Available online at www.elixirpublishers.com (Elixir International Journal)

Computer Engineering

Elixir Comp. Engg. 116 (2018) 50133-50135

ISSN: 2229-712X

Mitigating Insider Data Theft Attacks in the Cloud

B.Ramesh, J.Rakesh, Gurram Chandana, Ch.madhavi and K.Tirumaladevi
Department of CSE, Tirumala Engineering College.

ARTICLE INFO

Article history:
Received: 12 February 2018;
Received in revised form:
8 March 2018;
Accepted: 19 March 2018;

Keywords
Cloud Computing,
User Data Security.

ABSTRACT

Cloud computing promises to significantly change the way we use computers and access and store our personal and business information. With these new computing and communications paradigms arise new data security challenges. Existing data protection mechanisms such as encryption have failed in preventing data theft attacks, especially those perpetrated by an insider to the cloud provider. We propose a different approach for securing data in the cloud using offensive decoy technology. We monitor data access in the cloud and detect abnormal data access patterns. When unauthorized access is suspected and then verified using challenge questions, we launch a disinformation attack by returning large amounts of decoy information to the attacker. This protects against the misuse of the user’s real data. Experiments conducted in a local file setting provide evidence that this approach may provide unprecedented levels of user data security in a Cloud environment.

© 2018 Elixir All rights reserved.

I. INTRODUCTION

Businesses, especially startups, small and medium businesses (SMBs), are increasingly opting for outsourcing data and computation to the Cloud. This obviously supports better operational efficiency, but comes with greater risks, perhaps the most serious of which are data theft attacks. Data theft attacks are amplified if the attacker is a malicious insider. This is considered as one of the top threats to cloud computing by the Cloud Security Alliance .While most Cloud computing customers are well-aware of this threat, they are left only with trusting the service provider when it comes to protecting their data. The lack of transparency into, let alone control over, the Cloud provider’s authentication, authorization, and audit controls only exacerbates this threat.

The Twitter incident is one example of a data theft attack from the Cloud. Several Twitter corporate and personal documents were ex-filtrated to technological website TechCrunch and customers’ accounts, including the account of U.S. President Barack Obama, were illegally ac-cessed .The attacker used a Twitter administrator’s password to gain access to Twitter’s corporate documents, hosted on Google’s infrastructure as Google Docs.

The damage was significant both for Twitter and for its customers. While this particular attack was launched by an outsider, stealing a customer’s admin passwords is much easier if perpetrated by a malicious insider. Rocha and Correia outline how easy passwords may be stolen by a malicious insider of the Cloud service provider The authors also demonstrated how Cloud customers’ private keys might be stolen, and how their confidential data might be extracted from a hard disk. After stealing a customer’s password and private key, the malicious insider get access to all customer data, while the customer has no means of detecting this unauthorized access.

Much research in Cloud computing security has focused on ways of preventing unauthorized and illegitimate access to data by developing sophisticated access control and encryption mechanisms. However these mechanisms have not been able to prevent data compromise. Van Dijk and Juels have shown that fully homomorphic encryption, often acclaimed as the solution to such threats, is not a sufficient data protection mechanism when used alone.

We propose a completely different approach to securing the cloud using decoy information technology, that we have come to call Fog computing. We use this technology to launch disinformation attacks against malicious insiders, preventing them from distinguishing the real sensitive customer data from fake worthless data. In this paper, we propose two ways of using Fog computing to prevent attacks such as the Twitter attack, by deploying decoy information within the Cloud by the Cloud service customer and within personal online social networking profiles by individual users.

II. SECURING CLOUDS WITH FOG

Numerous proposals for cloud-based services describe methods to store documents, files, and media in a remote service that may be accessed wherever a user may connect to the Internet. A particularly vexing problem before such services are broadly accepted concerns guarantees for securing a user’s data in a manner where that guarantees only the user and no one else can gain access to that data. The problem of providing security of confidential information remains a core security problem that, to data, has not provided the levels of assurance most people desire.

Many proposals have been made to secure remote data in the Cloud using encryption and standard access controls. It is fair to say all of the standard approaches have been demonstrated to fail from time to time for a variety of

Tele:
E-mail address: Ramesh.guts@gmail.com

© 2018 Elixir All rights reserved

reasons, including in-sider attacks misconfigured services, faulty implementations, buggy code, misconfigured services, faulty implementations, buggy code, and the creative construction of effective and sophisticated attacks not envisioned by the implementers of security procedures. Building a trustworthy cloud computing environment is not enough, because accidents continue to happen, and when they do, and information gets lost, there is no way to get it back. One needs to prepare for such accidents.

The basic idea is that we can limit the damage of stolen data if we decrease the value of that stolen information to the attacker. We can achieve this through a 'preventive' disinformation attack. We posit that secure Cloud services can be implemented given two additional security features

1)User Behavior Profiling: It is expected that access to a user's information in the Cloud will exhibit a normal means of access. User profiling is a well-known technique that can be applied here to model how, when, and how much a user accesses their information in the Cloud. Such 'normal user' behavior can be continuously checked to determine whether abnormal access to a user's information is occurring. This method of behavior-based security is commonly used in fraud detection applications. Such profiles would naturally include volumetric information, how many documents are typically read and how often. These simple user-specific features can serve to detect abnormal Cloud access based partially upon the scale and scope of data transferred

2)Decoys: Decoy information, such as decoy documents, honey files, honey pots, and various other bogus information can be generated on demand and serve as a means of detecting unauthorized access to information and to 'poison' the thief's ex-filtrated information. Serving decoys will confound and confuse an adversary into believing they have ex-filtrated useful information, when they have not. This technology may be integrated with user behavior profiling technology to secure a user's information in the Cloud. Whenever abnormal access to a cloud service is noticed, decoy information may be returned by the Cloud and delivered in such a way as to appear completely legitimate and normal. The true user, who is the owner of the information, would readily identify when decoy information is being returned by the Cloud, and hence could alter the Cloud's responses through a variety of means, such as challenge questions, to inform the Cloud security system that it has inaccurately detected an unauthorized access. In the case where the access is correctly identified as an unauthorized access, the Cloud security system would deliver unbounded amounts of bogus information to the adversary, thus securing the user's true data from unauthorized disclosure. The decoys, then, serve two purposes:

(1) validating whether data access is authorized when abnormal information access is detected, and

(2) confusing the attacker with bogus information. We posit that the combination of these two security features will provide unprecedented levels of security for the Cloud. No current Cloud security mechanism is available that provides this level of security.

We have applied these concepts to detect illegitimate data access to data stored on a local file system by masqueraders, i.e. attackers who impersonate legitimate users after stealing their credentials. One may consider illegitimate access to Cloud data by a rogue insider as the malicious act of a masquerader. Our experimental results in local file system setting show that combining both techniques can yield better detection results, and our results suggest that this approach may work in a Cloud environment, as the Cloud is intended to be as transparent to the user as a local file system. In the following we review briefly some of the experimental results achieved by using this approach to detect masquerade activity in a local file setting.

Combining User Behavior Profiling and Decoy Technology for Masquerade Detection

User Behavior Profiling: Legitimate users of a computer system are familiar with the files on that system and where they are located. Any search for specific files is likely to be targeted and limited. A masquerader, however, who gets access to the victim's system illegitimately, is unlikely to be familiar with the structure and contents of the file system. Their search is likely to be widespread and untargeted.

Based on this key assumption, we profiled user search behavior and developed user models trained with a one-class modeling technique, namely one-class support vector machines. The importance of using one-class modeling stems from the ability of building a classifier without having to share data from different users. The privacy of the user and their data is therefore preserved.

We monitor for abnormal search behaviors that exhibit deviations from the user baseline. According to our assumption, such deviations signal a potential masquerade attack. Our previous experiments validated our assumption and demonstrated that we could reliably detect all simulated masquerade attacks using this approach with a very low false positive rate of 1.12%

Decoy Technology: We placed traps within the file system. The traps are decoy files downloaded from a Fog computing site, an automated service that offers several types of decoy documents such as tax return forms, medical records, credit card statements, e-bay receipts, etc. [10]. The decoy files are downloaded by the legitimate user and placed in highly-conspicuous locations that are not likely to cause any interference with the normal user activities on the system. A masquerader, who is not familiar with the file system and its contents, is likely to access these decoy files, if he or she is in search for sensitive information, such as the bait information embedded in these decoy files. Therefore, monitoring access to the decoy files should signal masquerade activity on the system. The decoy documents carry a keyed-Hash Message Authentication Code (HMAC), which is hidden in the header section of the document. The HMAC is computed over the file's contents using a key unique to each user. When a decoy document is loaded into memory, we verify whether the document is a decoy document by computing a HMAC based on all the contents of that document. We compare it with HMAC embedded within the document. If the two HMACs match, the document is deemed a decoy and an alert is issued.

The advantages of placing decoys in a file system are three-fold: (1) the detection of masquerade activity (2) the confusion of the attacker and the additional costs incurred to distinguish real from bogus information, and

(3) The deterrence effect which, although hard to measure, plays a significant role in preventing masquerade activity by risk-averse attackers.

Combining the Two Techniques: The correlation of search behavior anomaly detection with trap-based decoy files should provide stronger evidence of malfeasance, and therefore improve a detector's accuracy. We hypothesize that detecting abnormal search operations performed prior to an detecting abnormal search operations performed prior to an unsuspecting user opening a decoy file will corroborate the suspicion that the user is indeed impersonating another victim user. This scenario covers the threat model of illegitimate access to Cloud data. Furthermore, an accidental opening of a decoy file by a legitimate user might be recognized as an accident if the search behavior is not deemed abnormal. In other words, detecting abnormal search and decoy traps together may make a very effective masquerade detection system.

Combining the two techniques improves detection accuracy.

We use decoys as an oracle for validating the alerts issued by the sensor monitoring the user's file search and access behavior. In our experiments, we did not generate the decoys on demand at the time of detection when the alert was issued. Instead, we made sure that the decoys were conspicuous enough for the attacker to access them if they were indeed trying to steal information by placing them in highly conspicuous directories and by giving them enticing names. With this approach, we were able to improve the accuracy of our detector. Crafting the decoys on demand improves the accuracy of the detector even further. Combining the two techniques, and having the decoy documents act as an oracle for our detector when abnormal user behavior is detected may lower the overall false positive rate of detector.

We trained eighteen classifiers with computer usage data from 18 computer science students collected over a period of 4 days on average. The classifiers were trained using the search behavior anomaly detection described in a prior paper [9]. We also trained another 18 classifiers using a detection approach that combines user behavior profiling with monitoring access to decoy files placed in the local file system, as described above. We tested these classifiers using simulated masquerader data. Figure 1 displays the AUC scores achieved by both detection approaches by user model1. The results show that the models using the combined detection approach achieve equal or better results than the search profiling approach alone.

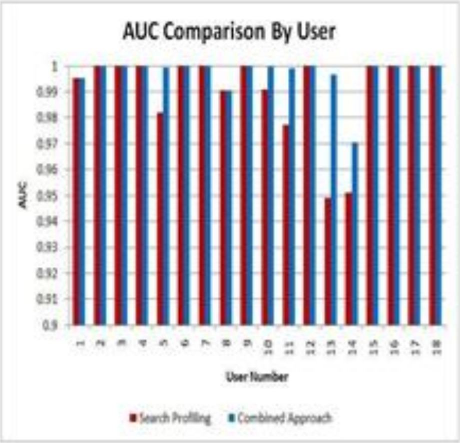

Fig 1. AUC Comparison By User Model for the Search Profiling and Integrated Approaches.

The results of our experiments suggest that user profiles are accurate enough to detect unauthorized Cloud access [9]. When such unauthorized access is detected, one can respond by presenting the user with a challenge question or with a decoy document to validate whether the access was indeed unauthorized, similar to how we used decoys in a local file setting, to validate the alerts issued by the anomaly detector that monitors user file search and access behavior.

III. CONCLUSION

In this position paper, we present a novel approach to securing personal and business data in the Cloud. We propose monitoring data access patterns by profiling user behavior to determine if and when a malicious insider illegitimately accesses someone's documents in a Cloud service. Decoy documents stored in the Cloud alongside the user's real data also serve as sensors to detect illegitimate access. Once unauthorized data access or exposure is suspected, and later verified, with challenge questions for instance, we inundate the malicious insider with bogus information in order to dilute the user's real data. Such preventive attacks that rely on disinformation