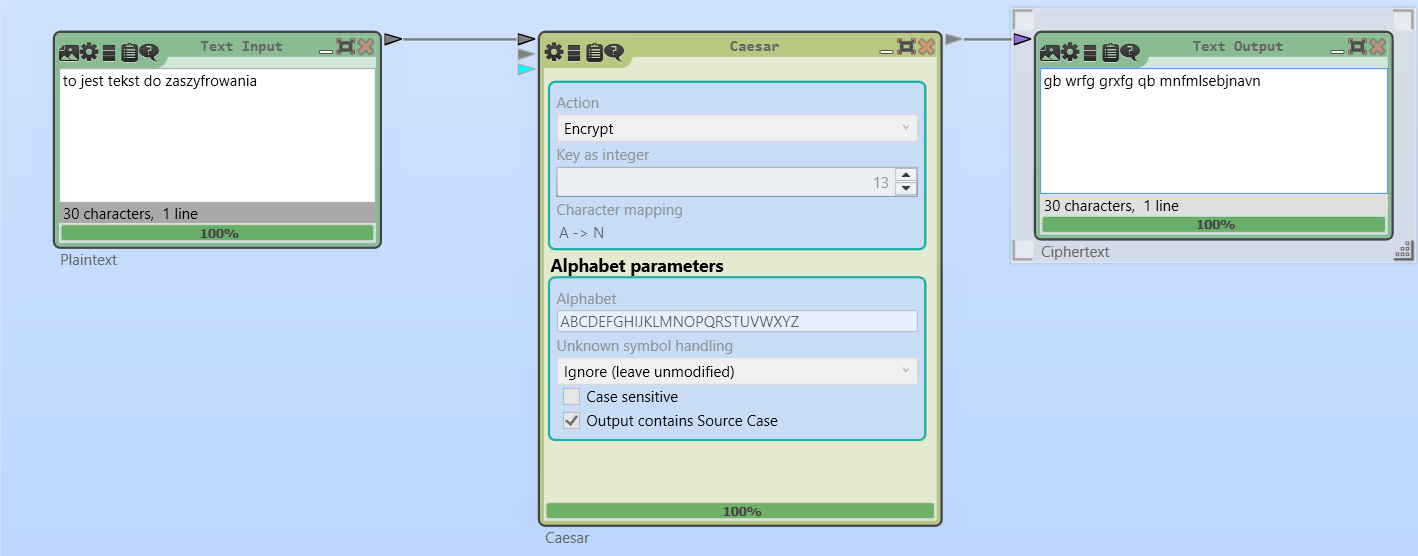
Polecenie: Zaszyfrować tekst **„to jest tekst do zaszyfrowania”** za pomocą szyfru Cezara z przesunięciem równym 13.

Na początku w zakładce Templates wyszukaliśmy **Caesar Cipher**.



Następnie w pole Text\_Input wpisaliśmy tekst jawny **„to jest tekst do zaszyfrowania”**.

Następnie w pole Key as integer wpisaliśmy liczbę **13**

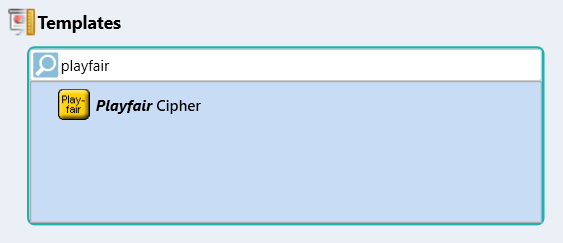


Po włączeniu programu w polu Output\_Text otrzymaliśmy szyfr: **gb wrfg grxfg qb mnfmlsebjnavn**.

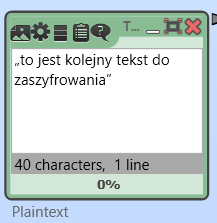
Zadanie 2 – Playfair

Polecenie: Zaszyfrować tekst **„to jest kolejny tekst do zaszyfrowania”** za pomocą szyfru Playfair z kluczem **„key”**.

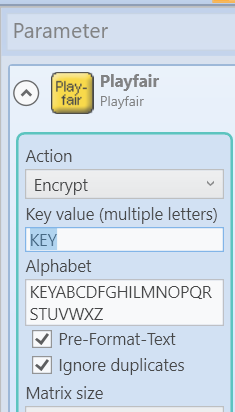
Na początku w zakładce Templates wyszukaliśmy **Playfair Cipher**.

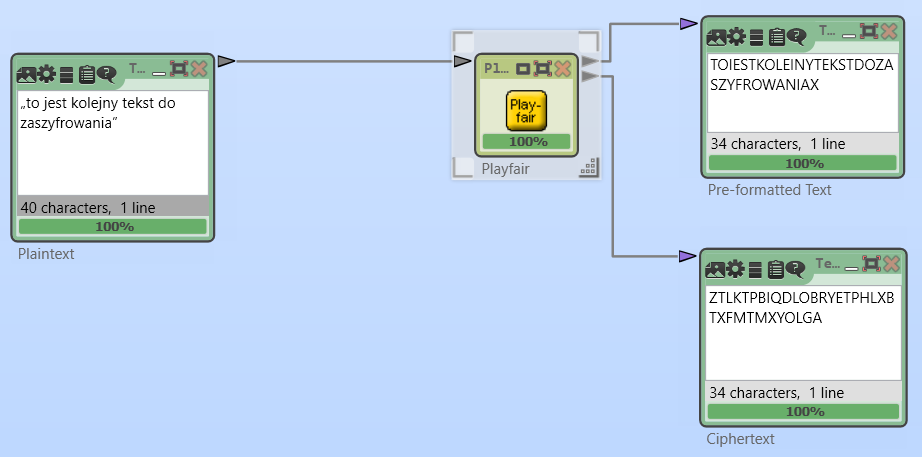


Następnie w pole Text\_Input wpisaliśmy tekst jawny **„to jest kolejny tekst do zaszyfrowania”**.



Następnie w pole Key value (multiple letters) wpisaliśmy wyraz **key.**



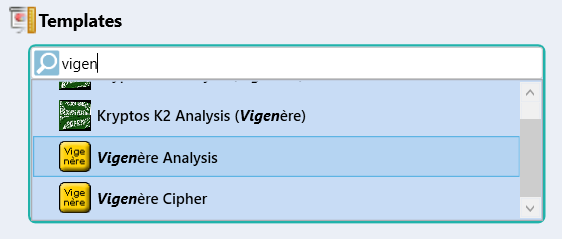


Po włączeniu programu w polu Output\_Text otrzymaliśmy szyfr: **ZTLKTPBIQDLOBRYETPHLXBTXFMTMXYOLGA**.

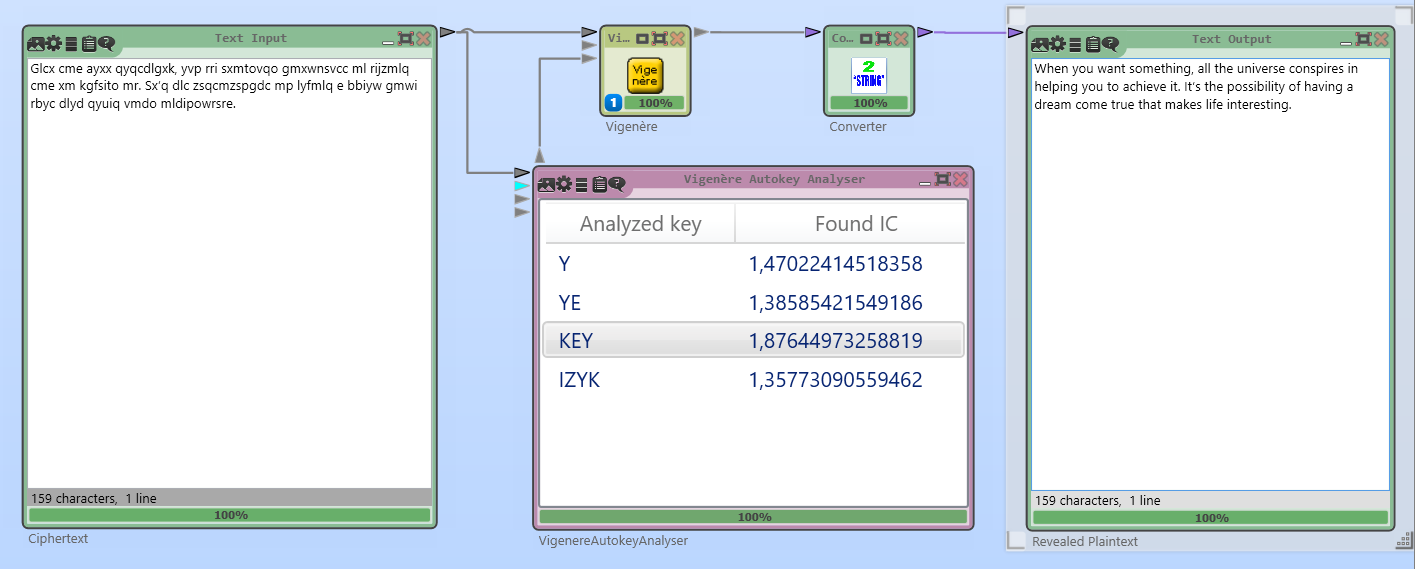
Zadanie 3 – Vigenère

Polecenie: Dokonać kryptoanalizy tekstu zaszyfrowanego szyfrem Vigenère: **„Glcx cme ayxx qyqcdlgxk, yvp rri sxmtovqo gmxwnsvcc ml rijzmlq cme xm kgfsito mr. Sx’q dlc zsqcmzspgdc mp lyfmlq e bbiyw gmwi rbyc dlyd qyuiq vmdo mldipowrsre.”**

Na początku w zakładce Templates wyszukaliśmy **Vigenère Analysis**.



Następnie w pole Text\_Input wpisaliśmy tekst zaszyfrowany **„Glcx cme ayxx qyqcdlgxk, yvp rri sxmtovqo gmxwnsvcc ml rijzmlq cme xm kgfsito mr. Sx’q dlc zsqcmzspgdc mp lyfmlq e bbiyw gmwi rbyc dlyd qyuiq vmdo mldipowrsre.”**.

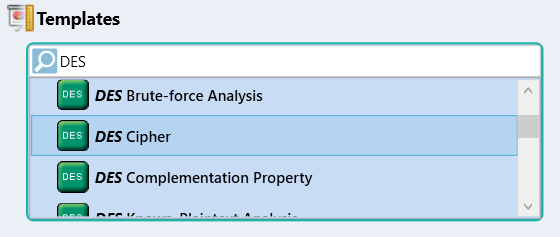


Po włączeniu programu w polu Output\_Text otrzymaliśmy: **When you want something, all the universe conspires in helping you to achieve it. It’s the possibility of having a dream come true that makes life interesting.**

Zadanie 4 – DES

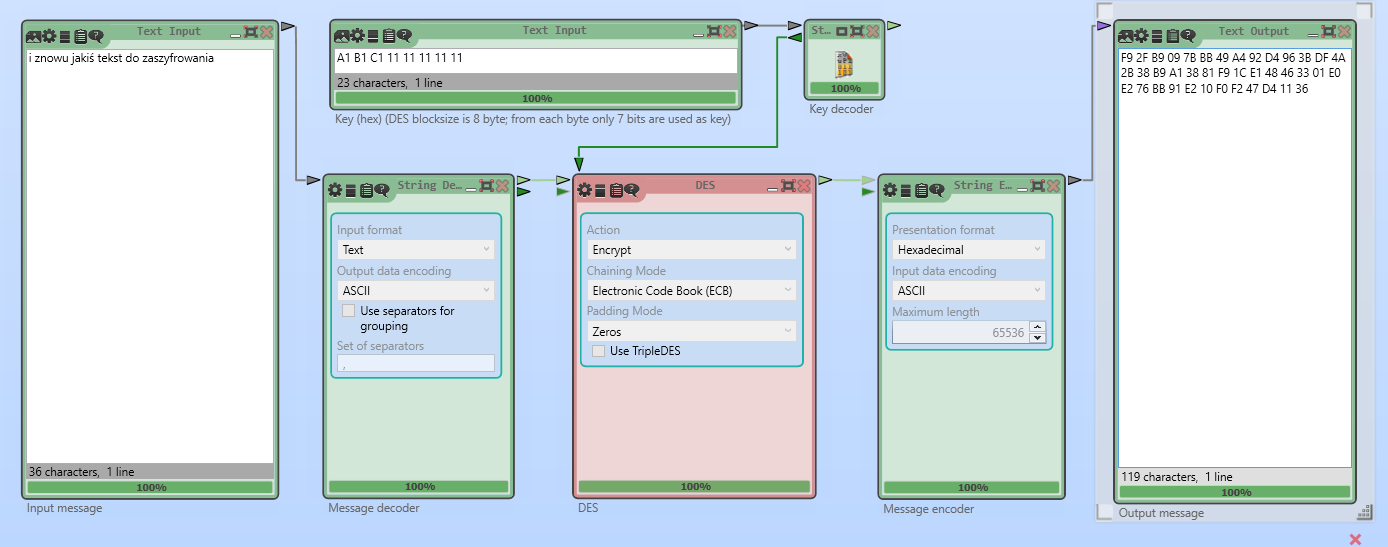
Polecenie: Zaszyfrować tekst **„i znowu jakiś tekst do zaszyfrowania”** używając DES w trybie ECB i CBC. Porównać szyfrogramy.

Na początku w zakładce Templates wyszukaliśmy **DES Cipher**.



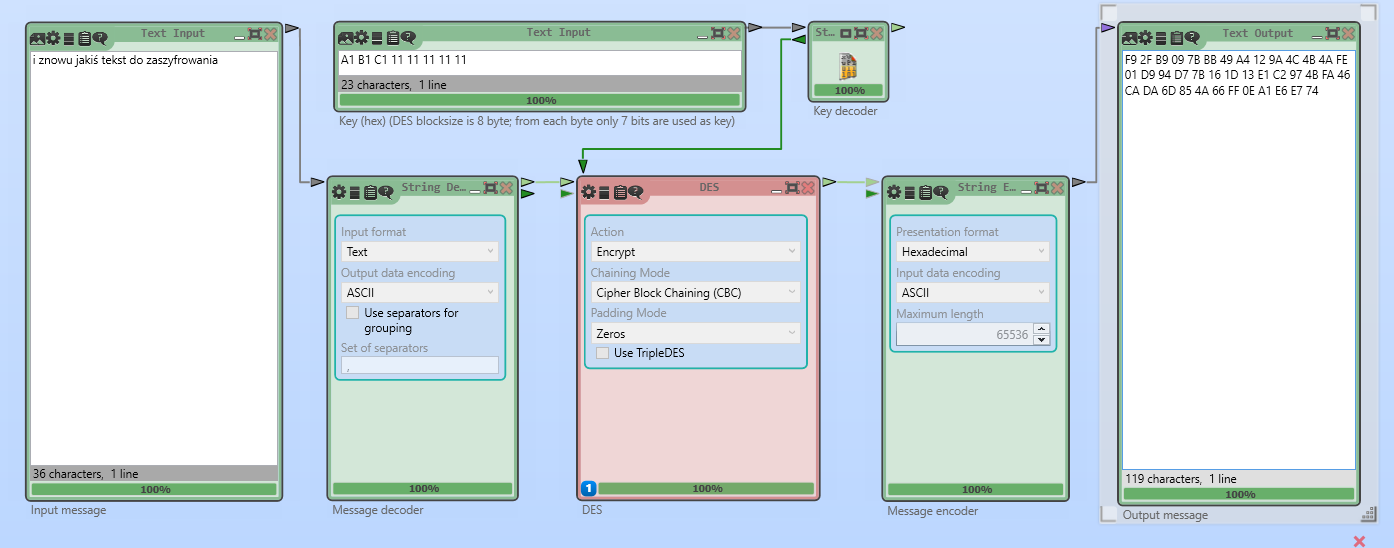
Następnie w pole Text\_Input wpisaliśmy tekst jawny **„i znowu jakiś tekst do zaszyfrowania”**.

Następnie w polu Chaining mode wybraliśmy opcję **Electronic Code Book (ECB).**



Po włączeniu programu w polu Output\_Text otrzymaliśmy szyfr: **F9 2F B9 09 7B BB 49 A4 92 D4 96 3B DF 4A 2B 38 B9 A1 38 81 F9 1C E1 48 46 33 01 E0 E2 76 BB 91 E2 10 F0 F2 47 D4 11 36**

Następnie w polu Chaining mode wybraliśmy opcję **Cipher Block Chaining (CBC).**



Po włączeniu programu w polu Output\_Text otrzymaliśmy szyfr: **F9 2F B9 09 7B BB 49 A4 12 9A 4C 4B 4A FE 01 D9 94 D7 7B 16 1D 13 E1 C2 97 4B FA 46 CA DA 6D 85 4A 66 FF 0E A1 E6 E7 74**

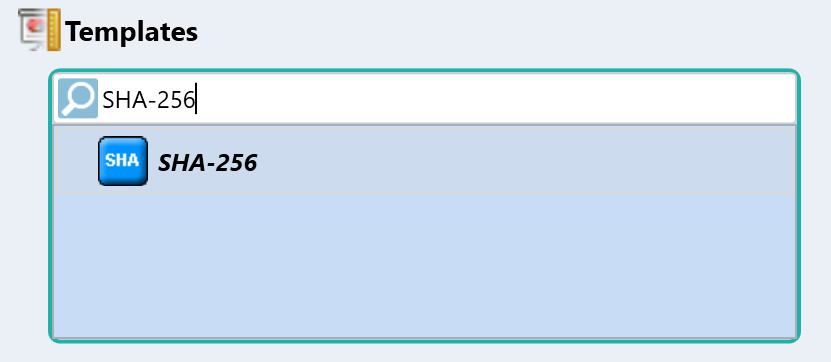
Porównanie szyfrogramów:

W obu szyfrogramach sekwencja **F9 2F B9 09 7B BB 49 A4** jest taka sama na początku szyfrogramów. Po tej sekwencji tekst się znacznie różni.

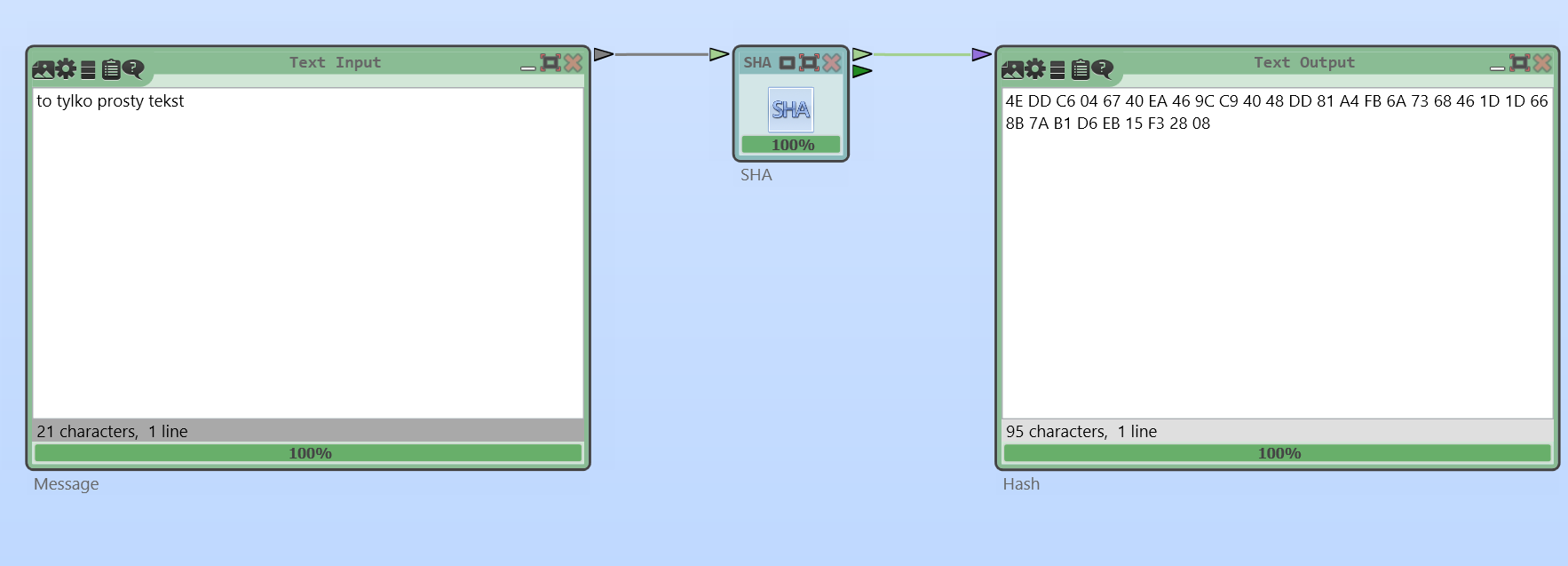
Zadanie 5 – SHA-256

Polecenie: Obliczyć wartość funkcji skrótu SHA-256 dla tekstu **„to tylko prosty tekst”.**

Na początku w zakładce Templates wyszukaliśmy **SHA-256**.



Następnie w pole Text\_Input wpisaliśmy tekst jawny **„to tylko prosty tekst”**.



Po włączeniu programu w polu Output\_Text otrzymaliśmy szyfr: **4E DD C6 04 67 40 EA 46 9C C9 40 48 DD 81 A4 FB 6A 73 68 46 1D 1D 66 8B 7A B1 D6 EB 15 F3 28 08**.

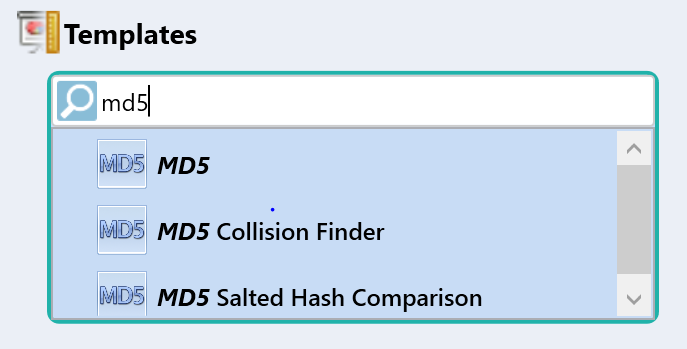
Zadanie 6 – MD5

Polecenie: Obliczyć skrót MD5 dla dwóch ciągów znakowych zapisanych heksadecymalnie i skomentować wynik:

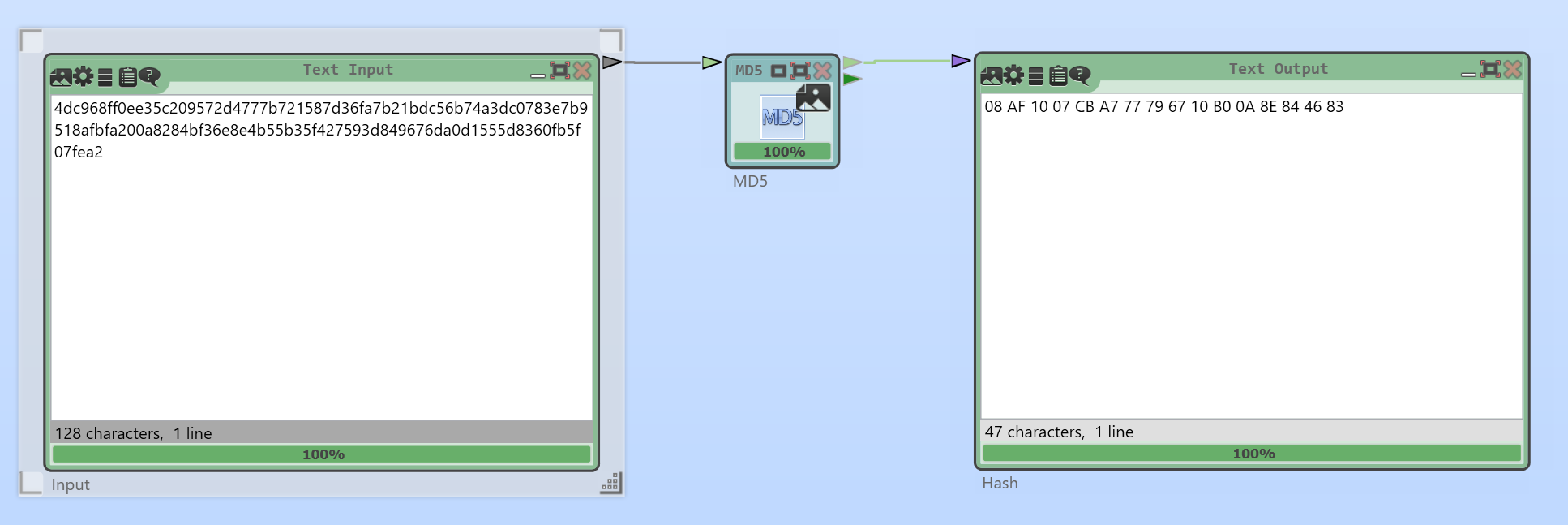
**4dc968ff0ee35c209572d4777b721587d36fa7b21bdc56b74a3dc0783e7b9518afbfa200a8284bf36e8e4b55b35f42759 3d849676da0d1555d8360fb5f07fea2**

**4dc968ff0ee35c209572d4777b721587d36fa7b21bdc56b74a3dc0783e7b9518afbfa202a8284bf36e8e4b55b35f42759 3d849676da0d1d55d8360fb5f07fea2**

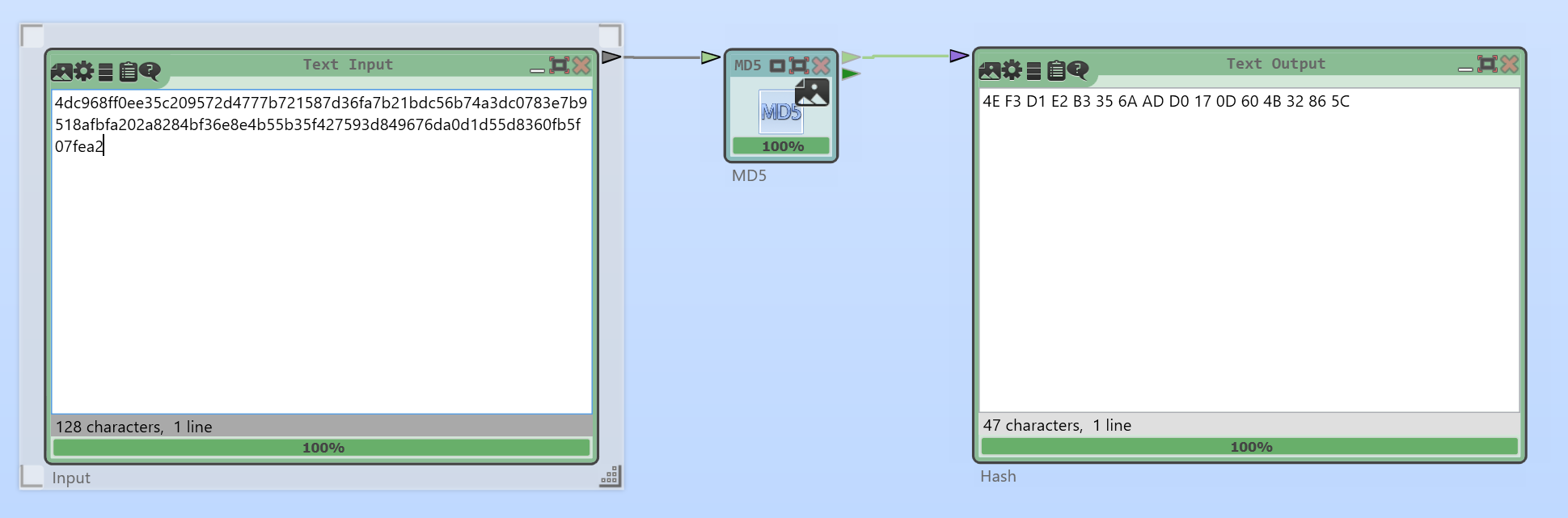
Na początku w zakładce Templates wyszukaliśmy **MD5**.



Potem w pole Text\_Input wpisaliśmy Tekst 1.



A następnie Tekst 2.



Otrzymany Output po Tekście 1: **08 AF 10 07 CB A7 77 79 67 10 B0 0A 8E 84 46 83**.

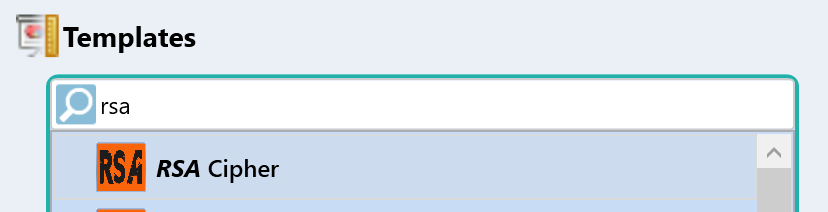
Otrzymany Output po Tekście 2: **4E F3 D1 E2 B3 35 6A AD D0 17 0D 60 4B 32 86 5C**.

Jak można zauważyć nasze teksty jawne, mające 130 znaków różniły się jedynie na dwóch pozycjach, jednak szyfry otrzymane w MD5 różnią się od siebie znacznie. Pokazuje to, że nawet przy niewielkiej zmianie tekstu jawnego szyfr znacznie się zmieni.

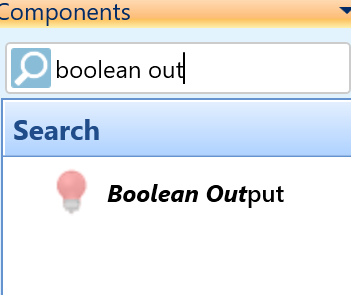
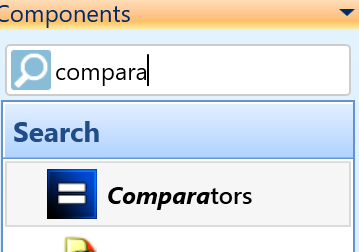
Zadanie 7 – RSA

Polecenie: Stworzyć schemat podpisu wiadomości RSA. Wybrać bezpieczne parametry podpisu. Zweryfikować działanie podpisu na wybranej wiadomości.

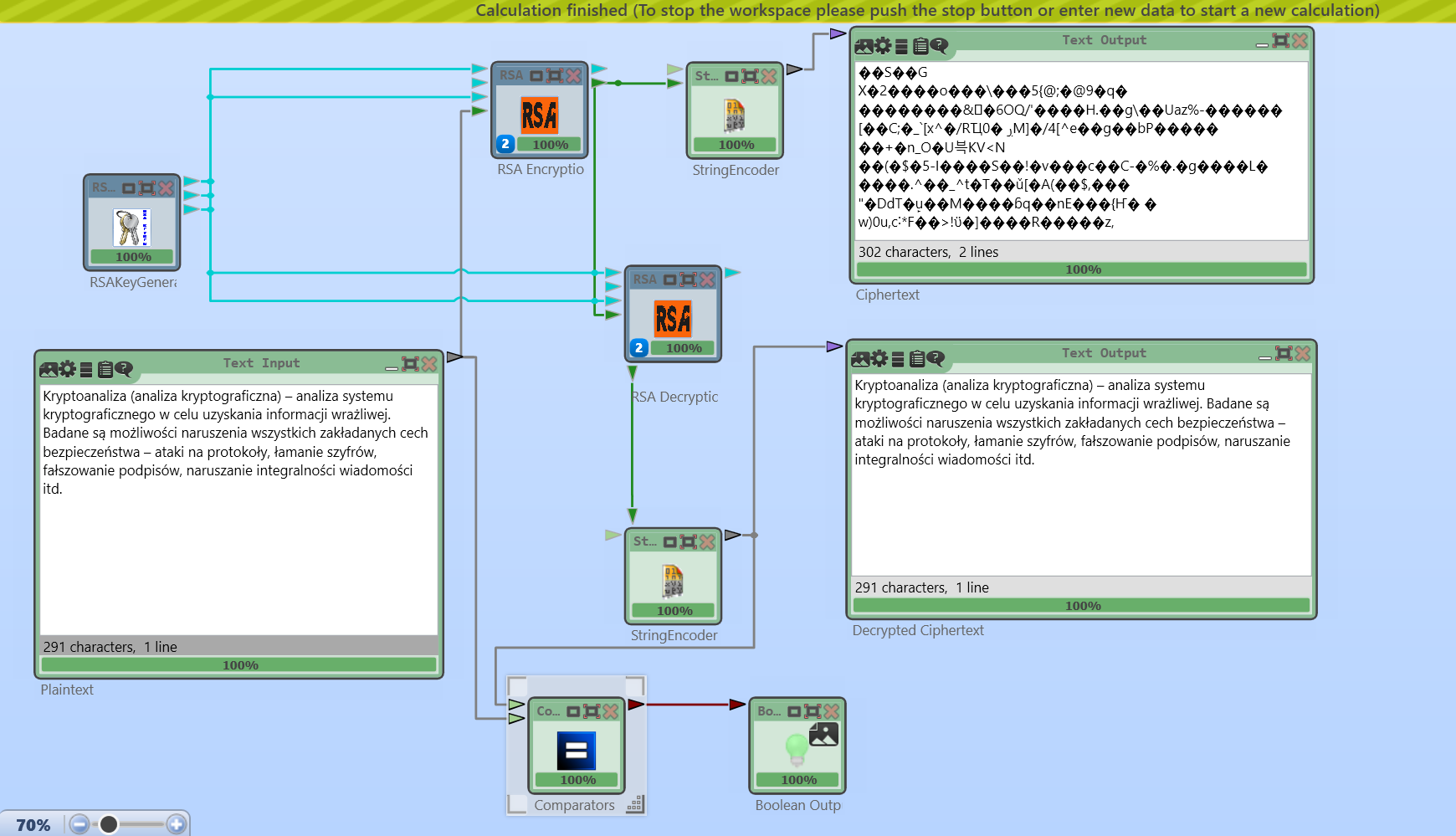
Na samym początku w zakładce Templates wyszukaliśmy RSA Cipher.



Następnie w zakładce Tools wybraliśmy narzędzie **Comparators**, które pozwoli nam na porównanie tekstu przed szyfrowaniem oraz po zaszyfrowaniu oraz **Boolean Output**, które pozwoli nam na potwierdzenie działania szyfru.



Po naciśnięciu przycisku **Play** sprawdziliśmy czy nasz program rzeczywiście poprawnie szyfruje tekst.



Wnioski

Wykonywanie tego projektu umożliwiło nam na lepsze poznanie urządzenia Cryptool oraz tego, jak się z niego korzysta. Dowiedzieliśmy się również o wielu różnych szyfrach oraz tego, jakie są ich wady i zalety.