网络环境下一次性口令身份认证的研究与实现

吴和生 范训礼 谢俊元
(南京大学计算机科学与技术系 软件新技术国家重点实验室 南京210093)

Research and Implementation of One-Time Password Authentication in Computer Networks

WU He-Sheng FAN Xun-Li XIE Jun-Yuan
(Department of Computer Science, State Key Lab for Novel Software Technology, Nanjing University, Nanjing 210093)

Abstract Some usual one-time password authentication protocols are analyzed. A practical efficient one-time password authentication implementing method is presented based on the symmetric algorithm, which conquers usual challenge-response protocol weakness and can protect user's identity and avoid replay attack etc. It also can boost up the security of the application security system by integrating with it in networks. And the correlative issues are discussed deeply such as security, reliability and so on.

Keywords Authentication, One-time passwords, Symmetric secret key, Hash function

1 引言

在开放式的网络环境中,身份认证往往是许多应用系统安全保护的第一道防线,也是保证应用系统安全的关键。进行身份认证的方式很多,通常分为三类:(1)只有该主体知道的秘密,如口令、密钥;(2)主体拥有的物品,如智能卡和令牌卡;(3)只有该主体具有的独一无二的特征或能力,如指纹、声音等。更强大的认证可以是几种方法的组合。

无疑,口令是最简单也是最常用的一种身份认证方法。一个好的口令对于保证用户数据的完整性、可靠性以及安全性十分重要。但通常使用的静态的口令有许多固有的弱点:易于猜测或窃听;不能进行共享控制等。而且也存在实现上的弱点:在分布式网络系统中,若不加密,可以被清晰地看见明文;即使加密,也易受重放攻击、差分密码分析等其他攻击手段的影响。从而给系统的安全性埋下隐患。而使用一次性口令系统则可以显著地增加系统的安全性,一次性口令系统允许用户每次登录时使用不同的口令,它可以防止重放攻击、词典攻击等常用的攻击手段,为对付窃听者以及公开的登录会话提供了强有力的保护。

本文首先分析了当前较为常用的几种一次性口令身份认证方案。在质询响应方案基础上,基于单钥体制设计和实现了一种实用有效的一次性口令身份认证方案。该方案克服了通常情况下的质询响应方案的弱点,有效地保护了用户的身份,可以用在分布式网络环境下,与其它应用系统集成,实现对用户身份的认证,有效地防止了重放攻击、词典攻击等攻击手段,执行效率较高,能显著增强应用系统的安全性。

2 常用的一次性口令身份认证方案分析

有许多方法可以实现一次性口令方案。常用的有如下四种:

(1)Lamport 方案 Lamport 提出的一次性口令方案[1],不需要特殊的硬件。假设存在某个函数 F,很容易进行正向计算,而不可能有效地进行逆向计算(密码散列函数就是一个很好的候选算法)。进一步假设用户有某个秘密(或许是一个口令)x。为了保证用户以某个次数进行登录,主机计算出次数 $F(x)$。因此,在口令改变前假设允许100次登录,那么主机应计算出 $F^{100}(x)$,并只存储该值。用户第一次登录,他应提供 $F^{99}(x)$。系统通过计算 $F(F^{99}(x))=F^{100}(x)$ 进行验证。如果登录正确,所提供的口令—$F^{99}(x)$—就变成新的存储值。它又被用于对 $F^{98}(x)$ 进行验证,它是用户下一次提供的口令。可使用手持鉴别器、可信工作站或便携机计算用户的 $F^{n}(x)$。

(2)Bellcore 方案 Bellcore 的方案[2],叫做 S/Key,它实现了 Lamport 方案,但对 Lamport 方案做了一些改进。当用户登录到一台安全机器上时,它可以运行一个程序计算出下几次登录的序列值,并将其编码成一系列的短字码。在进行旅行时可带上打印的列表。但用户使用时必须小心,不要在使用过的口令上作记号,以免打印的列表遭到偷窃后造成损失。

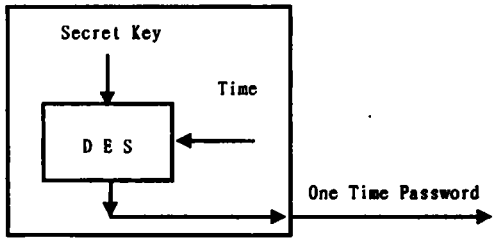

图1 时钟同步原理

(3)时钟同步方案(Time Synchronized) 这是最熟知的一种方案,它需要使用手持式鉴别器,也叫做 dongle 或标记(token)。鉴别器的通用形式包含一个内部时钟、某种类别的一个密钥以及一个显示器。显示器输出的是当前的时间和密钥的某种函数。其输出值常用于鉴别报文,约每分钟变化一次。而且"口令"绝不会重复。主机通过使用其秘密密钥的副本及其时钟计算出所希望的输出值,对用户进行证实。如果相匹配,则登录被接收。时钟同步原理见图1。实际上,设备和主机

吴和生 硕士生,主要研究领域:计算机网络与信息安全。谢俊元 教授,博士生导师,主要研究领域:计算机网络与信息安全,人工智能。范训礼 博士后,主要研究领域:计算机网络与信息安全。

之间的时钟偏差可能成为一个问题。为了保证这一点，计算出几个候选口令，用户口令值与这一组口令去进行匹配。一个可访问主机的数据库对设备时钟的偏差变化进行跟踪，以帮助最小化该时间窗口。这又引入另一个问题：一个口令可以在时钟偏差间隔期间被重用。正确的办法是高速缓存在口令有效生命期间所有接收到的口令，试图重用的口令应予以拒绝并记录在案。

(4)*质询响应方案*(Challenge Response) 一种不同的一次性口令系统使用来自主机而不是时钟的非重复质询。用户拥有的是一台用秘密密钥编程的带键板的设备。这个质询被键入设备中，这台设备利用秘密密钥的副本计算它的某种函数值，然后这个值就作为口令。质询响应方案原理如图2所示。由于这里不涉及时钟，也就不存在时钟的偏差问题，因此不需要高速缓存。RADIUS(Remote Authentication Dial In User Service(RADIUS)远程认证拨入用户服务)协议[3,4]详细描述了这种方案。但质询响应方案直接用在网络环境下也存在一些不足，比如：(1)需要特殊的硬件支持，增加了购买鉴别器硬件的代价；(2)要手工地输入质询到鉴别器中，从鉴别器获取响应后再手工输入到主机，使用起来十分不便；(3)虽然用户的密钥不用在网络传输，但用户的身份标识却直接在网络上明文传输，窃听者可以很容易地获取用户身份标识，留下了安全隐患。

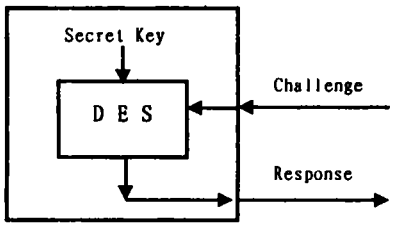

图2 质询响应原理

3 一次性口令身份认证的设计与实现

虽然 Lamport 方案和 Bellcore 方案易于实现，而且无需特殊的硬件，但其安全性依赖于函数 F，而且不宜用在分布式网络环境下。时钟同步方案对时钟偏差的解决会引起其他相关的安全问题。虽然质询响应方案直接用在网络环境下也存在一些不足，但其优点是明显的：(1)采用 C/S 模式，易于在网络环境下实现；(2)在客户端和服务器之间通过秘密密钥鉴别，密钥不用在网络传送；(3)灵活的认证机制，可支持 PPP，PAP，CHAP，UNIX login 等；(4)可扩展性(变长的属性-长度-数据三元组，其中在属性字段中，可以引入新属性值，而不必考虑已存在的实现)。而且它也是目前用于认证远程用户的强有力的方式之一。因而本文的设计采用了质询响应方案。

为了克服质询响应方案的不足，并确保网络上传输的安全性，实现时必须对其进行必要的改进：(1)用软件方法代替鉴别器硬件的功能；(3)采用 Hash 函数对用户的身份标识进行加密后再在网上传输。

3.1 设计原则

由于本方案主要用在网络环境下，用于远程登录会话的认证，故设计时应遵循以下原则：(1)尽可能使用计算简单而安全的密码算法，以减少计算量，如 Hash 函数可以选择 MD5、SHA 等；(2)尽可能使传送的消息简短，减少相互传递认证信息的个数，减少网络通信量。(3)虽然在客户端和服务器之间通过秘密密钥鉴别，但秘密密钥不能在网络上传送。(4)用户的身份标识不能在网络上传输。

3.2 协议流程

本方案是基于 RADIUS 协议的，在实现时虽作了一些改进，但为了保持通用性和兼容性，并未影响到协议层，因而可以看作是 RADIUS 协议的一种较安全的实现。在本方案中，用户 U 和服务器 S 共享信息为：U 的标识符 uid 和对称密钥 k；安全单向的 Hash 函数 H。为了提高认证效率和安全性，服务器 S 中存放的是用 H(uid)处理后的用户标识列表 List。本方案的协议流程如图3所示。

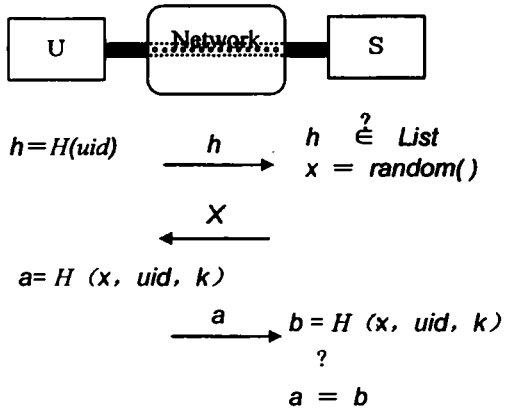

图3 协议流程

步骤1：U 计算 $h=H(uid)$，将 h 发送给 S。

步骤2：S 在收到上述信息后，判断：

$h \overset{?}{\in} List$，若 $h \in List$，表明此 uid 合法，则由随机数生成器生成随机数 $x=random()$，然后将 x 发送给 U。否则，若 h 不属于 $List$，则说明 U 为非法用户，终止与 U 的会话。

步骤3：U 收到 S 发送的消息之后，得到 x，计算 $a=H(x,uid,k)$，然后发送 a 给 S。

步骤4：S 收到 a 之后，计算 $b=H(x,uid,k)$，并验证 $a=b$ 是否成立，若成立，则 U 的身份得到了验证。否则，说明 U 为非法用户，终止与 U 的会话。

3.3 认证过程

为了保持通用性和兼容性，本方案的认证报文均采用 RADIUS 协议的标准报文格式，报文格式的细节请参考文[3,4]。报文封装在 UDP 数据域中，报文类型由代码域决定。有如下几种报文类型：

(1)Access-Request 由客户端发给服务器端的认证请求报文。必须包含 User-Name 属性，也应该包含 NAS-IP-Address 属性或 NAS-Identifier 属性(或两者都，尽管不推荐)，还必须包含 User-Password 属性或 CHAP-Password 属性。应该包含 NAS-Port 或 NAS-Port-Type 属性(或两者都)除非访问类型不涉及端口或 NAS 不区分端口。可以包含附加属性，但 server 不必理会它。

(2)Access-Accept 由服务器端发给客户端的认证成功报文。一旦 Access-Accept 被收到，其 Identifier 域必须与未决的 Access-Request 相应域匹配，另外，Response Authenticator 域必须包含对未决的 Access-Request 的正确的响应。无效的 Access-Accept 报文将被客户端丢弃。

(3)Access-Reject 如果任何被接收的属性值不可被接受，则服务器必须发送一个 Access-Reject 报文告知客户端，它可以包含一个或多个带有文本信息的响应消息属性。

(4)Access-Challenge 由服务器发送给客户端的质询报文。一旦 Access-Challenge 被收到,其 Identifier 域必须与未决的Access-Request相应域匹配,另外,ResponseAuthenticator 域必须包含对未决的 Access-Request 的正确的响应。无效的 Access-Challenge 报文将被客户端丢弃。

为了清晰起见,下面对认证通过和认证未通过的认证过程(图4)进行简要的描述。

①client 发送 Request($H(uid)$,password,client ID,port ID,etc)。这里,password 可以为固定字符串,或不填。在等待响应报文一段时间后,若无响应则重发一定次数,或发向其他认证服务器。

②server 收到 Request 之后,在数据库中查询用户名的散列值及其它请求中包含的属性,也可以作为一个 client 再向其他认证服务器发出认证请求。

③当 server 判断有条件不满足时,发出 Access-Reject 响应,该响应中可以包含一个文本消息提示用户,但不能包含其它属性。

④client 收到 Reject 后,结束此次认证过程。(可以重新进行认证)

⑤当 server 判断全部条件满足时,产生一随机数 x ,发送 Access-Challenge 响应,它可以包含一个提示用户响应 Challenge 的文本信息,以及一些状态属性。

⑥client 收到 Access-Challenge,以一个新的 request ID 发送它的原始的 Access-Request,这时 password 字段被重置为用 MD5算法进行加密后的字符串。

⑦server 收到新 Request 后,对收到的 password 与自己进行加密的 password 进行比较。如果一致,则发送 Access-Accept;如果不一致,则发送 Access-Reject(已经尝试了一定次数)或 Access-Challenge(未达到尝试次数)。

⑧client 收到 Accept 后,认证通过。

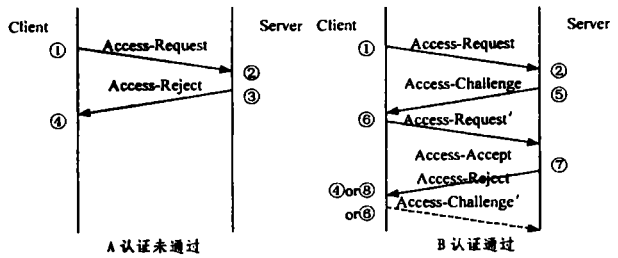

图4 认证过程

4 实现中的问题分析

无疑,要在分布式网络环境下较好地实现此身份认证方案需要考虑许多方面的问题,下面就我们实现中重点考虑的一些问题进行较为深入的分析。

4.1 安全性

安全系统自身的安全性无疑是必须首要考虑的问题。本系统为分布式网络环境提供了一种对用户身份进行验证的方法。它的安全机制在于采用了双重认证机制。首先对发出请求的用户标识进行验证,确认其是否是有效的用户;然后通过质询/响应报文,对用户的真实性进行进一步的确认。下面从三个方面对方案的安全性进行分析:

(1)*方案能抗重放攻击* 由于每次认证所生成的随机数 x 是各不相同的,而且绝不重复,所以入侵者重放已经截获的信息是无法通过认证的。

(2)*用户身份得到保护* 用户的身份信息发给服务器时,使用单向哈希函数进行处理,由于安全散列函数的特性:对于任意的随机输出值,不可能构造其输入,因而也不可能由截获的 h 获得用户的 uid 。

(3)*协议过程是安全的* 假设入侵者可以截获合法认证过程中的任何通信报文进行分析,他从截获的 h 无法获取 uid,当然他直接发送截获的 h 也能通过服务器的初步认证,但由于密钥 k 根本不通过网络传输,因而他不可能从截获的报文中解析出 k,从而无法通过下一步的认证,故协议的整个过程是安全的。

事实上,由于用户标识 uid 和密钥 k 不通过网络传输以及随机数 x 的随机和不重复特性,因而窃听报文进行分析和词典攻击等常用攻击手段对此是无能为力的。而且这种方案可以对付来自客户端的冒充攻击,这是因为入侵者不知道密钥 k(而且由于密钥 k 并不在网上传输,所以入侵者即使分析所截获的报文也不可能解析出 k。)而无法完成协议的认证功能,而且由于 x 的随机和不重复特性,即使对 hash 值进行生日攻击也不可能通过身份认证。

但是这种方案不能有效抵抗来自服务器端的冒充攻击。虽然入侵者冒充服务器接收来自客户端的报文,并不能从截获的报文中分析出密钥 k 甚至用户标识 uid,但至少使得合法用户的合理要求得不到满足,也有可能造成重大的损失,而且也留下了安全隐患。一种可行的解决方案是客户端要求服务器端发来的质询报文中包含随机数 x 和密钥 k 的 hash 值。用户可以对此 hash 值进行校验来防止来自服务器端的欺骗。

4.2 可靠性

认证报文均封装在 UDP 数据域中。众所周知,UDP 是一个简单的面向数据报的传输层协议,它不提供可靠性:它把应用程序传给 IP 层的数据发送出去,但是并不保证它们能到达目的地。既然缺乏可靠性,那我们为何弃用能可靠传输的 TCP 协议而采用它呢?这主要充分考虑了本身份认证方案所固有的特性:(1)如果到主认证服务器的请求失败,必须查询从服务器:为满足这个要求,请求报文的复本必须保留在传输层以便可选传输,这意味着需要重传定时器。(2)此方案的定时要求与 TCP 提供的有显著差别。极端地,此方案不要求一个应答检测丢失的数据。用户乐意为完成认证等几秒。因而不需要 TCP 的重传(基于平均环回时间),也不需要 TCP 确认的开销。另一种极端,用户将不愿意为认证而等几分钟。因此 TCP 的可靠传输一两分钟后便会毫无用处。(3)此方案的无状态特性。(4)UDP 简化了服务器的实现。

虽然采用了不可靠的 UDP 协议传输报文,但本方案通过如下机制保证了认证的可靠性:建立了不同于 TCP 的报文重传机制。由应用程序提供重传定时器,当预定时间到了仍未收到回应报文则重发数据报。客户端和服务器端通过校验报文序号来确定报文传输的时序和区分重发的报文。

结束语 本文分析了当前较为常用的几种一次性口令身份认证方案。在质询响应方案基础上,基于单钥体制设计和实现了一种有效的适用于网络系统的一次性口令身份认证方案,该方案能够对用户身份的真实性进行可靠的鉴别,并且克服了通常的质询响应认证方案的弱点,有效保护了用户的身份,防止重放攻击等常规的攻击手段的攻击,实现简单,执行效率高。可与应用系统(特别是安全系统如防火墙)集成,用于

应用系统本身的身份认证，以增强应用系统的安全性。而且实践证明该方案是有效的。

参考文献

1 Lamport L. Password authentication with insecure communication. Communications of the ACM，1981，24(11)：770～772

2 Haller N M. The S/Key one-time password system. In：Proc. of the Internet Society Symposium on Network and Distributed System Security，San Diego，CA，Feb. 1994

3 Rigney C，Rubens A，Simpson W，Willens S. Remote Authentication Dial In User Service(RADIUS). RFC 2138，April 1997

4 Rigney C. RADIUS Accounting. RFC 2139，April 1997

5 Steiner J，Neuman B C，Schiller J I. Kerberos：An authentication service for open network systems. In：Proc. Winter USENIX Conference，Dallas，TX，1988. 191～202

6 Kohl J，Neuman C. The Kerberos Network Authentication Service(V5). RFC 1510，Sep. 1993

7 Tsudik G. Message authentication with one-way hash functions. In：Proc. of IEEE Infocom' 92，Florence，Italy，May 1992

8 Schneier B. Applied Cryptography：Protocols，Algorithms，and Source Code in C. John Wiley & Sons，New York，1994

9 Needham R，Schroeder M. Using encryption for authentication in large networks of computers. Communications of the ACM，1978，21(12)：993～999

10 Davies D W，Price W L. Security for Computer Networks. John Wiley & Sons，second edition，1989

(上接第93页)

个方面进行研究：构件信息的共享机制；基于 Web 的构件检索机制；分布构件库的一体化；构件表示的标准化、多样化；多库的协同检索；构件的反馈和后期度量。

参考文献

1 NATO Standard for Management of a Reusable Software Component Library，1991

2 Sauer L D，Clay R L ，Armstrong，R. Meta-component architecture for software interoperability，Software Methods and Tools，2000. SMT 2000. Proceedings. International Conference on，2000. 75～84

3 Conn R. The Ada Software Repository and the Defense Data Network：A Resource Handbook，New York Zoetrope，1987

4 NATO Standard for Development of Reusable Software，1991

5 Seacord R C，Hissam S A，Wallnau K C. Agoro - a search engine for component. IEEE Internet Computing，Nov./Dec. 1998. 62～70

6 常继传，郭立峰，马黎. 可复用软件构件的表示和检索. 计算机科学，1999. 45～49

7 Cha J-E，Yang Y-J，Song M-S，Kim H-G. Design and implementation of component repository for supporting the component based development process，2001 IEEE International Conference on Systems，Man，and Cybernetics. In：2001 IEEE Intl. Conf. on，Volume：2，2001. 735～740

8 RIG Basic Interoperability Da ta Model(BIDM)：[RPS-0001]，1993

9 Frakes W B，Pole T P. An Empirical Study of Representation Methods for Reusable Software Components，IEEE Trans. On Software Engineering，1994，20(8)

10 Sauer L D，Clay R L，Armstrong R. Meta-component architecture for software interoperability，Software Methods and Tools，2000. SMT 2000. proceedings. International Conference on，2000. 75～84

11 Prieto-Diaz R. Implementing Faceted Classification. Communication of ACM，1991，34(5)：88～97

12 Mary Shaw：Truth vs Knowledge：The Difference Between What a Component Does and What We Know It Does，Proceedings of 8th International Workshop on Software Specification and Design，1996 pp. 181-185

13 Partners of the SCREEN Project：Service Creation Engineering Environment，Sema Group Telecom，France Telecom CNET，ACTS Ref.：AC227 SCREEN，1999

14 Atkinson S. A Unifying Model for Retrieval from Reusable Software Libraries：[Technical Report No. 95-41]. the University of Queensland，1995

15 Frakes W B，Pole T P. An Empirical Study of Representation Methods fro Reusable Software Components，IEEE Trans. On Software Engineering，1994，20(8)

16 Fischer B. Specification-Based Browsing of Software Component Libraries. Journal of Automated Software Engineering，2000，7(2)：79～200

17 Atkinson S，Duke R. Behavioural Retrieval from Class Libraries：Australian Computer Science Communications，1995，17(1)：13～20

18 Podgurski A，Pierce L. Behaviour Sampling：A Technique for Automated Retrieval of Reusable Components. In：Proc. of the 14th Intl. Conf. on Software Engineering，1992. 349～360

19 Mili A，Mili R，Mittermeir R. Storing and Retrieving Software Components：A Refinement Based System. In：Proc. 16th ICSE，IEEE Computer Society Press，1994. 91～100

20 Kontio J. Case Study in Applying a Systematic Method for COTS Selection. In：Proc. of the 18th Intl. Conf. on Software Engineering. Berlin，Germany，Los Alami-tos，CA：IEEE Computer Society Press，1996. 1996. 201～209

21 Morel J M，Faget J. The REBOOT Environment，BULL S. A. Rue Jean JAURES，F-78340 LESCLAYES-SOUS-BIOS，France

22 杨燕燕. 基于数据仓库技术的构件库系统研究：[博士研究生学位论文]. 北京大学计算机科学技术系，1999

23 Reuse Library Interoperability Group. RIG Uniform Data Model for Reuse Libraries(UDM)：[RPS-0002]，1994

24 Sorumgard L S，Sindre G，Stokke F. Experiences from Application of a Faceted Classification Scheme，In：Proc. Reuse '93，Lucca，Italy，1993，IEEE CS Press，1993

25 Chichester E-A. Software Reuse：A Holistic Approach -Measuring the Effect of Reuse Chapter. New York：Wiley，1995. 113～180

26 Ye Yunwen. An Active and Adaptive Reuse Repository System. In：Proc. of 34th Hawaii Intl. Conf. on System Sciences(HICSS-34)，Software Technology Track ，Maui，HI，IEEE Press，CD-ROM，2001. 10

27 Guo J，Luqi. A Survey of Software Reuse Repositories，In：Proc. of the IEEE Intl. Conf. and Workshop on the Engineering of Computer Based Systems(IEEE ECBS'2000)，Edinburgh，Scotland，UK，2000

28 Leymann F. Web Services Flow Language(WSFL 1.0)，IBM Software Group，2001