基于 SSL 协议的可信应用及实现*

金　仑，谢俊元
(南京大学 软件新技术国家重点实验室，江苏 南京 210093)

摘　要：首先分析了安全套接层(SSL)协议在握手阶段的信任不足问题，然后讨论了信任协商机制的原理，提出了采用信任协商机制来扩充 SSL 协议，并给出了具体的实现方案。

关键词：访问控制；SSL；信任协商
中图法分类号：TP393.08　　文献标识码：A　　文章编号：1001-3695(2006)01-0103-03

Credible Application and Realization Based on SSL

JIN Lun，XIE Jun-yuan
(State Key Laboratory for Novel Software Technology，Nanjing University，Nanjing Jiangsu 210093，China)

Abstract: This paper analyses the problem of distrust in the handshake of SSL protocol，discusses the principle of trust negotiation，puts forward an extension of SSL using trust negotiation and the implementation of it.

Key words: Access Control；SSL；Trust Negotiation

安全套接层(SSL)协议是一种在 Internet 基础上提供保证私密性的安全通信协议，越来越多的 Web 应用服务采用了 SSL 来保护通信数据。然而，在 SSL 握手协商产生通信双方加密密钥的过程中，存在着各种不够信任的漏洞，从而导致了虽然双方之间的通信数据有了密钥的保护，但是对于双方本身而言，仅仅依赖于在握手阶段交换的证书，他们并不能对通信另一方的身份产生足够的信任，双方也不能进一步获取对方的隐秘资源。在传统的网络应用中，通常采用访问控制(Access Control)的方式来对资源访问作限制。但是，在分布式网络环境中，传统的访问控制机制就暴露出了很大的问题:绝大多数的访问控制都是基于访问者的身份认证，然而在双方完全陌生的情况下，这种方式将无法起作用。本文将结合 SSL 协议，使用信任协商机制来解决这一问题，从而得到一种既能进行保密通信，又能建立双方信任的网络应用方案。

1　SSL 协议的原理和信任缺陷

SSL 协议是基于连接的，并且通过 TCP 的支持来提供一种可靠的端到端的安全服务。SSL 支持可靠性、数据完整性和 C/S 认证，它包括了两层协议:

(1) SSL 记录协议(SSL Record Protocol)。在 TCP 的基础上，对高层协议提供数据封装、压缩、加密等基本功能的支持。

(2) SSL 握手协议(SSL Handshake Protocol)。在双方正式传输数据前进行身份认证、协商加密算法、交换加密密钥。

本文重点考察的是 SSL 协议中的信任缺陷，而 C/S 认证存在于握手协议中。一个完整的带有认证功能的握手过程(以 RSA 方式交换密钥为例)如图 1 所示。

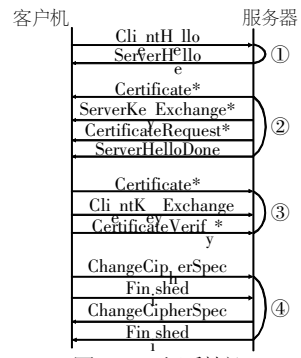

图 1　SSL 握手协议

SSL 握手协议一共有四个阶段(带* 的报文表示可选发送):①客户端和服务器端交换 Hello 报文，以用来创建在握手阶段将要用到的安全参数;②服务器可以发送证书、密钥交换和证书请求、服务器发出结束 Hello 报文阶段的信号;③客户端在服务器端要求客户端证书的情况下发送证书以及一个包含了用于生成密钥的材料的报文 ClientKeyExchange，另外可以发送包含对服务器端证书验证结果的 CertificateVerify 报文;④双方结束握手，准备开始用协商的密钥对交换数据进行加密。关于 SSL 握手协议的详细描述超出本文的范围，具体内容可以参见文献[1]。

仔细考察一下 SSL 握手协议中的身份认证，可以发现在通信双方完全陌生的情况下，存在以下一些信任不足的问题:

(1) 握手阶段客户机和服务器以明文形式交换证书，虽然不会导致由于被窃听后冒用证书的危险，但是证书的隐私内容会被非信任的对象所获知。

(2) 客户机和服务器之间的认证只依赖于一条证书链。

(3) 服务器在客户机之前提交自己的证书，这样使服务器总是要对完全陌生的对象提交证书。

收稿日期：2004-11-03；修返日期：2005-03-02
基金项目：国家“863”计划资助项目(2001AA142010)

(4) 如果客户机或者服务器收到的证书链中没有完全满足他们认证需求的信息,协议没有提供额外的证书来满足他们的认证需求。

产生以上问题的根源是 SSL 协议本身对证书使用的机制还不够完善,这说明如果仅依赖于 SSL 握手协议中的证书交换来确立完全陌生的通信双方之间的信任是不可靠的。为了解决这些问题,有必要采用更严密的认证机制来使通信的双方在保证数据通信安全的同时能获取对另一方身份的信任,以便更好地访问对方的资源。

2 信任协商机制的原理

信任协商(Trust Negotiation)机制[2]作为一种新的认证机制,能对 Internet 上完全陌生的通信双方通过不断交换证书和策略来建立他们之间的信任。下面来看一个信任协商的例子。

病人 A 因为昏迷而被送入医院,医生 B 决定对 A 进行手术,需要麻醉 A,但必须知道 A 对麻醉药是否过敏。假设存在一个病人资料库 K,于是,B 需要和 K 进行通信,得到病人 A 对某些药物排斥情况的资料。通常情况下,这个资料库 K 只对具有合法行医证书的人才开放病人资料。现在 B 具有合法的行医证明 D_1,不过他只对合法的病人资料库提供他的行医证明。假设 K 也是经授权的,它具有由合法医疗保障机构颁发的能证明它身份的证书 D_2(D_2 是可以自由公布的)。

图 2 描绘了一次医生 B 和资料库 K 之间的信任协商过程:①B 的客户端软件向 K 请求获取 A 的病人资料;②K 请求获取 B 的合法行医证明;③B 的客户端请求获取能证明 K 的合法性的证书;④由于 K 能不受限地提供证明其身份的证书 D_2,它便将 D_2 直接发送给 B;⑤B 的客户端收到 D_2 后经过验证,确信是 K 是一个经过授权的病人资料库,于是 B 将自己的行医证明 D_1 发送给 K;⑥K 验证了 D_1 之后确信 B 是一个具有合法行医证明的医生,于是 K 将 A 的病人资料发送给 B,结束了本次协商。

仔细考察一下上述协商过程,发现其中最重用的是双方交换的证书(或证明)以及访问控制的策略。这里,可以把证书(或证明)看作是电子证书。在信任协商中,证书是逐步交换的,因为通常人们不会直接把包含有自己隐私信息的证书提交给陌生人,这些信息一般包含在电子证书的属性中。如果一个证书含有隐私信息,那么需要有一条相应的访问控制策略来控制证书的提交。上例中,医生 B 使用一条访问控制策略来保证对方必须是一个经过授权的病人资料库才能得到 B 的行医证明,这条策略定义了请求 B 的行医证明的主体的必备属性。

在上述例子中看到的信任协商是简单而又直接的,但是,如果双方都具有隐私信息和访问控制策略的时候,信任协商的交换证书就需要进行多轮。

3 信任协商与 SSL 协议的结合和实现

3.1 问题的解决

如果陌生的双方要使用握手过程交换的证书来确定信任关系,那么也就是在对方完全陌生的情况下,一方就不得不将包含有它的隐私信息的证书发送给对方,并且这种隐私信息还有可能被网络上其他用户所窃听。为了让通信在有了 SSL 的加密保护之后,还能够进一步使双方获取信任,以便更好地访问对方资源,采用在 SSL 协议之后补充信任协商机制。

(1) 必须确保数据传输的保密性。由于 SSL 协议本身就是一种保密传输的机制,所以我们有必要先建立一条 SSL 连接。由于这个阶段的发送证书报文是可选择的,我们可以不使用任何证书,以防止证书中的隐私信息泄漏。

(2) 开始做信任协商的工作。假定从服务器端发起请求信任协商的报文,其中指明了对客户机哪种隐私资源的请求,如果客户机愿意进行,就回应。然后服务器发送第一阶段的(证书,策略)信息,其中证书可能包含多个。客户机收到此二元组后,验证此次收到的证书的正确性,并且根据收到的策略来生成回应的策略。如果证书验证通过,并且客户端也需要新的证书来进一步达到信任,则客户端也发送(证书,策略)这样的一组信息给服务器,服务器收到以后重复客户机的过程。不断这样往复协商直到服务器获取它所需要的资源,或者中间出现了某些失败的情形,协商过程就可以结束。

具体协商过程可以用图 3 来表示(我们假定由服务器发起协商请求)。

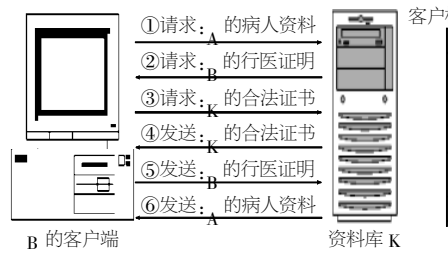

图 2 信任协商举例

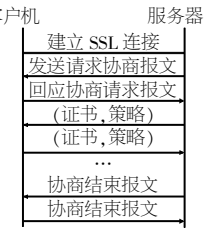

图 3 在 SSL 之后加入信任协商机制

在对 SSL 连接增加了这种信任协商机制以后,通信双方能有效地建立起信任关系,从而解决前面提到的 SSL 握手协议中的信任缺陷问题。

(1) 由于是在已经建立的 SSL 加密通道中传输数据,所以牵涉隐私信息的证书是以加密形式传送给对方的,避免了被窃听的危险;

(2) 在每一个来回中客户机和服务器可以交换多个证书,这些来自多个认证机构的证书可以减少单一证书带来的危险;

(3) 允许服务器发起协商请求,并且服务器可以不被强迫地在客户机没有提交证书前提交它自己的证书;

(4) 客户机和服务器有相同的机会向对方提交证书,并提出它的信任需求;

(5) 通过互相发送访问控制策略,陌生的双方获知对方为建立信任关系所需要的证书。

3.2 实现方案

在给出了解决 SSL 协议信任缺陷的方法之后,本文详细给出这种方法的实现细节。将上文中提到的二元组(证书,策略)记为(Y, Q),其中 Y 为证书集合,Q 为对方发来的索取本地证书的策略。

假设客户机方和服务器方各有一个自身的信任材料集合, 其中包含了各种能证明自己身份的信任材料(如证书)。另外, 它们还各自有一个验证模块 *T*, 用于对对方发送过来的信任材料进行检验、判断, 用于决定下一步的动作。同时双方还各自有一个策略判别和生成模块 P, 用于解释对方发送过来的策略, 并生成本地策略。

为简单起见, 我们假定信任材料集合中只含有证书形式的信任材料, 这些证书根据不同的 ID 号来区别。每个证书都有它对应的获取策略, 即必须满足某个条件的情况下才可以将这个证书提交给对方。

策略判别和生成模块 P 包含了判断执行对方策略和生成本地策略两部分。对方策略一般是要求取得本地哪些证书, 而产生的本地策略就是为了提供对方所需要的这些证书, 希望从对方取得相应的证书。于是, 我们需要对每一个本地证书设置一个获取策略, 指明对应于该证书, 获取它所必须得到的证书, 可以设计成如表 1 形式的本地策略表。

表 1　本地策略表

本地证书	获取策略
X_1	$Y_{11}, Y_{12}, \cdots, Y_{1k}$
X_2	$Y_{21}, Y_{22}, \cdots, Y_{2m}$
X_3	NULL
…	…
X_n	$Y_{n1}, Y_{n2}, \cdots, Y_{ns}$

表 1 中每个本地证书对应的获取策略中可能有多个证书, 这些证书之间的关系可以是“与”或者“或”, 表示是需要全部获取还是只需要获得其中的一个就可以提交相应的本地证书。当获取策略为“NULL”时, 表示该本地证书可以直接发送给对方。为了方便讨论, 我们假设每个本地证书对应的获取策略中至多只有一个证书。

服务器为获得客户机的资源 *R*, 双方开始进行协商。客户机端设置有一个存放收到的服务器证书的集合 *W*, 则初始状态时 *W* 为空。下面描述当客户机方收到对方发送来的二元组 (Y_i, Q_i) 后的处理算法:

```
void ProcessCP ( struct Cert Y_i, struct Policy Q_i)
{
  //首先处理二元组中的证书 Y_i
  if ( T 模块验证 Y_i 的正确性 == False) {
    发送错误信息报文;
    return;
  }
  else {
    将 Y_i 加入到 W 中, 检查 W 中的所有证书是否能满足 R 的提交条件;
    if( 可以提交 R) {
      发送 R 给对方;
      return;
    }
    else
      GetCert = 由 Y_i 中所有的证书经过匹配本地策略表的获取策略中的证书, 进行反向查找获取的本地证书集合;   (1)
  }
  //下面处理二元组中的策略 Q_i
  P 模块执行策略 Q_i, Q_i 要求发送本地证书 X_j;   (2)
  对证书 X_j 在本地策略表的本地证书中进行查找;
  if( 查找不到 X_j 对应的条目) {
    发送错误信息报文;
    return;
  }
  else if( X_j 对应的获取策略 == NULL) {
      组合新二元组, 其中证书为( GetCert∪ X_j), 策略为调用 P 模块生成的新策略(不要求对方证书);
      return;
    }
    else {
      Y_j = X_j 对应的获取策略中的证书;   (3)
      组合新二元组, 其中证书为 GetCert, 策略为调用 P 模块生成的新策略(要求对方证书 Y_j);
      return;
    }
}
```

上述算法只描述了客户端对二元组报文的处理情况, 在服务器端, 相应可以做适当的微小改动。另外, 如果每个本地证书对应的获取策略可以包含多个证书, 需要进行如下改动: (1) 中反向查找时要用到 *W* 中的所有已经获得的证书; (2) 中的 X_j 和 (3) 中的 Y_j 都可以是多个证书。

不过, 如果仅依靠上述处理二元组的算法, 协商是有可能无法终止的。因为这里通信双方是完全陌生的, 可能其中的策略设置会出现循环请求证书, 比如客户端请求服务器证书 C_1, 服务器端 C_1 的获取策略是客户端的证书 C_2, 而客户端 C_2 的获取策略又是 C_1, 这样会来回请求这两个证书, 却永远不会终止。为了避免这种情况的产生, 我们可以在本地设置一个判断是否产生循环证书请求的过程加入到协商流程中。限于篇幅, 这里就不再描述这个过程。另外, 考虑到多轮交换可能让双方耗时太厉害, 可以在算法中确定一个发送二元组次数的阈值, 一旦超过这个值, 协商就以失败告终。

4　结束语

SSL 协议虽然由于它的简单、安全而在如今的网络软件中得到了广泛的应用, 但是, 它的信任机制是不完善的。本文介绍了一种逐步建立双方信任的新方法: 信任协商机制, 并运用它来补充 SSL 协议, 使得原本完全陌生的通信方可以在实现利用 SSL 加密通信的基础上取得互相信任。这种结合的方式将在实际应用中使 SSL 协议产生更大的应用价值。

参考文献:

[1] Dierks, C Allen. The TLS Protocol version 1.0[S]. RFC 2246, 1999.

[2] W Winsborough, K Seamons, V Jones. Automated Trust Negotiation [C]. DARPA Information Survivability Conference and Exposition, Hilton Head, South Carolina, 2000.

[3] William Stallings. 密码编码学与网络安全(原理与实践)[M]. 杨明, 等. 北京: 电子工业出版社, 2001.

[4] M Blaze, J Feigenbaum, J Ioannidis, *et al*. The KeyNote Trust Management System version 2[S]. RFC 2704, 1999.

[5] Eric Rescorla. SSL 与 TLS[M]. 崔凯. 北京: 中国电力出版社, 2002.

作者简介:

金仑(1978-), 男, 浙江嘉兴人, 硕士研究生, 研究方向为计算机网络安全; 谢俊元(1961-), 男, 教授, 博士生导师, 研究方向为信息安全和人工智能。