**密码学课程第 3次实验报告**

**实验名称：分析校园无线身份认证密码协议**

学号： 2011428 姓名： 王天行 班级： 密码科学与技术

### 实验目的

使用抓包软件捕获登录南开大学无线WIFI——NKU\_WLAN的过程，根据抓包结果对所涉及的身份认证密码协议及其安全性进行分析。

抓包工具：Wireshark&Microsoft Network Monitor

实验环境：Windows

### 实验内容说明

使用抓包软件捕获登录南开大学无线WIFI——NKU\_WLAN的四次握手数据包，并进行分析

### 实验原理

本节以捕获到密钥信息报文EAPoL-Key为例，即分析WiFi身份认证中四次握手的过程。

**1) 名词解释**

Authenticator：认证者，指AP，即无线接入点

Supplicant：请求者，指Station，即任何企图接入AP服务集的设备

ANonce：由AP生成的随机数

SNonce：由Station生成的随机数

Mac(AA)：AP的Mac 地址

Mac(SA)：Station的Mac地址

PRF：Pseudo-Random Function，表示伪随机函数

MSK：Master Session Key，主会话密钥

PMK：Pairwise Master Key，成对主密钥，由MSK生成，用于生成PTK

GMK：Group Master Key，组主密钥，同样由MSK生成，用于生成 GTK

PTK：Pairwise Transit Key，成对临时密钥，用来加密 AP 和 Station 通讯的单播数据包，AP与每个Station 通讯用的PTK都是唯一的。PTK的生成函数如下：

GTK：Group Temporal Key，组临时密钥，用来加密 AP 和 Station 通讯的多播/广播数据包，连接该 AP 的所有 Station 共享一个 GTK。GTK的生成函数如下：

MIC：Message Integrity Check，消息完整性校验码，针对一组需要保护的数据计算出的散列值，用来防止数据遭篡改

**2) 协议描述**

四次握手是 AP（Authenticator）和Station（Supplicant）为了生成一个用于加密无线数据的密钥而进行四次消息交换的过程。协议流程可以参考图7。

（1）第一次握手：AP—>Station：ANonce

AP产生随机数Anonce，用EAPOL-KEY帧发送给Station，Station在收到消息后用它生成 PTK，前面已经提到PTK的生成函数：。其中，PTK的前128位是KCK（EAPOL-Key Confirmation Key），用来校验EAPOL-Key帧的完整性。

（2）第二次握手：Station—>AP：SNonce，MIC

Station 创建了自己的PTK后，会立即响应一条包含SNonce和MIC的 EAPoL 消息给 AP，AP将进行同样的计算得到PTK， 然后用得到的前128位对EAPoL

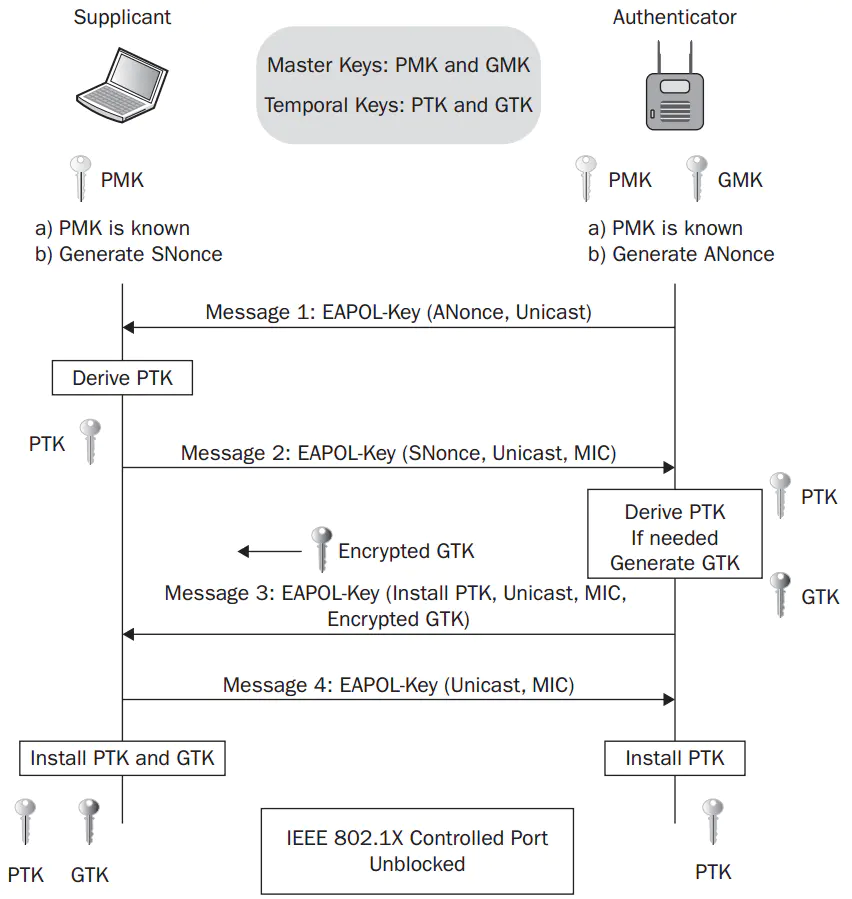


图7 四次握手流程图

报文进行完整性校验：比较所得到的值是否和收到报文中的WPA Key MIC的值一致，如果一致，则验证成功，说明Client端（即Station）拥有正确的PMK，否则判定Client端拥有的PMK错误，整个握手就此停止。

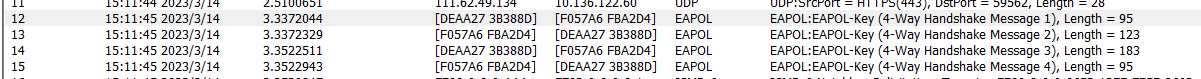
（3）第三次握手：AP—>Station：MIC，Encryted GTK

验证成功后，AP将 MIC和用PTK加密后的GTK 发送给 Station，并且告知 Station 安装 PTK 和 GTK。Station收到消息后检查MIC，同第二次握手的检查流程一样，如果不一致，将默默丢弃第三个报文，握手就此停止。验证成功后，Station可以使用掌握的PTK进行解密，恢复GTK的值，并安装PTK和GTK。

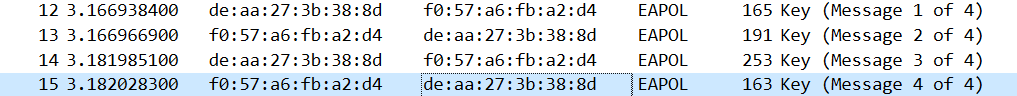
（4）第四次握手：Station—>AP：MIC

Station向AP发送EAPOL-Key消息，确认密钥已经安装，AP 收到该消息后再次检查MIC，验证成功后也安装PTK。安装的意思是指使用PTK 和GTK来对数据进行加密。

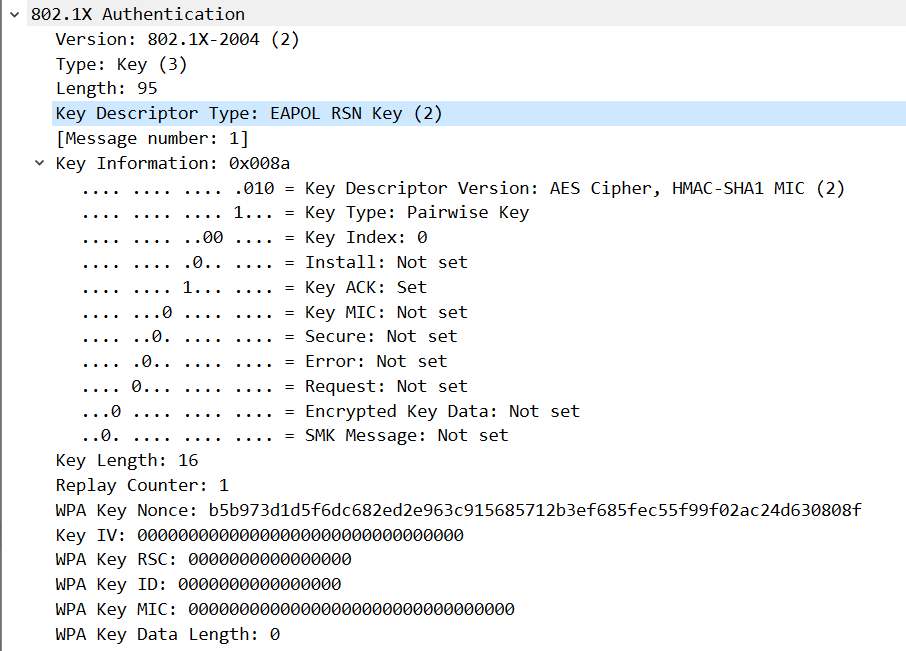
### 实验步骤



用MNM抓取到连接手机热点WIFI的四次握手数据包如上图。在wireshark中打开：



1. 第一次握手AP—>Station:ANonce



Key Descriptor

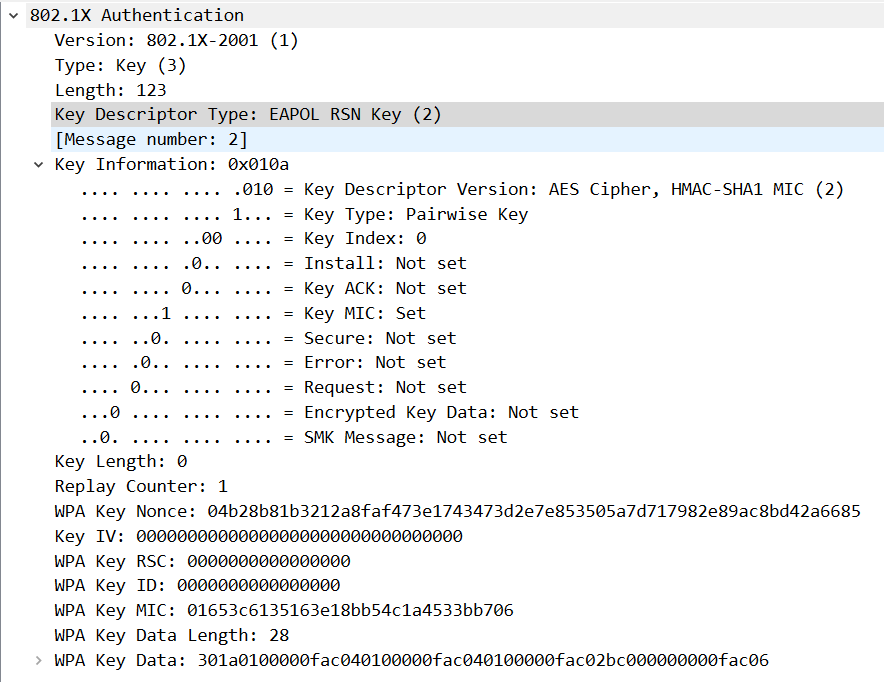
Key Descriptor部分用来描述EAPOL-Key帧的Key信息

Key Info：Secure位为0表示该帧没有加密的数据；Key ACK为1表示AP要求STA回复此帧；Install为0表示现在还无法安装PTK。Key Type为1表示当前时Pairwise密钥派生；Key Mic为0，表示该帧不包含MIC数据。

Replay Counter：STA需要保存这个值（这里为1）以检测重放攻击

WPA Key Nonce存储的就是Nonce，这里由AP生成的Nonce叫做ANonce。

1. 第二次握手Station—>AP:SNonce，MIC



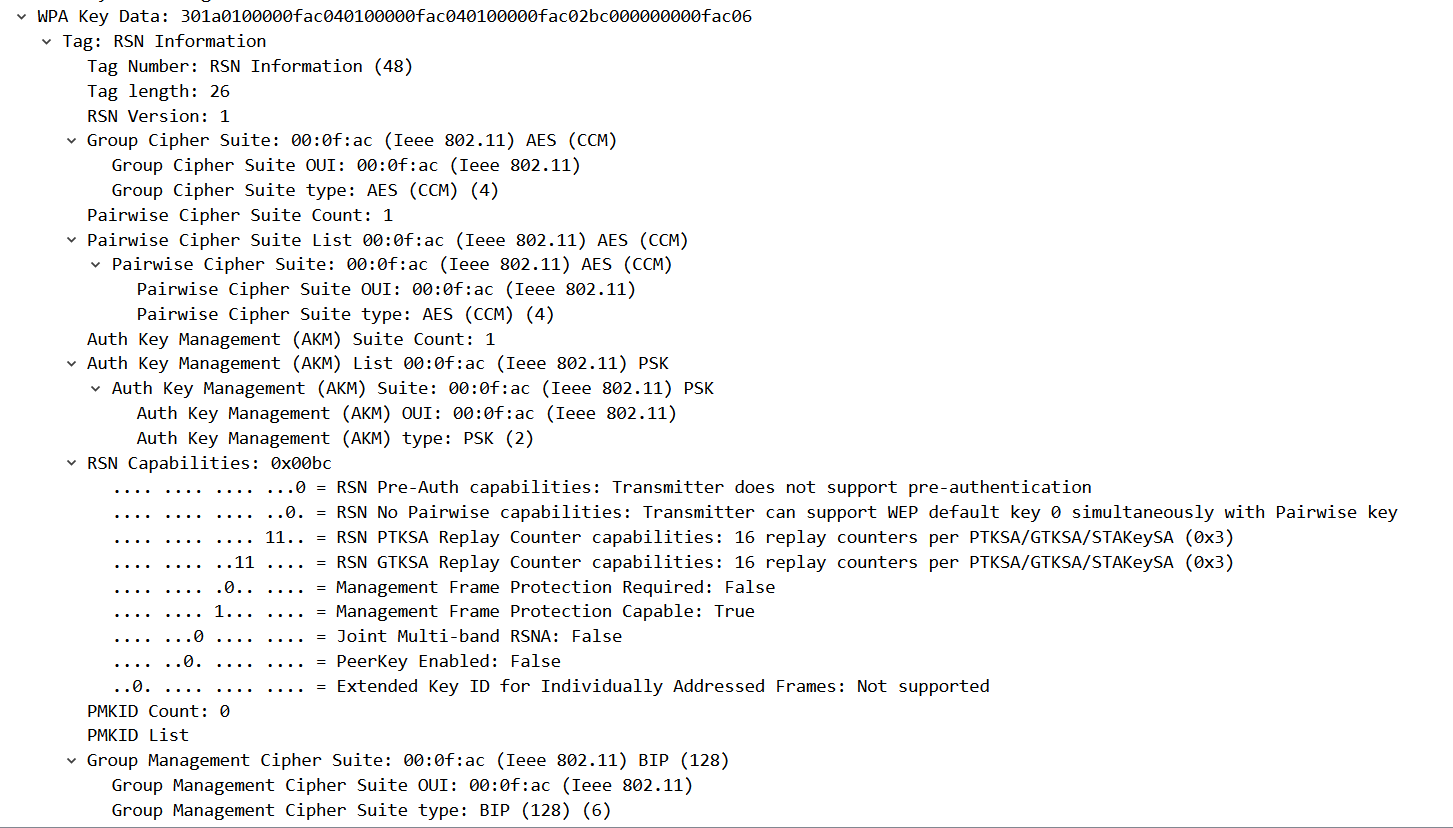
Key Info：Key Mic为1，表示该帧包含MIC数据。

Replay Counter：等于第一个帧中的Replay Counter。

WPA Key Mic是对整个EAPOL-Key帧进行计算而来，计算方法由Key Descriptor Version指定

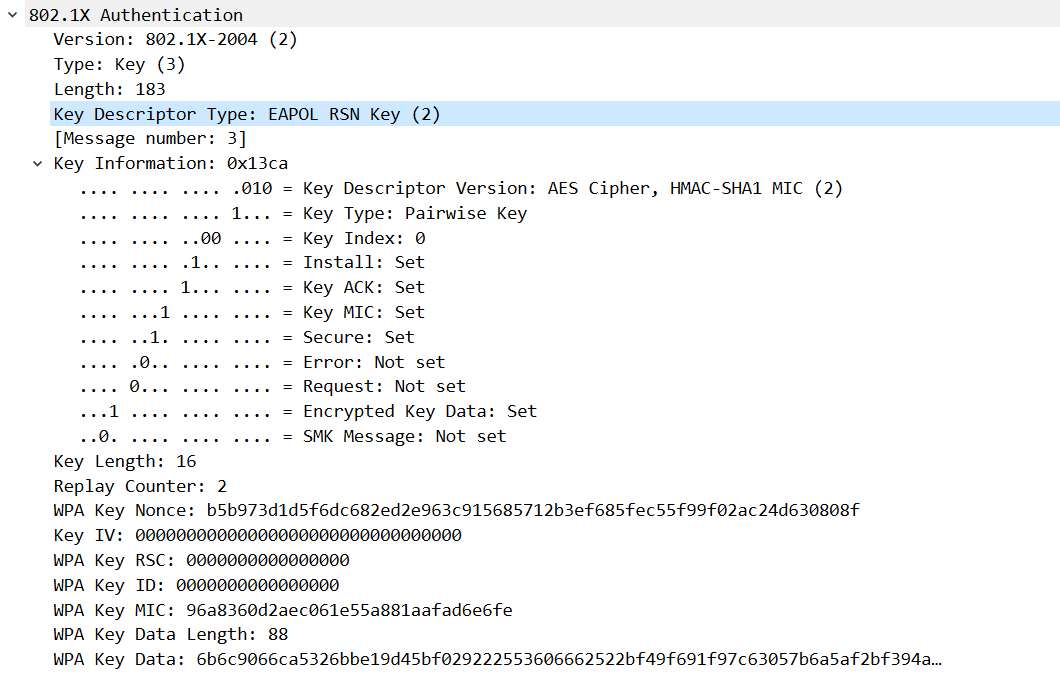
WPA Key Nonce存储Station生成的SNonce。

WPA Key Data



包含一个RSN Information Element。该RSN IE来自STA之前和AP在关联操作时获得的RSN IE。

1. 第三次握手AP—>Station:MIC，Encryted GTK



Key Info：Install为1表示STA收到该帧后可以安装PTK了；Secure位为1表示AP已经派生了PTK；Encrypted Key Data设为1，表示Key Data被加密了。

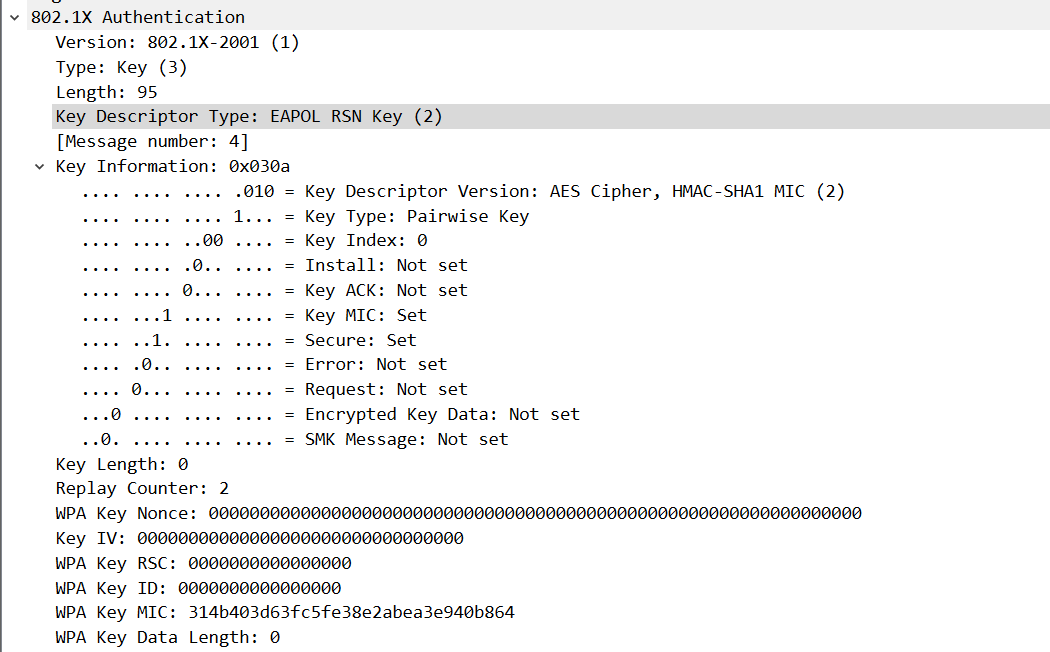
Replay Counter为2，比前面的值增加1。

Key Nonce的值和第一帧一样。

MIC对EAPOL-Key整个进行计算得来。

Key Data由PTK加密后而来，其解密后的内容包括GTK信息

1. 第四次握手Station—>AP:MIC



### 实验结果分析

**对于手机热点WIFI的安全性分析**

**1）数据安全性**

协议使用的成对临时密钥PTK包含3个部分，KCK（Key Confirmation Key），KEK（Key Encryption Key），TK（Temporal Key）。其中，KEK和KCK在四次握手中用于数据加密和完整性验证。协议还使用MIC来防止数据遭篡改，保证数据完整性。

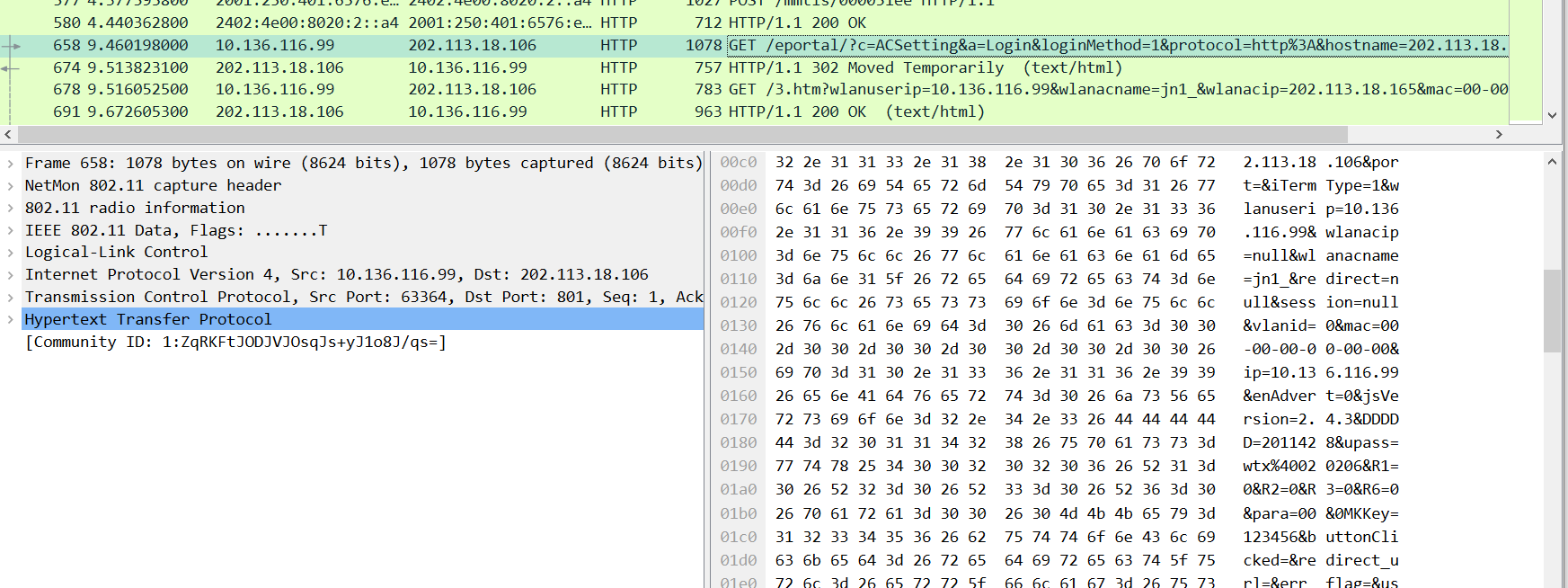
1. **拒绝服务攻击（DoS）**

消息1并没有使用任何加密或完整性校验措施，而且如果AP在规定时间内没有收到客户端发回的应答消息2，就会启动超时装置，重传刚才发送的消息1。因此，客户端会对收到的每一个消息1计算SNonce和PTK。攻击者可以根据这一点伪造消息1，使客户端重新计算PTK，由于前后的Nonce值不一样，因此很容易造成PTK生成的混乱。若攻击者重复这样的攻击过程，那么客户端永远无法与AP完成四次握手过程，更不可能通过AP访问其他合法资源，从而实施DoS攻击。

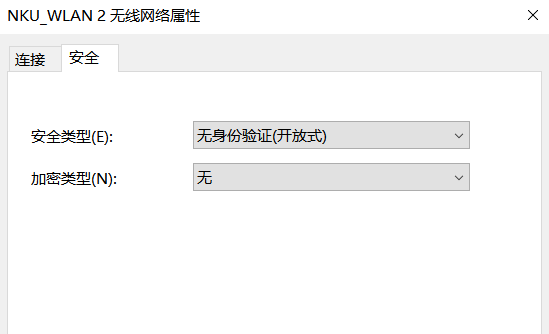
**对于连接校园网的安全性分析**

连接校园网捕获的数据包中，并没有EAPOL数据包和802.11协议相关数据包。

通过注销页（网页）进行登录，在捕获的数据包中可以获得学号与密码。



在电脑中查询网络相关信息，发现校园网的安全类型为无身份验证（开放式）



### 总结感想

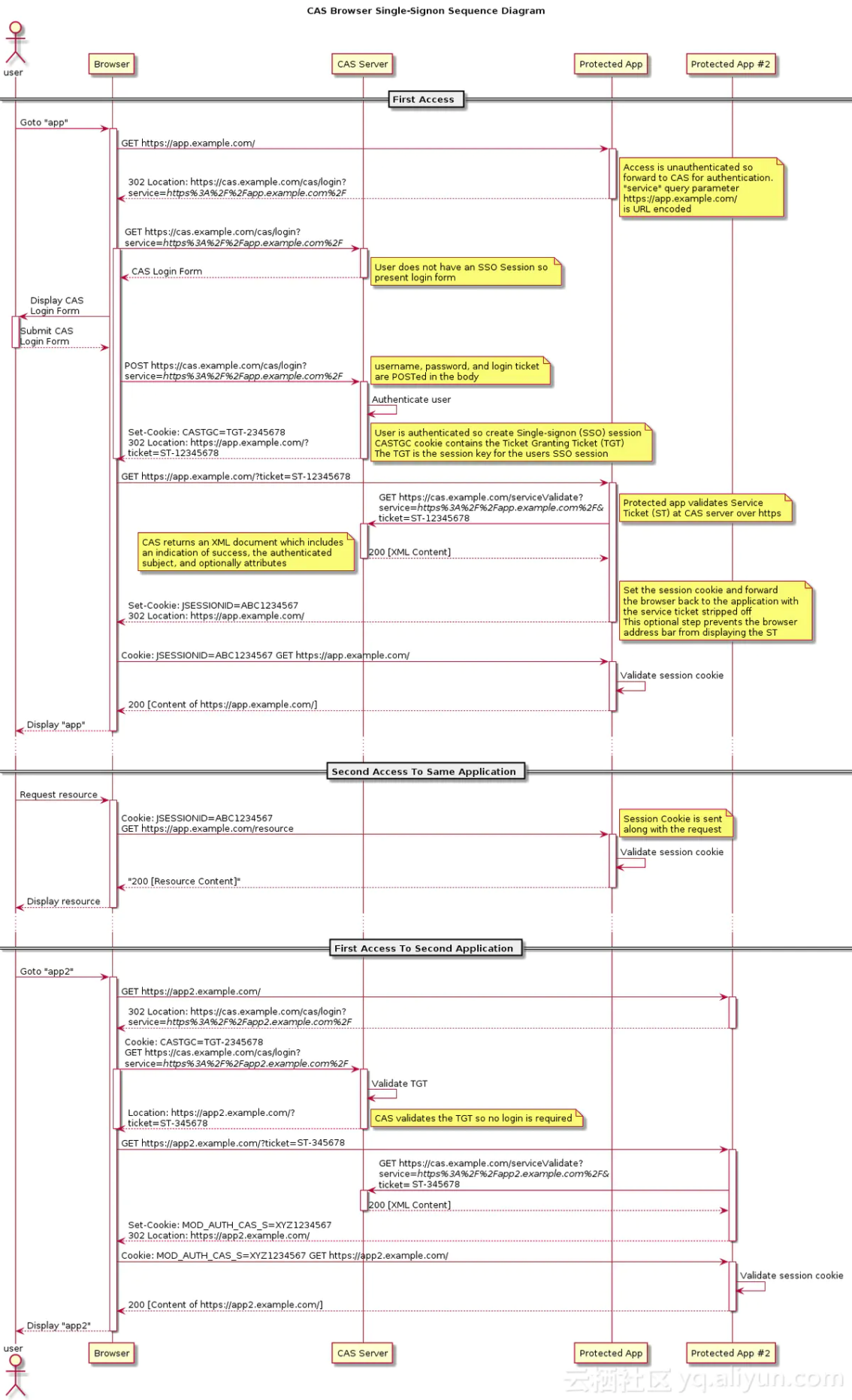
**Q.通过抓包对南开大学SSO（Single Sign On单点登录）协议进行分析。**

**A.**通过[https://sso.nankai.edu.cn/sso/login](https://sso.nankai.edu.cn/sso/login" \t "http://www.91yixue.com/daxue/jwgl/_blank)看到SSO的定义：

“Single Sign On”，简称SSO。主要是解决学校师生访问学校的应用系统时，都需要去输入一遍账号密码进行登录，不方便而且不容易记住（应用系统特别多，密码账号还不一样）。单点登录，只需记住一个登录账号，登录一次后，可进入其他系统，不需要再次登录。

下图是CAS官网上的标准流程，具体流程如下：

1. 用户访问app系统，app系统是需要登录的，但用户现在没有登录。
2. 跳转到CAS server，即SSO登录系统，以后图中的CAS Server我们统一叫做SSO系统。SSO系统也没有登录，弹出用户登录页。
3. 用户填写用户名、密码，SSO系统进行认证后，将登录状态写入SSO的session，浏览器（Browser）中写入SSO域下的Cookie。
4. SSO系统登录完成后会生成一个ST（Service Ticket），然后跳转到app系统，同时将ST作为参数传递给app系统。
5. app系统拿到ST后，从后台向SSO发送请求，验证ST是否有效。
6. 验证通过后，app系统将登录状态写入session并设置app域下的Cookie。



**Q.在安全协议设计中，常使用时间戳和随机数，分析这两种机制的作用和优缺点。**

**A.**时间戳（Timestamp）

优点：

1. 时间戳服务器与权威的国家授时中心对接，通过与数字签名的有效结合，可为互联网活动任何电子文件或网上交易所产生的电子数据提供保密性、完整性、防抵赖等功能，通过数字签名保证内容和签发人的不可抵赖性，通过时间戳提供准确的、权威的、不可篡改的时间证明和内容完整性证明。时间戳已成为网络数据安全不可或缺的一部分。
2. 防dos攻击（第三方使用正确的参数，不停请求服务器，使之无法正常提供服务）
3. 操作简单、迅速，存储容量小，验证简单

缺点：

1. 双方可以进行串通篡改或伪造时间戳
2. 需要双方调整时间一致，对于时钟的准确性要求高

随机数（Nonce）

优点：

1. Nonce是由服务器生成的一个随机数，在客户端第一次请求页面时将其发回客户端；客户端拿到这个Nonce，将其与用户密码串联在一起并进行非可逆加密（MD5、SHA1等等），然后将这个加密后的字符串和用户名、Nonce、加密算法名称一起发回服务器；服务器使用接收到的用户名到数据库搜索密码，然后跟客户端使用同样的算法对其进行加密，接着将其与客户端提交上来的加密字符串进行比较，如果两个字符串一致就表示用户身份有效。这样就解决了用户密码明文被窃取的问题，攻击者就算知道了算法名和nonce也无法解密出密码。每个nonce只能供一个用户使用一次，这样就可以防止攻击者使用重放攻击，因为该Http报文已经无效。可选的实现方式是把每一次请求的Nonce保存到数据库，客户端再一次提交请求时将请求头中得Nonce与数据库中得数据作比较，如果已存在该Nonce，则证明该请求有可能是恶意的。

缺点：

1. 有可能在两次正常的资源请求中，产生的随机数是一样的，这样就造成正常的请求也被当成了攻击，随着数据库中保存的随机数不断增多，这个问题就会变得很明显
2. 所有使用的随机数需要进行存储，存储量大