COMSM0007 Assignment: A Summary of Public Key Encryption with keyword Search

Dominic Moylett – dm1905@my.bristol.ac.uk

May 7, 2015

1 Introduction

Public Key Encryption with keyword Search [4] is a paper by Dan Boneh, Giovanni Di Crescenzo, Rafail Ostrovsky and Giuseppe Persiano published at Eurocrypt 2004. This paper looks at the ability to search for items contained within encrypted data without the private key.

An example application for this problem is an email server: Alice can grant the server the ability to check if an email has the keyword 'Urgent'. When the server receives an email, it checks if it has the keyword: if so the email is routed to Alice's pager, otherwise it is sent to her laptop. Non-interactive public key encryption with keyword search (or 'searchable encryption') schemes mean that the server can check a ciphertext for keywords without learning what the other keywords are, or any information about the plaintext besides whether or not it contains those keywords.

Due to the computational cost of public key encryption in practice, a searchable encryption scheme does not allow a computer to search the entire encrypted message for keywords. Instead, Alice gives the machine access to a list of keywords which can be searched for. When Bob sends an email, he selects a number of keywords and sends them using a *Public-Key Encryption with keyword Search* (PEKS) of each keyword. When the server receives the email, it checks if any of the keywords in its collection are in the list of keywords sent by Bob. This is one of the disadvantages of using searchable encryption for emails; because each keyword is essentially an inividual public key encryption, it becomes very costly as more keywords are added.

Another disadvantage of searchable encryption in this form is that there is no way of verifying that the keywords are actually relevant to the email itself. It could be the case that Bob is sending useless spam to Alice, yet it is being sent directly to her pager just because he adds the keyword 'Urgent' as a searchable encryption. The server has no way of checking if this is the case, as it cannot decrypt the message without Alice's private key.

2 Definitions

Boneh et al. describe a non-interactive public key encryption with keyword search scheme as featuring four algorithms, which must run in random polynomial time. Alongside the key-generation algorithm, there must be: a function PEKS, which given Alice's public key returns a searchable encryption for a keyword; a function Trapdoor, which given Alice's private key returns a trapdoor for a keyword; and a function Test, which given a public key, checks that a searchable encryption and trapdoor were generated for the same keyword.

When Bob sends an email to Alice, he encrypts the email with Alice's public key and then selects a number of keywords. He uses PEKS with Alice's public key to produce a searchable encryption for each keyword and then sends the encrypted message followed by each of the keywords to Alice. The email is received by Alice's server, which has trapdoors for a number of keywords generated by Alice using her private key. The server then uses Test to see if the email contains one of the keywords in its database.

One of the disadvantages of this definition is that the overall scheme (encryption scheme of the message and searchable encryption scheme of the keywords) is not IND-CCA secure. Because the ciphertext is simply the encrypted message followed by a list of keywords, an adversary could reorder the keywords arbitrarily to get a different yet still valid ciphertext, and send this to the decryption oracle to retrieve the underlying plaintext. So some further work is required to ensure the indistinguishability of the overall email.

Accompanying this semantic definition is a security definition, specified as a PEKS Security game against an active attacker $\mathcal{A}$. The game is defined similarly to IND-CCA. At the start of the game public and private keys are generated and the adversary is given access to the public key. The adversary eventually selects two keywords and sends them to the challenger. The challenger picks one of the two keywords and sends a searchable encryption of the keyword to the attacker. The attacker wins if it can eventually guess which of the two keywords was chosen.

Throughout the whole game, the adversary has access to a Trapdoor oracle. The adversary is allowed to adaptively send keywords to the oracle at any point in the game and will receive in turn a trapdoor for those keywords. The only caveat is that the adversary cannot send the same keyword to both the challenger and the oracle, otherwise our adversary could request the trapdoor for one of the challenge keywords and use Test to trivially determine the keyword selected.

A searchable encryption scheme is thus defined as 'semantically secure against an adaptive chosen keyword attack' if no polynomial time attacker can win with non-negligible advantage.

Note that since the adversary has the public key, they can adaptively call PKES a polynomial number of times. Thus in order for the adversary to not win this game trivially via exhaustive search, PKES must be a random function.

3 Constructions

The remainder of the paper gives two constructions of public key searchable encryption schemes.

The first is based on bilinear maps. Bilinear maps are functions $e : G_1 \times G_1 \rightarrow G_2$ for groups G_1, G_2 of prime order p which are efficiently computable and satisfy these properties:

- For integers $x, y \in \{1, ..., p\}$, $e(g^x, g^y) = e(g, g)^{xy}$.
- If g is a generator of G_1 then $e(g, g)$ is a generator of G_2.

Using e and two hash functions $H_1 : \{0,1\}* \rightarrow G_1$ and $H_2 : G_2 \rightarrow \{0,1\}^{\log p}$, a searchable encryption scheme can be devised.

The scheme is proven semantically secure against an adaptive chosen keyword attack in the random oracle model. This is proven via a reduction to the Bilinear Diffie-Hellman Problem: if an adversary $\mathcal{A}$ can break the scheme in polynomial time with non-negligible advantage ϵ, an adversary $\mathcal{B}$ can be constructed which, given $g, g^a, g^b, g^c \in G_1$ as input, can compute $e(g, g)^{abc} \in G_2$ with advantage $\frac{\epsilon}{e q_T q_{H_2}}$, where q_T and q_{H_2} are the number of calls to Trapdoor and H_2 respectively.

The second construction is based on any trapdoor permutation. A result from Bellare et al. [3] states that given a trapdoor permutation, we can construct a source indistinguishable public key encryption scheme. A public key encryption scheme is source indistinguishable if, given a ciphertext, it is computationally difficult to determine the public key that encrypted it. The scheme works by A_{pub} being a set of public keys – one for each keyword – and A_{priv} being their corresponding private keys. The security of this scheme is shown by a reduction: If the keywords can be distinguished, the underlying encryption scheme is not source indistinguishable.

The advantage of this second scheme is that its proof of security does not rely on random oracles. But it has a cost in that the public and private keys grow linearly in size as the number of the keywords increase. Instead of using a single source indistinguishable public key for each keyword, we use a set of keys. These keys are combined to form a cover-free family, which allows us to re-use the keys. As long as Alice only creates t trapdoors for the server, a cover-free family can be created for each keyword using q keys, where q is a parameter defined by [8].

A reduction shows that if the PKES scheme has an adversary with advantage ϵ, then there exists an algorithm that breaks source indistinguishable public key encryption scheme and aborts with probability $\frac{1}{poly(t,q,d)}$, where q and d are parameters from [8]. Running this algorithm repeatedly until it doesn't abort yields an expected polynomial time adversary with advantage $\frac{\epsilon}{q^2}$. This reduces the key size, but the scheme is now limited as Alice can only create t trapdoors.

4 Importance and Impact

Boneh et al. were not the first people to investigate the topic of searchable encryption, which was instead done by Song, Wagner and Perrig [12]. But Song, Wagner and Perrig's research looked into private key searchable encryption, as opposed to public key. Boneh et al. were also the first to provide definitions of syntax and security for searchable encryption, whereas Song, Wagner and Perrig instead argued that their scheme was secure by proving it was computationally indistinguishable from random text.

Boneh et al. were also not the only people working on this subject at the time. A similar concept was developed by Goh [9] and submitted to the Cryptology ePrint Archive a matter of weeks later. The authors acknowledged each other's work in later versions of both papers.

Work inspired by this paper has focused on two general themes. The first theme is improving the ideas proposed in the original paper. One example of this is Abdalla et al. [1], who looked at the concept of consistency within the schemes, where if $\mathsf{Test}(m, C) =$ 'yes' and $\mathsf{Test}(m', C) =$ 'yes' then $m = m'$. Baek, Safavi-Naini and Susilo [2] improved the efficiency of the bilinear maps construction by removing the need for the trapdoors to be sent from Alice to the email server via a secure (encrypted and authenticated) channel. And both papers investigated extending schemes so that keywords and trapdoors are only temporary, and expire after a given time such as the end of the day. The benefit of public key encryption with temporary keyword search (PETKS) schemes is that when a server is given a trapdoor for a keyword, it is not able to learn if previous ciphertexts contain that keyword.

The second theme is investigating what other predicates can be performed on encrypted data. The above papers by Song et al. and Boneh et al. provided methods for testing equality predicates on encrypted data. Boneh and Waters [5] expanded this by developing encryption schemes catering for three more predicates: Comparison, subset and conjunctive queries. Katz, Sahai and Waters [10] developed an encryption scheme that allowed the evaluation of inner products over $\mathbb{Z}_{\mathbb{N}}$ for some large integer N, which in turn allows the evaluation of polynomial equations and disjunctions.

It is worth noting that while Public Key Encryption with keyword Search has had a large impact in the theory world, it has seen little utilisation in practice. In particular, I have been unable to find a single email system which implements a PEKS scheme, or even a paper with an empirical analysis of such a scheme. A possible reason for this might be because of the practicality of the constructions for this problem; we are unlikely to see data large enough for this to be beneficial sent via email.

But not all is lost; since 2004 when the paper was first published, a new paradigm has appeared. Boneh et al.'s work has given rise to investigations around the topic of keyword searches on encrypted data which is stored on a remote server [7]. In particular, keyword searches on data stored in the cloud is a growing trend, with research expanding to problems such as approximate keyword searching [11] and searching files for multiple keywords [6]. This area has seen experimental analysis, due to the larger amounts of data that can be stored in the cloud.

5 Conclusion

With over 1100 citations on Google Scholar, Public Key Encryption with keyword Search has had a large impact on the theory of cryptography. While it was not the first paper to investigate to topic of searchable encryption, it provided a framework for searchable encryption schemes, along with a means of defining security. This paper – along with its symmetric predecessor – has led to a large body of research in granting permission to make certain computations on encrypted data.

Although the original proposed application of an email server has seen little support, subsequent schemes have seen benefits in practice on other problems. Most notably, the growth of cloud computing has given rise to papers investigating computation on encrypted data that is stored remotely. As more and more companies switch to the cloud for their computation needs, we are likely to see even more research in this area.

References

[1] Michel Abdalla, Mihir Bellare, Dario Catalano, Eike Kiltz, Tadayoshi Kohno, Tanja Lange, John Malone-Lee, Gregory Neven, Pascal Paillier, and Haixia Shi. Searchable encryption revisited: Consistency properties, relation to anonymous ibe, and extensions. In Victor Shoup, editor, *Advances in Cryptology–CRYPTO 2005*, volume 3621 of *Lecture Notes in Computer Science*, pages 205–222. Springer Berlin Heidelberg, 2005.

[2] Joonsang Baek, Reihaneh Safavi-Naini, and Willy Susilo. Public key encryption with keyword search revisited. In Osvaldo Gervasi, Beniamino Murgante, Antonio Lagan, David Taniar, Youngsong Mun, and MarinaL. Gavrilova, editors, *Computational Science and Its Applications–ICCSA 2008*, volume 5072 of *Lecture Notes in Computer Science*, pages 1249–1259. Springer Berlin Heidelberg, 2008.

[3] Mihir Bellare, Alexandra Boldyreva, Anand Desai, and David Pointcheval. Key-privacy in public-key encryption. In Colin Boyd, editor, *Advances in Cryptology–ASIACRYPT 2001*, volume 2248 of *Lecture Notes in Computer Science*, pages 566–582. Springer Berlin Heidelberg, 2001.

[4] Dan Boneh, Giovanni Di Crescenzo, Rafail Ostrovsky, and Giuseppe Persiano. Public key encryption with keyword search. Cryptology ePrint Archive, Report 2003/195, 2003. `http://eprint.iacr.org/`.

[5] Dan Boneh and Brent Waters. Conjunctive, subset, and range queries on encrypted data. In SalilP. Vadhan, editor, *Theory of Cryptography*, volume 4392 of *Lecture Notes in Computer Science*, pages 535–554. Springer Berlin Heidelberg, 2007.

[6] Ning Cao, Cong Wang, Ming Li, Kui Ren, and Wenjing Lou. Privacy-preserving multi-keyword ranked search over encrypted cloud data. *Parallel and Distributed Systems, IEEE Transactions on*, 25(1):222–233, Jan 2014.

[7] Yan-Cheng Chang and Michael Mitzenmacher. Privacy preserving keyword searches on remote encrypted data. In John Ioannidis, Angelos Keromytis, and Moti Yung, editors, *Applied Cryptography and Network Security*, volume 3531 of *Lecture Notes in Computer Science*, pages 442–455. Springer Berlin Heidelberg, 2005.

[8] Ding-Zhu Du and Frank K. Hwang. *Combinatorial Group Testing and Its Applications.* World Scientific, 1993.

[9] Eu-Jin Goh. Secure indexes. Cryptology ePrint Archive, Report 2003/216, 2003. `http://eprint.iacr.org/`.

[10] Jonathan Katz, Amit Sahai, and Brent Waters. Predicate encryption supporting disjunctions, polynomial equations, and inner products. *Journal of Cryptology*, 26(2):191–224, 2013.

[11] Jin Li, Qian Wang, Cong Wang, Ning Cao, Kui Ren, and Wenjing Lou. Fuzzy keyword search over encrypted data in cloud computing. In *INFOCOM, 2010 Proceedings IEEE*, pages 1–5, March 2010.

[12] Dawn Xiaoding Song, D. Wagner, and A. Perrig. Practical techniques for searches on encrypted data. In *Security and Privacy, 2000. S P 2000. Proceedings. 2000 IEEE Symposium on*, pages 44–55, 2000.