移动互联网安全

第二章 无线接入网监听

黄 玮

内容提纲

- 实战无线网监听
- 无线通信数据报文分析

实战无线网监听

无线网络监听的基本条件

- 无线网卡
- 无线网卡驱动
  - 无线网卡设置工具
    - 例如：iwconfig
- 抓包器
  - 例如：wireshark、tshark、airodump-ng等

无线网络监听的进阶条件

- 操作系统支持设置无线网卡进入monitor模式
  - 无需加入任何一个BSS
    - 无需绑定到一个AP或进入Ad-Hoc模式
  - 无线网卡通过channel hopping技术在多个channel之间快速切换
  - 捕获802.11数据帧

无线网络监听的限制因素

- 无线网卡只能工作在一个确定的频道上
  - 不能同时监听所有频道和波段
- 无线网卡对802.11协议的支持有硬件差异
  - 指定范围：a/b/g/n/ac，其中b/g最为常见

无线网络安全实验的操作系统选择

- Windows
  - 驱动支持少
  - 相关工具软件少
- Linux
  - 驱动支持全面
  - 相关工具软件多

BackTrack VS. Kali Linux

- 面向网络安全审计定制化的Linux操作系统
- Kali Linux是BackTrack的下一代系统代号
- 官方已经放弃对BackTrack的继续维护
- 目前推荐使用的是Kali Linux

无线嗅探 ON LINUX

无线网卡设置查看基本工具（1/2）

- ifconfig
  - 通用网卡设置工具（IP、MAC相关设置）
- iwconfig / iw
  - 网卡特性、工作模式查看与设置
    - iw phy
- lspci / lsusb / dmesg
  - 查看硬件接口信息（USB、外设是否连接正常等）

无线网卡设置查看基本工具（2/2）

- iwlist
  - 控制无线网卡获取详细无线网络信息
    - 主动获取可用AP信息列表
      - iwlist wlan0 scanning
    - 查看无线网卡的可用工作频率 传输速率 加密标准等

常见故障排查

- tail -f /var/log/messages
  - 设备运行时变化会产生消息并被记录到上述日志文件
- 抓不到包
  - 更换USB口
  - 虚拟机的USB设备共享设置
    - 启用USB 3.0兼容
  - 重启虚拟机
    - 关闭虚拟机再重新打开

aircrack-ng

- 无线网络安全审计的瑞士军刀
  - 802.11 WEP / WPA-PSK
  - 包含一系列小工具
  - 选择密文攻击
  - 支持多种典型攻击算法
    - FMS / KoreK / PTW

aircrack-ng包含的主要工具

- dpkg -L aircrack-ng
  - airdriver-ng
  - tkiptun-ng
  - easside-ng
  - besside-ng
  - aircrack-ng
  - airdecloak-ng
  - aireplay-ng
  - airodump-ng
  - airmon-ng
  - airtun-ng
  - airbase-ng
  - airmon-zc
  - 。 。 。

设置网卡进入monitor模式

- usage
  - airmon-ng <start|stop|check> <interface> [channel or frequency]
  - iw dev wlan0 interface add mon0 type monitor && ifconfig mon0 up
    - iw dev mon0 del
  - iwconfig wlan0 mode monitor

无线数据帧监听（抓包）

- usage
  - airodump-ng <options> <interface>[,<interface>,…]
  - tshark -i mon0 -w mon0.pcap
  - tshark -I -i wlan0 -w wlan0.pcap

无线通信数据报文分析

内容提纲

- 802.11协议提供的网络服务类型
- 802.11协议中的重要理论基础
- 802.11帧结构与Wireshark过滤器语法
- 802.11加密与认证机制原理

802.11网络服务

- 数据封包传送
- 身份验证(authentication)
- 解除验证(de-authentication)
- 隐私(privacy)保护
- 连接(association)服务
- 重连(re-association)服务
- 取消连接（dis-association）服务
- 分发(distribution)服务
- 整合(integration)服务

STA

DS

数据封包传送服务

- 此服务为最基本的功能
- IEEE802.11使用自身协议将数据进行封装和传送

身份验证服务

- 主要用来确认每个主机（STA）的身份
- 802.11通常要求双向式的身份确认，它也允许同一时间一个主机和多个主机（包括AP）进行身份验证

解除验证

- 已完成身份认证的STA可以用这个服务来取消身份认证，一旦取消后连接也同时被取消

隐私保护

- 通过加密机制保护通信数据的机密性

连接服务

- 目的：在STA和AP之间建立一个通信链路
- 当分布式系统要将数据传送给主机时，必须事先知道这个主机目前是通过哪个AP接入分布式系统的，这些信息都可以由连接服务提供
- 一个主机在被允许经由某个AP传送数据给分布式系统前，必须先和此AP进行连接
- 通常在一个BS内有一个AP，因此在这个区域内的任意主机若想要与外界进行通信，就必须先与此AP进行连接。这个过程类似注册，当主机完成连接后，AP就会记住这台主机目前在它的管辖范围之内。连接服务通常都由主机启动，用它来与AP进行连接。（注意，在任何时刻一台主机只会和一个AP进行连接，这样才能使分布式系统知道哪个主机是由哪个AP所管辖的，然而一个AP却可以同时与多台主机进行连接。）

重连服务

- 目的：将一个移动中的主机连接由一个AP转移至另一个AP
- 当主机从一个服务区移动到另一个服务区时，它将启动重连服务
- 重连服务会将主机与它所移入的服务区内的AP进行连接，使分布式系统知道此主机已经转移至另一个AP的管辖区域内
- 重连服务通常也是由主机启动

取消连接服务

- 当一台主机数据传送结束时，可以使用取消连接服务对当前已有的连接进行取消
- 当主机在服务区内移动时，它除了会对新的AP启动重连服务外，还会对旧的AP启动取消连接服务
- 此服务可以由主机或AP任一方来启动，不论是哪一方启动的另一方都不能拒绝。（需要注意的是AP可能因网络负荷过重而是用取消连接服务对主机取消连接）

分发服务

- 此服务主要由BSS中的主机使用
- 当主机需要传送数据时，会先将数据传送至AP，再由AP通过分布式系统传送至目的地
- IEEE802.11并没有规定分布式系统要如何将数据正确的送至目的地，但它说明了在连接、取消连接和重连等服务中，数据应该由哪个AP进行输出以将数据送达至正确的目标地点

整合服务

- 目的：让数据能够在分布式系统和现有的局域网之间进行传送
- 整合服务的任务就是将数据从分布式系统转送到相连的局域网络媒介，其主要工作就是将不同的地址空间做一个转换

CSMA/CD

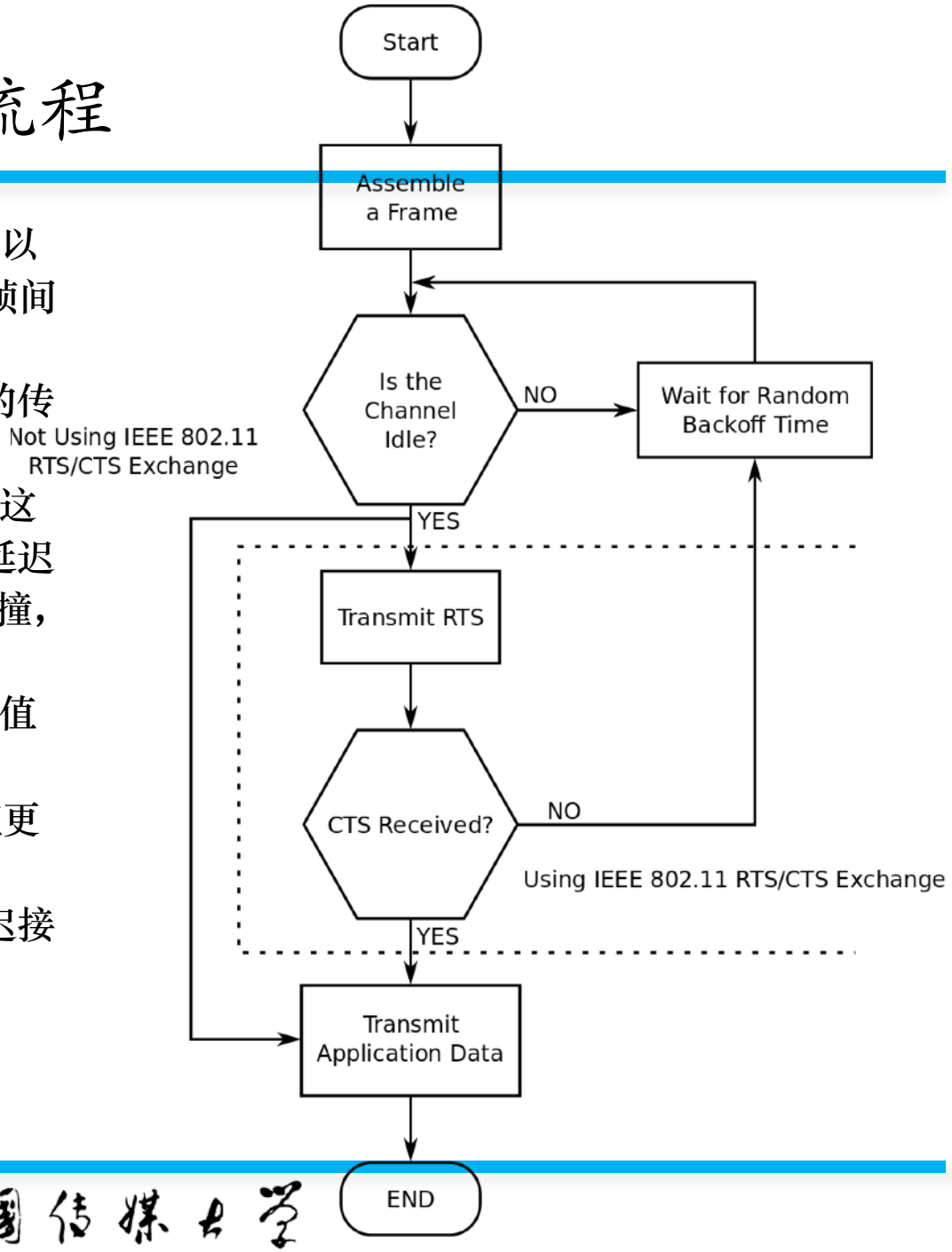
- 802.3
- 基于碰撞检测的载波监听多路访问
  - Carrier Sense Multiple Access With Collision Detection

CSMA/CA

- 802.11
- 基于碰撞规避的载波监听多路访问
  - Carrier Sense Multiple Access with Collision Avoidance
- 无线局域网数据链路层最基本的接入方法
- 分布协调功能（DCF）的基础
  - Distributed Coordination Function

CSMA/CA简易流程

- STA发送信息之前，检测信道是否空闲以及空闲的时间是大于IEEE802.11规定的帧间隔时间
- 如果否，该STA就延迟接入，直到当前的传输结束
- 之后，也就是一次成功的传输刚结束，这时碰撞发生率最高，因为所有的STA都延迟等待这一时刻的到来，为进一步减少碰撞，STA选择随机退避再次延迟接入
- 在检测信道的同时倒数计数器，直到其值为0
- 这时，如果其它的Backoff计时器的数值更短，它就赢得了信道的占用权
- 其他的STA检测到信道忙，只有再次延迟接入
- 否则，只有在信道空闲时再发送信息



四次握手协议

- 解决数据链路层的传输过程中丢帧问题
  - 检测并重发
    - 进一步，避免重发时的碰撞，解决隐藏节点问题，引入了RTS/CTS + ACK协议
- RTS：发送请求控制
  - Request to send
- CTS：清除发送控制
  - Clear to send

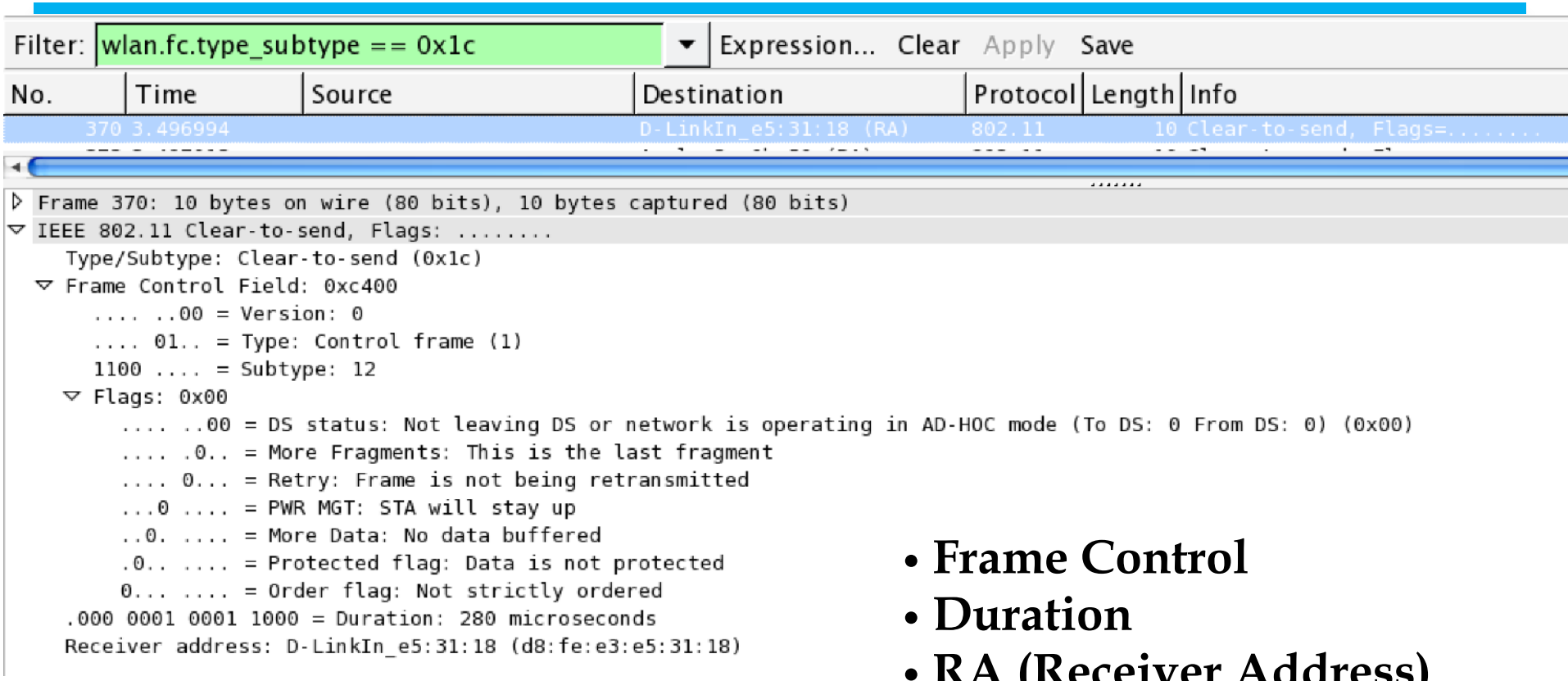
http://en.wikipedia.org/wiki/Hidden_node_problem

RTS


```
Frame 372: 16 bytes on wire (128 bits), 16 bytes captured (128 bits)
IEEE 802.11 Request-to-send, Flags: ........
    Type/Subtype: Request-to-send (0x1b)
  Frame Control Field: 0xb400
      .... ..00 = Version: 0
      .... 01.. = Type: Control frame (1)
      1011 .... = Subtype: 11
    Flags: 0x00
        .... ..00 = DS status: Not leaving DS or network is operating in AD-HOC mode (To DS: 0 From DS: 0) (0x00)
        .... .0.. = More Fragments: This is the last fragment
        .... 0... = Retry: Frame is not being retransmitted
        ...0 .... = PWR MGT: STA will stay up
        ..0. .... = More Data: No data buffered
        .0.. .... = Protected flag: Data is not protected
        0... .... = Order flag: Not strictly ordered
    .000 0001 0111 1010 = Duration: 378 microseconds
    Receiver address: D-LinkIn_e5:31:18 (d8:fe:e3:e5:31:18)
    Transmitter address: Apple_2a:0b:50 (b8:e8:56:2a:0b:50)
```

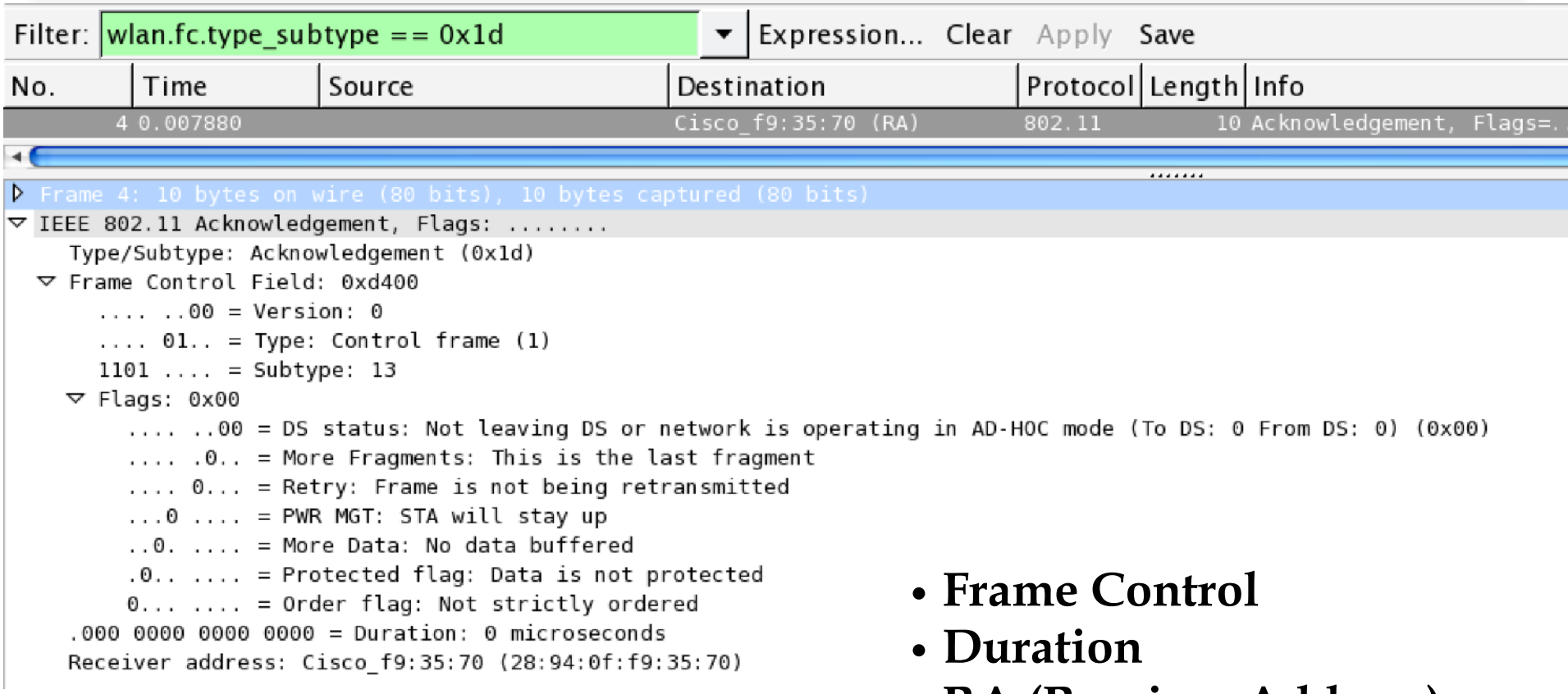
- **Frame Control**
- **Duration**
- **RA (Receiver Address)**
- **TA (Transmitter Address)**
- **FCS**

CTS



- **Frame Control**
- **Duration**
- **RA (Receiver Address)**
- **FCS**

ACK



- **Frame Control**
- **Duration**
- **RA (Receiver Address)**
- **FCS**

802.11帧类型

- 三大类
  - 管理帧
  - 控制帧
  - 数据帧
- 每个大类下都有子类型

管理帧

- 管理帧负责监督无线网络状态，它主要用于建立第二层，即链路层，主机间的连接，管理数据包包括身份认证数据包、关联数据包和Beacon数据包等。（为了限制广播或组播管理帧所造成的副作用，收到管理帧后，必须加以查验。只有广播或者组播帧来自工作站当前所关联的BSSID时，它们才会被送至MAC管理层，唯一例外的是Beacon帧。）

管理帧

Filter: wlan.fc.type eq 0 | Expression... | Clear | Apply | Save

No.	Time	Source	Destination	Protocol	Length	Info
9809	158.599234	D-LinkIn_e5:31:2e	Broadcast	802.11	296	Beacon frame, SN=60, FN=0, Flags=........, BI=100, SSID=dlink85cuccs
9860	159.410875	D-LinkIn_e5:31:2e	Broadcast	802.11	296	Beacon frame, SN=75, FN=0, Flags=........, BI=100, SSID=dlink85cuccs
9872	159.827272	D-LinkIn_e5:31:2e	Broadcast	802.11	296	Beacon frame, SN=79, FN=0, Flags=........, BI=100, SSID=dlink85cuccs
9883	159.927297	D-LinkIn_e5:31:2e	Broadcast	802.11	296	Beacon frame, SN=80, FN=0, Flags=........, BI=100, SSID=dlink85cuccs
9899	160.331560	D-LinkIn_e5:31:2e	Broadcast	802.11	296	Beacon frame, SN=84, FN=0, Flags=........, BI=100, SSID=dlink85cuccs
10014	162.880627	D-LinkIn_e5:31:2e	Apple_9d:5b:5c	802.11	421	Probe Response, SN=151, FN=0, Flags=........, BI=100, SSID=dlink85cuccs
10017	162.891126	D-LinkIn_e5:31:2e	Apple_9d:5b:5c	802.11	421	Probe Response, SN=153, FN=0, Flags=........, BI=100, SSID=dlink85cuccs
10018	162.902324	D-LinkIn_e5:31:2e	Broadcast	802.11	296	Beacon frame, SN=152, FN=0, Flags=........, BI=100, SSID=dlink85cuccs
10022	162.925095	D-LinkIn_e5:31:2e	Apple_9d:5b:5c	802.11	421	Probe Response, SN=154, FN=0, Flags=........, BI=100, SSID=dlink85cuccs
10024	162.947036	D-LinkIn_e5:31:2e	Apple_9d:5b:5c	802.11	421	Probe Response, SN=155, FN=0, Flags=........, BI=100, SSID=dlink85cuccs
10025	162.947248	D-LinkIn_e5:31:2e	Apple_ad:b8:35	802.11	421	Probe Response, SN=156, FN=0, Flags=........, BI=100, SSID=dlink85cuccs
10027	162.969719	D-LinkIn_e5:31:2e	Apple_ad:b8:35	802.11	421	Probe Response, SN=157, FN=0, Flags=........, BI=100, SSID=dlink85cuccs
10030	163.001160	D-LinkIn_e5:31:2e	Broadcast	802.11	296	Beacon frame, SN=158, FN=0, Flags=........, BI=100, SSID=dlink85cuccs
10036	163.153955	D-LinkIn_e5:31:2e	Apple_ad:b8:35	802.11	421	Probe Response, SN=165, FN=0, Flags=........, BI=100, SSID=dlink85cuccs
10047	163.209465	D-LinkIn_e5:31:2e	Broadcast	802.11	296	Beacon frame, SN=168, FN=0, Flags=........, BI=100, SSID=dlink85cuccs
10060	163.407586	D-LinkIn_e5:31:2e	Broadcast	802.11	296	Beacon frame, SN=171, FN=0, Flags=........, BI=100, SSID=dlink85cuccs
10098	163.596599	D-LinkIn_e5:31:2e	SamsungE_c8:4a:7f	802.11	421	Probe Response, SN=173, FN=0, Flags=........, BI=100, SSID=dlink85cuccs

```
▷ Frame 10014: 421 bytes on wire (3368 bits), 421 bytes captured (3368 bits)
▽ IEEE 802.11 Probe Response, Flags: ........
    Type/Subtype: Probe Response (0x05)
  ▽ Frame Control Field: 0x5000
      .... ..00 = Version: 0
      .... 00.. = Type: Management frame (0)
      0101 .... = Subtype: 5
    ▽ Flags: 0x00
```

```
0000  50 00 3a 01 70 11 24 9d  5b 5c d8 fe e3 e5 31 2e   P.:.p.$. [\....1.
0010  d8 fe e3 e5 31 2e 70 09  b7 bf 74 0c 00 00 00 00   ....1.p. ..t.....
0020  64 00 31 04 00 0c 64 6c  69 6e 6b 38 35 63 75 63   d.1...dl ink85cuc
0030  63 73 01 08 82 84 8b 96  0c 12 18 24 03 01 07 2a   cs...... ...$...*
0040  01 00 32 04 30 48 60 6c  2d 1a 6e 18 1e ff 00 00   ..2.0H`l -.n.....
0050  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ........ ........
0060  00 00 00 00 3d 16 07 07  00 00 00 00 00 00 00 00   ....=... ........
0070  00 00 00 00 00 00 00 00  00 00 00 00 dd 16 00 50   ........ .......P
0080  f2 01 01 00 00 50 f2 04  01 00 00 50 f2 04 01 00   .....P.. ...P....
0090  00 50 f2 02 30 14 01 00  00 0f ac 04 01 00 00 0f   .P..0... ........
```

控制帧

- 控制帧通常与数据帧搭配使用，负责清空区域获取信道和载波监听的维护，并在收到数据时予以确认以提高工作站之间数据传送的可靠性（因为无线收发器通常只有半双工工作模式，即无法同时收发数据，为防止冲突，802.11允许工作站使用request to send和clear to send信号来清空传送区域）

控制帧

Filter: wlan.fc.type eq 1 | Expression... | Clear | Apply | Save

No.	Time	Source	Destination	Protocol	Length	Info
10012	162.869361		D-LinkIn_e5:31:18 (RA)	802.11	10	Clear-to-send, Flags=........
10015	162.890915		MurataMa_aa:72:eb (RA)	802.11	10	Acknowledgement, Flags=........
10020	162.913661		D-LinkIn_e5:31:18 (RA)	802.11	10	Clear-to-send, Flags=........
10029	162.991037		D-LinkIn_e5:31:18 (RA)	802.11	10	Clear-to-send, Flags=........
10031	163.075859		D-LinkIn_ce:4d:1c (RA)	802.11	10	Acknowledgement, Flags=........
10033	163.098320		D-LinkIn_e5:31:2e (RA)	802.11	10	Acknowledgement, Flags=........
10035	163.120776		D-LinkIn_e5:31:2e (RA)	802.11	10	Acknowledgement, Flags=........
10041	163.165556		D-Link_e0:a0:46 (RA)	802.11	10	Acknowledgement, Flags=........
10043	163.187130		D-Link_e0:a0:46 (RA)	802.11	10	Acknowledgement, Flags=........
10045	163.187220		D-Link_e0:a0:46 (RA)	802.11	10	Acknowledgement, Flags=........
10046	163.187225		D-LinkIn_e5:31:2e (RA)	802.11	10	Acknowledgement, Flags=........
10052	163.362665		MurataMa_aa:72:eb (RA)	802.11	10	Clear-to-send, Flags=........
10054	163.362873		MurataMa_aa:72:eb (RA)	802.11	10	Acknowledgement, Flags=........
10056	163.374037		D-LinkIn_e5:31:18 (RA)	802.11	10	Clear-to-send, Flags=........
10062	163.418864		D-LinkIn_78:c6:77 (RA)	802.11	10	Acknowledgement, Flags=........
10063	163.418917		D-LinkIn_e5:31:18 (RA)	802.11	10	Clear-to-send, Flags=........
10066	163.419958		Apple_9d:5b:5c (RA)	802.11	10	Acknowledgement, Flags=........

```
▷ Frame 10012: 10 bytes on wire (80 bits), 10 bytes captured (80 bits)
▽ IEEE 802.11 Clear-to-send, Flags: ........
    Type/Subtype: Clear-to-send (0x1c)
  ▽ Frame Control Field: 0xc400
      .... ..00 = Version: 0
      .... 01.. = Type: Control frame (1)
      1100 .... = Subtype: 12
    ▽ Flags: 0x00
```

```
0000  c4 00 c4 00 d8 fe e3 e5  31 18                    ........ 1.
```

数据帧

- 数据帧中包含实际需要传送的数据，并且是能够从无线网络转发到有线网络的唯一帧类型。

数据帧

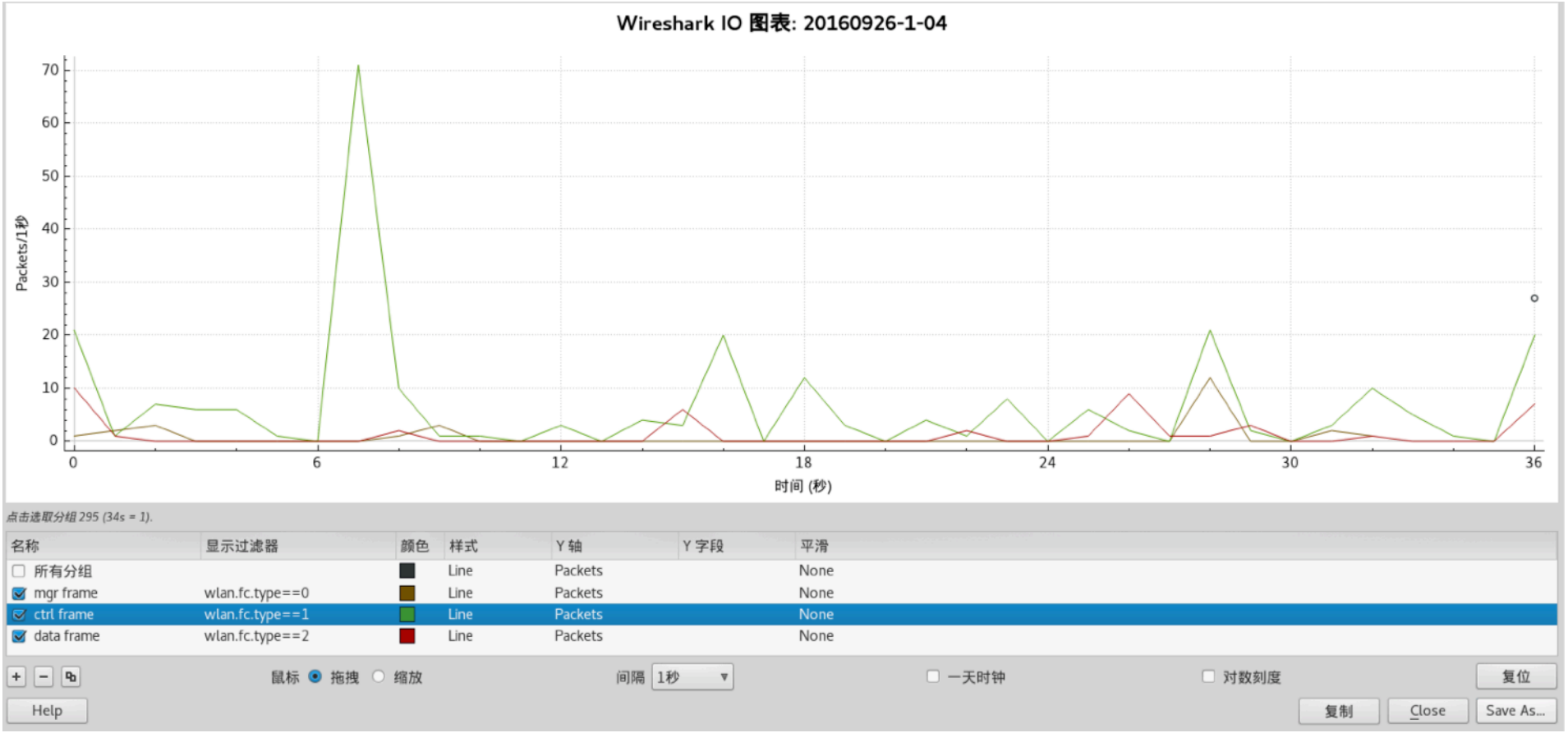
Filter: wlan.fc.type eq 2 | Expression... | Clear | Apply | Save

No.	Time	Source	Destination	Protocol	Length	Info
13530	211.881752	52:1a:a9:c1:fc:39	Zte_10:80:5f	802.11	26	QoS Null function (No data), SN=262, FN=0, Flags=.....F.
17441	274.820930	Apple_07:17:53	Broadcast	802.11	76	Data, SN=2815, FN=0, Flags=.p....F.
17468	275.132509	Apple_07:17:53	Broadcast	802.11	76	Data, SN=2825, FN=0, Flags=.p....F.

```
Frame 17441: 76 bytes on wire (608 bits), 76 bytes captured (608 bits)
IEEE 802.11 Data, Flags: .p....F.
    Type/Subtype: Data (0x20)
  Frame Control Field: 0x0842
      .... ..00 = Version: 0
      .... 10.. = Type: Data frame (2)
      0000 .... = Subtype: 0
    Flags: 0x42
        .... ..10 = DS status: Frame from DS to a STA via AP(To DS: 0 From DS: 1) (0x02)
        .... .0.. = More Fragments: This is the last fragment
        .... 0... = Retry: Frame is not being retransmitted
        ...0 .... = PWR MGT: STA will stay up
        ..0. .... = More Data: No data buffered
        .1.. .... = Protected flag: Data is protected
        0... .... = Order flag: Not strictly ordered
    .000 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: D-LinkIn_e5:31:2e (d8:fe:e3:e5:31:2e)
    BSS Id: D-LinkIn_e5:31:2e (d8:fe:e3:e5:31:2e)
    Source address: Apple_07:17:53 (48:74:6e:07:17:53)
    Fragment number: 0
    Sequence number: 2815
  CCMP parameters
      CCMP Ext. Initialization Vector: 0x00000000003B
      Key Index: 1
Data (44 bytes)
```

```
0000  08 42 00 00 ff ff ff ff  ff ff d8 fe e3 e5 31 2e   .B...... ......1.
0010  48 74 6e 07 17 53 f0 af  3b 00 00 60 00 00 00 00   Htn..S.. ;..`....
0020  af 38 40 52 de 8d a2 bc  fb 45 58 c8 7e 1b 31 2e   .8@R.... .EX.~.1.
0030  e0 2b 34 be 3b 67 2b 24  e9 88 1e b3 4d b5 9b da   .+4.;g+$ ....M...
0040  f6 64 b3 8a 3b dc 31 13  d7 c9 0c 39               .d..;.1. ...9
```

用wireshark统计分析



802.11 MAC地址类型

- DA = Destination MAC Address
- SA = Source MAC Address
- RA = Receiver Address indicate MAC Address of station in WM that have to receive frame
- TA = Transmitter Address indicate station which have transmitted frame in WM
- BSSID

802.11帧头

byte

Frame control	Duration/ID	Addr1	Addr2	Addr3	Seq-ctl	Addr4	body	FCS
2	2	6	6	6	2	6	0~2312	4

Frame Control

bit

Protocol	Type	Sub type	To DS	From DS	More Frag	Retry	Pwr Mgmt	More Data	Protect Frame	Order
2	2	4	1	1	1	1	1	1	1	1

From DS和To DS的取值组合决定了Addr1-Addr4的意义

802.11帧头

To DS	*From DS*	*Addr 1*	*Addr 2*	*Addr 3*	*Addr 4*
0	*0*	*DA*	*SA*	*BSSID*	*N/A*
0	*1*	*DA*	*BSSID*	*SA*	*N/A*
1	*0*	*BSSID*	*SA*	*DA*	*N/A*
1	*1*	*RA*	*TA*	*DA*	*SA*

注意在wireshark的packet字段解析时，同一个字段地址可能会显示2个或多个等价地址类型名称

802.11帧结构与Wireshark过滤器语法

- wlan.*
- wlan_mgt.*

Demo time

802.11加密与认证机制原理

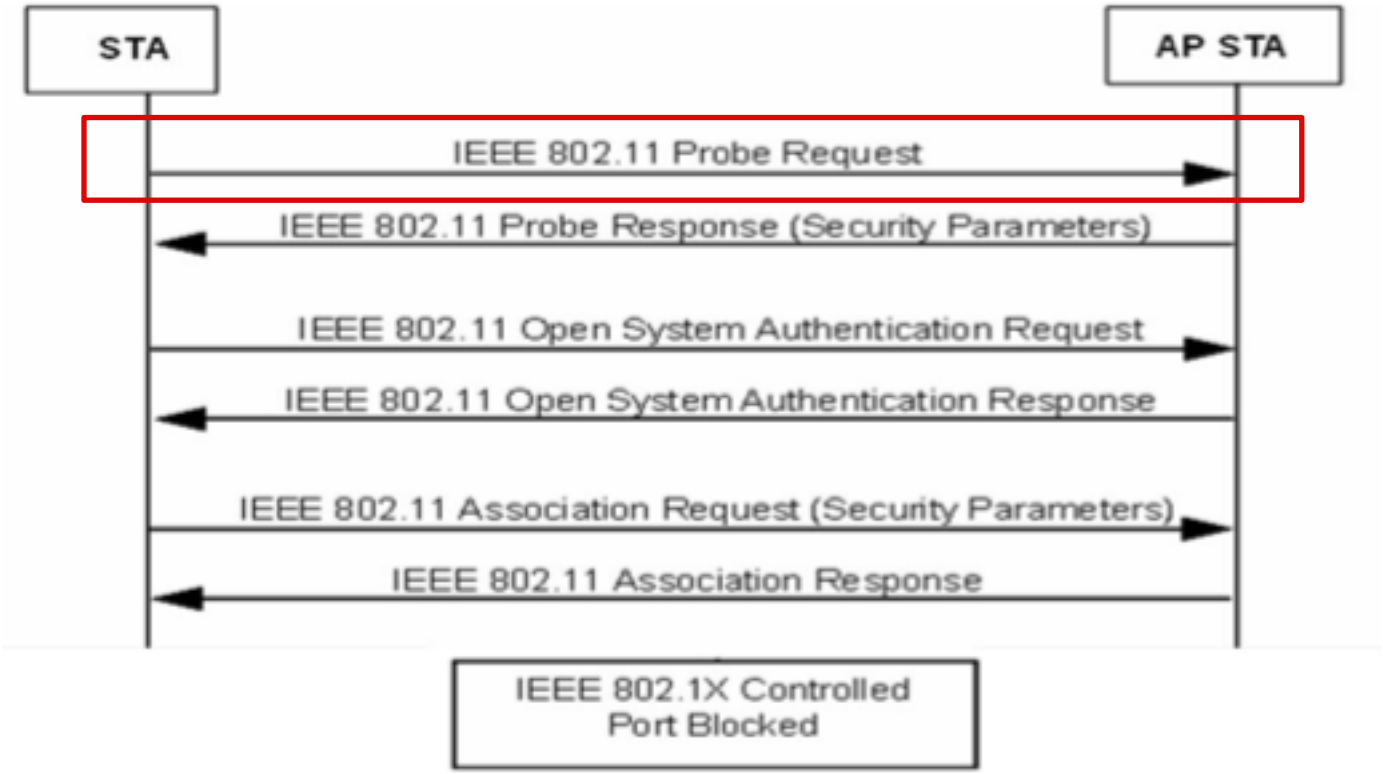
- 开放式认证（无认证）
- ~~WEP - Wired Equivalency Protocol~~
- WPA - Wi-Fi Protected Access
- WPA2 - 802.11i
- WPS - Wi-Fi Protected Setup
- WPA/WPA2 企业级认证

  —基于802.1X EAP

开放式认证

- 开放式认证本身是一种无效的认证算法，如果没有配合数据加密技术AP将会准许任何来自客户端的认证要求，即任何知道基站SSID的客户端都可以连接到此网络
  - 但如果基站使用WEP加密算法对数据进行加密，WEP密钥就成为了另一种存取控制机制，即客户端即使通过了认证，但若没有WEP密钥也无法将资料传送到基站亦或是将基站传送出的数据进行解码

802.11关联过程（无加密，开放认证）



可选步骤，AP如果开启了SSID广播，则STA可以通过beacon frame得到认证相关信息

WPA

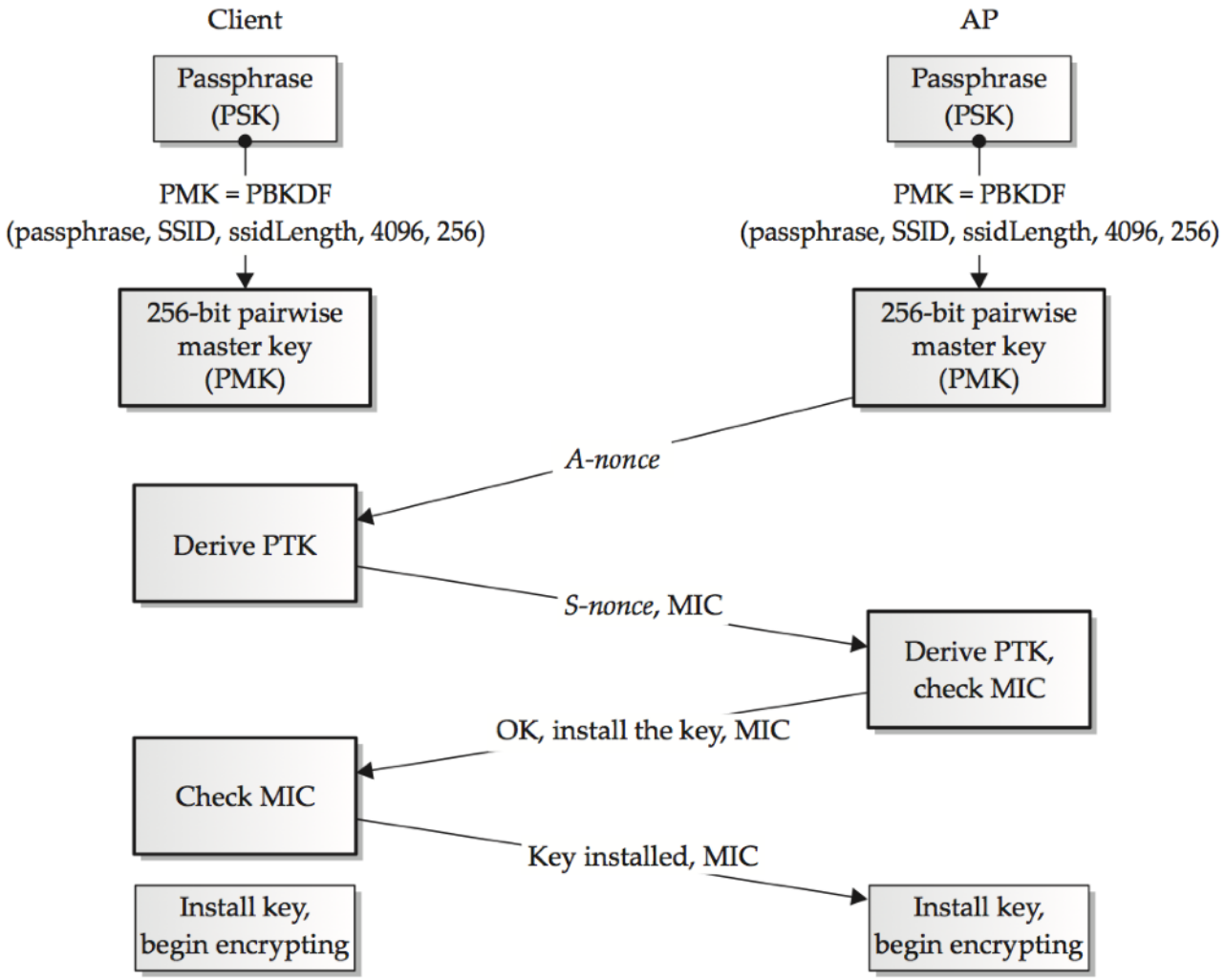
- 基于802.11i协议第3版草稿
- 使用TKIP: Temporal Key Integrity Protocol
- 向前兼容所有支持WEP机制的硬件设备

WPA2

- 802.11i正式版
- 使用CCMP（AES）加密协议
  - Counter Cipher Mode with block chaining message authentication code Protocol
- 不向后兼容老旧设备
- 所有新设备获得Wi-Fi认证标志的必备条件之一

四次握手认证

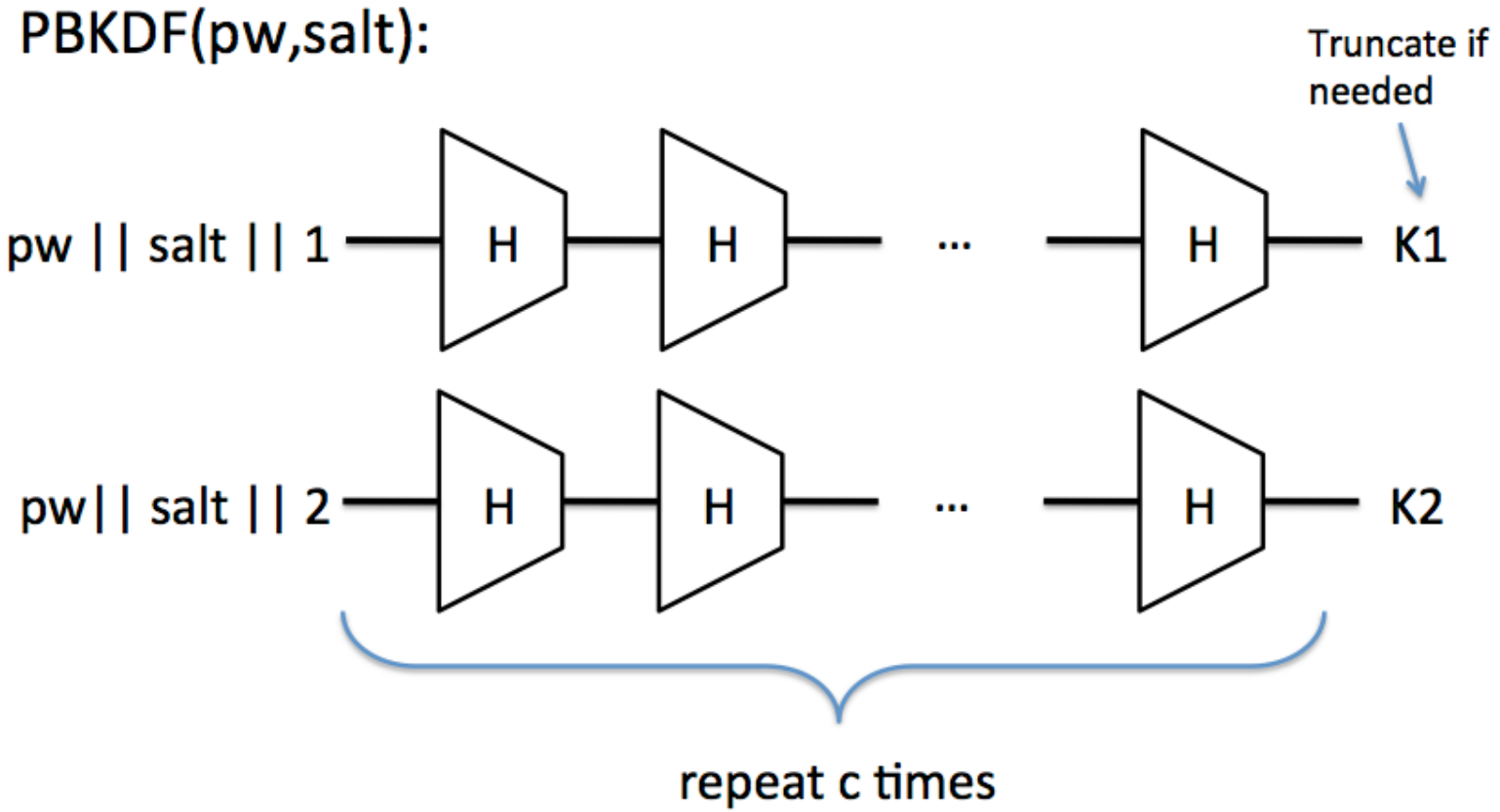
- PSK
- PBKDF
- PMK
- A-nonce
- PTK
- S-nonce
- MIC



核心概念

- PSK: Pre-Shared Key
  - passpharse
- PBKDF: Password-Based Key Derivation Function
- PMK = PBKDF(PSK, SSID, ssidLength, c)
  - CCMP: c=4096, 输出长度：256bit
- MIC=HMAC-MD5(PTK, 2nd Msg)

PBKDF


PBKDF(pw,salt):
Truncate if needed
pw || salt || 1
H
H
...
H
K1
pw|| salt || 2
H
H
...
H
K2
repeat c times

PBKDF的密码学意义

- 相同的密码输入，经过PBKDF运算之后每次的结果都不相同
- 通过增大迭代参数c，降低暴力破解的速度
- salt的选择如果做到不可预测，则可以抵御预先计算PBKDF字典的加速暴力破解攻击方法
  - WPA/WPA2 PSK使用的salt是SSID和ssidLength

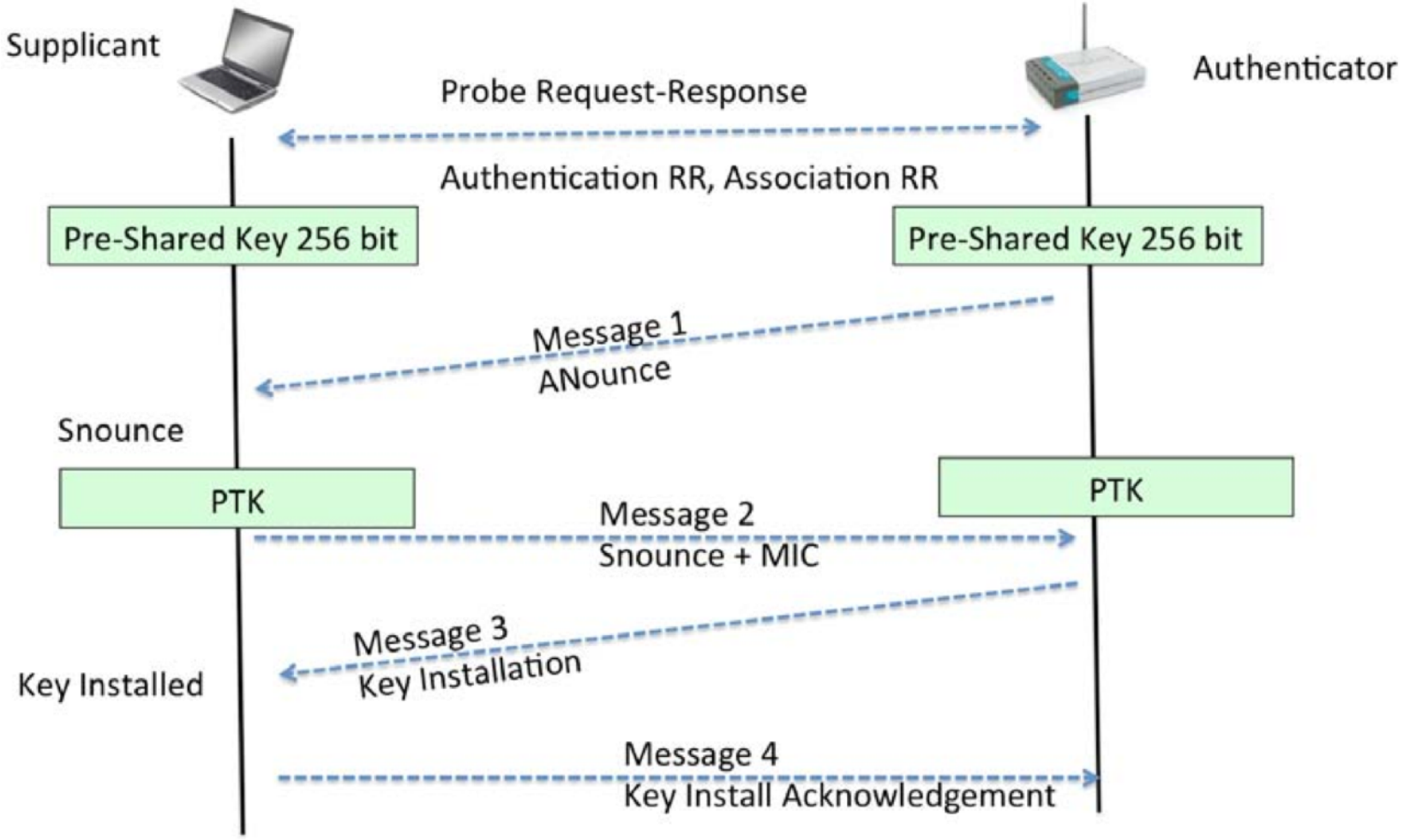
核心概念

- A-nonce: Authenticator (generated) nonce
- S-nonce: Supplicant (generated) nonce

PTK - Pairwise Transient Key

- PTK = Function(PMK, A-nonce, S-nonce, Authenticator MAC, Supplicant MAC)
  - Hash(PMK||A-nonce|| S-nonce || AP MAC || STA MAC)

完整的WPA/WPA2 PSK认证流程



Wi-Fi Direct（Wi-Fi直连）

- 最开始被称为Wi-Fi P2P
- 使用WPS协议为每个联网设备分配一个受限的无线接入点
- 发展自Ad-hoc（IBSS），但有区别：
  - 安全性：Ad-hoc 默认WEP， 而Wi-Fi Direct 默认WPA2
  - 连接：Wi-Fi Direct支持连接Wi-Fi Direct同时连接已存在的网络，Ad-hoc只能工作在其中一种

WPS - Wi-Fi Protected Setup

- 来自于协议规范：Wi-Fi Simple Config
  - 传统方式加入一个无线局域网需要手动输入SSID和密码
  - WPS能帮助用户自动获得并设置SSID、认证方式和密钥
  - Wi-Fi Alliance在2006年提出协议规范
  - 最新版WPS协议规范仅支持配置WPA2-PSK网络
- 无线设备制造商使用自己产品线的别名
  - QSS - Quick Secure Setup, TP-Link设备的功能别名
  - Push 'N' Connect ， Netgear设备的功能别名

WPS - Wi-Fi Protected Setup

- 优点
  - 改进了用户使用无线网络的体验，简单化
- 缺点
  - 认证机制存在已知脆弱性，大大降低了网络接入认证的保护强度

WPS - 应用场景

- 主要场景
  - 设置一个新的安全的 WLAN，并为该 WLAN 添加无线设备
  - PIN、PBC
- 次要场景
  - 从 WLAN 中移除某个无线设备、通过添加新的AP或路由器来扩充WLAN 的覆盖范围、密钥信息更换（Re-keying credentials）等

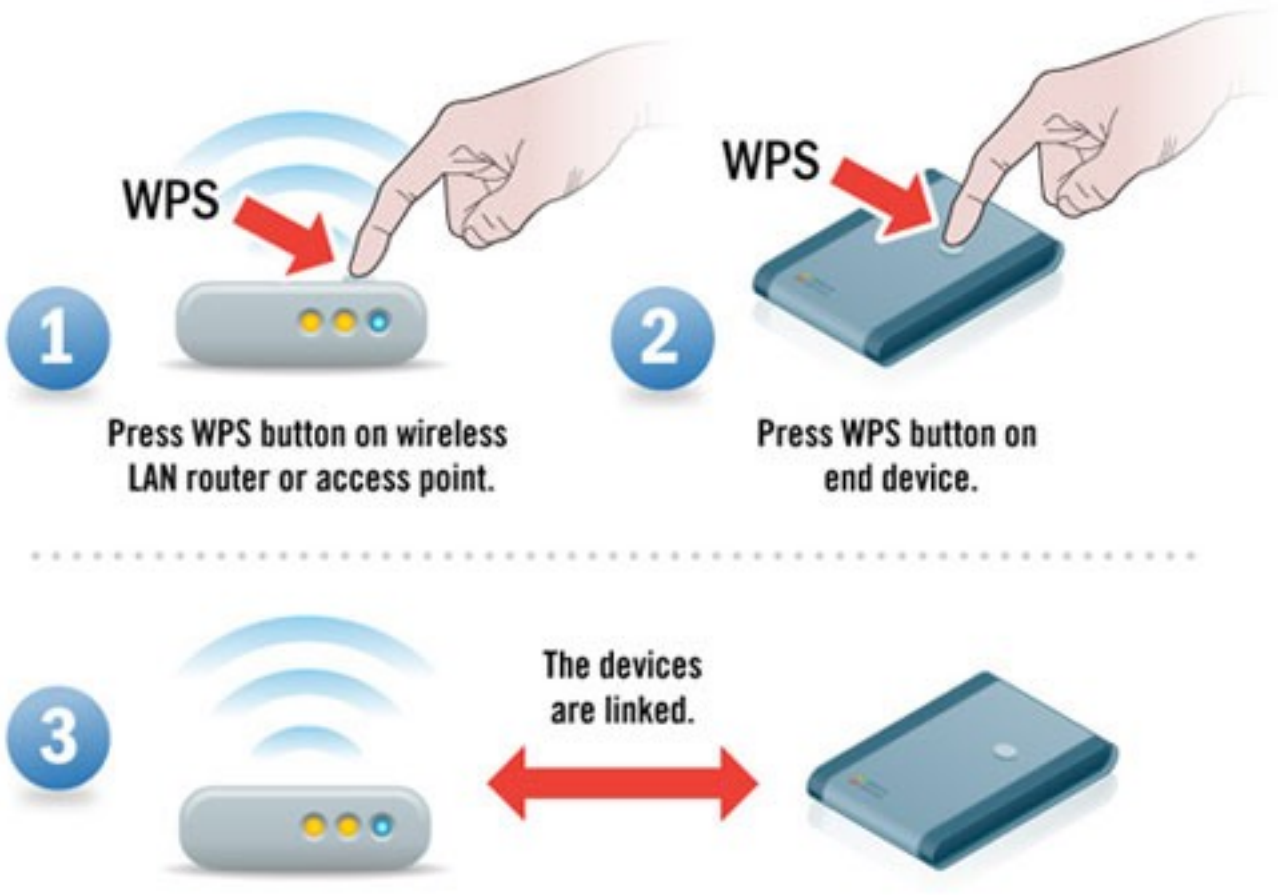
WPS - Wi-Fi Protected Setup 工作模式

- PIN
  - —Personal Identification Number
- PBC
  - —Push Button Configuration
- 带外（out-of-band）方法（非协议标准）
  - —NFC
    - Near Filed Communication
  - —USB
    - 已过时

使用WPS加入无线网络——PIN

- 静态（贴纸）或动态显示PIN在无线设备上
- 在AP上输入无线客户端上产生的PIN（内部注册）或在无线客户端上输入AP上的PIN（外部注册）
- STA基于上述约定好的PIN完成扫描、关联、四次握手等工作以加入目标无线网络
- 所有WPS兼容设备必须支持的工作模式
- Wi-Fi直连规范要求所有包含键盘或显示屏的设备都必须支持该模式

使用WPS加入无线网络——PBC方式



使用WPS加入无线网络——PBC方式

- 使用物理按键或虚拟按键
- AP和无线客户端应同时或在先后间隔的一小段时间（通常是2分钟或更短时间）内先后按下设备上的Push按钮
- 所有AP必须支持，无线客户端可选支持
- Wi-Fi直连规范要求所有无线设备都必须支持该模式
  - —Wi-Fi直连规范最早被称为Wi-Fi P2P，无需无线AP连接无线设备的协议规范

WPS协议术语定义

- Registrar（作为外部Registrar角色的无线客户端设备或AP）
  - 具有授权或取消网络接入能力的设备，还可以配置AP
  - 可以被集成到AP内或作为一个独立设备
- External Registrar
  - 作为一个独立设备运行的Registrar
- Enrollee（无线客户端设备或AP）
  - 准备加入到一个无线网络的设备，类似supplicant

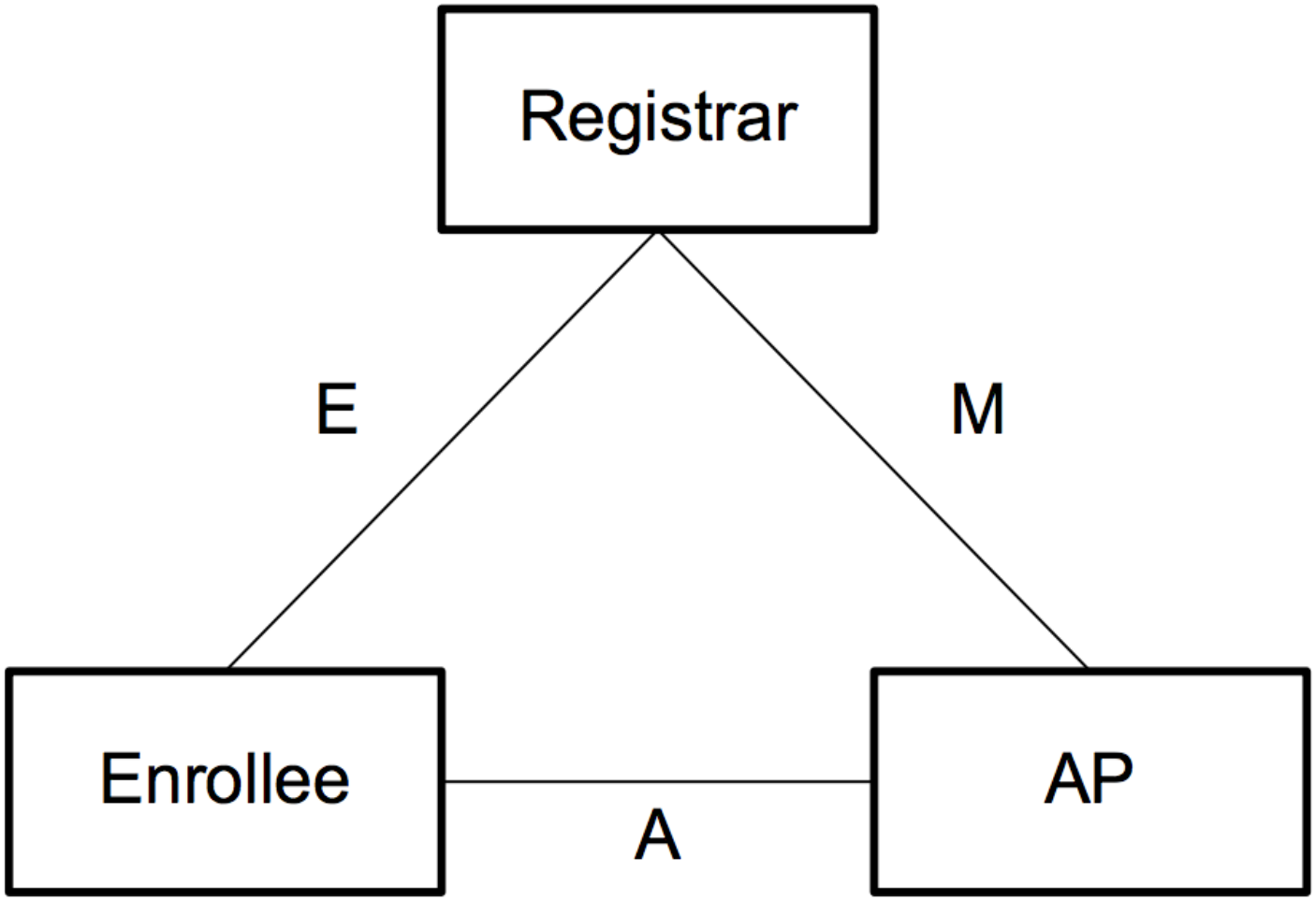
WPS协议术语定义

- Headless Device
  - 没有显示屏的无线网络设备
- 设备密码
  - 用于Registrar和Enrollee之间通过带内（In-band）双向认证的共享密钥
  - 官方协议标准(Wi-Fi Simple Configuration spec v2.0.5)建议PIN认证模式下手工输入的设备密码是8位数字

Device Password

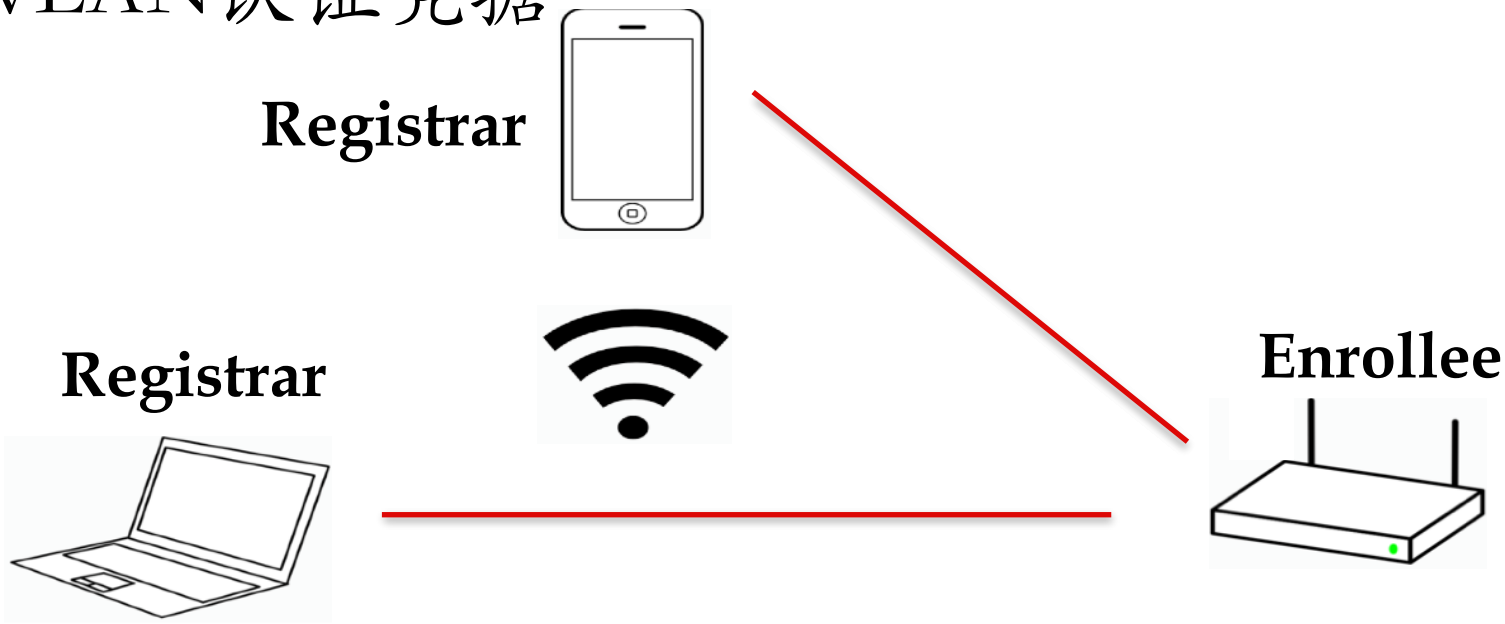
All devices supporting Wi-Fi Simple Configuration shall provide at least one numeric Device Password (PIN) for initial setup that is unique and randomly generated per device. Although it is possible and permitted for two devices to have the same Device Password, a group of devices should not intentionally be assigned the same Device Password, and the Device Password SHALL not be based on other characteristics of the device, such as MAC address or serial number. 下一章我们会深入讲这个安全风险

WPS技术架构



WPS技术架构——接口E（最常见）

- 逻辑上是连接Enrollee和Registrar，物理上AP在这2个逻辑实体对象之间扮演代理角色。该接口的目的是让Registrar发现和给Enrollee签发WLAN认证凭据

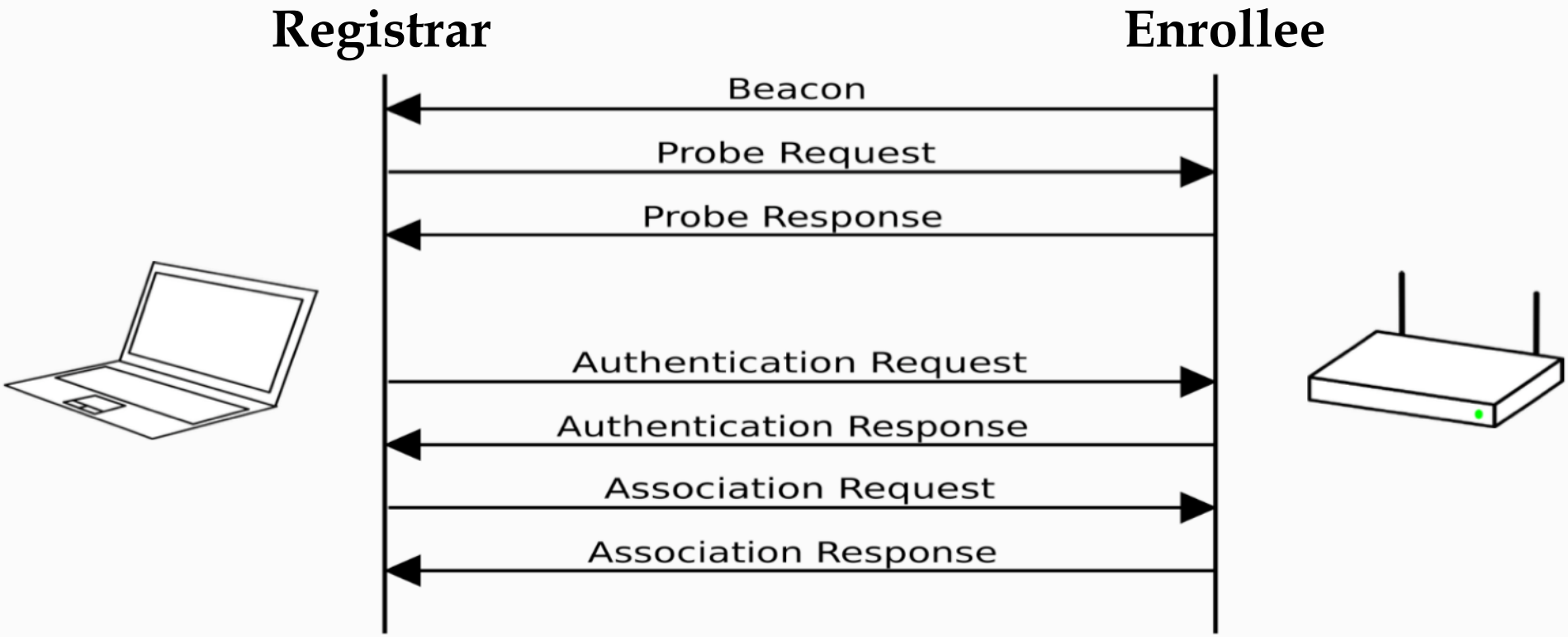


WPS技术架构——接口E（注册协议）

- 用于故障排查无线传输通道的基本连接问题
- 使用带外（out-of-band）信息提供Enrollee和Registrar之间双向识别，支持认证凭据配置
- 建立每个设备（AP，Registrar和Enrollee）之间的角色
- 从Registrar向Enrollee安全传输WLAN配置和其他配置信息
- 建立起扩展主会话密钥（EMSK，Extended Master Session Key），可以用于保护其他应用层相关配置函数的安全性

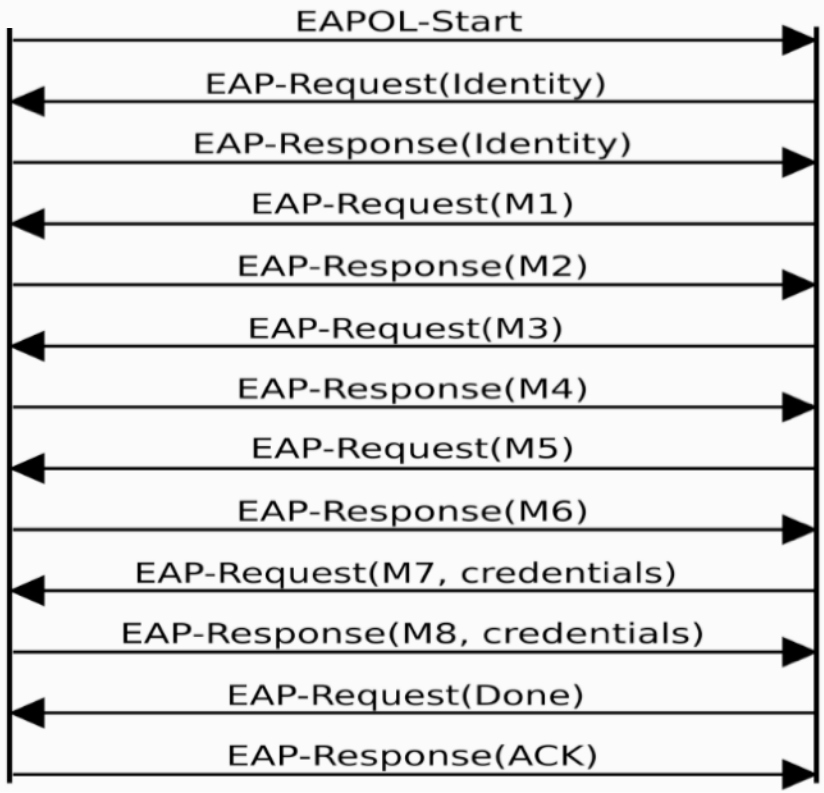
WPS PIN外部注册协议流程

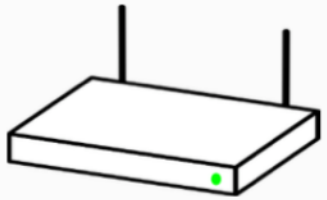
以外部注册（在AP上输入无线客户端上产生的PIN码）为例



WPS PIN外部注册协议流程

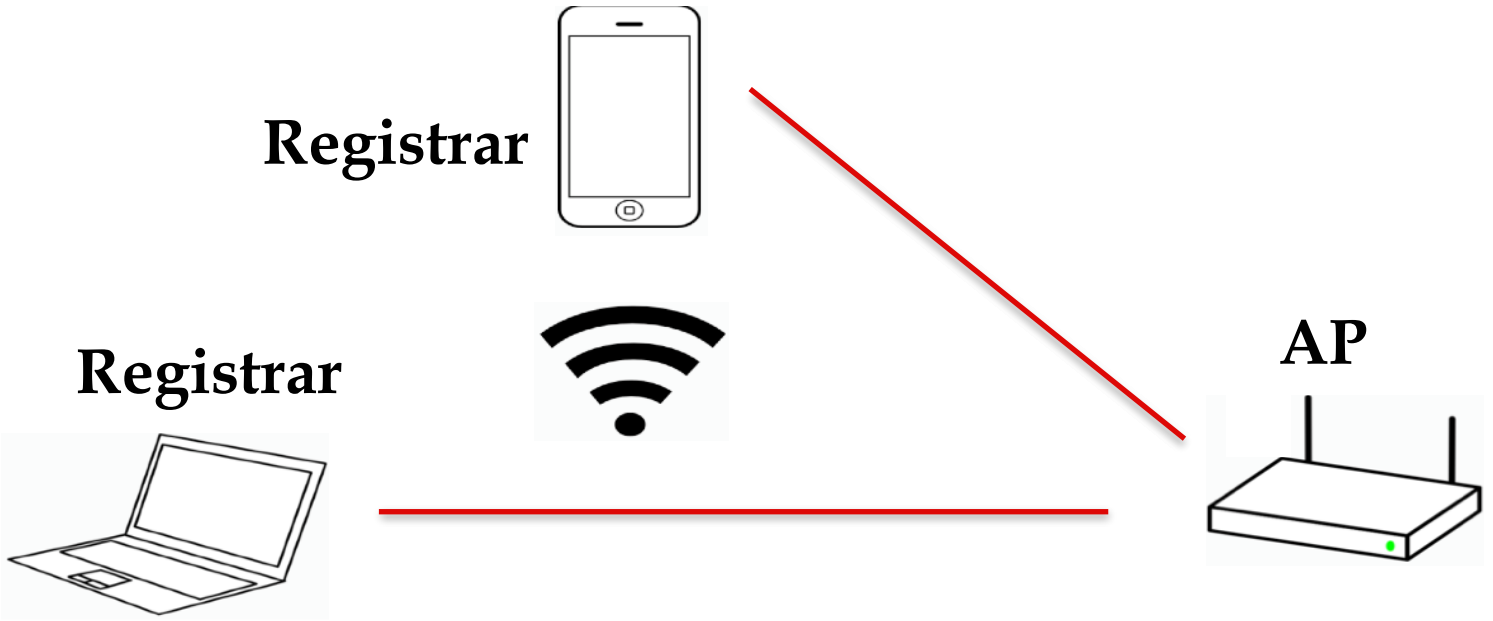
以外部注册（在AP上输入无线客户端上产生的PIN码）为例





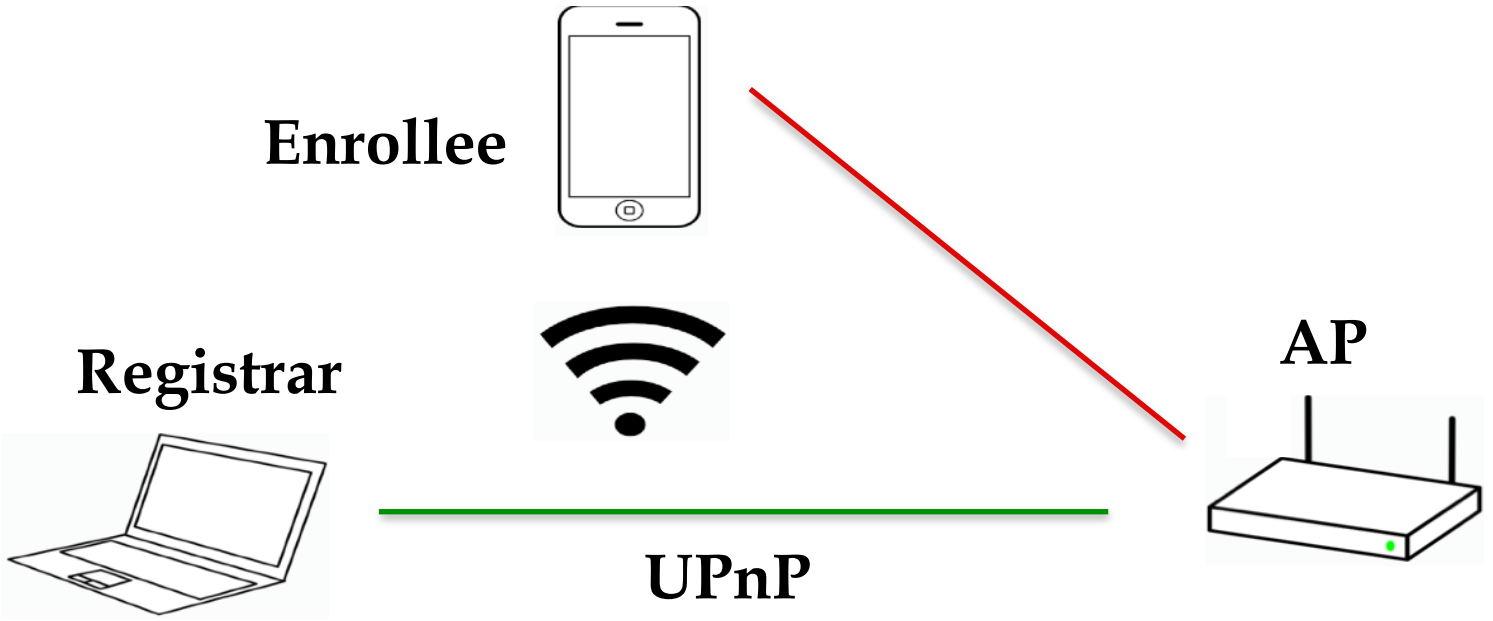
WPS技术架构——接口M

- AP和Registrar之间的接口。使得外部Registrar能管理一个WPS AP。WPS使用相同协议配置AP管理接口用于向Enrollee设备发送认证凭据



WPS技术架构——接口A

- AP和Enrollee之间的接口。支持WPS WLAN发现和支持Enrollee和IP协议Registrar设备之间的通信



WPS认证注册协议关键消息定义

- Enrollee -> Registrar: M1 = Version || N1 || Description || PKE
- Enrollee <- Registrar: M2 = Version || N1 || N2 || Description || PKR [|| ConfigData] || HMAC_AuthKey(M1 || M2*)
- Enrollee -> Registrar: M3 = Version || N2 || E-Hash1 || E-Hash2 || HMAC_AuthKey(M2 || M3*)
- Enrollee <- Registrar: M4 = Version || N1 || R-Hash1 || R-Hash2 || ENC_KeyWrapKey(R-S1) || HMAC_AuthKey (M3 || M4*)

WPS认证注册协议关键消息定义

- Enrollee -> Registrar: M5 = Version || N2 || ENC_KeyWrapKey(E-S1) || HMAC_AuthKey (M4 || M5*)
- Enrollee <- Registrar: M6 = Version || N1 || ENC_KeyWrapKey(R-S2) || HMAC_AuthKey (M5 || M6*)
- Enrollee -> Registrar: M7 = Version || N2 || ENC_KeyWrapKey(E-S2 [||ConfigData]) || HMAC_AuthKey (M6 || M7*)
- Enrollee <- Registrar: M8 = Version || N1 || [ENC_KeyWrapKey(ConfigData)] || HMAC_AuthKey (M7 || M8*)

WPS认证注册协议关键消息定义：符号和字段说明

- || 字符串拼接操作
- Mn* 不包含HMAC-SHA-256签名值的Mn消息
- Version 对应注册协议消息版本号
- N1 Enrollee设置的128bit随机数
- N2 Registrar设置的128bit随机数
- Description 包含一个人类可读的发送设备描述信息（UUID，设备商名称，型号，MAC地址等）和设备能力信息（支持算法范围，I/O信道，注册协议角色等等）。在802.11 probe request和probe response消息中也有

WPS认证注册协议关键消息定义：符号和字段说明

- HMAC_AuthKey(…) 使用AuthKey作为密钥的HMAC函数。为了减小消息长度，256bit HMAC输出的前64bit被包含在认证属性字段，默认使用 HMAC-SHA-256
- ENC_KeyWrapKey(…) 使用KeyWrapKey作为对称加密密钥的对称加密函数，默认使用AES-CBC
- PKE和PKR分别是 Enrollee和Registrar的Diffie-Hellman公钥。如果在未来使用了新的密钥交换算法（例如椭圆曲线），则会使用不同的协议版本号标识
- AuthKey由DH算法中产生的会话密钥、随机值N1和N2以及Enrollee的MAC地址计算推导出。如果M1和M2通过不受中间人攻击威胁的通道传输，Enrollee的设备密码可以在密钥推导过程中被省略

WPS认证注册协议关键消息定义：符号和字段说明

- E-Hash1和E-Hash2是Enrollee用于证明它掌握它自己的设备密码（分别对应前半部分和后半部分）
  - $\text{E-Hash1} = \text{HMAC}_{\text{AuthKey}}(\text{E-S1} \,||\, \text{PSK1} \,||\, \text{PKE} \,||\, \text{PKR})$
  - $\text{E-Hash2} = \text{HMAC}_{\text{AuthKey}}(\text{E-S2} \,||\, \text{PSK2} \,||\, \text{PKE} \,||\, \text{PKR})$
- R-Hash1和R-Hash2是Registrar用于证明它掌握Enrollee的设备密码（分别对应前半部分和后半部分）
  - $\text{R-Hash1} = \text{HMAC}_{\text{AuthKey}}(\text{R-S1} \,||\, \text{PSK1} \,||\, \text{PKE} \,||\, \text{PKR})$
  - $\text{R-Hash2} = \text{HMAC}_{\text{AuthKey}}(\text{R-S2} \,||\, \text{PSK2} \,||\, \text{PKE} \,||\, \text{PKR})$

WPS认证注册协议关键消息定义：符号和字段说明

- 任何一方证明自己掌握设备密码时
  - 设备密码首先被转换格式为2个128bit的PSK
    - PSK1 = first 128 bits of $HMAC_{AuthKey}$(1st half of DevicePassword)
    - PSK2 = first 128 bits of $HMAC_{AuthKey}$(2nd half of DevicePassword)
  - 例如，如果PIN码是39358448，则转换之后的存储数据结构是8字节ASCII字符串“39358448”。PSK1来自于对“3935”计算HMAC取前128bit，PSK2来自于对“8448”计算HMAC取前128bit

WPS认证注册协议关键消息定义：符号和字段说明

- R-S1和R-S2均是128bit随机数，分别对应和R-Hash1和R-Hash2一起被用于Enrollee验证Registrar已经掌握Enrollee设备密码的前半部分和后半部分
  - Enrollee解密R-S1后，代入公式计算结果是否等于收到的R-Hash1。R-S2和R-Hash2的验证过程类似
- E-S1和E-S2均是128bit随机数，分别对应和E-Hash1和E-Hash2一起被用于Registrar验证Enrollee已经掌握Enrollee设备密码的前半部分和后半部分
  - Registrar类似上述R-S1和R-Hash1的验证过程，分别验证E-S1和E-Hash1匹配与否、E-S2和E-Hash2匹配与否

WPS认证注册协议关键消息定义：符号和字段说明

- ConfigData包含提供给Enrollee的WLAN配置和认证凭据，其他配置信息也可能包含于此。虽然ConfigData在这里被定义为加密存储，但只有密钥和密钥绑定信息是强制要求加密的。其他配置数据可选择加密。发送者自行决定是否加密ConfigData哪一部分。

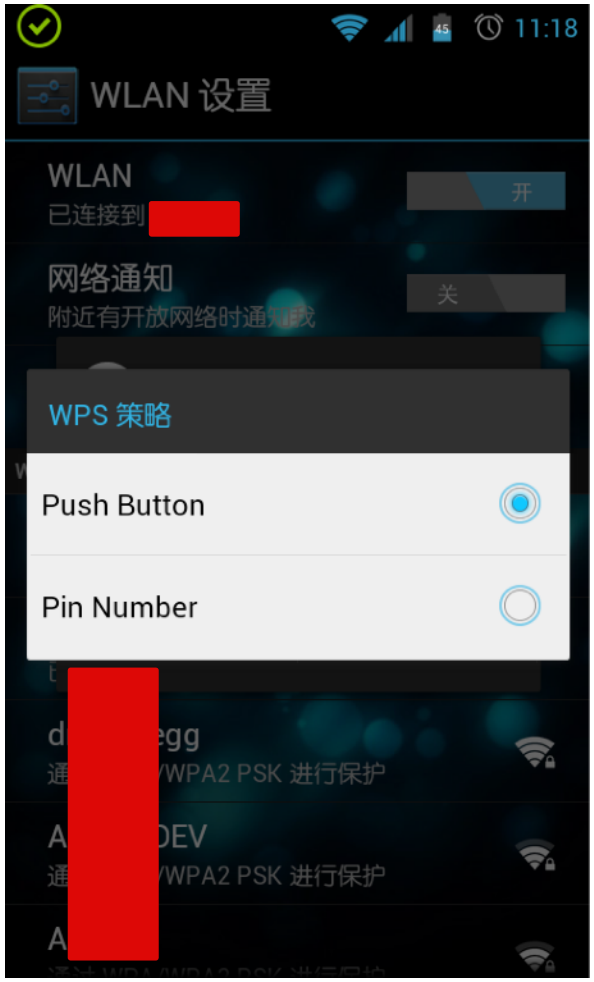
WPS的设备支持情况


F12
Backspace
Enter
Shift
Ctrl


F9
F11
TP-LINK
Scroll Lock
Pause Break
Enter
Pg Up
Pg Dn
QSS
Ctrl
4
1
End

WPS的系统支持情况


11:17
WLAN 设置
WLAN
已连接到
开
网络通知
附近有开放网络时通知我
关
添加 WPS 网络
WLAN 网络
WPA/WPA2 PSK 进行保护
WPA/WPA2 PSK 进行保护
ala cuccs


11:18
WLAN 设置
WLAN
已连接到
开
网络通知
附近有开放网络时通知我
关
WPS 策略
Push Button
Pin Number
WPA2 PSK 进行保护
WPA2 PSK 进行保护


11:17
WLAN 设置
WLAN
已连接到
开
网络通知
附近有开放网络时通知我
关
添加 WPS 网络
WPS 策略
Push Button
取消
开始
WPA2 PSK 进行保护
cuccs

WPS的系统支持情况


D-LINK SYSTEMS, INC. | WIRELESS AP : ADVANCED / WI-FI PROTECTED SETUP - Windows Internet Explorer
http://192.168.0.50/adv_wpssetting.htm
D-LINK SYSTEMS, INC. | WIRELESS AP : ADV...
DWL-2000AP+A
设置
高级
维护
状态
帮助
高级无线
MAC地址过滤器
WIFI保护安装
注销
WIFI保护安装 :
WPS(一键加密）功能可以通过使用PIN码或按键的方式很方便的将设备添加到网络中，要通过此种方式设定，设备必须支持WPS功能。
如果PIN码有改变，新的PIN码将在随后的WPS设置中生效。点击“不保存设定”按钮，PIN码将不会被重置。
不过，如果新PIN码未经保存，在设备重启或掉电后该PIN将失效。
保存设置
不保存设置
帮助提示...
启用如果你想包括局域网中其他支持Wi-Fi保护设置（WPS一键加密）的无线设备
点击添加无线设备向导使用Wi-Fi保护设定（WPS一键加密）将无线设备添加到无线网络
WIFI保护安装（在WINDOWS VISTA中也叫WCN 2.0 ） :
启用 :
关闭WPS-PIN方法 :
重置为未配置
PIN码设定
当前PIN: 10030103
将PIN重置为缺省设置
生成新的PIN
添加无线设备向导
添加无线设备向导
Internet | 保护模式: 启用
100%


TP-LINK
状态
QSS
网络
配置文件
高级
QSS
本应用程序将指导你完成无线网络配置。
请选择一种接入无线网络的方法:
按下接入点或无线路由器的按钮
输入接入点或无线路由器的PIN
输入设备的PIN
连 接

使用WPS加入无线网络——NFC

- 无线设备可以使用RFID兼容标签
- 无线认证参数和密钥数据可以通过NFC在2个无线设备之间进行近距离的传输

真实场景抓包分析

动手时间！

抓包任务

- AP广播的beacon frame
- STA主动发出的probe request frame
- 开放认证: 认证成功、解除认证
- WEP: 认证成功、认证失败、解除认证
- WPA-PSK: 认证成功、认证失败、解除认证
- WPA2-PSK: 认证成功、认证失败、解除认证
- WPS: 认证成功、认证失败、解除认证
  - PIN码模式、push to connect模式

无线网络分别配置DHCP和禁用DHCP状态下进行抓包

分析任务

- 哪些无线帧里包含SSID字段信息
- 绘制WEP的认证流程图
- WPA和WPA2在认证流程上是否有区别
- 绘制WPS的认证流程图

WPA/WPA2 企业级认证

802.11与802.1X

- 802.11是无线网络链路层协议规范
- 802.1X是物理层无关的基于端口的（链路层）访问控制协议
- 两者组合后可以提高无线网络安全性

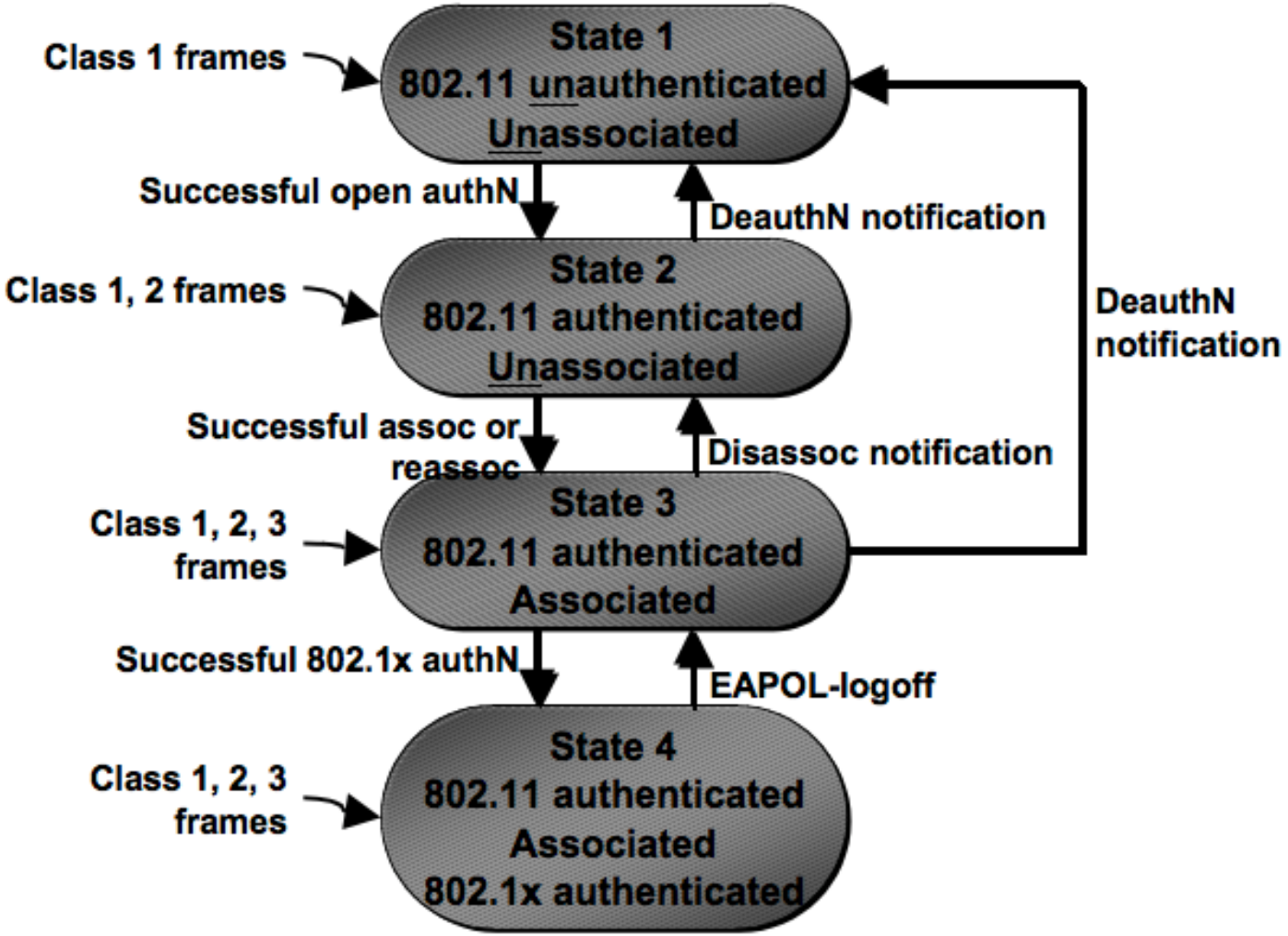
802.1X与802.11身份认证需求

- 可以解决
  - 不同无线客户端使用独立认证凭据
  - 伪造AP和中间人攻击
  - 精细化授权
- 无法解决
  - 伪造数据报文和伪造断开连接请求进行DoS

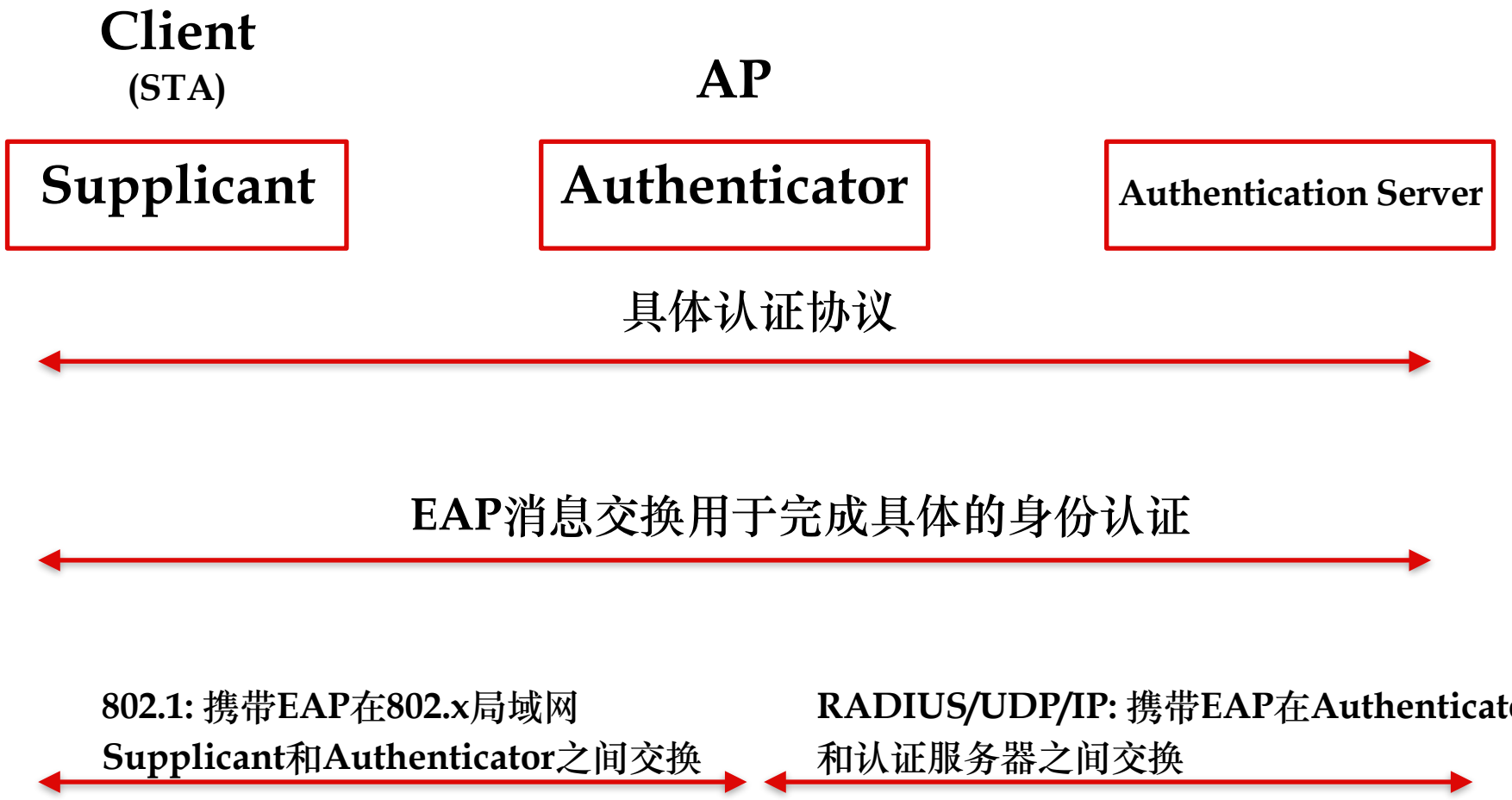
802.1X

- 解决用户身份认证问题
- 定义了有线和无线局域网传送EAP的标准
- EAP消息被封装在以太帧负载
- 提供基于（交换机）端口的访问控制
- 在不改动现有网络设备的前提下提供高层应用新的认证方式
- 保证最新的安全技术可以兼容现有网络基础设施

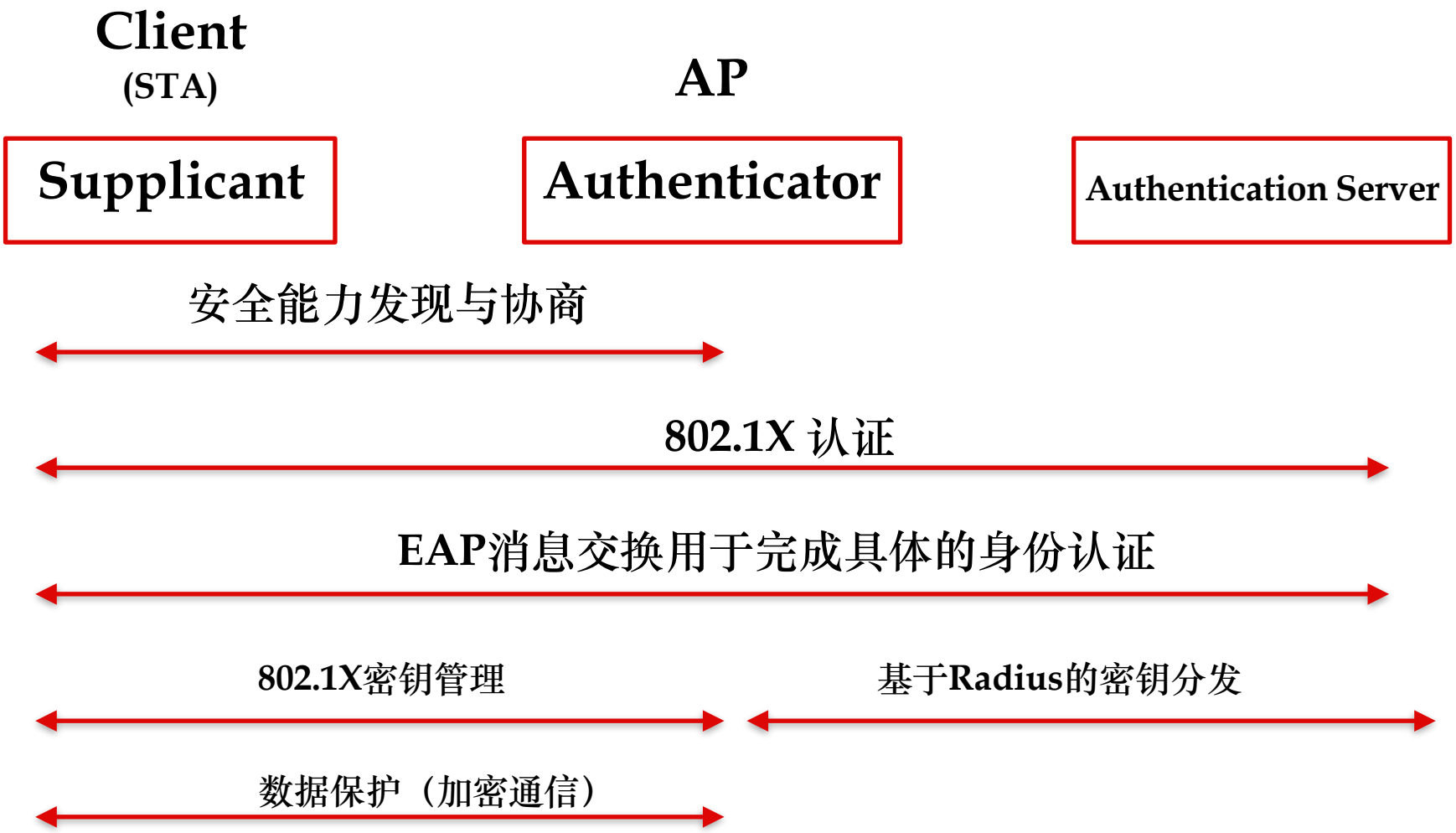
802.11/802.1X状态机



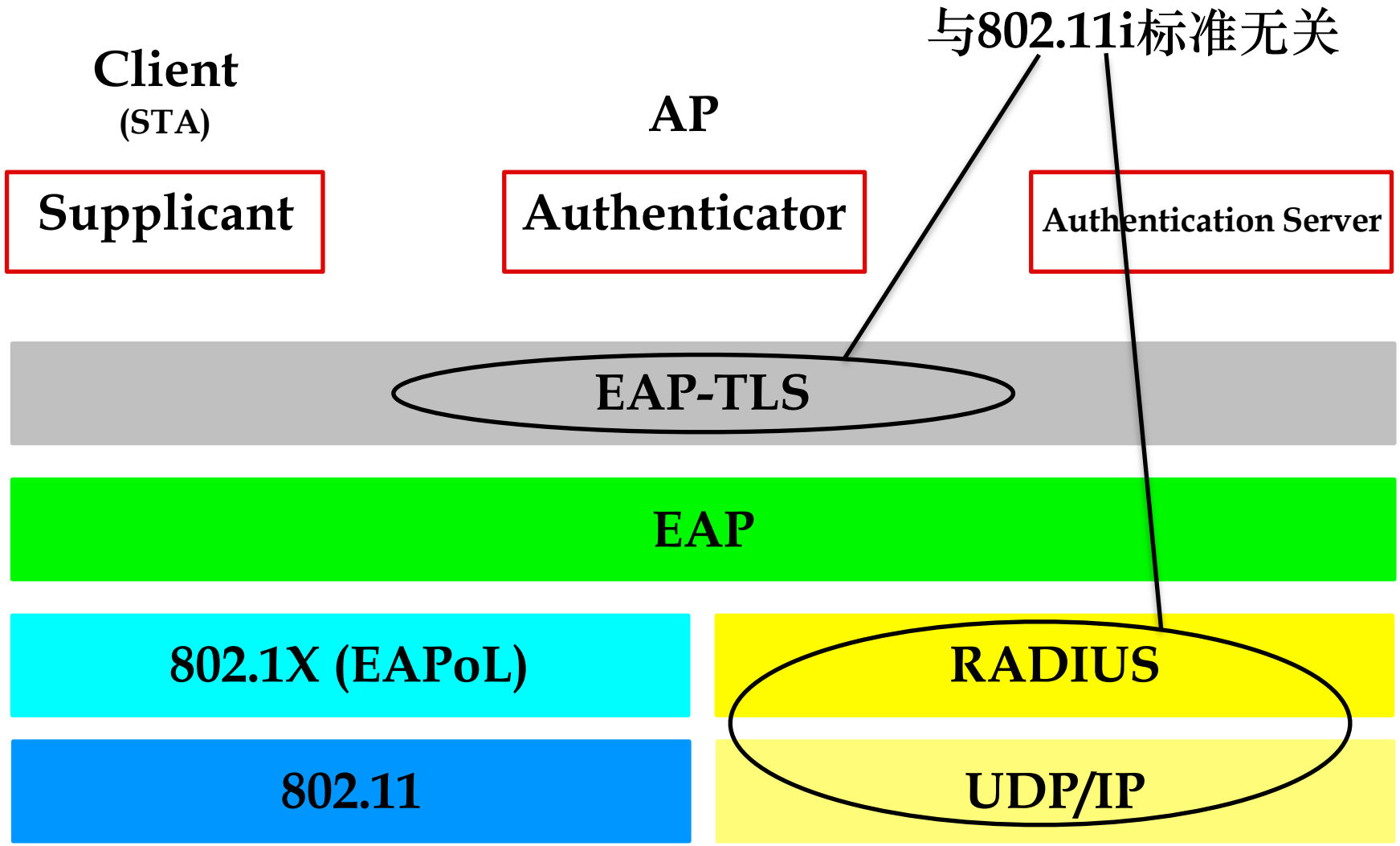
802.1X架构概览


Client
(STA)
AP
Supplicant
Authenticator
Authentication Server
具体认证协议
EAP消息交换用于完成具体的身份认证
802.1: 携带EAP在802.x局域网
Supplicant和Authenticator之间交换
RADIUS/UDP/IP: 携带EAP在Authenticator
和认证服务器之间交换

基于802.1X的802.11企业级认证流程简化



802.11企业级认证架构



EAP - Extensible Authentication Protocol

- 非简单用户名和密码
- 很容易封装到数据链路层协议数据帧
- 提供了一个适用于所有认证方法的通用框架
- 更简单的不同认证方法的互操作和兼容性
- 在802.11协议中提供STA和认证服务器之间的端到端认证
  - —AP在这个过程中扮演认证代理角色
  - —Radius是EAP在IP网络中传输的事实标准

EAP: https://en.wikipedia.org/wiki/Extensible_Authentication_Protocol

EAP的常见可选认证方法

- 按普及程度从高到低排序
  - PEAP
  - EAP-MD5
  - EAP-TTLS
  - EAP-TLS（安全等级最高）
  - LEAP （安全性最差）
  - EAP-FAST

EAP-MD5 (RFC 2284)

- 第一个基于EAP的IEEE RFC标准
- 提供最低等级的基本安全
  - 密码散列值被明文传送
- MD5散列算法存在字典攻击威胁
- 中间人攻击威胁
  - 缺少客户端和服务器的双向身份认证机制
- Windows 2000开始支持，Windows Vista开始禁用该危险认证协议

LEAP

- 轻量级扩展认证协议
  - Lightweight Extensible Authentication Protocol
- Cisco的私有协议
- 已被放弃
  - 易遭受字典攻击
    - http://www.cisco.com/en/US/tech/tk722/tk809/technologies_security_notice09186a00801aa80f.html

EAP-FAST

- Cisco私有协议，用于替换LEAP
- 服务端证书是可选项
- 通过TLS隧道交换认证凭据
- Faked AP可用于诱骗捕获到认证MSCHAPv2报文，然后进行离线字典攻击暴力破解

EAP-TLS

- 安全等级最高
- 部署难度最高
  - 客户端和服务端都需要预先安装和配置SSL证书

PEAP

- 基于EAP的专用于无线局域网的认证协议
- 基于2个已知著名协议
  - EAP
  - TLS
    - 基于SSL 3.0协议规范进行了大量改进
- 2个常见认证方法版本
  - MS-CHAPv2 口令
  - TLS 证书

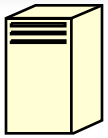
PEAP

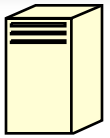
- 所有用户相关敏感认证数据都被加密后在TLS隧道中传输
  - 解密TLS数据需要TLS对称加密主密钥
- 客户端认证失败后连接会被AP丢弃
- TLS对称加密主密钥不在AP上存储，避免Rogue AP解密PEAP保护的终端通信数据
- 客户端通过EAP服务器上部署的服务器证书验证认证服务器真实性
  - 避免有良好安全意识的用户客户端连入虚假伪造AP

PEAP工作流程简图


Client


WAP


EAP Server


Authentication Server

Request Connection → Request Connection →

← Do you support PEAP?

Yes →

← Server PKI certificate & server’s TLS preferences

Certificate verified & client’s TLS preferences or OK →

← TLS settings accepted & TLS finished

PEAP工作流程简图

Response to TLS tunnel established

Request client’s identity

Client’s identity (tells server domain to contact)

Server’s requested EAP authentication type

Client’s requested EAP authentication type or OK

EAP method accepted, request authentication

Client’s UserID and Password → UserID & password

EAP authentication success ← Success

PEAP面临的主要威胁

- AP和认证服务器之间的数据传输没有加密，面临中间人攻击威胁
- TLS隧道仅被用于PEAP认证过程，PEAP认证之后的数据传输没有加密
- PEAP配置缺陷和使用人安全意识低
  - 网络使用人忽略对服务器证书的签名和指纹校验

参考资料

- https://wifipineapple.com/index.php
- Introduction to WiFi Security and Aircrack-ng by Thomas D'Otreppe
- WikiDevi
- http://www.aircrack-ng.org/doku.php

延伸阅读

- wireshark年度开发者大会