(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 101222331 B
(45)授权公告日 2013.04.24

(21)申请号 200710079799.8
(22)申请日 2007.03.09
(66)本国优先权数据
200710063346.6 2007.01.09 CN
(73)专利权人 华为技术有限公司
地址 518129 广东省深圳市龙岗区坂田华为总部办公楼
(72)发明人 黄浩 张军平 莫良耀 张慧敏 冯丹凤
(74)专利代理机构 北京凯特来知识产权代理有限公司 11260
代理人 郑立明
(51)Int.Cl.
H04L 9/32(2006.01)
H04L 29/06(2006.01)
H04L 12/28(2006.01)

(56)对比文件
CN 1863090 A,2006.11.15, 摘要,权利要求1-14,说明书第17页第17行-第20页第13行.
CN 1558584 A,2004.12.29, 摘要、说明书第10页第15行-第13页第11行,图3-4.
CN 1630269 A,2005.06.22, 全文.
CN 1691603 A,2005.11.02, 全文.

审查员 白芳芳

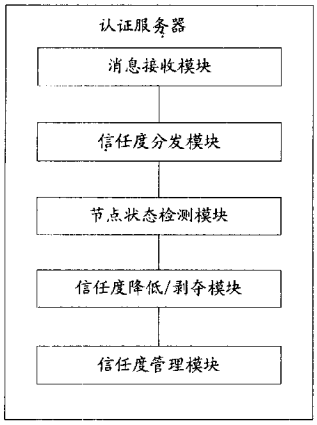
权利要求书2页 说明书11页 附图4页

(54)发明名称

一种认证服务器及网状网中双向认证的方法及系统

(57)摘要

本发明涉及通信技术领域中一种认证服务器及网状网中双向认证的方法及系统。所述认证服务器包括:信任度分发模块,用于根据接收到的请求认证的节点信息确定所述节点的安全等级,并在认证成功后产生信任度时戳,签名后连同签名算法标识下发给所述请求认证的节点;节点状态检测模块,用于检测各节点是否受到攻击;信任度降低/剥夺模块,用于在接收到断开消息或节点受到攻击后降低/剥夺节点的信任度;信任度管理模块,用于存储Mesh网络中所有认证成功的节点的用户身份信息以及AS认可的信任度、签名和时间戳信息。本发明实现了Mesh网络中各节点可以基于信任度来选择关联节点,并通过验证关联节点反馈的信任度签名一次实现双向认证。



CN 101222331 B

1. 一种网状网中双向认证的方法，其特征在于，包括：

网状网 Mesh 中的节点基于信任度选择关联节点，所述关联节点对节点进行认证，并将本身的信任度签名反馈给所述节点，所述节点根据所述信任度签名对所述关联节点进行认证；所述信任度由认证服务器 AS 下发及管理。

2. 如权利要求 1 所述的方法，其特征在于，所述 AS 采用双向认证下发信任度的方法包括：

Mesh 中的节点发送探测请求给邻居节点，并根据各邻居节点反馈的消息中携带的信任度选择与 AS 交互的中间认证者；

所述节点与所述选择的中间认证者进行关联，根据中间认证者关联过程中下发的信任度签名验证所述中间认证者的合法性，并获得所述中间认证者的认证后，将所述节点接入 AS；

AS 对所述请求接入的节点进行认证，下发信任度给所述请求接入的节点。

3. 如权利要求 2 所述的方法，其特征在于，所述 Mesh 中各节点选择中间认证者的方法包括：

在所有反馈消息的邻居节点中选择信任度最大的或满足事先约定信任度范围的节点作为中间认证者。

4. 如权利要求 2 所述的方法，其特征在于，所述 Mesh 中的节点与所述选择的中间认证者进行关联的方法具体包括：

Mesh 中的节点向选择的中间认证者发送认证请求；

所述中间认证者将自己支持的安全参数、AS 使用的公钥和签名算法、对信任度的签名反馈给所述节点；

所述节点验证所述中间认证者的有效性，并与所述中间认证者进行开放式或者预共享密钥认证；

所述中间认证者返回认证结果；

所述节点根据中间认证者支持的安全参数发送自己需要的安全参数组合给中间认证者；

所述中间认证者返回关联成功或失败结果给所述节点。

5. 如权利要求 4 所述的方法，其特征在于，所述节点验证中间认证者合法性的方法包括：

各节点检查是否支持签名算法，并且整个 Mesh 的签名算法是否一致，如果不支持认证者反馈的签名算法，或整个 Mesh 的签名算法不一致，则认为认证者无效，否则有效。

6. 如权利要求 2 所述的方法，其特征在于，所述 AS 对请求接入的节点进行认证的方法包括：

节点验证 AS 身份的合法性；

验证通过之后，所述节点将自己的信息提交给所述 AS 来进行认证；

AS 由所述节点的信息确定所述节点的安全等级，并产生信任度时戳，签名之后连同 AS 使用的签名算法标识发送给所述节点；

所述节点验证 AS 的合法性，在所述 AS 合法情况下所述节点获得信任度。

7. 如权利要求 2 所述的方法，其特征在于，所述 AS 对请求接入的节点进行认证的方法

包括：

节点验证 AS 身份的合法性；

验证通过之后，所述节点将自己的信息提交给所述 AS 来进行认证；

所述 AS 对所述节点进行认证后，向所述节点发送认证结束消息；

所述节点接收到所述认证结束消息后，发送响应消息，同时请求 AS 颁发认证签名；

所述 AS 接收到所述请求后，产生信任度时戳、签名之后连同 AS 使用的签名算法标识，承载于认证成功或失败消息中发送给所述节点；

所述节点验证 AS 的合法性，在所述 AS 合法情况下所述节点获得信任度。

8. 如权利要求 1 所述的方法，其特征在于，所述 AS 对信任度的管理方法包括：

在某个节点想要离开 Mesh 或 AS 检测到某个节点受到攻击时，降低或剥夺所述节点的信任度。

9. 如权利要求 1 所述的方法，其特征在于，所述 AS 采用动态目录下发和管理信任度，所述动态目录包含 Mesh 中认证成功的所有节点的用户身份信息以及 AS 认可的信任度、签名和时间戳信息。

10. 一种网状网网络系统，其特征在于，包括：至少一个 AS 及多个节点，所述 AS 用于根据接收到的请求认证的节点的信息对所述节点进行认证，以确定节点的安全等级，并在认证成功后产生信任度时戳，签名后连同签名算法标识下发给所述请求认证的节点；

Mesh 中的节点，基于所述 AS 下发的信任度选择关联节点进行认证，并根据所述关联节点反馈的信任度签名对所述关联节点身份进行验证。

11. 如权利要求 10 所述的系统，其特征在于，所述 AS 进一步包括：

信任度分发模块，用于根据接收到的请求认证的节点信息确定所述节点的安全等级，并在认证成功后产生信任度时戳，签名后连同签名算法标识下发给所述请求认证的节点；

信任度管理模块，用于存储 Mesh 中所有认证成功的节点的用户身份信息以及 AS 认可的信任度、签名和时间戳信息。

12. 如权利要求 10 所述的系统，其特征在于，所述 Mesh 中的节点上设置有：

认证者身份验证模块，用于根据所述关联节点反馈的信任度签名对所述关联节点身份进行验证。

一种认证服务器及网状网中双向认证的方法及系统

技术领域

[0001] 本发明涉及通信技术领域，尤其涉及一种认证服务器及网状网中双向认证的方法及系统。

背景技术

[0002] 无线 Mesh（无线网状网）网络也称为“多跳”网络，它是一种与传统无线网络完全不同的新型无线网络技术。传统的无线网络必须首先访问集中的 AP(Access point，接入点）才能进行无线连接。这样，即使两个 802.11b 的节点互相挨着，它们也必须通过接入点才能通信。而在无线 Mesh 网络中，任何无线设备节点都可以同时作为 AP 或路由器，每个节点都可以发送和接收数据，都可以与一个或者多个对等节点进行直接通信。

[0003] 无线 Mesh 网与传统 BSS（基本业务组）的角色划分不同。在 BSS 域中只包括 STA(station，站点或终端）、AP 和 AS(Authentication Server，认证服务器）三种角色。认证时 AS 强制 STA 充当请求者的角色，AP 充当认证者的角色。现有的 802.11i 体系结构中只定义了这三种角色：STA、AP 和 AS，以及执行角色功能的机制：(a) 安全的 AS-STA 对等认证；(b) 安全的 AS-AP 频道；(c)AS 访问控制列表。

[0004] 而在 Mesh 中的主要功能包括：

[0005] (1) 传输 (Transport)：与邻居节点之间传送非 Mesh 的应用数据。只发送、接收，而不转发。

[0006] (2) 内部路由 (Internal Routing)：Mesh 设备之间建立转发路径。

[0007] (3) 外部路由 (External Routing)：与 Mesh 外部设备建立转发路径。

[0008] (4) 转发 (Forwarding)：发送收到的其他设备产生的帧。

[0009] (5) 接入 (Access point)：允许一个 802.11 的 STA 通过 AP 连到 Mesh 中。

[0010] 基本层次结构如图 1 所示，Mesh 中各种设备所完成的功能是不一样的，按照上述功能划分的角色为：

[0011] MAP(Mesh 接入点）：传输 + 内部路由 + 转发 + 接入；

[0012] MPP(Mesh 与外网的网关，图中未示出）：传输 + 内部路由 + 转发 + 外部路由；

[0013] MP(Mesh 节点）：传输 + 内部路由 + 转发；

[0014] LWMP（轻量级 Mesh 节点，图中未示出）：传输；

[0015] STA：数据源。

[0016] 可见 Mesh 中的角色划分与 BSS 域有很大差别。而目前 802.11s 中关于角色的定义与执行中只对 STA 和 MAP 有较全面的定义，而 LWMP、MP、MPP 以及 AS 在 Mesh 中的角色则没有定义任何机制来执行。要使用 802.1x 来完成认证则存在如下问题：

[0017] (1) 协商中每个 MP 可能有三种角色要完成：Supplicant（请求者），Authenticator（认证者），和 Authentication Server(AS，认证服务器），因此，存在如何协商并确定双方联合的安全角色和安全策略问题。

[0018] (2) 需要一个管理行动来给各个设备分配角色和选择一个关联的 MP 来安全地完

成认证等问题。

[0019] 目前，关于角色协商问题提案的主要思路是：在 RSN IE（健壮安全网络的信息元素）中通过一个可达性的元素“AS Reachability”来做为衡量协商角色的依据。具体实现如图 2 所示包括：

[0020] 首先在发现阶段，需要关联的节点通过 Beacon（信标）或者探测响应来发送对于 AS 的可达性。然后通过可达性来协商认证的角色：

[0021] (1) 如果双方只有一个对 AS 可达，那么可达这方就是认证者，而另一方就是请求者。

[0022] (2) 如果双方都可达或都不可达，则选择较大 MAC 地址的一方作为认证者，而另一方作为请求者。

[0023] 角色确定后用 802.1x 的方式来完成认证，接入 Mesh 网络。

[0024] 上述认证方式存在如下缺点：

[0025] 认证中在只有一方可达 AS 时的情况类似于普通的 BSS，但仍存在恶意 MP 攻击时切换认证点的问题。而且与 BSS 非对称结构所不同的是，原有一轮 802.1x 认证只能完成对请求者的认证，是一个单向认证，现在 Mesh 中 MP 的层次结构是对称平等的，要完成双向认证需两轮 802.1x 认证，不但浪费时间，而且大多数情况下第二轮认证是已接入 Mesh 的 MP 所并不想要的。

[0026] 另外，对于 BSS 认证而言不存在中间道路选择的问题，AP 直接连到了 AS。而对于 Mesh 而言相对复杂，认证中间节点的选择也要考虑安全，必须建立一条最安全（或者是用户最想要的安全）的认证路径。但存在如下情况会使得可达性不可靠。例如某个中间 MP 的路由性能是最好的，此时无论想要加入 Mesh 的 MP 怎么选择认证点认证数据都会交到该 MP 来转发，但是如果此中间 MP 遭到了攻击被破坏，那么攻击者可能大量获取用户认证信息，并根据 EAP-Success（扩展认证协议成功消息）消息是明文的特点伪造或改为失败消息返回给认证者（没有使用 EAP-TLS），认证结果就是失败的。认证者不知道哪个地方出错了只是反复地重新切换认证点，这时可达性就不可靠了。

发明内容

[0027] 本发明实施例提供一种认证服务器及 Mesh 网络中双向认证的方法及系统，解决了现有技术 Mesh 网络中间节点不可靠及单项认证带来的弊端。

[0028] 本发明实施例是通过以下技术方案实现的：

[0029] 本发明实施例提供一种网状网中双向认证的方法，包括：

[0030] 网状网 Mesh 中的节点基于信任度选择关联节点，所述关联节点对节点进行认证，并将本身的信任度签名后反馈给所述节点，所述节点根据所述信任度签名对所述关联节点进行认证。

[0031] 可选地，所述 AS 对信任度的管理方法包括：

[0032] 在某个节点想要离开 Mesh 或 AS 检测到某个节点受到攻击时，降低或剥夺所述节点的信任度。

[0033] 可选地，所述 AS 采用动态目录下发和管理信任度，所述动态目录包含 Mesh 中认证成功的所有节点的用户身份信息以及 AS 认可的信任度、签名和时间戳信息。

[0034] 本发明实施例提供一种网状网网络系统，包括：至少一个 AS 及多个节点，所述 AS 用于根据接收到的请求认证的节点的信息对所述节点进行认证，以确定节点的安全等级，并在认证成功后产生信任度时戳，签名后连同签名算法标识下发给所述请求认证的节点；

[0035] 所述 Mesh 中的节点，基于所述 AS 下发的信任度选择关联节点进行认证，并根据所述关联节点反馈的信任度签名对所述关联节点身份进行验证。

[0036] 本发明实施例提供一种认证服务器，包括：

[0037] 信任度分发模块，用于根据接收到的请求认证的节点信息确定所述节点的安全等级，并在认证成功后产生信任度时戳，签名后连同签名算法标识下发给所述请求认证的节点；

[0038] 信任度管理模块，用于存储 Mesh 中所有认证成功的节点的用户身份信息以及认证服务器认可的信任度、签名和时间戳信息。

[0039] 由上述本发明提供的技术方案可以看出，本发明实施例提出了一种节点最初接入 Mesh 网络时根据信任度来选择接入点，然后双方协商认证角色，并通过验证认证者的信任度签名校验认证者的合法身份，一次实现原有 802.1x 需要两轮才能完成的双向认证。

[0040] 同时，本发明实施例实现了 AS 对请求者身份合法认证之后，对信任度进行管理，对用户和设备信任度进行了简单的分类，然后分发相应的签名消息，实现集中式的认证管理。任意节点接入认证时也可选择最好的认证道路来完成认证而与路由机制无关，不会随着路由的改变而改变认证的安全性。当检测到攻击时也可降低某些点的信任度，从而减少受到攻击的风险。

附图说明

[0041] 图 1 为现有技术 Mesh 网络基本层级结构示意图；

[0042] 图 2 为现有技术认证角色选择方法示意图；

[0043] 图 3 为 RSN IE 格式结构图；

[0044] 图 4 为本发明所述方法接入认证过程关联阶段一种实施例流程图；

[0045] 图 5 为本发明所述方法 AS 进行认证下发信任度一种实施例流程图；

[0046] 图 6 为本发明所述方法 AS 进行认证下发信任度另一种实施例流程图；

[0047] 图 7 为本发明所述认证服务器一种实施例模块示意图。

具体实施方式

[0048] 本发明实施例基于信任度来选择关联节点，并且通过验证信任度的签名来实现请求者和认证者之间双向认证的目的。由 AS 对信任度进行统一的分发管理。请求者由 AS 认证通过后下发信任度和相关签名信息，用于作为接入 Mesh 网络的依据。当某个节点离开 Mesh 网络或受到攻击时，AS 负责信任度的剥夺或降低。

[0049] 下面的实施例以 Mesh 网络中的节点接入认证过程为例对双向认证过程进行说明，具体实现过程如下所述：

[0050] 步骤 1，请求者选择与 AS 交互的中间节点，即选择认证接入节点，作为接入 AS 的中间认证者：

[0051] 当 MP、MAP 或 STA 最初想要接入 Mesh 网络时，它会向周围所有的邻居发送探测请

求，而周围的邻居则以 beacon 或者探测响应的方式反馈携带信任度的消息。

[0052] 当所述节点 (MP、MAP 或 STA) 收到周围所有邻居节点发过来的 Beacon 或者探测响应中携带的信任度后，会根据自己的策略来选择最想要关联的节点，如可以选择信任度最大的或者满足 MPP 全局策略的节点进行 802.1x 认证。例如某个 Mesh 网络已经事先约定某些范围的信任度来完成接入，其他范围来转发或其他功能，这时选择所述认证接入节点需要满足所述预先约定的策略。

[0053] 所述信任度由 AS 进行统一分发管理，可以通过对当前 RSN IE 的格式进行改动来实现，所述当前 RSN IE 格式如图 3 所示，可以对其中的能力信息 Capabilities 进行修改，其中修改前后的 Capabilities 信息如下表所示：

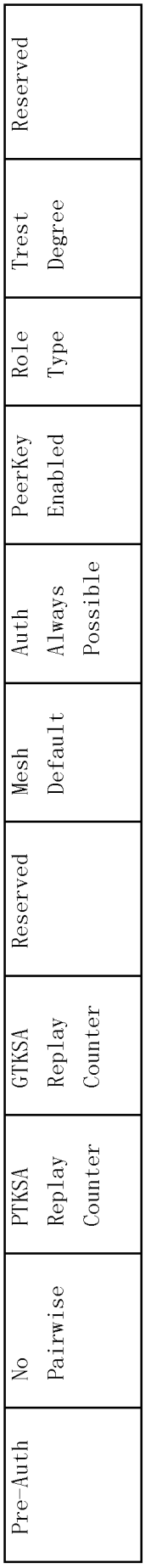
[0054] 修改前：

[0055]

Pre-Auth	No Pairwise	PTKSA Replay Counter	GTKSA Replay Counter	Reserved	Mesh Default	Auth Always Possible	PeerKey Enabled

[0056] 修改后：

[0057]


Pre-Auth
No Pairwise
PTKSA Replay Counter
GTKSA Replay Counter
Reserved
Mesh Default
Auth Always Possible
PeerKey Enabled
Role Type
Trest Degree
Reserved

[0058] 本发明实施例增加的两个字段包括：Role Type 和 Trust Degree。其中 RoleType 表示信任度角色类型，占有一个 bit 位。当取值为 0 时，表示用户类型信任度，如 STA 信任度；取值为 1 时表示 Mesh 设备节点，如 MP 或 MAP 信任度，此时保证大多不会通过 STA 来作为认证者，而只会通过 Mesh 设备节点作为认证者，例如 MP、MAP 等来作为认证者，完成认证中间人的角色。

[0059] 其中 Trust Degree 表示信任度，占有四个 bit 位，认证通过的用户或设备都会获得由 AS 下发的信任度。默认为 0 时，表示未通过认证或者信任级别最低。

[0060] 每个节点根据已有的信任度信息来选择相对最安全的认证接入节点，那么任意节点的认证路径都是最可靠的。对于数据传输则可以根据认证协商的密钥加密，并可选择路由最好的路径来转发。

[0061] 步骤 2，中间认证者选择完成后，请求者 (MP、MAP 或 STA) 与所述选择的中间认证者进行关联；

[0062] 具体流程如图 4 所示，包括如下步骤：

[0063] (1) 首先请求者向选定的中间认证者发送认证请求；

[0064] (2) 所述中间认证者将自己支持的安全参数、AS 使用的公钥和签名算法、对信任度的签名反馈给所述请求者。

[0065] 请求者检查是否支持签名算法，并且整个 Mesh 网络的签名算法是否一致，如果不支持认证者反馈的签名算法，或整个 Mesh 网络的签名算法不一致，则认为认证者无效，否则有效。然后根据 AS 的信息来验证信任度签名是否合法、有效，具体验证方法见后续介绍；所述签名算法使用一个标识位来表示，不同的数值表示不同的算法，例如：设置为 1 时，表示默认的公钥算法 RSA（基于整数因子分解的公钥系统）；设置为 2 时表示 DSS（数字签名标准）；设置为 3 时，表示 WAPI（无线局域网鉴别与保密基础结构）的 ECC（椭圆曲线公钥系统）；设置为 0 时，表示其它支持可扩充的签名算法。之后所有请求者向 AS 发送的认证相关消息都可以使用 AS 的公钥来加密，以及后来对称密钥的协商均可保证对认证者的透明。

[0066] (3) 请求者与所述中间认证者之间进行开放式或者预共享密钥认证；

[0067] 所述开放式系统认证属于非基于密码学认证，也称为“零认证”，请求接入的节点只要使用一个空字符串作为 SSID 进行响应，即可与中间认证者建立关联。

[0068] 所述预共享密钥认证则是基于是否具有共享密钥的“请求 / 响应”机制。请求接入的节点首先向中间认证者发送认证请求，然后中间认证者产生一个随机数响应给所述请求认证的节点，请求认证的节点然后通过共享的密钥加密该随机数响应给中间认证者，所述中间认证者解密比较后，最后发送确认消息。

[0069] (4) 中间认证者返回认证结果；

[0070] (5) 请求者根据中间认证者支持的安全参数发送自己需要的安全参数组合；

[0071] (6) 认证者返回关联成功或失败结果。

[0072] 步骤 3，请求者与中间认证者关联之后，将请求者接入 AS，由 AS 完成对请求者的 802.1x 认证及分发信任度；

[0073] 请求者根据现有 802.11 的标准进行基于端口的 802.1x 接入控制认证。所述认证相关消息在请求者和认证者之间通过 LAN 传递，即使用 EAPOL（基于 LAN 的 EAP) 技术，所述 EAPOL 是支持多认证机制的通用协议，可以采用的 EAP 类型主要包括：EAP-TLS（基于传输层

安全的扩展认证协议）、EAP-TTLS（基于隧道传输层安全的扩展认证协议）、PEAP（保护的扩展认证协议）、LEAP（基于轻量级的扩展身份认证协议）、EAP-MD5（基于消息摘要算法的扩展认证协议）以及结合移动通信的 EAP-SIM（基于用户身份模块的扩展认证协议）。

[0074] 其中 EAP-TLS 可以为客户端和服务器之间的数据交换分配会话 ID、选择合适的完整性保护加密机制、分配动态会话密钥、并能够有效的保护 802.1x 认证过程中交互消息的安全。其对于中间认证者是透明的，即认证接入点看不到任何认证的内容。因此可以采用 EAP-TLS 来完成信任度的下发和 AS 对请求者的认证。

[0075] 为了便于 Mesh 的部署时降低终端或设备的开销，可以只令 AS 有一个表明身份的证书和公钥能够提供 AS 身份的认证即可，而其他节点可以没有 X.509 证书。

[0076] 其 AS 下发信任度和认证流程第一种实施例如图 5 所示，包括如下步骤：

[0077] 1) 请求者在接收到中间认证者（图中未示出，因为之后的交互对于其而言是透明的所以不用考虑）的身份验证请求后，发送身份相关响应，由所述中间认证者发送给 AS；

[0078] 2)AS 接收到所述响应后发送 TLS 开始请求给所述请求者；

[0079] 3) 请求者反馈 TLS 开始响应给 AS；

[0080] 4)AS 接收到所述响应后将自己的身份信息，包括证书和公钥一同发送给请求者；

[0081] 5) 请求者验证 AS 身份的合法性，验证通过之后，请求者正式将自己的用户、密码等信息提交给 AS 来进行认证，请求者能够完全识别 AS 的真实性；

[0082] 所述验证方式可根据具体的 TLS 的协商验证方法，如果是用户密码 MD5 的方法，则只需比较一下就可以；如果是公钥证书验证，则双方需要证书的交互验证和对方公钥加密。

[0083] 6)AS 可以由请求者节点的安全能力信息、用户服务级别，或者通过集中访问控制服务器的策略来决定请求者的安全等级。认证成功将产生信任度时戳 TimeStamp(AS 可以保证时间戳的唯一性)，签名之后连同 AS 使用的签名算法标识发送给请求者。

[0084] 所述 AS 下发的签名包含如下参数：

[0085] RoleType：角色类型，取值为 0 时，表示节点为用户类型，如 STA，取值为 1 时表示节点为 Mesh 网络设备类型，如 MP 或 MAP；

[0086] TrustDegree：表示用户或设备的信任度；

[0087] ID_{AS}、$ID_{Suplicant}$：AS、请求者的身份标识符，例如各自的 MAC 地址；

[0088] SK_{AS}、PK_{AS}：AS 的公钥、私钥；

[0089] TimeStamp：认证成功产生的时间戳，具有唯一性。

[0090] 所述的签名算法如下：

[0091] $$S = Sig_{SK_{AS}}(RoleType \| TrustDegree \| ID_{AS} \| ID_{Suplicant} \| TimeStamp);$$

[0092] 将所述签名算法连同信任度 TrustDegree 一同发送给请求者；

[0093] 7) 请求者检查所述签名算法是否与认证接入点使用的算法相同，即是否与中间认证者使用的签名算法相同，在确认统一性后，再验证信任度签名，并发送验证反馈响应消息给 AS。

[0094] 首先判断信任度合法性；

[0095] 包括：判断 RoleType 的值是 0 还是 1，为 1 表示该节点为一个设备。这样的作用是不会把用户作为认证者，认证时根据自己选择的方法判断是否合法，因为会有某些节点只要求能被设备认证接入，而不允许终端作为认证者；还包括：判断 ID_{AS} 及 $ID_{Suplicant}$；

[0096] 再判断信任度有效性；

[0097] 主要根据校验所得的 TrustDegree 和 TimeStamp 判断有效性。其中 TrustDegree 主要用于判断是否接受新信任度值取代原有信任度值。由于每次关联都要进行一次 802.1x 认证，但是 Mesh 中有多关联的情况，已经接入的两个节点之间关联时没有必要再获取新的信任度而只是判断一下相互的信任度合法性即可。

[0098] 而 TimeStamp 判断主要用于校验认证者的信任度，可以抵抗信任度的重放攻击。如果检测到 TimeStamp 重用，则可认定认证者是个伪造者；如果没有重复，则可通过与 AS 的交互来判断信任度的唯一指定性，在认证阶段 AS 同时记录了与某个时间戳相关联的其他所有节点。

[0099] 8)AS 发送认证成功或失败消息给请求接入的节点。

[0100] 通过这样的过程一个设备 (MP 或 MAP) 或用户 (STA) 就获得了 AS 的认可，使用所述信任度可以接入 Mesh 网络。其中通过认证的设备可以作为新的认证者来认证其他节点。

[0101] 其 AS 下发信任度和认证流程第二种实施例如图 6 所示，包括如下步骤：

[0102] (1) 请求者在接收到中间认证者（图中未示出，因为之后的交互对于其而言是透明的所以不用考虑）的身份验证请求后，发送身份相关响应，其中包括认证者的时间戳和用 AS 公钥加密的随机数，由所述中间认证者发送给 AS；

[0103] (2)AS 接收到所述响应后，检测所述请求者非重放（非被盗用）之后，解开所述随机数，并签名后发送 TLS 开始请求给所述请求者；

[0104] (3) 请求者反馈 TLS 开始响应给 AS；

[0105] (4)AS 接收到所述响应后将自己的身份信息，包括证书和公钥一同发送给所述请求者；

[0106] (5) 请求者验证 AS 身份的合法性，验证通过之后，请求者正式将自己的用户、密码等信息提交给 AS 来进行认证，请求者能够完全识别 AS 的真实性；

[0107] (6)AS 验证请求者节点的身份信息，之后发送认证结束消息给所述请求者。

[0108] 不局限于 EAP-TLS，还包括其他使用证书进行认证的协议例如 EAP-PEAP、EAP-TTLS 等（步骤总数也可不一样）。

[0109] 所述具体相互验证方式可根据具体采用的协商验证方法，如果是用户密码 MD5 的方法，则只需比较一下就可以；如果是公钥证书验证，则双方需要证书的交互验证和对方公钥加密。

[0110] (7) 请求者收到所述认证结束消息之后，反馈响应消息给 AS，并在需要获取所述信任度时戳时，发送请求 AS 颁发签名的标识，以成为新的合法认证者。

[0111] AS 将产生信任度时戳 TimeStamp(AS 可以保证时间戳的唯一性)，并由请求者节点的安全能力信息、用户服务级别，或者通过集中访问控制服务器的策略来决定请求者的安全等级，产生签名，签名方法与实施例 1 相同。

[0112] (8)AS 发送认证成功（或失败消息）消息连同签名 S、签名算法标识和信任度 TrustDegree 一同发送给所述请求者给请求者。

[0113] 请求者检查所述签名算法是否与认证接入点使用的算法相同，即是否与中间认证者使用的签名算法相同，在确认统一性后，再验证信任度签名，并验证后判断是否接受。

[0114] 判断方法与实施例 1 相同。

[0115] 步骤4,AS在某个节点想要离开Mesh网络或AS在检测到某个节点收到攻击后,降低或剥夺所述节点的信任度;

[0116] AS降低或剥夺信任度包括如下两种情况:

[0117] (1)节点主动想要离开Mesh网络时:节点在结束关联之前以广播的方式告知所有与之关联的其他节点和单播发送给AS,消息都使用统一的GTK(组临时密钥)或与AS的PMK(协商好的对称主密钥)加密,保证了消息源的可靠。这时主要在deassociation帧(断开关联的帧)之前发送两个消息:给所有已关联的节点用GTK广播加密发送断开消息;给AS用PMK单播发送断开消息。之后,AS删除与之相关时间戳的信任度和其他关联节点的信息,表示这个信任度不再可用了。其他节点也可记录时间戳作废信息,以保证信任度的后向安全。

[0118] (2)当AS检测到某些节点受到恶意攻击时:由AS强制剥夺其信任度,AS通过单播发送强制断开、信任度时间戳作废消息给与所述受攻击的节点相关联的所有节点,这时主要在deassociation帧之前发送一个消息:给所有已与所述受攻击节点关联的节点用各自PMK广播发送断开消息。接收到所述消息的节点在验证AS的身份可靠性之后,主动断开与所述受攻击节点的关联,并记录时间戳作废消息。同时AS通知受攻击节点如想继续接入,必须重配置之后再次认证。此时AS也可降低其信任度,减少受到攻击的风险,保证整个网络接入的安全。

[0119] 所述信任度由AS进行统一分发和管理。AS对信任度的管理和发送使用动态目录机制,在Mesh网络运行中随着节点的加入、离开或强制离线等情况的发生,AS会不断被更新保存的相关信息。所述AS中保存了Mesh网络中认证成功的所有节点的用户身份信息以及AS认可的信任度、签名和时间戳信息,所述动态目录可以使用一个三元组来定义:TD_Mesh ＜M,td_r,TimeStamp＞其中:

[0120] M中包含已成功认证的所有节点;

[0121] td_r中标记所有关联节点之间的信任关系;

[0122] timestamp包含所有已分发信任度的时间戳,所述时间戳是唯一的,而信任度是可以重用的。时间戳和信任度签名是一一对应的关系。

[0123] 在AS的策略库中可以采用如下格式的策略定义语言:

[0124] {RolesID}IF{conditions}THEN{actions}

[0125] 其中对于M策略库可以基于角色进行管理,分为两类:

[0126] (1)设备角色的信任度:仅仅表示该设备的安全等级或被攻击的难易,主要为MP或者MAP的信任度;

[0127] (2)用户角色的信任度:不但可以表示该节点的安全等级,同时可以将用户服务分级,用来调整基于级别的公平性或进行用户的访问控制等,主要为STA的信任度。

[0128] 实际应用中用户可根据实际需要增加更多的功能来延伸该策略库,包括相应基于用户的负载均衡等都可以添加到其中,需要对condition(条件字段)和action(执行字段)进行不断扩充。

[0129] 而对于安全策略的定义主要依赖于实际网络情况以及想要提供的服务类型等参数来确定,策略的粒度需进一步细化以满足不同用户级别的要求。

[0130] 上述实施例以接入认证为例进行说明。实际操作中当任意两个MP想要建立关联

的时候，也可以根据信任度的大小来选择请求者和认证者。小的一方为请求者，大的一方为认证者。如果双方都有信任度，那么表示两者只是想要建立关联而与接入认证无关，这时可以通过 802.1x 的方法来建立关联，而不接受新发出的信任度，或者仅仅两者交互信任度及签名就可关联，那么关联可以更快的建立，这样交互的过程能够更快的完成，其主要包括下列步骤：

[0131] (1) 请求者将自己的信任度签名信息交给认证者来验证；

[0132] (2) 认证者验证成功之后，用自己的信任度签名响应给请求者；

[0133] (3) 请求者验证成功之后，返回成功或失败消息；

[0134] (4) 利用原有 802.11i 的机制来进行密钥管理。

[0135] 如果双方都没有信任度，则需要本地数据库来完成认证，结果可能失败，则需要等待其他能够完成认证的节点来接入 Mesh 网络。

[0136] 本发明实施例提供一种 Mesh 网络中双向认证系统，所述系统包括至少一个 AS 及多个节点，所述节点包括：MP、MAP 和 STA。所述 AS 用于为 Mesh 网络中各节点分发信任度，并对所述信任度进行统一管理，其一种实施例模块示意图如图 7 所示，包括：

[0137] 消息接收模块，用于接收 Mesh 网络中节点发送的消息，包括：认证请求消息、断开消息等；

[0138] 信任度分发模块，用于根据接收到的请求认证的节点的安全能力信息、用户服务级别、或通过集中访问控制器的策略确定所述节点的安全等级，分发认证消息给请求者。所述认证消息中包括：信任度时戳，认证者签名、使用的签名算法等。所述请求者包括：MP、MAP 或 STA；

[0139] 节点状态检测模块，用于检测 Mesh 网络中节点 (MP、MAP 或 STA) 是否受到攻击；

[0140] 信任度降低 / 剥夺模块，用于在接收到断开消息或检测到节点受到攻击后降低 / 剥夺节点的信任度；

[0141] 信任度管理模块，用于存储 Mesh 网络中所有认证成功的节点的用户身份信息以及系统中认可的信任度、签名和时间戳信息。

[0142] 所述 Mesh 网络中的多个节点基于所述 AS 分发的信任度进行关联认证，其进一步包括如下模块：

[0143] 认证者身份验证模块，用于根据接收到认证消息中的认证者签名验证认证者身份是否合法。

[0144] 所述Mesh网络中的各设备节点包括MP或MAP，在获得认证后可以作为认证者来认证其他节点。

[0145] 综上所述，本发明实施例提出了一种节点最初接入 Mesh 网络时根据邻居信任度来选择接入点，然后双方协商认证角色，并通过验证认证者的信任度签名校验认证者的合法身份，一次实现原有 802.1x 需要两轮才能完成的双向认证。

[0146] 以上所述，仅为本发明较佳的具体实施方式，但本发明的保护范围并不局限于此，任何熟悉本技术领域的技术人员在本发明揭露的技术范围内，可轻易想到的变化或替换，都应涵盖在本发明的保护范围之内。因此，本发明的保护范围应该以权利要求的保护范围为准。

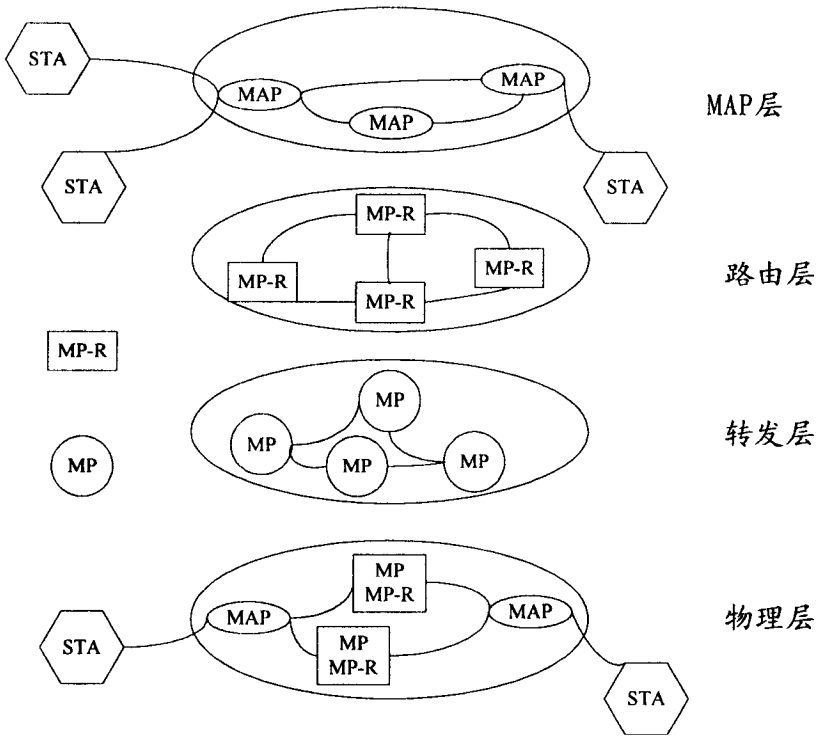

STA
MAP
MAP
MAP
STA
STA
MAP层
MP-R
MP-R
MP-R
MP-R
路由层
MP-R
MP
MP
MP
MP
转发层
MP
MP-R
MAP
MAP
STA
MP
MP-R
STA
物理层

图 1

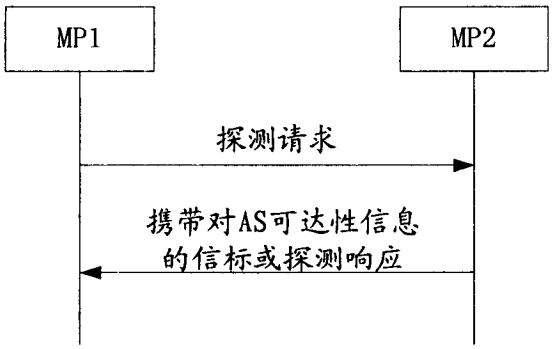

MP1
MP2
探测请求
携带对AS可达性信息
的信标或探测响应

图 2

Element ID	Length	Version
Group Key Ciphersuite Selector		
Pairwise Ciphersuite Count		Pairwise Ciphersuite List
Pairwise Ciphersuite List		AKM Count
AKM List		
Capabilities		PMK ID Count
PMK ID List		

图 3

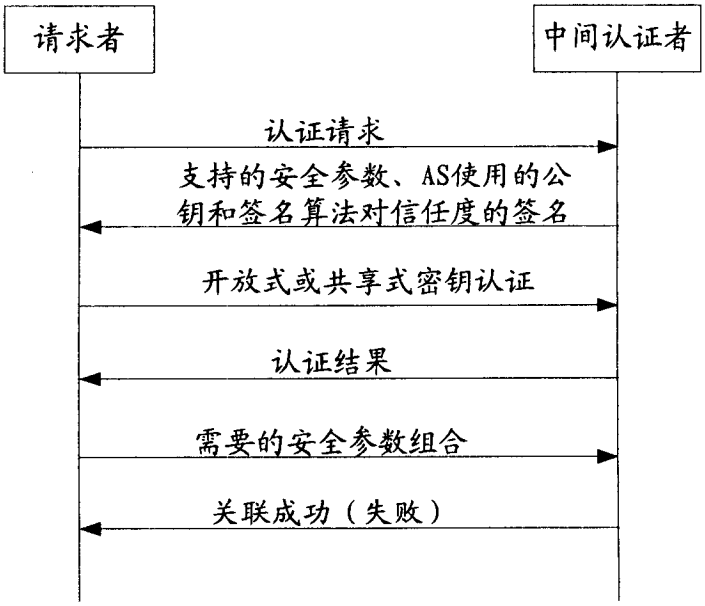


图 4

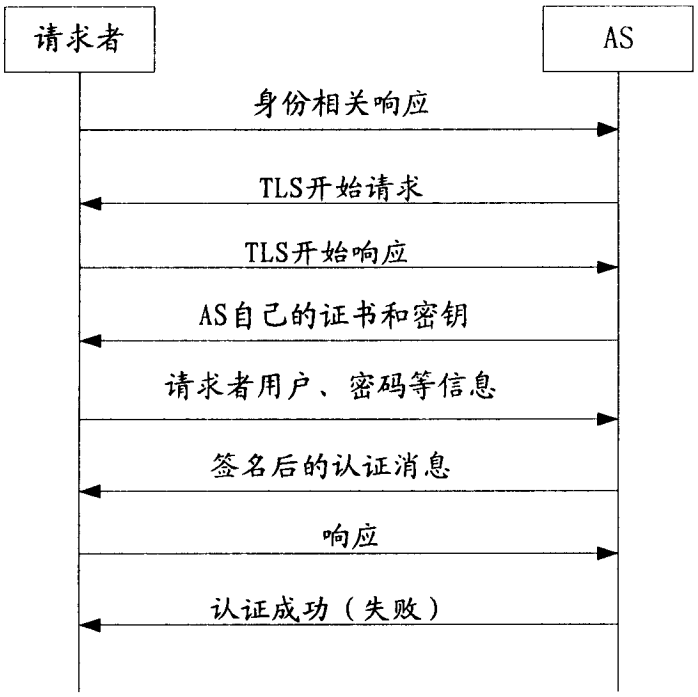

请求者
AS
身份相关响应
TLS开始请求
TLS开始响应
AS自己的证书和密钥
请求者用户、密码等信息
签名后的认证消息
响应
认证成功（失败）

图 5

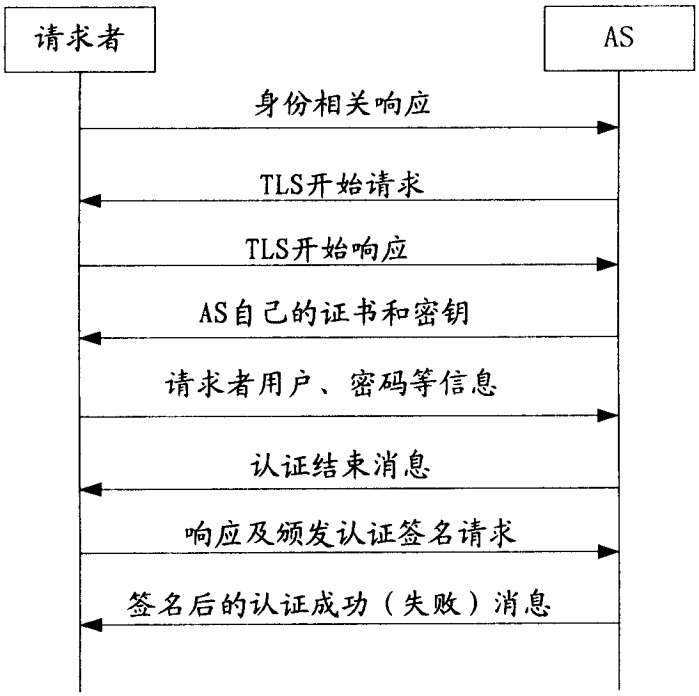

请求者
AS
身份相关响应
TLS开始请求
TLS开始响应
AS自己的证书和密钥
请求者用户、密码等信息
认证结束消息
响应及颁发认证签名请求
签名后的认证成功（失败）消息

图 6

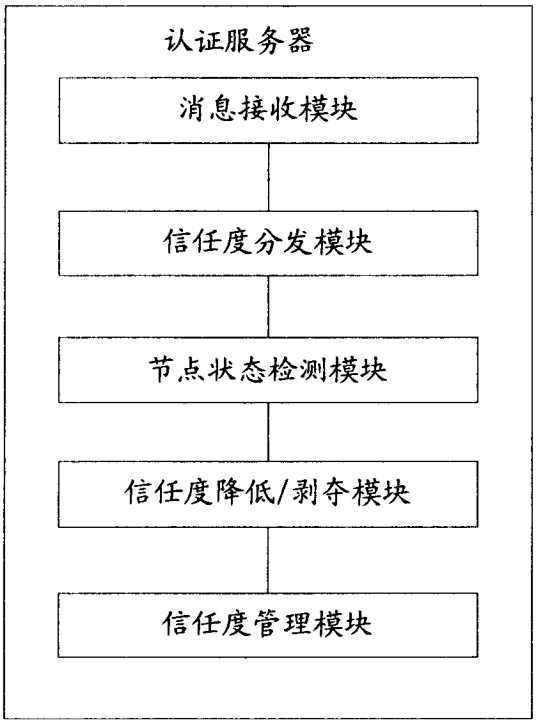


图 7