课程项目：重安装键攻击

1. 项目简介

1.1 项目背景

WPA2 是经由 Wi-Fi 联盟验证过的 IEEE 802.11i 标准的认证形式。它是目前应用最为广泛的保护无线网络安全的协议。

2017 年，比利时鲁汶大学的安全研究员发表了针对 WPA2 协议的重安装键攻击（Key Reinstallation Attacks）。该攻击通过重放 Client 和 Access Point (AP) 之间传递的信息，迫使 Client 多次使用原本应为一次性的随机数，进而破坏加密方案的安全性。通过该攻击，攻击者可以窃听或篡改 Client 和 AP 之间的所有通信流量，进而窃取信用卡、密码、聊天消息、电子邮件、照片等信息。

1.2 项目目标

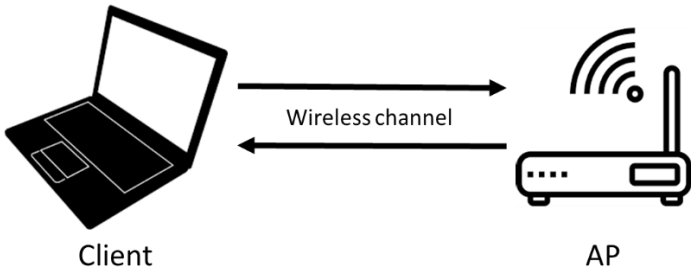
通过仿真的方式实现简化版的 WPA2 协议并重现重安装键攻击。

1.3 开发环境

C，C++ 或 Java。

2. WPA2 实现

2.1 系统模型



i. Client:

一个命令行程序，负责将用户提供的数据通过加密的方式传输给 AP。

Client 程序接收四个命令行参数：

1. 目标 AP 的 IP 地址。
2. 目标 AP 的端口。
3. 目标 AP 的密码 (MasterKey)。
4. 需要传递的数据（文本文档，与 Client 程序在同一目录下）。

例如，如果程序命名为 Client, 则启动命令如下

- ./Client 192.168.0.35 1234 z3dg35dg Packet.txt

ii. AP:

一个命令行程序，负责验证 Client 的身份以及数据的接收

AP 程序接收两个命令行参数：

1. 密码 (MasterKey)
2. 端口

例如，如果程序命名为 AP, 则启动命令如下

- ./AP z3dg35dg 1234

iii. Channel:

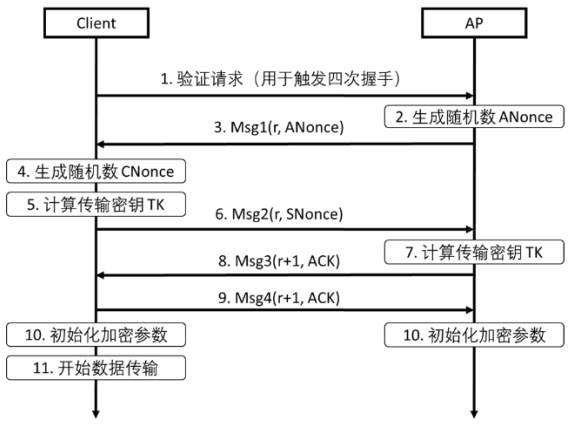
Client 与 AP 通过 TCP Socket 进行交互。

2.2 协议流程

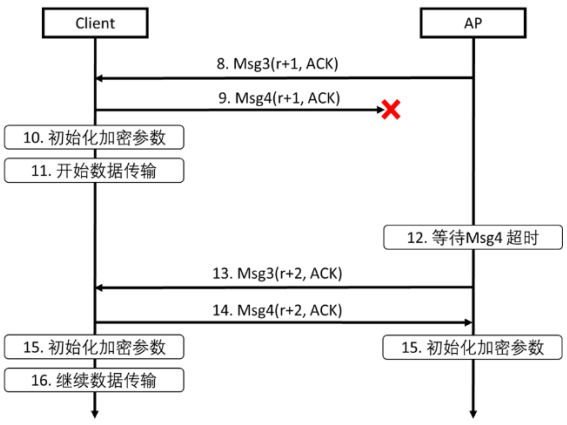
在 WPA2 中，Client 和 AP 通过四步握手流程生成一个共享的密钥 TK。该 TK 用于加密/解密之后传输的数据。

为了便于理解和实现，本项目仅要求实现简化版 WPA2 流程。此流程省去了部分细节（例如 GroupKey），且只考虑一种异常，即 Msg4 丢失。完整版 WPA2 流程可参考论文。简化版的流程如下：

正常流程



Msg4丢失



2.2.1 正常流程

在不考虑数据包丢失且无攻击者干扰的情况下，Client 与 AP 的交互流程如下

1. Client 向 AP 发出验证请求，这里验证请求可以设置为任意字符串，例如 Authentication_Request.
2. AP 收到请求后，生成一个随机数 ANonce。
3. AP 将 ANonce 和 r 发送给 Client，这里 r 为计数器，AP 每发送一次信息给 Client, r 增加 1（四步握手第一步）。
4. Client 收到 ANonce 后，生成另外一个随机数 CNonce。
5. Client 根据 ANonce, CNonce 以及 MasterKey（见 2.1）, 计算出传输密钥 TK。TK 将被用于加密用户想要上传的数据包。
6. Client 将 CNonce 和 r 发送给 AP, 这里的 r 即是 AP 在四步握手的第一步使用的 r（四步握手第二步）。
7. AP 收到 CNonce 后，和已有的 ANonce 和 MasterKey 一起，计算出传输密钥 TK。这里算出的 TK 应与 Client 算出的 TK 相同。
8. AP 发送 ACK 和 r+1 给 Client。r+1 表示这是 AP 发送的第二条信息。ACK 为一个字符串，用于告知 Client，已经收到 CNonce。（四步握手第三步）。
9. Client 收到 ACK 后，发送 ACK 和 r+1 给 AP。r+1 表示这是 Client 发送的第二条信息。ACK 为一个字符串，用于告知 AP，四步握手完成。（四步握手第四步）。
10. Client 和 AP 各自初始化加密需要用到的两个参数（细节见 2.3）。
    - EncryptionKey = TK
    - Nonce = 0
11. Client 开始使用设置好的参数加密并传输数据。每加密一次数据，Nonce 增加 1。

2.2.2 Msg 4 丢失

9. Client 发送 ACK 和 r+1 给 AP, 但是数据包丢失了。
10. 由于 AP 未收到信息，只有 Client 开始初始化加密参数。
11. Client 开始使用设置好的参数加密并传输数据。
12. AP 长时间未收到 Msg4，触发等待超时。这里的计时从第 8 步发出 Msg 3 后开始，经过一定时间未收到 Msg 4 则进入步骤 13。实现时可自行选择合理的 AP 等待时间。
13. AP 重发 Msg 3。在重发的 Msg 3 中，计数器变为 r+2，表示这是重发的 Msg 3。
14. Client 收到重发的 Msg 3 后， 重发 Msg 4。这里 Msg4 的计数器也变为 r+2.（四步握手第四步）。
15. Client 重新进行参数初始化，EncryptionKey 不变，Nonce 归 0。如果 AP 收到了 Msg 4，则也进行参数初始化。如果 AP 仍未收到 Msg 4，等待超时后，重新发送 Msg 3, 计数器变为 r+3。
16. Client 继续数据传输（并非重新传输）。

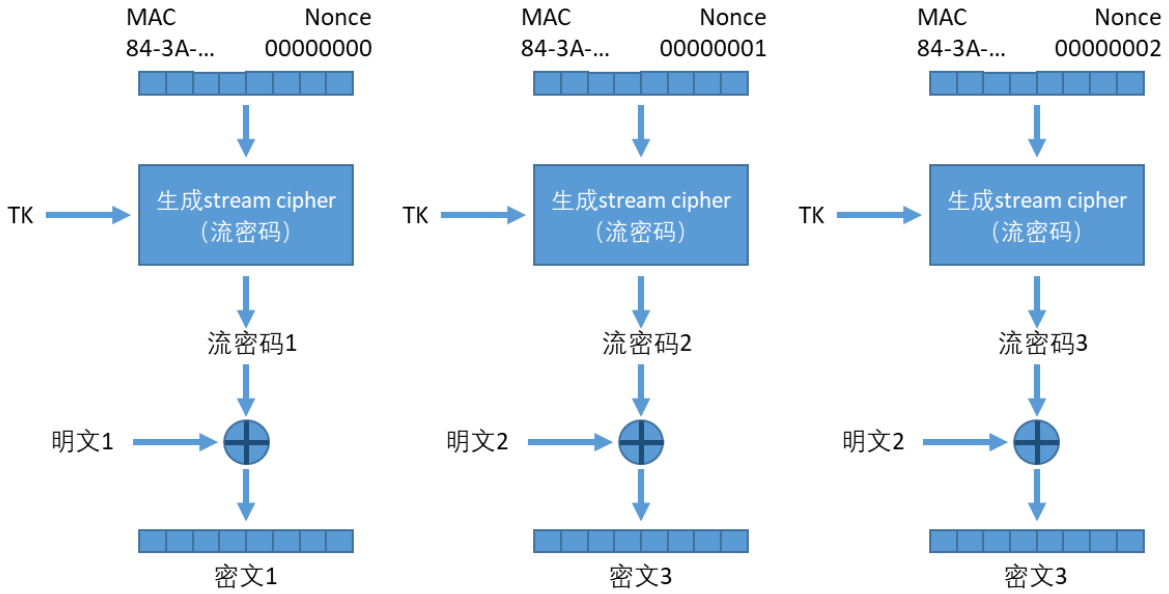
2.3 加密算法

2.3.1 生成 TK

TK 是一个长度为 128bit 的密钥，用于加密用户想要传输的数据。它由 ANonce, CNonce 和 MasterKey 产生。在本项目中，简便起见，可直接将 ANonce, CNonce 和 MasterKey 字符串串联后，计算所得字符串的哈希值。将所得哈希值作为 TK 使用。

2.3.2 使用 TK 进行数据加密

WPA2 中应用最广泛的加密算法是 CCMP。它采用的是一种计数器模式下的 AES。在本次项目中，我们采用简化版实现，忽略 AES 中的细节：



加密流程如下：

1. 获取本机 MAC 地址，如 84-3A-72-3D-3A
2. 将 MAC 和当前 Nonce 串联，作为 IV 使用。
3. 通过 IV 和 TK 生成本次加密的流密码（128bit）。简便起见，可直接将 IV 和 TK 串联，计算所得字符串的哈希值。
4. 将所得哈希值作为流密码与本次加密的明文异或，生成密文。这里明文的长度也为 128bit。当需要发送超过 128bit 的数据时，应将数据分成多个 128bit 的块（不足 128bit 的块在末尾补 0），分块加密。由于 Nonce 每次使用过后都会增加 1，因此加密每个数据块的流密码都不同。

2.4 实现要求

实现以上协议并按照以下要求将输出打印至命令行窗口：

- 在四步握手阶段，Client 程序和 AP 程序应将每一步发送和接收的数据包打印出来。
- 在数据传输阶段，Client 程序将发送出去的密文和对应的明文打印到命令行窗口。AP 程序将收到的密文打印出来。如果此时四步握手已经成功，则 AP 还应将解密后的明文打印出来。

Note: 在数据传输阶段，应限制 Client 发送数据的速度，例如在每发送一条数据后 Sleep 1 秒。这样可以帮助控制 Client 在 AP 重发 Msg 3 前发送的加密数据的条数（方便后续 Adversary 程序的设计）。

3. 重安装键攻击

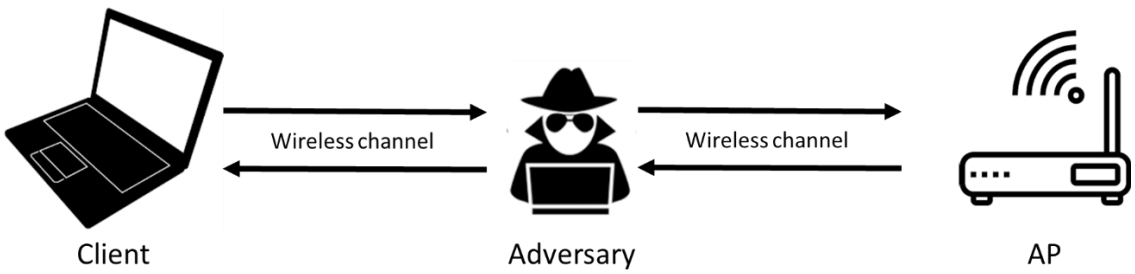
3.1 攻击原理

重安装键攻击主要针对的是四步握手协议的第四步。如上所述，当 Msg 4 丢失时，AP 会重发 Msg 3 给 Client，导致 Client 将 Nonce 的值重置为 0。如果 Client 在收到重发的 Msg 3 之前已经发送了一些加密的数据，那么接下来的数据传输就会重用部分 Nonce。

举例来说，Client 在发出 Msg 4 后，又使用 TK 加密并发送了三条信息 M1, M2, 和 M3。根据上述加密算法，它们对应的流密码分别为 Hash(TK||0), Hash(TK||1)和 Hash(TK||2)，密文为$M1 \oplus Hash(TK||0)$, $M2 \oplus Hash(TK||1)$, 以及 $M3 \oplus Hash(TK||2)$（这里 A||B 表示 A 与 B 进行字符串串联）。此时，AP 由于未收到 Msg4，触发超时并重发 Msg 3。Client 收到重发的 Msg 3 后，将 Nonce 值重置为 0。接下来，Client 又使用 TK 加密并发送了三条信息 M4, M5 和 M6。由于 Nonce 被重置，这三条信息会重用流密码 Hash(TK||0), Hash(TK||1)和 Hash(TK||2)。对应的密文为$M4 \oplus Hash(TK||0)$, $M5 \oplus Hash(TK||1)$, 以及 $M6 \oplus Hash(TK||2)$。

在 WPA2 中，由于明文是通过与流密码异或进行加密的，Nonce 的重用将导致严重的后果。例如，攻击者获取到了 $M1 \oplus Hash(TK||0)$和$M4 \oplus Hash(TK||0)$。通过将两者进行简单地异或，则可以得到$M1 \oplus M4$。而 M1 和 M4 这类明文数据通常会符合特定的语言规律。在已知$M1 \oplus M4$的情况下，攻击者可以通过字典攻击，找出 M1 和 M4, 进而计算出流密码$Hash(TK||0)$。

3.2 攻击模型



Adversary：

一个命令行程序，作为中间人转发 Client 与 AP 之间传递的数据包。

Client 程序接收三个命令行参数：

1. 目标 AP 的 IP 地址。
2. 目标 AP 的端口（用于建立与 AP 之间的交互）
3. 本程序监听的端口（用于接收用户数据包）

例如，如果程序命名为 Adversary, 则启动命令如下

- ./Adversary 192.168.0.35 1234 4567

Client:

采用之前实现的 Client 程序。在实际应用中，攻击者需要欺骗 Client 将自己当成 AP，从而实现中间人攻击。在本项目中的攻击模型中，简化起见，我们假设 Client 在启动时输入的就是 Adversary 的 IP 和端口。即用 Adversary 的 IP 地址和端口替代 AP 的 IP 地址和端口。

AP:

采用之前实现的 AP 程序，使用正常的参数启动。

3.3 实现要求：

根据上述攻击原理，设计并实现 Adversary 程序。所设计的 Adversary 程序应当能够：

1. 转发 Client 与 AP 之间的信息。（10 分）
2. 获取到多组由相同 Nonce 加密的数据（例如 3.1 例子中的 $M1 \oplus \text{Hash}(TK||0)$和 $M4 \oplus \text{Hash}(TK||0)$）。（15 分）
3. 根据获取到的密文，分析出对应的明文和流密码。如根据 3.1 例子中的 $M1 \oplus \text{Hash}(TK||0)$和$M4 \oplus \text{Hash}(TK||0)$）分析出 M1, M4 以及$\text{Hash}(TK||0)$。这里假设 Adversary 已知 Client 发送的所有数据均由五个特定词汇排列组合而成：POST, GET, HTTP, INPUT, OUTPUT。（15 分）

4. 提交内容

1. Client 程序源代码
2. AP 程序源代码
3. Adversary 程序源代码
4. 电子版 Report，包含以下内容

a. 程序运行说明
b. WPA2 协议正常运行的截图。
c. 重安装键攻击具体设计
d. 重安装攻击运行截图及效果分析

5. 评分标准

所有提交的代码均会进行查重，抄袭 0 分

1. WPA2 协议实现： 40 分
2. 重安装键攻击：40 分
3. Report：20 分

6. 参考文献

[1] Vanhoef, M., & Piessens, F. (2017, October). Key reinstallation attacks: Forcing nonce reuse in WPA2. In Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security (pp. 1313-1328). ACM.