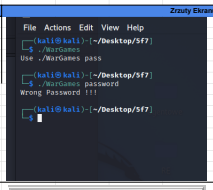
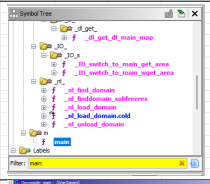
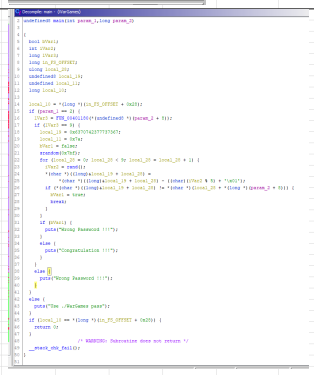
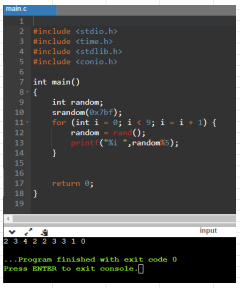
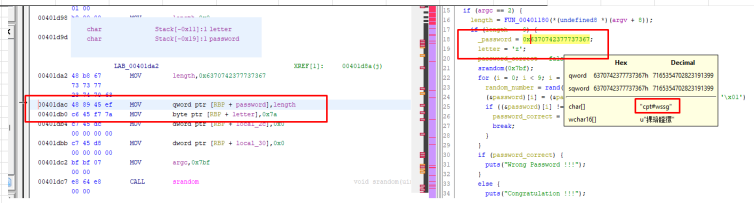
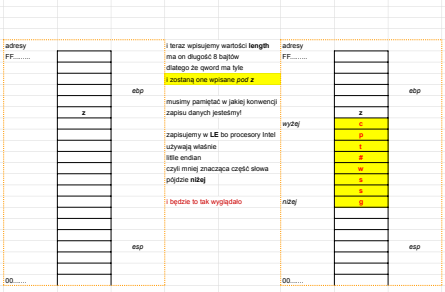
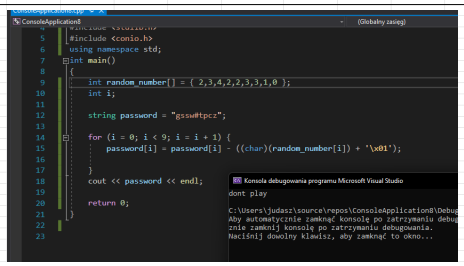
Autor:	YugnatD
Tytuł:	WarGames
Język:	C/C++
Platforma:	Unix/linux
Trudność:	3.0
Jakość:	4.2
Arch:	x86-64
Link:	https://crackmes.one/crackme/5f78ae7533c5d4367b3b03b8
Opis:	Use ./WarGames pass
Plik:	https://crackmes.one/static/crackme/5f78ae7533c5d4367b3b03b8.zip

Krok	Rozwiązanie	Zrzuty Ekranu
1	Sprawdzenie jak działa plik .exe Działanie polega na fladi jako argumenty. Zwraca czy wartość jest poprawna czy nie.	
2	Zacznijmy od odnalezienia funkcje **main**, jeżeli taka istnieje Z pomocą służy **Symbol Tree**	
3	Tak wygląda funkcja **main** po **prawej** stronie jest **zmodyfikowana funkcja main** względem nazwa i typow zmiennych Po analize głównej **pętli** w mainie widzimy że nasza flaga jest generowana za pomocą funkcji **rand()** dodatkowo znamy **seed'a** do **srandom** a nasza flaga do modyfikacji jest w zmiennej **password** tutaj błędnie nazwałem tą zmienną boolean **password_correct** z logicznych względów następnych if'ów odpowiedniejsza byłaby nazwa **password_incorrect** *oczywiscie ta literkowka nie przeszkadza nam rozwiązać zadania*	 
4	na początek zajmiemy się tą funkcją random w pętli jest ona używana do zmieniania wartości naszej flagi odtworzyłem program w C nadając tego samego **seed'a** przez co generowane "losowo" liczby będą takie same jak w naszym .exe dodatkowo uwzględniłem **modulo 5** żeby operować już na docelowych wartościach Oto one **2,3,4,2,2,3,3,1,0**	
5	Teraz zajmiemy się tym co jest w zmiennej **password** ponieważ domyślnie jest tam tylko wartość **"cpt#wssg"** nasza pętla iteruje **9 razy** a długość tego wynosi tylko **8** żeby bardziej to zrozumieć warto spojrzeć na kod w assemblerze widzimy na początku 1 zrzutu ekranu że letter i password są gdzieś na **stosie** napoczątku przesuwamy **0x6370742377737367** co w ascii wygląda **"cpt#wssg"** a potem przesuwamy **0x7a** oznacza **'z'** Patrząc po wartościach *movow* widzimy że będą one obok siebie	
6	Teraz by nie popełnić błędu przy odczytywaniu tego stringa myślac sobie o to hasło pewnie wygląda coś w stylu: **"cpt#wssgz"** albo **"zcpt#wssg"** co jest kluczowe przy dalszym procesie odwracania tej flagi bo na tym będziemy bazować trzeba sobie wyjasnić sobie parę rzeczy ze stosem zaprezentuje **uproszczoną** jego wersję napoczątku ***mov* 'z'** *(w przykładowe miejsce!)* i teraz uwzględniając że password = **-0x19** to wartość **length** będzie pod **z** Podsumowanie tego kroku: czyli nasza flaga do obróbki wygląda tak **gssw#tpcz**	
6	Teraz znając tablice liczb losowych oraz nasze password stworzyłem program w .cpp że odtworzyć proces otrzymałem flagę **dont play** sprawdźmy czy działa	

7

Flaga jest podawana jako argument przy wywolaniu funkcji

dodatkową uwagą jest że trzeba znak specjalny
taki jak **spacja** polaczyć ze znakiem \

Flaga działa
koniec zadania