# 加密XML文档的计算和搜索

摘要——随着行业继续采用云技术，对隐私保护搜索的需求得到了越来越大的兴趣。最近的许多努力都是针对更先进的搜索技术。尽管许多人提出了在未加密数据中进行搜索和计算的解决方案，但开发基于加密文档的有效解决方案仍然是困难的。在本文中，我们研究了处理大量类似XML格式的加密文档的问题，其中用户可能希望基于XML树中的某些元素搜索或计算。我们的解决方案利用索引表来实现快速地查询关键字和位置。为了实现在不信任的服务器上执行计算，提出了同态加密，并与对称加密一起使用，以减少计算和存储成本。

1. 介绍

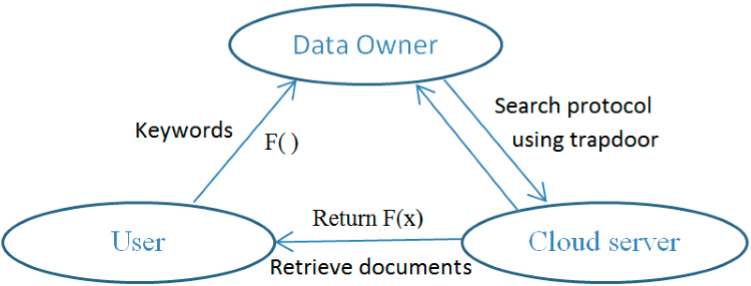
云技术保证了巨大的可伸缩性和可访问性，但是它的流行也突出了对更好的隐私性和安全性的需求。虽然人们普遍认为加密是必要的保护数据安全性，但用加密数据进行工作是很有挑战性的。我们考虑这样一个问题，即托管在云服务器上的Web服务使用大量类似XML/JSON格式的文档。为了确保隐私，文档被加密。然而，Web服务提供者希望从这些文档中提取聚合数据。一个简单的解决方案是下载和解密所有文档，以提取相关信息，这可能在带宽和计算上很昂贵，并且对于具有显著流量的流行服务来说可能是不可行的。相反，我们提出了一种解决方案，允许从加密文档本身提取有用信息。

据我们所知，从加密的XML文件获得聚合数据的问题并没有得到解决，虽然存在这些文件的查询工作。在[1 ]中，作者描述了一种用树形结构对XML文档进行加密和查询的技术。而方案是合理有效的，它的限制是它是选择类型的查询，而且使用树暴露了XML文档的结构。Sung[2]提出了使用椭圆曲线密码体制对XML文档进行加密，Juan[3]研究了减少XML数据解密开销的d技术。这两种解决方案都只考虑了部分加密的XML文档，并且也局限于选择类型的查询。许多研究人员正在积极研究加密数据的搜索问题。在[4]中，Boneh提出了一个解决方案，涉及使用公钥加密进行关键字搜索，它允许在不泄露电子邮件内容的情况下搜索一组关键字。Waters [5]解决了在加密审计日志上搜索的问题。虽然早期的工作集中在单个关键字搜索上，但是最近的工作已经考虑了涉及多个关键字的联合关键字搜索[6]，[7]。更先进的搜索技术，如短语搜索[8 ]，[9 ]和模糊关键字搜索[10 ]也进行了研究。关于加密数据上的数据挖掘技术，Du[11]提出了一种安全近似匹配技术，而Goethals[12]研究了标量积的计算，Damgard[13]提供了安全比较的解决方案。

本文提出了一种将加密文档的搜索和计算结合起来从XML文档中提取聚合数据的方案。与现有方案不同，我们的方法可以用于部分或完全加密的XML文档，并允许在服务器端执行计算，扩展到包括sum、avg等函数的查询。我们首先在第二节介绍所提出方案的通信模型，在第三节中简单描述XML格式，然后在第四节和第五节介绍关键字搜索所需的基本协议。最后，在第六节中，简要描述了同态加密以及用于搜索和计算的全部协议。

二、基于加密数据的关键字搜索与计算模型

这个方案的协议的通信模型最多涉及到三方：数据所有者、云服务器和用户。在私有云中，用户只是数据所有者。图1表明了一个以必须的关键字和函数作为参数的搜索的标准协议。首先，用户通过向数据所有者发送关键字kwi 和函数F(x)来发起请求。然后数据所有者生成陷门并将其发送给云。最终，云向用户响应所请求文档的索引和F(x)的结果。

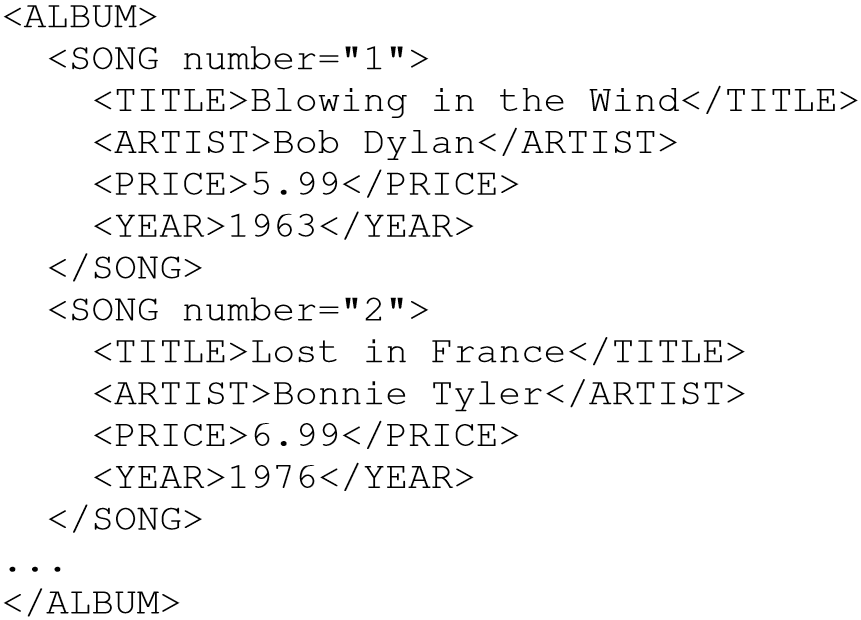


图一、加密数据搜索与计算的通信模型

三、XML 格式

XML是一种标记语言，它说明文档是如何以简单易用的格式编码的[14 ]。自成立以来，它在互联网上得到了广泛的应用，它的许多变体，如JSON，此后也一直被人们研究。

典型的XML语言包括标签、属性和内容。标签是以“<”开头并以“>结尾”的标记。属性有时被包含在标签中，以便更好地描述内容。以下是描述一张专辑中的音乐的XML 文件的示例，其中有使用诸如标题和艺术家的各种标记，和描述歌曲顺序的数字属性。



重要的是要注意文档中标签的位置不是固定的。标签的排序可以在不影响文档内容的情况下有所不同。某些元素可能存在于某些而不是所有文档中。内容的大小也可能是不一致的。因此，处理XML文件需要一种搜索算法。

1. 基本联合关键字搜索协议

在这里，我们描述了一个简单的基于索引的关键字搜索方案。

给定一个文档集合D，D = {D1,D2,...,Dn}，每个文档Di被解析为关键字kwj的列表。为了生成索引条目I(kwj)，I(kwj)={d1,d2,...,dn}，将关键字映射到文档，如果关键字kwj链接到文档Di，就将di设置为1。然后，我们将文档加密并上传到服务器。关键字到文档的索引也被加密并存储在云服务器上：

I(EK(kwj)) = {EK(d1,d2,...,dn)}. (1)

为了执行搜索，用户向数据所有者发送一组关键字kw’ = {kw1,kw2,...,kwq}。数据所有者加密搜索项，EK(kw’)，并将它们发送到云服务器。然后云服务器定位并返回加密的索引条目给数据所有者。最后，数据所有者解密并从索引条目的交集处找到匹配的文档：

I(EK(kw1)) & I(EK(kw2))…& I(EK(kwq)), (2)

其中＆表示按位和操作。

1. 加密XML文档的搜索

基本的联合关键字方案将允许用户识别与一组关键字匹配的文档。但是，它不能访问XML文档中的特定元素。例如，为了回答“Bob Dylan的歌曲的平均价格是多少？”这个问题，人们首先需要识别艺术家标签中包含Bob Dylan的专辑并检索相应的价格。注意，为了确定价格，必须首先确定列出了艺术家歌曲的文档，并且艺术家的名字包含两个连续的关键字。研究人员最近才研究加密数据中连续术语的匹配或短语搜索。虽然唐[9]和Zittrower [8]最近的工作可以在这里使用，但我们提出了一个轻量级的替代品。感兴趣的读者可以参考[ 15 ]中对短语搜索算法更详细的讨论。

联合关键字搜索和短语搜索之间的关键区别在于，后者除了包含所请求的关键字之外，它们还必须以文档中指定的顺序连续出现。为了提供短语搜索功能，我们使用关键字位置索引。为了生成关键字位置索引，我们计算

IL(H(Di|kwj)) = EK(l1,l2,...,ln), (3)

其中H()是一个安全的散列函数，lx是kwj在Di中的位置。给定用户短语搜索请求kw’ = {kw1,kw2,...,kwq}，数据所有者如第四部分所述，识别包含所有关键字的n个文档。然后，它通过向云发送i = 1到n和j = 1到q的H(Di|kwj)来查询关键字的位置。给定位置，如果对于任何 lx ∈ IL(H(Di|kw1))， (lx + j) ∈ IL(H(Di|kwj+1))，其中j = 1到q，那么数据所有者可以判定短语存在。

应该注意的是，即使在查询单个关键字时，标签或属性也可能包含查询关键字。当关键字同时属于标记和内容时，返回结果的情况很少，因此会导致错误的匹配。在将基本短语搜索协议应用于XML文档时，重要的是将某些符号作为关键字来索引，尤其是<、>和/，以区分标记和内容，并便于从XML树中识别兄弟元素、子元素和父元素。

1. 加密XML文档的计算

先前的短语搜索方案足以执行最常见的搜索。 然而，尽管AES等对称加密具有计算和存储效率，但是在没有解密的情况下无法对对称加密数据执行计算，从而导致检索数千个值的潜在昂贵过程，例如，如果要计算目录中所有相册的平均价格。

1. 同态加密

同态加密允许在密文上执行计算，其中将结果解密就相当于在明文上执行相应计算。例如，加性同态加密方案将具有以下属性：

E(A)+E(B)=E(A + B) , (4)

这一特性允许第三方在不暴露保密信息的情况下执行计算。

Paillier密码系统：Paillier密码系统[16]是文献中最流行的概率同态加密算法之一。该方案涉及三种算法：

1. 密钥生成：生成两个等长的大素数p和q。令n = p \* q, g = n + 1 , λ = φ(n) 和μ = φ(n) ^(-1) mod n，其中 φ(n) = ( p − 1)(q − 1)；
2. 加密：对于消息，m∈Zn，计算密文，c = mod ，其中r∈IMG_256；
3. 解密：对于密文，c∈IMG_256，计算消息，m = L(mod)mod n，其中L = IMG_256。

该方案是加性同态的：

E(m1,r1)+E(m2,r2)mod n^2= E(m1 + m2,r3)mod n^2. (5)

虽然没有已知的方法来计算两个密文的乘法，但是可以通过(6)来执行和明文的乘法。

E(m1,r1)^m2mod n^2 = E(m1m2,r3)mod n^2. (6)

后者用于功能更复杂的协议，例如安全标量积[12 ]。

1. 结合对称加密和同态加密

为了实现对加密数据的计算，同态加密是一种自然的选择。然而，在存储和计算中，它比对称加密要昂贵得多。为了在降低成本的同时从其功能中受益，我们建议根据文档内容调整加密方案。也就是说，只有可以用于计算的内容（这里假设为数字数据）被同态加密，而剩余内容是对称加密的。使用第三部分中的示例，我们可能有S(</ARTIST> )，(<PRICE>)，(5.99)，(</PRICE>)。 在不失一般化的情况下，我们将使用Paillier的密码系统来描述我们的方案。

实现加密XML文档的计算：

1. 文档加密：对每个文档Di进行解析和加密，以便使用(m)对数字数据进行加密，如第六部分第（1）节所述，其余内容使用AES，(m)进行对称加密。
2. 文档索引：每个文档Di都按第四和第五部分中的描述进行解析和索引。
3. 关键字搜索：要计算F(x)，其中tagm = kw1，kw2 ... kwq和x∈tagx，我们首先必须解析子句，我们可以使用关键字到文档的索引I(kwj) 和关键字位置索引IL(H(Di|kwj))（如果查询的是短语）搜索关键字kwi。然后我们查找匹配文档中tagx的位置。最后还要查询标记符号位置以解决关键字和标记之间的潜在冲突。
4. 函数计算：一旦确定了函数参数的位置，云服务器就使用(x)计算F(x)，其中x∈tagx。例如，如果想要得到平均价格，那么tagx = ‘price’，F(x)=(x)。用户接收到F(x)后，对其进行解密求得总和，然后除以匹配元素的数量求得平均值。

注意，函数F(x)的总和没有限制。只要修改较少的部分，我们就可以使用其它更复杂的函数，例如标量积[12]和比较[13]。我们的解决方案需要交换两轮查询和函数的加密结果。 在计算方面，查询由二进制搜索，哈希计算和相等比较组成，每个查询都可以有效地完成。计算F(x)的效率取决于所使用的函数和加密算法。在求和方面，Paillier的密码系统由一个简单的整数加法模组成。由于大多数XML文档相对于文本由很少的数值组成，因此文档加密的计算成本并不被认为比纯对称加密差。

值得注意的是，仅通过检查密文来确定密文是由AES加密还是Paillier加密产生的是不可行的。因此，云服务器不可以从加密文档中得到额外信息。如果需要进一步的安全性，可以请求在其他对称加密数据上计算F(x)来隐藏所需数值的位置，但代价是计算和通信成本。其他安全措施包括，在关键字到文档的索引和关键字位置索引的设计中，以牺牲效率为代价来防御统计分析[9]，[8]。

1. 总结

我们描述了一种搜索和计算类似XML格式的加密文档的方案，其计算和通信成本都比较低。由于XML文档中的标签位置不固定，所以我们需要一个搜索算法。特别是，许多常见搜索术语需要短语搜索算法。索引提供了一种访问文档和关键字位置的有效方法。为了实现计算，使用了同态加密。然而，在实际应用中，它的计算和存储成本可能过高。为了最小化它对方案的影响，我们限制它只用于加密数值。尽管效率很高，但该方案很容易受到统计分析的攻击。特别是，在已知明文模型中用常用标签设置的文档可以向好奇的云服务器泄露有意义的信息。虽然[9]，[8]中描述的方法可以提供一些安全措施，但它们很昂贵并且会大大降低算法的实用性。未来的工作，我们打算研究一些技术，以提高统计分析方面的安全性，同时保持效率。