

개인 공부

2020.12.01 김현진

airodump-ng

모니터링 모드를 통해 네트워크 정보를 출력해주는 프로그램

Aircrack-ng

aircrack-ng , airdecap-ng , airmon-ng
aireplay-ng , airodump-ng

airodump-ng

Monitor Mode



공기 주변에 뿌려지는 모든 패킷을 볼 수 있는 모드

* Managed 모드 : 나에게 오는 패킷만 볼 수 있는 모드

airodump-ng

AP 의 MAC

AP의 이름

```
root@kali:~# airodump-ng wlan0
ioctl(SIOCSIWMODE) failed: Device or resource busy

CH 9 ][ Elapsed: 0 s ][ 2020-12-01 17:41
```

BSSID	PWR	Beacons	#Data, #/s	CH	MB	ENC CIPHER	AUTH	ESSID
FA:8F:CA:5A:D2:34	-89	2	0 0	9	65	OPN		<length: 0>
88:36:6C:CB:E2:E0	-89	3	0 0	9	270	WPA2 CCMP	PSK	iptime1469
88:3C:1C:74:5F:E4	-77	3	0 0	9	360	CCMP	PSK	KT_GiGA_2G_5FE0
88:3C:1C:B3:6D:56	-61	2	0 0	3	360	CCMP	PSK	KT_GiGA_2G_Wave2_6D52
0A:5D:DD:F6:AC:BB	-89	3	0 0	8	130	WPA2 CCMP	PSK	<length: 7>
42:23:AA:DE:99:92	-45	4	0 0	8	130	WPA2 CCMP	PSK	SK_WiFiGIGA9990_2.4G
00:23:AA:DE:99:92	-45	4	0 0	8	130	WPA2 CCMP	PSK	SK_WiFiGIGA9990
12:23:AA:DE:99:92	-44	6	0 0	8	130	WPA2 CCMP	PSK	<length: 7>
E0:3F:49:9D:70:A8	-83	3	0 0	7	195	WPA2 CCMP	PSK	dollyi_RPT2G
08:5D:DD:B4:38:2E	-87	2	0 0	1	130	OPN		grace-hospital_2708
C0:4A:00:76:B8:A0	-59	4	0 0	1	270	WPA2 CCMP	PSK	CHADE-2.4GHz

BSSID	STATION	PWR	Rate	Lost	Frames	Notes	Probes
00:23:AA:DE:99:92	50:77:05:62:51:8F	-28	0 -24	0	1		
(not associated)	20:3D:BD:DA:57:64	-90	0 - 1	29	7		KT_GiGA_2G_Wave2_912B

```
[1]+ Stopped airodump-ng wlan0
root@kali:~#
```

beacon 갯수

부가적인 정보

Beacon packet

AP가 보내는 방송 프레임

프레임 타입

MAC 주소

SSID

부가적인 정보

```
Frame 24: 385 bytes on wire (3080 bits), 385 bytes captured (3080 bits) on int
Radiotap Header v0, Length 24
802.11 radio information
IEEE 802.11 Beacon frame, Flags: .....
Type/Subtype: Beacon frame (0x0008)
Frame Control Field: 0x8000
.0000 0000 0000 0000 = Duration: 0 microseconds
Receiver address: Broadcast (ff:ff:ff:ff:ff:ff)
Destination address: Broadcast (ff:ff:ff:ff:ff:ff)
Transmitter address: Mercury_b3:6d:56 (88:3c:1c:b3:6d:56)
Source address: Mercury_b3:6d:56 (88:3c:1c:b3:6d:56)
BSS Id: Mercury_b3:6d:56 (88:3c:1c:b3:6d:56)
.... .... 0000 = Fragment number: 0
0011 0001 0010 .... = Sequence number: 786
IEEE 802.11 Wireless Management
Fixed parameters (12 bytes)
Tagged parameters (325 bytes)
Tag: SSID parameter set: KT GiGA 2G Wave2 6D52
Tag: Supported Rates 1(B), 2(B), 5.5(B), 11(B), 9, 18, 36, 54, [Mbit/sec]
Tag: DS Parameter set: Current Channel: 3
Tag: Traffic Indication Map (TIM): DTIM 0 of 0 bitmap
Tag: Country Information: Country Code KR, Environment Any
```

프로그램

```
Input Your Interface !!
\n);
sample: wlan0
Input : wlan0
Interface : wlan0

**** What Choice? ****
* 1.Scanning          *
* 2.Stop Scanning     *
*****
```

no	BSSID	Ch	SSID
1	8a:3c:1c:8f:36:21	4	
2	ba:3c:1c:8f:36:21	4	SK_WiFiGIGA361E_2.4G
3	88:3c:1c:b3:6d:56	3	KT_GiGA_2G_Wave2_6D52
4	c0:4a:00:76:b8:a0	1	CHADE-2.4GHz
5	08:5d:dd:b4:38:2d	1	
6	18:c5:01:80:29:3a	1	U+Net293B
7	70:2c:1f:5f:5c:26	1	[refrigerator] Samsung
8	88:3c:1c:35:ee:37	1	KT_GiGA_2G_EE33_Info
9	08:5d:dd:50:b2:35	1	
10	08:5d:dd:50:b2:36	1	grace-hospital_2805
11	08:5d:dd:b4:13:52	1	grace-hospital_2608
12	42:23:aa:de:99:92	8	SK_WiFiGIGA9990_2.4G
13	00:23:aa:de:99:92	8	SK_WiFiGIGA9990
14	12:23:aa:de:99:92	8	
15	70:5d:cc:6e:bc:ae	2	kiss

interface 입력 : wlan0

네트워크 정보 : BSSID / CH / SSID

BSSID : 공유기 MAC 주소를 알기 위함

CH : 채널을 알기 위함

SSID : WiFi 이름을 알기 위함

프로그램 코드

패킷 구조체

```
46 #pragma pack(push, 1)
47 struct IEEE80211_radiotap_header {
48     u_int8_t    it_version;
49     u_int8_t    it_pad;
50     u_int16_t   it_len;
51     u_int32_t   it_present;
52 } __attribute__((__packed__));
53 #pragma pack(pop)
54
55 #pragma pack(push, 1)
56 struct Beacon_Frame {
57     uint16_t    Type;
58     uint16_t    Dur;
59     u_char      dst_mac[6];
60     u_char      src_mac[6];
61     u_char      bssid[6];
62     uint16_t    number;
63 };
64 #pragma pack(pop)
65
66 #pragma pack(push, 1)
67 struct Fixed {
68     uint8_t     Time[8];
69     uint16_t    Interval;
70     uint16_t    capabilities;
71 };
72 #pragma pack(pop)
```

패킷 스캔

```
33 printf("\nno    BSSID                Ch
34 printf("-----
35
36 while(my_th_info->th_ex) {
37     struct pcap_pkthdr* header;
38     const u_char* packet;
39
40     int res = pcap_next_ex(handle, &header, &packet);
41     if (res == 0) continue;
42     if (res == -1 || res == -2) {
43         printf(" pcap_next_ex error \n");
44         exit (0);
45     }
46
47     struct IEEE80211_radiotap_header * radiotab;
48     radiotab = (struct IEEE80211_radiotap_header *)packet;
49     packet += radiotab->it_len;
50
51     struct Beacon * beacon_frame;
52     beacon_frame = (struct Beacon *)packet;
53     u_short b_type = ntohs(beacon_frame->type);
54
55     if (b_type != Beacon_type) continue;
56
57     memcpy(&ssid_size, &beacon_frame->tag, 1);
```

프로그램 동작

```
19 int Num, Num2, res, status;
20 pthread_t p_thread[2];
21 struct Thr_info * th_info;
22 th_info = (Thr_info *)malloc(sizeof(Thr_info));
23
24 memcpy(th_info->Dev, Dev, sizeof(Dev));
25
26 while(true) {
27
28     printf("Choice : ");
29     scanf("%d", &Num);
30
31     sleep(1);
32
33     if (Num == 1) {
34         th_info->th_ex = 1;
35         res = pthread_create(&p_thread[0], NULL, scan, (void *)th_info);
36         if (res < 0) {
37             perror("thread create error : ");
38             exit (0);
39         }
40     }
41     if (Num == 2) {
42         printf("\nStop Scanning....\n\n");
43         th_info->th_ex = 0;
44         pthread_join(p_thread[0], (void **)&status);
45     }
46 }
```

프로그램 코드

```
Input Your Interface !!
v);
sample: wlan0
Input : wlan0
Interface : wlan0

**** What Choice? ****
* 1.Scanning *
* 2.Stop Scanning *
*****
```

기능

1. 패킷 스캔
 2. 패킷 길이 읽기
 3. 패킷 정보 출력
 4. 중복된 패킷 처리
 5. 스캔하고 나서 중지
- ...

프로그램 코드

```
Input Your Interface !!
v);
sample: wlan0
Input : wlan0
Interface : wlan0

**** What Choice? ****
* 1.Scanning *
* 2.Stop Scanning *
*****
```

기능

1. 패킷 스캔
2. 패킷 길이 읽기
3. 패킷 정보 출력
- 4. 중복된 패킷 처리**
- 5. 스캔하고 나서 중지**

...

MAP

Key , value 가 쌍으로 저장되는 함수

`#include <map>`

생성 > `map<std:string , int> m;`

key : string 자료형 / value : int 자료형

삽입 > `m.insert({key,value});`

key : string 자료형 / value : int 자료형

MAP

```
5 map<std::string, std::string> m; map 생성
6 //map<std::string, std::string> :: iterator it;
7
8 bool dump(u_char BSSID[],std::string ss) { //}, u_char *SSID) {
9
10     char bssid[] = "";
11     sprintf(bssid,"%02x%02x%02x%02x%02x%02x",BSSID[0],BSSID[1],BSSID[2],BSSID[3],BSSID[4],BSSID[5]);
12
13     std::string bs(bssid);
14
15     /* print key, value
16     for (it=m.begin();it !=m.end();it++) {
17         cout << "\nkey : " << it->first << "Value : " << it->second ;
18     }
19     */
20     if (m.find(bs) != m.end()) { 키가 존재하는지 확인 : find(key)
21         return true;
22     }
23     else { // not find
24         m.insert(pair<std::string,std::string>(bs,ss)); map에 key, value를 삽입
25         return false;
26     }
27 }
```

MAP

```
5 map<std::string, std::string> m; map 생성
6 //map<std::string, std::string> :: iterator it;
7
8 bool dump(u_char BSSID[],std::string ss) { //},
9
10     char bssid[] = "";
11     sprintf(bssid,"%02x%02x%02x%02x%02x%02x",BSSID[0],BSSID[1],BSSID[2],BSSID[3],BSSID[4],BSSID[5]);
12
13     std::string bs(bssid);
14
15     /* print key, value
16     for (it=m.begin();it !=m.end();it++) {
17         cout << "\nkey : " << it->first << "Value : " << it->second ;
18     }
19     */
20     if (m.find(bs) != m.end()) { 키가 존재하는지 확인 : find(key)
21         return true;
22     }
23     else { // not find
24         m.insert(pair<std::string,std::string>(bs,ss)); map에 key, value를 삽입
25         return false;
26     }
27 }
```

```
78 if (dump(BSSID,ssid)!=false) continue;
79
80
81 // print
82 printf("%d ",k);
83 if (k<10) printf(" ");
84 for (i=0;i <6;i++) {
85     printf("%02x",beacon_frame->b_frame.bssid[i]);
86     if(i<5) {
87         printf(":");
88     }
89 }
```

Tread

하나의 프로그램에서 여러가지 기능을
동시에 작업 할 수 있게 하는 함수

`#include <pthread.h>`

생성

```
int pthread_create( pthread_t *th_id, const pthread_attr_t *attr, void* 함수명,  
void *arg );
```

```
void pthread_exit( void* ret_value );
```

종료

```
int pthread_join( pthread_t th_id, void** thread_return );
```

종료시 자원해제

Tread

```
20 pthread_t p_thread[2];
21 struct Thr_info * th_info;
22 th_info = (Thr_info *)malloc(sizeof(Thr_info));
23
24 memcpy(th_info->Dev, Dev, sizeof(Dev));
25
26 while(true) {
27
28     printf("Choice : ");
29     scanf("%d", &Num);
30
31     sleep(1);
32
33     if (Num == 1) {
34         th_info->th_ex=1;
35         res = pthread_create(&p_thread[0], NULL, scan, (void *)th_info);
36         if (res < 0) {
37             perror("thread create error : ");
38             exit (0);
39         }
40     }
41     if (Num == 2) {
42         printf("\nStop Scanning...\n\n");
43         th_info->th_ex=0;
44         pthread_join(p_thread[0], (void **)&status);
45     }
```

현재 스레드의 식별자 정보를 담고 있음

Tread

```
20 pthread_t p_thread[2];
21 struct Thr_info * th_info;
22 th_info = (Thr_info *)malloc(sizeof(Thr_info));
23
24 memcpy(th_info->Dev, Dev, sizeof(Dev));
25
26 while(true) {
27
28     printf("Choice : ");
29     scanf("%d", &Num);
30
31     sleep(1);
32
33     if (Num == 1) {
34         th_info->th_ex=1;
35         res = pthread_create(&p_thread[0], NULL, scan, (void *)th_info);
36         if (res < 0) {
37             perror("thread create error : ");
38             exit (0);
39         }
40     }
41     if (Num == 2) {
42         printf("\nStop Scanning...\n\n");
43         th_info->th_ex=0;
44         pthread_join(p_thread[0], (void **)&status);
45     }
```

현재 스레드의 식별자 정보를 담고 있음

scan 함수

```
8 void * scan(void *th_info) {
9
10     char errbuf [PCAP_ERRBUF_SIZE];
11     int i,k=1;
12
13     struct Thr_info * my_th_info = (Thr_info *)th_info;
14     char dev[50] ;
15     memcpy(dev, my_th_info->Dev, sizeof(my_th_info));
```

Tread

```
20 pthread_t p_thread[2];
21 struct Thr_info * th_info;
22 th_info = (Thr_info *)malloc(sizeof(Thr_info));
23
24 memcpy(th_info->Dev, Dev, sizeof(Dev));
25
26 while(true) {
27
28     printf("Choice : ");
29     scanf("%d", &Num);
30
31     sleep(1);
32
33     if (Num == 1) {
34         th_info->th_ex=1;
35         res = pthread_create(&p_thread[0], NULL, scan, (void *)th_info);
36         if (res < 0) {
37             perror("thread create error : ");
38             exit (0);
39         }
40     }
41     if (Num == 2) {
42         printf("\nStop Scanning...\n\n");
43         th_info->th_ex=0;
44         pthread_join(p_thread[0], (void **)&status);
45     }
```

현재 스레드의 식별자 정보를 담고 있음

scan 함수

```
129 pcap_close(handle);
130 pthread_exit((void *)0);
131 }
```


Tread

```
20 pthread_t p_thread[2];
21 struct Thr_info * th_info;
22 th_info = (Thr_info *)malloc(sizeof(Thr_info));
23
24 memcpy(th_info->Dev, Dev, sizeof(Dev));
25
26 while(true) {
27
28     printf("Choice : ");
29     scanf("%d", &Num);
30
31     sleep(1);
32
33     if (Num == 1) {
34         th_info->th_ex=1;
35         res = pthread_create(&p_thread[0], NULL, scan, (void *)th_info);
36         if (res < 0) {
37             perror("thread create error : ");
38             exit (0);
39         }
40     }
41     if (Num == 2) {
42         printf("\nStop Scanning...\n\n");
43         th_info->th_ex=0;
44         pthread_join(p_thread[0], (void **)&status);
45     }
```

현재 스레드의 식별자 정보를 담고 있음

scan 함수

```
36 while(my_th_info->th_ex) {
37     struct pcap_pkthdr* header;
38     const u_char* packet;
```

```
129 pcap_close(handle);
130 pthread_exit((void *)0);
131 }
```

완성!

사용자가 입력한 번호

```
sample: wlan0
Input : wlan0
Interface : wlan0

**** What Choice? ****
* 1.Scanning          *
* 2.Stop Scanning     *
*****

Choice : 1
Choice :
no  BSSID                Ch  SSID
-----
1   00:27:1c:e2:fc:7e    2
2   70:5d:cc:6e:bc:ae    2   kiss
3   00:27:1c:e2:fc:7f    2   U+NetFC81
4   88:3c:1c:b3:6d:56    3   KT_GiGA_2G_Wave2_6D52
2
5   88:36:6c:c0:c9:d6    2   oym

Stop Scanning....
Choice : 
```

감사합니다

2020.12.01 김현진