一种基于 HASH 函数的 EAP 认证协议①

陈凤其　姚国祥 (暨南大学 信息科学技术学院 广东 广州 510632)

摘　要： 随着安全认证技术的发展，网络认证已成为保障网络安全的重要环节。当前被广泛使用的 IEEE 802.1x 是建立在可扩展认证协议(EAP)基础上的一种认证框架。EAP 提供了许多认证协议，每个认证协议都有自身的优缺点。有些没有提供用户名的保护，有些没有提供双向认证；有些部署较困难等。针对上述缺陷，提出了一种基于哈希函数的认证协议。阐述了该协议的具体认证过程，并对其进行了安全性分析，最后与当前一些认证协议作了比较。

关键词： 802.1x；远程鉴别拨入用户服务协议；可扩展认证协议；哈希；信息摘要算法

A Hash-Based EAP Authentication Protocol

CHEN Feng-Qi, YAO Guo-Xiang

(School of Information Science and Technology, Jinan University, Guangzhou 510632, China)

Abstract: With the development of secure authentication technologies, network authentication has already become an important approach to protect network security. IEEE802.1x, the widely used authentication protocol, is based on the Extensible Authentication Protocol(EAP). EAP provides a lot of specific protocols, most of which have their own advantages and disadvantages. The disvantages include lacking protection of user name and mutual authentication and difficulty in deployment. This paper presents a new hash-based authentication protocol, which aims to avoid the above disadvantages of protocols. Authentication process and security analysis is explained in detail. Comparison with some protocols is also provided.

Keywords: 802.1x; RADIUS; EAP; Hash; MD5

1　引言

802.1x 是基于端口的访问控制协议，由 IEEE 802 LAN/WAN 委员会于 20 世纪 90 年代后期提出，通过该协议可以实现对连接到局域网上的客户端或设备提供认证和授权。Radius 是一种基于客户端/服务器端的安全拨号认证协议，目前被广泛使用在网络准入控制领域中。该协议支持多种认证协议，能够满足大多数情况下的网络需求。该协议将用户认证信息存储在认证服务器中，认证系统通过与用户通信获取用户的认证信息并将其发送到认证服务器验证其合法性。如果用户合法，则允许其访问网络资源，否则予以拒绝。

现有的 802.1x 认证协议主要分为基于密码的认证和基于证书的认证，基于密码的验证往往在安全上存在较多的缺陷，有些甚至不符合 802.1x 安全性要求，例如没有提供用户名保护，没有提供双向认证，抵御重放攻击和字典攻击的能力较差等；基于证书的认证虽然在安全上有较大改进，但需要有可信赖的证书发放中心提供严格的证书管理，并需要有安全的传输信道，因此开销相当大，且部署过程较为复杂。

2　802.1x和EAP认证协议的缺陷

2.1 802.1x 体系结构

802.1x 主要由 3 个部分组成[1]：客户端(Suppli-

① 基金项目:广东省部产学研项目(2008B090500201, cgzhzd0807);广东省科技计划(2008A010100001)
收稿时间:2009-10-13;收到修改稿时间:2009-11-14

cant System)，认证系统(Authenticator System)，认证服务器(Authenticator Server)。图 1 描述了这 3 者之间的关系。

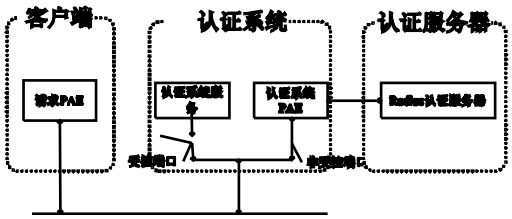

图 1 802.1x 体系结构

客户端一般装有认证请求系统，在认证开始阶段，以广播的形式发送认证请求，并等待认证系统的响应。认证系统通常指那些支持 802.1x 的网络设备，如交换机等。认证系统需要有两种逻辑端口：受控端口和不受控端口。不受控端口始终处于打开状态，主要用于传递用户认证信息；受控端口只有在认证通过的情况下才打开，为用户传递网络资源。其中认证系统的任务是收集客户认证信息，并将其通过安全的方式传递给认证服务器，同时接受来自认证服务器的响应；认证服务器的主要任务是接受认证请求，验证客户认证信息，返回客户验证信息。为了保证认证系统和认证服务器之间信息传输的安全性，认证系统和认证服务器之间共享一密钥，用此密钥对重要的认证信息加密。

2.2 EAP 认证协议的缺陷

认证系统与认证服务器之间传输的信息一般封装在 EAP 中[2]，EAP(Extensible Authentication Protocol 可扩展认证协议)本身并不提供某种具体的认证方法，它通过支持多种认证协议来提供通信安全，如 EAP-MD5，EAP-TLS 等。认证系统和认证服务器最终将其封装成 Radius 包，通过 Radius 协议在两者之间传输信息。

RFC3748 中关于 EAP 安全性说明中要求 EAP 所使用的认证协议应当具备双向认证、抵御重放攻击、防范密码猜测等安全性策略[3]。然而当前的 EAP 认证协议大多在这些方面存在着不同程度的缺陷，EAP-MD5 协议的主要缺陷如下[4]：

(1) 双向认证：双向认证是指除了认证服务器对客户端的认证外，客户端能够对认证服务器或认证者进行认证，以确保只与一个合法的认证服务器或认证者进行通信。EAP-MD5 中客户端没有对认证服务器或认证者进行认证，而只提供了认证服务器对客户端的单向认证。

(2) 抵御重放攻击：重放攻击能够使入侵者在不知道用户密码甚至用户名的情况下通过认证。虽然 Radius 协议通过使用随机码抵御重放攻击，但很多认证服务器并不都会去检验相隔较长时间的两个随机码是否重复，这使得某些攻击者可以成功实施重放攻击。另外使用某些较差的伪随机数发生器也可能会产生相同的随机码，使得协议抗重放攻击能力更差，EAP-MD5 协议同样存在重放攻击问题。

(3) 防范猜测攻击：在 EAP-MD5 的挑战-响应认证模式中，认证服务器端发送的挑战码是以明文的形式出现的，攻击者在得到该挑战码以及客户端发送的经过该挑战码哈希处理的响应值以后，就可以进行离线的猜测攻击，使其成功获取密码的概率大大增加。

3 改进的EAP-MD5协议

尽管 EAP-MD5 协议有上述不足之处，但在密码验证领域，与传统的 PAP 等认证方式相比还是有相当大的优势。本文所提出的改进的协议就是基于 EAP-MD5 协议。

在 EAP 框架内，认证流程涉及到 3 种角色：认证客户端(认证请求者)、认证系统(认证者)、认证服务器，分别用 U、A、S 代表它们，符号 H 表示一次性哈希函数，⊙表示异或运算。

3.1 EAP-MD5 协议认证流程

EAP-MD5 认证采取先验证用户名，后验证用户密码的方法。其具体认证流程如下：

(1) U——>A ：ID

客户端将用户身份信息 ID 发送至认证系统

(2) A——>S ：ID

认证系统将用户信息以 Access-Radius 包的形式发送至认证服务器

(3) S——>A ：chapID, chanllenge

认证服务器对用户信息进行验证，如果合法就进一步发送挑战码要求确认密码信息，否则，发送拒绝包。

(4) A——>U :chapID, chanllenge

认证系统将认证服务器的挑战码转发至认证客户端。

(5) U——>A :ID, H(chapID, chanllenge, password)

客户端将挑战码和认证密码计算成哈希值H(chapID,chanllenge,password)，与 ID 一起发送至认证系统

(6) A——>S : ID, H(chapID,chanllenge, password)

认证系统将客户端发送的信息以 Access-Request 包的形式传送至认证服务器

(7) S——>A :Access-Accept/Access-Reject 包

认证服务器验证用户的登录信息，合法则发送 Access-Accept 包，否则发送 Access-Reject 包。

3.2 改进的 EAP-MD5 认证协议

为了避免用户名以明文的形式直接传输，本协议由认证客户端生成的随机数代替用户名，而实际的用户名则利用认证系统和认证服务器之间的共享密钥加密，这样只有合法的认证服务器才能获取认证的用户名。

在认证过程中使用随机数可以有效地避免重放攻击，本协议在整个认证过程中使用了两个随机数，且分别由认证客户端和认证服务器产生，理论上这两个随机数在每次认证过程中都是不同的，攻击者要想同时产生两个相同的随机数是有相当大困难的，因此增强了抵御重放攻击的能力。

在 EAP-MD5 认证协议中，随机数经常以挑战码的形式由认证服务器经认证系统发往认证客户端，认证客户端依据与认证服务器共享的密钥和此挑战码使用 MD5 哈希函数生成响应值供认证服务器验证。这种方法实际上是在认证的第一阶段认证用户名的合法性，接着在第二阶段使用挑战码验证密码的合法性。近些年随着密码认证学的发展和进一步研究，一些研究人员提出使用哈希函数的密码认证同样可以实现双向认证，例如 Tzung-her Chen[4]等人提出的密码认证模式可以很好地解决双向认证的问题，该认证模式即在认证的第一阶段将用户名及密钥信息一起发送至认证服务器认证(保证非法认证服务器无法从中提取有效信息的前提下)，而在第二阶段则是利用哈希函数由认证客户端验证认证服务器合法性。

本文通过两个随机数和共享密钥保护结合的方式，可以很好的保护认证客户端的用户名和密码[5]，增加猜测攻击的难度。改进的 EAP-MD5 认证协议具体认证流程如图 2 所示，其具体的认证过程如下：

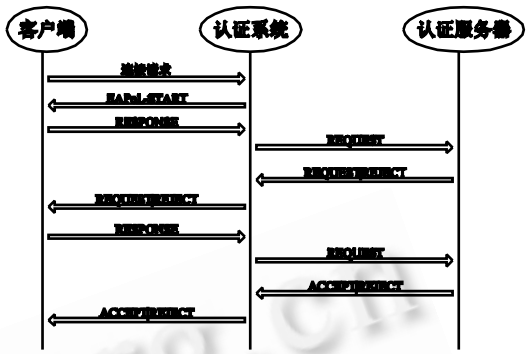

图 2 新的认证流程

(1) U ——> A :R1, ID, H(R1,ID,PW)

认证客户端生成一随机数 R1，计算 H(R1, ID,PW)，并将 R1、用户名 ID 以及 H(R1，ID，PW)一起发送给认证系统，其中 PW 为认证客户端与认证服务器之间的共享密钥。

(2) A——> S :R1，ID ⊙ H(R1,K), H(R1,ID,PW)⊙H(ID,K)

认证系统收到到认证客户端发送的认证信息以后，计算 H(R1,K)和 H(ID,K)，然后将 R1，ID⊙H(R1,K)和 H(R1,ID,PW)⊙H(ID,K)一起发送给认证服务器，其中 K 为认证系统与认证服务器之间的共享密钥，⊙表示进行一次异或处理。

(3) S ——> A :R2⊙H(R1,K), H(R1,R2), R2⊙H(ID,PW,R1)

认证服务器收到认证系统发来的认证信息后，计算 H(R1,K)，通过 ID⊙H(R1,K)⊙H(R1,K)得到用户名 ID，再计算 H(ID,K)，进行一次异或处理得到 H(R1,ID,PW)；认证服务器通过用户名 ID 验证其对应的密钥 PW，并计算 H(R1,ID,PW)，比较其与认证系统发送的信息是否一致。如果不一致，则进一步认证将被拒绝，响应的 Radius 包被标记为 REJECT 包；如果一致，继续认证过程，响应的 Radius 包被标记为 REQUEST 包，此时认证服务器将生成一随机数 R2，计算 H(R1,R2)，将 R2⊙H(R1,K), H(R1,R2), R2⊙H(ID,PW,R1)一起发送给认证系统。

(4) A ——> U :H(R1,R2), R2⊙H(ID,PW,R1)

认证系统收到认证服务器发送的认证信息后，计算 R2⊙H(R1,K)⊙H(R1,K)得到随机数 R2，再计算

H(R1,R2)，检验其与认证服务器发送的是否一致。如果不一致，则认证系统将拒绝转发至认证客户端；如果一致，则将认证服务器响应信息连同 H(R1,R2)，R2⊙H(ID，PW，R1)发送给认证客户端确认。

(5) U——> A　:ID, H(R2,R1)

认证客户端收到认证系统的响应信息后，计算 H(ID,PW,R1)，进行一次异或处理得到随机数 R2，计算 H(R1,R2)验证其是否一致。如果一致且认证服务器发送的 Radius 包为 REQUEST 包，则将用户名 ID 与 H(R2,R1)一起发送给认证系统。

(6) A——> S　:R1, ID⊙H(R2,K), H(R2,R1)

认证系统将 ID⊙H(R2,K)和 H(R2,R1)发送给认证服务器。

(7) S——>A　:Access-Accept/Access-Reject 包

认证服务器收到认证系统发送的信息后，检验 H(R2，R1)是否正确，如果正确，则发送 ACCEPT 消息，通知认证系统为该用户打开访问权限，否则，发送 REJECT 消息。

4　安全性分析

针对改进的 EAP-MD5 认证协议，从双向认证、抵御重放攻击、机密性、防范猜测攻击这几个方面的进行安全性分析。

(1) 双向认证：在 EAP-MD5 协议中，为了避免传统的 PAP 认证协议中密码以明文形式传输的缺点，使用了通过产生随机数挑战客户端的形式，这些既可以保证密码的安全，又可以验证认证客户端的合法性。本文所提出的协议由客户端生成一随机数作为挑战码传输至认证服务器，认证服务器响应时须据此生成哈希值，供认证客户端验证，从而实现双向认证的功能[6]。

(2) 抵御重放攻击：对每一次新的认证请求，协议都要求认证客户端和认证服务器分别产生新的随机数 R1 和 R2。在认证过程中，认证服务器产生新的随机 R2，并计算出 H(R1,R2)供认证系统和认证客户端验证，以避免欺骗认证系统或者认证客户端的攻击。认证客户端将计算 H(R2,R1)供认证服务器验证，以防止欺骗认证服务器的攻击，即改进的方案可以有效地抵御重放攻击。

(3) 机密性：以往的认证协议中用户名一般直接由从 Radius 包中的 USER-NAME 属性得到，本方案中使用随机数 R1 取代用户名，具体的用户名需要由认证服务器通过计算 H(R1,K)才可得出，而密钥 K 只有合法的认证服务器才拥有，这样无论是截取 Radius 包的攻击者还是非法的认证服务器都无法得到用户名，从而保证了认证客户端的机密性。

(4) 猜测攻击：如果攻击者企图得到认证客户端的用户名和密码，可以通过窃听和截取认证系统与认证服务器之间的通信数据，计算 H(R1,K)，再与 ID⊙H(R1,K)异或得到用户名，然后再计算 H(ID,K)，与 H(R1,ID,PW)⊙H(ID,K)异或后得到 H(R1,ID,PW)，最后还要计算所有可能的哈希值与 H(R1,ID,K)比较，才能得到用户的用户名 ID 和密码 PW。由于攻击者没有认证系统和认证服务器之间的共享密钥 K，因此无法计算出 H(ID,K)，由于攻击者没有用户名和密码中的任何一个，因此可以有效防止猜测攻击。

表 1　相关 EAP 认证协议的比较

	EAP-HMD5	EAP-MD5	EAP-LEAP	EAP-TLS	EAP-TTLS
双向认证	有	无	有	有	有
用户名保护	有	无	无	无	有
抵御重放攻击	较强	弱	弱	较强	强
防范猜测攻击	较强	弱	弱	较强	强
服务器认证	哈希值	无	哈希值	证书	证书
客户认证	哈希值	哈希函数	哈希值	证书	证书
部署难度	易	易	难	难	中等

5　改进的EAP-MD5(EAP-HMD5)与其他EAP协议的比较

EAP 支持多种具体的认证协议，每一个协议都有其优缺点。下面将本文所提出的改进的协议与当前一些主流的认证协议作对比，这些协议主要有 EAP-MD5，LEAP，EAP-TLS，EAP-TTLS。

EAP-MD5 协议利用挑战码进行密码认证，避免传统认证协议中密码以明文的形式直接传输。但从上文的分析中，可以看出其没有提供双向认证，用户名没有得到保护，抵御重放攻击和猜测攻击的能力也较差。

EAP-LEAP 是由 CISCO 公司开发的，它采用了哈希值提供认证客户端和认证服务器之间的双向认证，但在该协议中用户名也是以明文的形式的传输，这也导致其抵御重放攻击和猜测攻击的能力较差。

EAP-TLS 使用 x.509 证书提供双向认证，但是在使用证书认证之前必须先向对方提供身份验证，因此

(下转第 61 页)

(上接第 77 页)

用户名没有得到很好的保护，而且其部署及管理都比较复杂。

EAP-TTLS 是基于隧道的 EAP-TLS 协议，相对于 EAP-TLS，其主要优势是利用隧道在证书认证之前提供身份验证，从而保护了用户名不被泄露。

6 结语

本文分析了 IEEE802.1x 协议和其所依赖的 EAP 协议以及 EAP 协议的缺陷，在密码认证等理论基础上，结合 802.1x 认证协议的安全性要求，提出了一种新的认证方案 EAP-HMD5 认证协议。并分析了该协议的安全性，最后与当前主流的 EAP 认证协议作了比较，证明了本文所提出的协议能够在低开销的前提下保证认证过程的安全性。

参考文献

1 IEEE Standard 802.1X-2001. IEEE standards for local and metropolitan area networks: Port based network access control, 2001.

2 RFC3580. IEEE 802.1X remote authentication dial in user service(RADIUS) usage guidelines, 2003.

3 RFC3748.Extensible Authentication Protocol (EAP). 2004.

4 Chen T H, Lee W B. A new method for using hash functions to solve remote user authentication. Computers and Electrical Engineering, 2007l: 53－62.

5 Sun HM, Yeh H T. Password based authentication and key distribution protocols with perfect forward secrecy. Journal of Computer and System Sciences, 2006: 1002－1011.

6 Yoon E J, Ryu E K, Yoo K Y. A secure user authentication using hash functions. ACM SIGOPS Operating Systems Review, 2003: 62－68.