Lab5 -01-switching实验报告

姓名	学号	邮箱	完成时间
陈嘉昀	211220137	jiayunchen@smail.nju.edu.cn	2023年11月28日

1 代码实现

1.1 handle_packet

- 查看eth包头，查找有无包含目的mac的表项，若有，则转发；若无，则广播。
- 插入源mac的表项

```c
    struct ether_header *eh = (struct ether_header *)packet;

    iface_info_t* dst_port = lookup_port(eh->ether_dhost);
    if(dst_port) {
        iface_send_packet(dst_port, packet, len);
    }
    else {
        broadcast_packet(iface, packet, len);
    }

    insert_mac_port(eh->ether_shost, iface);
```

1.2.0

在编写mac.c中的代码时，需要注意并发bug，此处有一个编码小技巧。在此后的代码中只要是atmoic括号内的代码，都是上锁的。

```c
#define atomic \
for(int __i = (pthread_mutex_lock(&mac_port_map.lock), 0); __i < 1; __i += 1, pthread_mutex_unlock(&mac_port_map.lock))

// usage:
atmoic {
    do_something();
}
```

1.2.1 helper_func: lookup_entry

查找具有某个mac地址的条目，若无，则返回NULL

```c
static mac_port_entry_t* lookup_entry(u8 mac[ETH_ALEN]) {
    mac_port_entry_t* ret = NULL;
    u8 index = hash8((char*)mac, ETH_ALEN);
    atomic {
        mac_port_entry_t *entry, *q;
        list_for_each_entry_safe(entry, q, &mac_port_map.hash_table[index], list) {
            if(strncmp((char*)mac, (char*)entry->mac, ETH_ALEN) == 0) {
                ret = entry;
            }
        }
    }
    return ret;
}
```

1.2.2 lookup_port

先查找条目，若查找成功，则更新访问时间并返回port，否则返回NULL

```c
// lookup the mac address in mac_port table
iface_info_t *lookup_port(u8 mac[ETH_ALEN])
{
    mac_port_entry_t* ret_entry = lookup_entry(mac);
    iface_info_t* ret = NULL;
    if(ret_entry) {
        ret = ret_entry->iface;
        ret_entry->visited = time(NULL);
    }
    return ret;
}
```

1.2.3 insert

先查找有没有条目，若有则更新，若无则插入新节点。

```c
// insert the mac -> iface mapping into mac_port table
void insert_mac_port(u8 mac[ETH_ALEN], iface_info_t *iface)
{
    mac_port_entry_t* old = lookup_entry(mac);
    if(old == NULL) {
        // add a new entry that doesn't exist
        mac_port_entry_t* entry = malloc(sizeof(mac_port_entry_t));
        entry->iface = iface;
        entry->visited = time(NULL);
        strncpy((char*)entry->mac, (char*)mac, ETH_ALEN);

        u8 index = hash8((char*)mac, ETH_ALEN);
        atomic {
            list_add_head(&entry->list, &mac_port_map.hash_table[index]);
        }
    }
    else {
        // overwrite the old entry
        atomic {
            strncpy((char*)old->mac, (char*)mac, ETH_ALEN);
            old->visited = time(NULL);
        }
    }
}
```

1.2.5 sweep

删掉超时的条目，遍历即可

```c
// sweeping mac_port table, remove the entry which has not been visited in the
// last 30 seconds.
int sweep_aged_mac_port_entry()
{
    int ret = 0;
    time_t now_time = time(NULL);
    atomic {
        mac_port_entry_t *entry, *q;
        for (int i = 0; i < HASH_8BITS; i++) {
            list_for_each_entry_safe(entry, q, &mac_port_map.hash_table[i], list) {
                if(now_time - entry->visited > MAC_PORT_TIMEOUT) {
                    list_delete_entry(&entry->list);
                    free(entry);
                    ret += 1;
                }
            }
        }
    }
    return ret;
}
```

2 运行截图

b1节点在后台运行switch并输出日志：

```
(base) ubuntu@VM-4-8-ubuntu:~/2023-fall/CN/lab5-2023autumn-Chen-Jiayun/01-switching$ sudo ./three_nodes_bw.py
mininet> b1 ./switch > log-switch.txt 2>&1 &
mininet>
```

h1成功ping通h2：

```
(base) ubuntu@VM-4-8-ubuntu:~/2023-fall/CN/lab5-2023autumn-Chen-Jiayun/01-switching$ sudo ./three_nodes_bw.py
mininet> b1 ./switch > log-switch.txt 2>&1 &
mininet> h1 ping -c1 h2
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=0.184 ms

--- 10.0.0.2 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.184/0.184/0.184/0.000 ms
mininet>
```

查看b1的log，它正确删除了两个条目（h1和h2）

```
(base) ubuntu@VM-4-8-ubuntu:~/2023-fall/CN/lab5-2023autumn-Chen-Jiayun/01-switching$ bat log-switch.txt
       File: log-switch.txt
   1   DEBUG: find the following interfaces:  b1-eth0 b1-eth1 b1-eth2.
   2   DEBUG: 2 aged entries in mac_port table are removed.
(base) ubuntu@VM-4-8-ubuntu:~/2023-fall/CN/lab5-2023autumn-Chen-Jiayun/01-switching$
```