

一个组播源认证方案的安全分析

何永忠^{1,2}, 冯登国²

(1. 北京交通大学计算机学院, 北京 100044; 2. 中国科学院软件研究所信息安全国家重点实验室, 北京 100080)

摘 要: 组播源认证是组播通信中的一个研究热点。对一个基于不可靠通信信道的组播源认证方案进行了安全分析, 给出了通过选择性地截留部分通信数据包, 成功伪造了一个新的流签名的攻击方法。基于 Chernoff 界, 讨论并给出了对原方案的参数设置的改进和限制, 从而提高了方案的安全性, 避免选择性截留攻击。

关键词: 组播; 源认证; 选择性截留攻击

Security Analysis on a Multicast Source Authentication Scheme

HE Yongzhong^{1,2}, FENG Dengguo²

(1. School of Computer, Beijing Jiaotong University, Beijing 100044;

2. State Key Laboratory of Information Security, Institute of Software, Chinese Academy of Sciences, Beijing 100080)

【Abstract】 Multicast source authentication is one of the main challenges of securing multicast communication. The non-repudiation multicast source authentication scheme proposed by Li and Huai is claimed to be very efficient and robust to packet loss. However, with selectively intercepting and discarding of some packets, the paper shows that this scheme is vulnerable to forgery attacks. Based on Chernoff bound, the attack can be defeated by careful selection of security parameters of the scheme.

【Key words】 Multicast; Source authentication; Selective interception attack

组播技术是因特网中诸多应用的基础, 如网络会议、视频点播等。组播安全机制的研究对组播技术的推广有着重要意义。组播安全研究的方向主要有组播密钥管理、组播源认证、接收者访问控制、组播数字指纹等。其中, 组播源认证是组播安全研究的一个重要方面。组播源认证的主要要求有:

- (1) 可认证性, 数据接收者可以验证数据发起者的身份;
- (2) 完整性, 数据接收者可验证接收的数据未被修改过;
- (3) 不可否认性, 数据发起者不能否认相应的数据是他发出的;
- (4) 效率, 发送数据和接收数据需要的计算时间代价、通信代价和储存代价较小。

在组播通信中有大量的数据接收者, 仅仅使用消息认证码方案不能满足不可否认性的要求, 而使用一般的消息数字签名方案又不能满足实际系统对效率的需求^[1]。因此, 最近几年, 数据流组播的源认证方案成为研究热点, 并涌现出了多种方案^[1-3]。文献[4]中针对组播通信一般采用不可靠传输的特性, 提出了一个在部分数据包丢失的情况下依然能够进行高效源认证的方案EMAS, 然后在通常的签名安全定义下, 证明了方案的安全性。

本文通过对 EMAS 的安全性分析, 找到一种特殊的攻击方法, 该方法通过选择性地截留部分通信数据包, 从而成功地伪造数据流通过认证。虽然原方案并不安全, 但是, 根据 Chernoff 界, 发现适当地选择参数, 可以大大降低攻击成功的可能性。在此基础上, 本文还进一步讨论并给出了对原方案的参数设置的改进和限制, 从而提高方案的安全性, 避免选择性截留攻击。

1 EMAS方案简述^[4]

数据流组播的发送者将要发送的消息分割成一些链, 对每一条链的认证处理是相同的。假设一条链记为 M 。链 M 由 l 多个数据包列 P 组成, 而每个数据包列由 n 个数据包 p 组

成。形式化地, 数据链可表示为

$$M = \bigcup_{i=1}^l P_i, \quad P_i = \{p_{ij} \mid j=1, 2, \dots, n\}$$

方案中用到的其他记号定义如下: H 是抗碰撞的安全哈希函数; (Gen, Sig, Ver) 是消息签名体制; SK_A, PK_A 表示组播数据流的发送者 A 的私钥和公钥; 算法 $Sig(SK_A, m)$ 和 $Ver(PK_A, m, s)$ 分别表示用 A 的私钥和公钥对消息 m 的签名和验证, 如果 $s = Sig(SK_A, m)$ 则 $Ver(PK_A, m, s) = 1$; $F = \{f_{ij} \mid 1 \leq i \leq n, 1 \leq j \leq n\}$ 为单比特输出的哈希函数簇, $f_{ij}: M \rightarrow \{0, 1\}$ 。 N 为认证安全基数。当接收者接收到一个数据报列 P 中的数据包数目不小于 N 时, 可以进行安全的认证; 否则, 认证失败。

签字生成:

(1) 生成 P_l 的认证信息。

计算 $a_{lj} = H(p_{lj}), j=1, 2, \dots, n$, a_{lj} 作为 p_{lj} 的认证信息, 附加在数据报 p_{lj} 上传输。

(2) 依次生成 $P_{l-1}, P_{l-2}, \dots, P_1$ 的认证信息。

计算 $c_{rj}^i = f_{rj}(a_{i+1,r}, p_{ij})$,

其中 $i=l-1, l-2, \dots, 1; j=1, 2, \dots, n; r=1, 2, \dots, n$ 。

令 $a_{ij} = (c_{1j}^i \mid c_{2j}^i \mid \dots \mid c_{nj}^i) \in \{0, 1\}^n$, a_{ij} 作为 p_{ij} 的认证信息, 附加在数据报 p_{ij} 上传输。

(3) 生成 P_l 认证信息的数字签名。

基金项目: 国家自然科学基金资助项目(60025205, 90304007, 60373048)

作者简介: 何永忠(1969—), 男, 博士, 主研方向: 信息安全; 冯登国, 博士、研究员

收稿日期: 2005-09-26 **E-mail:** heyongzhong@bigfoot.com

$s_{1j} = \text{Sig}(SK_A, a_{1j}), j = 1, 2, \dots, n$, 附加在数据报 p_{1j} 上传输。

签字验证：

假设接收到的消息为

$$M' = \bigcup_{i=1}^l P'_i$$

其中, $P'_1 \subseteq \{(p_{1j}, k_{1j}, s_{1j}) \mid j = 1, \dots, n\}$,

$$P'_i = \{(p_{ij}, a_{ij}) \mid j = 1, \dots, n\}, (i=2, 3, \dots, l).$$

由于传输的不可靠性, 可能会丢失一些数据包, 因此每个收到的数据包列都是发送的列的子集。在下面每一步对每个数据报列 P'_i 验证时, 先检查其中数据报的数目, 如果小于 N , 则认证失败。

(1) 验证 P_1 认证信息的数字签名: $\text{Ver}(PK_A, a_{1j}, s_{1j}) = 1?$

(2) 验证 P_1, P_2, \dots, P_{l-1} 的认证信息。

令 $J_i = \{j \mid p_{ij} \in P'_i\}$, $1 \leq i < l$, 对所有的 $j \in J_i$, $r \in J_{i-1}$, 验证 $(a_{ij})_r = f_{rj}(a_{i+1,r}, p_{ij})$ 。其中, $(a_{ij})_r$ 表示 a_{ij} 的二进制表示的第 r 位 c_{rj}^i (从左边数)。

(3) 验证 P_l 的哈希值。对所有的 $j \in J_l$, 验证 $a_{lj} = H(p_{lj})$ 。

2 对 EMAS 的攻击

网络协议的经典安全模型假设, 网络通信完全由攻击者控制, 攻击者可以对网络通信的数据包直接转发、修改后转发、直接伪造或者截留。由于因特网上数据流传输一般采用不可靠的传输模式(如UDP协议), 因此接收者接收到的数据会发生随机丢失或者突发丢失(Burst Loss)的情况, 并且一般不会采用重新传输的措施。源认证协议EMAS针对数据报可能丢失的应用环境而设计, 目标是在每个数据报列 P_i 最多丢失 $n-N$ 个数据报时还能对数据源进行认证。

攻击者可以从一个数据报列中选择截留部分数据报不转发给接收者, 同时保证丢失率也小于协议规定的情况下, 伪造一个可以通过签字合法性验证的数据流。下面是该攻击方法的具体描述。

假设攻击者在攻击开始时, 已经获知一个没有丢失数据报的、有合法验证信息的签名数据链

$$M' = \bigcup_{i=1}^l P'_i$$

其中, $P'_1 = \{(p_{1j}, a_{1j}, s_{1j}) \mid 1 \leq j \leq n\}$,

$$P'_i = \{(p_{ij}, a_{ij}) \mid 1 \leq i < l, 1 \leq j \leq n\}。$$

攻击者通过网络监听的方式很容易获得这些数据。攻击者希望伪造一个签名流 \tilde{M} , 其中仅 \tilde{p}_{uv} 与 M' 中的 p_{uv} 不同 ($1 \leq u < l$, $1 \leq v \leq n$, 由攻击者任意指定), 即伪造的数据包是位于在第 u 列的第 v 个数据包。具体方法是, 随机生成一个数据报 $p \neq p_{uv}$, $\tilde{p}_{uv} = p$, 计算

$$b_{rv}^u = f_{rv}(a_{u+1,r}, \tilde{p}_{uv}) , r = 1, 2, \dots, n$$

$$\tilde{a}_{uv} = (b_{1v}^u \mid b_{2v}^u \mid \dots \mid b_{nv}^u) \in \{0, 1\}^n$$

如果 \tilde{a}_{uv} 与 a_{uv} 对应的比特位相同的数目小于 N , 就随机生成另一个 p , 重复上面的计算, 直到找到一个 p , 使得 \tilde{a}_{uv} 与 a_{uv} 对应的比特位相同的数目大于或等于 N 。假设:

$$(\tilde{a}_{uv})_{r_1} = (a_{uv})_{r_1}, (\tilde{a}_{uv})_{r_2} = (a_{uv})_{r_2}, \dots, (\tilde{a}_{uv})_{r_t} = (a_{uv})_{r_t} , t \geq N ,$$

那么攻击者可以伪造一个数据链:

$$\tilde{M} = \bigcup_{i=1}^l \tilde{P}_i$$

各个数据报列构成如下:

(1) 当 $i \neq u, i \neq u+1$, 则 $\tilde{P}_i = P'_i$;

(2) 当 $i = u$, $u \neq 1$, 则 $\tilde{P}_u = P'_u - \{(p_{uv}, a_{uv})\} + \{(p', a_{uv})\}$,

(3) 当 $i = u$, 且 $u=1$, 则

$$\tilde{P}_1 = P'_1 - \{(p_{1v}, a_{1v}, s_{1v})\} + \{(p', a_{1v}, s_{1v})\} ;$$

(4) 当 $i = u+1$ 时, 则

$$\tilde{P}_{u+1} = \{(p_{u+1,r_1}, a_{1r_1}), (p_{u+1,r_2}, a_{1r_2}), \dots, (p_{u+1,r_t}, a_{1r_t})\}$$

下面说明

$$\tilde{M} = \bigcup_{i=1}^l \tilde{P}_i$$

可以通过接收者的签字合法性验证。

(1) 检查所有数据包列中数据包的数目, 第 $i = u+1$ 个数据包列的数据包数目最少, 等于 t , 而 $t \geq N$, 所以满足条件。

(2) 验证 \tilde{P}_1 认证信息的签名。如果攻击者伪造的数据包在第 1 个数据包列中, 那么根据伪造的方法(见上述伪造方法第 3 条, 当 $u=1$ 时的规定), 认证信息 a_{1v} 和签名都没有更改, 所以可以通过认证。如果伪造的不是第 1 数据包列中的数据包, 则根据 $\tilde{P}_i = P'_i$, 第 1 个数据包列没有任何改变, 所以也可以通过认证。

(3) 验证 $\tilde{P}_1, \tilde{P}_2, \dots, \tilde{P}_{l-1}$ 的认证信息。与源数据包列相比, 在这些数据包列中, 除了 u 数据包列中的一个数据包被篡改, 和 $u+1$ 列中部分数据包被攻击者截留外, 其他数据包列没有变化。根据验证条件, 只需要检查 $(a_{uv})_r = f_{rv}(a_{u+1,r}, p_{uv})$ (对所有可能的 r) 是否成立即可。根据伪造方法, 这是成立的。

(4) 验证 \tilde{P}_l 的哈希值。因为伪造方法要求伪造的数据包不在 l 列 ($u < l$) , 所以第 l 列中的数据包只可能被截留, 不会被篡改, $a_{lj} = H(p_{lj})$ 成立。

根据上面的分析, 伪造的数据流可以成功通过签名验证。

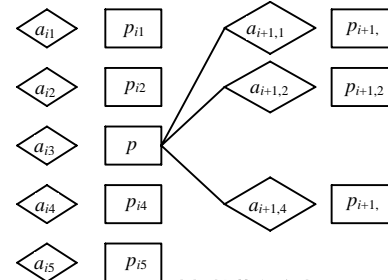


图 1 选择性截留攻击

图 1 是一个选择截留攻击的例子。假设 $n=5$, $N=3$, 图中, $\{(p_{i1}, a_{i1}), \dots, (p_{i5}, a_{i5})\}$ 是接收者收到的第 i 个数据包列, 其中 p_{i3} 被替换为攻击者伪造的数据包 p 。接收者收到的第 $i+1$ 个数据包列中只有第 1、第 2、第 4 个数据包, 而第 3 和第 5 个数据包被攻击者截留。伪造的数据包 p 满足:

$$(a_{i3})_1 = f_{13}(a_{i+1,1}, p), (a_{i3})_2 = f_{23}(a_{i+1,2}, p), (a_{i3})_4 = f_{43}(a_{i+1,4}, p)$$

可以看出, 伪造的数据流可以通过流签名验证。下面说明该攻击算法在某些合法的参数选择下是可行的。根据随机问答机模型^[5], 可以认为哈希函数是完全随机的, 那么, \tilde{a}_{uv} 与 a_{uv} 相同比特的个数 t 符合二项分布。根据原文推荐的参数, 比如 $n=256$, $N=128$, 那么对于任意生成的一个 \tilde{a}_{uv} , $t \geq N$ 的概率为 $1/2$ 。显然, 随机选择几次 p , 攻击者可以以很高的概率成功, 因此攻击是可行的。

3 对 EMAS 的改进

从前面的攻击可以看到, 如果不对流签名方案 EMAS 的参数选择进行限制, 攻击者可以很轻易地进行伪造攻击。但

(下转第 22 页)