BitDeposit: Deterring Attacks and Abuses of Cloud Computing Services Through Economic Measures

Jakub Szefer and Ruby B. Lee
Dept. of Electrical Engineering
Princeton University
Princeton, NJ 08544, USA
{*szefer, rblee*}*@princeton.edu*

***Abstract*—Dependability in cloud computing applications can be negatively affected by various attacks or service abuses. To come ahead of this threat, we propose an economic measure to deter attacks and various service abuses in cloud computing applications. Our proposed defense is based on requiring a service user to pay a small deposit, using digital currency, before invoking the service. Once they are done using the service, and there has been no detected abuse or attack, the deposit is paid back by the service provider to the service user. If an attack or an abuse is detected, the service user is not paid back and the service provider gets to keep the deposit. We propose the use of micropayments with a decentralized nature and small transaction fees, such as the Bitcoin digital currency. Moreover, thanks to the existence of money exchanges which convert the Bitcoin currency to real world currency, service providers can recoup looses when they exchange the confiscated deposits for real world currency.**

***Keywords*-attack deterrence; cloud computing; digital currency; Bitcoin; digital and real world money exchanges**

I. Introduction

Dependability in cloud computing applications can be negatively affected by various attacks or service abuses. The different attacks and abuses can be deterred by requiring service users to perform some work or make a payment before they can use the service. Requiring work, such as in proof-of-work systems, can be wasteful for the legitimate users as they need to perform some computation that is later then checked by the service provider and discarded. For example, users need to find a message M that will generate hash hM with n lower bits having the value 0. Requiring a monetary payment using real world money is also not desirable as legitimate users may not be willing to actually spend, even a small amount of money to use a service, e.g. GMail or Hotmail e-mail services. Furthermore, credit card or bank transaction costs make it prohibitive to use real world currencies to make the payments.

While users may not be willing to pay for a service, we propose many would be willing to make a deposit before using the service and then get repaid when they are finished with the service. For example, a user would deposit a small amount when logging into his or her cloud-based e-mail service, send some e-mails, and when done, he or she would log out. After some period when it was determined that none of the e-mails were spam, the user would get repaid.

From a legitimate user's perspective, they would be temporarily giving up a short-term deposit, which is later refunded, so there is no real cost. The illegitimate users, however, would be deterred because when the attack or abuse is detected, they would not get their deposit back.

If our BitDeposit scheme is to be implemented, however, real world currency can not be used due to the transaction costs. The payments, and later the repayments, would have to be made then through micropayments and digital currency that has low transaction costs. But two questions remain. To deter malicious users, the currency has to actually cost something for the malicious user, either in real world money or in computation (which would be similar to the proof-of-work systems proposed before). On the service provider side, what can a service provider do with micropayments that it withholds from attackers and abusers?

We came to a resolution of the two questions by examining the recently popular Bitcoin digital currency and the ecosystem that has evolved around it. For users to obtain Bitcoins, to later use to pay for a service, they either have to "mine" Bitcoins or purchase them with real money on one of the available exchanges. Mining (explained further in Section II) is a difficult activity often requiring a lot of compute power or purchase of dedicated computer hardware. Obtaining Bitcoins requires real world money (to buy the Bitcoins at one of the exchanges or to spend on electricity and equipment to mine them) that malicious users would have to spend, and lose.

Our BitDeposit scheme leverages the benefits of Bitcoin and the ecosystem that has developed around it. Unlike proof-of-work systems, the scheme is not wasteful since there is no computation that is thrown away after each interaction with the service. While there is a lot of computation involved in generating Bitcoins, it results in generation of new Bitcoins, rather than a one-time result that is checked and discarded. Because of use of digital currency with low transaction costs, very small amounts can be paid (and later repaid) for the services so the users are not burdened. Most interestingly, however, because of the exchanges, the service

Jakub Szefer and Ruby B. Lee,
"BitDeposit: Deterring Attacks and Abuses of Cloud Computing Services Through Economic Measures,"
in Proceedings of the Workshop on Assured Cloud Computing (ACC), May 2013.

providers can actually turn the micropayments withheld from the attackers and abusers back into real world money. As one example, collected money can then be used to help pay for electricity to run the servers.

The notion of having users make a deposit that is later returned when no malicious behavior is detected has been widely explored in the area of spam deterrence. Our contribution is in combining such ideas with digital currency, i.e. the Bitcoins, and applying them to a variety of cloud computing applications (not only e-mail spam) for which a service provider can run attack detection (e.g. use VM introspection to spot malicious VMs [1]). In addition, an important contribution is the proposal to use currency exchanges to convert between digital and real world currencies in deterrence scenarios.

The remainder of this paper is organized as follows. Section II provides background on the digital Bitcoin currency. Section III describes the details of the proposed scheme for deterring attacks and abuses of cloud computing services. Section IV lists numerous challenges which make this area fertile for further research. Section V presents opportunities for increasing assurance in cloud computing through the use of schemes such as our BitDeposit scheme. Related work is listed in Section VI. We conclude in Section VII.

II. Introduction to Bitcoin

Bitcoin [2] is a distributed digital currency which has attracted a substantial number of users. For over three decades, research on cryptographic-based digital currencies has not led to as large-scale of a deployment as is seen with Bitcoin today. This decentralized, peer-to-peer system was initially designed and developed by a pseudonymous entity Satoshi Nakamoto and proposed in a self-published paper in 2008 [2]. While there are some known problems [3] with the Bitcoin system, Bitcoin has many advantages and, most importantly, a large active user base with over 10M Bitcoins in circulation as of winter 2012. Below, we provide background on Bitcoin, focusing on aspects pertinent to our BitDeposit system.

A. Bitcoin Mining

Bitcoin is based on a peer-to-peer network of users. The users are pseudonymous and are identified only through their cryptographic public keys. The user's private keys are used to sign transactions, i.e. statements specifying whom they are sending money to, and what amount. The transactions are broadcast to the Bitcoin network and logged in a "blockchain", which is a public record of all transactions. Logging the transactions and creation of new Bitcoins is captured through "mining."

Bitcoins are created through a process known as "mining" whereby certain users, namely "miners" solve computational puzzles. However, the puzzles are not arbitrary. The puzzles to solve require the miners to find a cryptographic hash with a certain number of zeros. The hash input includes all the unacknowledged transactions the miner knows about, i.e. transactions which it saw broadcast to the network. The hash input also includes a transaction that generates some amount of Bitcoins for the miners – thus the incentive to mine Bitcoins. The exact value can change, as explained below. Finally there is a nonce input. The miner repeatedly tries to find a nonce such that the SHA-256 cryptographic hash output has the desired number of zeros. Both the number of required zeros in the hash as well as the amount of the Bitcoins that the miner can get for mining is specified ahead of time – but both of these change over the lifetime of the Bitcoin system.

When a miner finds a nonce with gives the right solution, it is announced to the network. At this point all the transactions, including the miner's own transaction of giving himself or herself some Bitcoins, become permanently logged in the blockchain. Forever from now, all Bitcoin users can look up these transactions in the blockchain. The value of Bitcoins comes from a community consensus that solving the computational puzzle is worth x Bitcoins. Finding a solution to the puzzle is a way of logging a set of transactions, and the miners are then rewarded for their effort to add entries to this log of transactions, i.e. the blockchain. The values of x is reduced as the lifetime of the whole Bitcoin system progresses. This is to limit the total number of Bitcoins in circulation and also to offset advances in software and hardware implementation of the algorithms that solve the puzzles so that Bitcoins are not created too fast. The current value of x is 25, and it is projected to be halved every 4 years starting with 2017.

B. Transactions and Transfer of Money

Transactions are the means for spending Bitcoins. A user, identified by their cryptographic public key, has a number of Bitcoins. The Bitcoins are specified by the various transactions which reside in the blockchain. To spend money, the user first selects inputs: a number of transactions which gave them Bitcoins. As outputs, the user selects a recipient and the amount of Bitcoins to give to him or her. Sometimes, the total inputs are not exactly the same as the payment amount. In such cases, the transaction can also include specification of any left over change that should be returned back to the user. Hence, a transaction can have many inputs and at most two outputs (recipient and user). The user then signs the transaction with his or her private key and broadcasts it to the network.

For example, assume there are transactions T_a giving a user 1 Bitcoin, T_b giving him or her 2 Bitcoins and T_c giving him or her 3 Bitcoins. To spend 2.5 Bitcoins, the user could select T_a and T_b as inputs (total input is 3 Bitcoins). Then the user could select to give another user 2.5 Bitcoins and pay himself or herself the remaining 0.5 (total output is 3 Bitcoins). Now a net transaction T_d would be logged and it would say that the user got 0.5 Bitcoins while the other user got 2.5. In future, T_d could be used as inputs to further transactions.

It should be noted, that through a community consensus, it is customary to leave a small amount of change for the miners of the Bitcoins. In the above example, the user could actually get back 0.49, giving 0.01 Bitcoins to the miner or miners of the Bitcoins from transactions T_a and T_b. This is effectively a small transaction fee that is further motivation for miners to spend effort on mining Bitcoins.

C. Bitcoin Exchanges

A very interesting feature of the Bitcoin ecosystem is the existence of exchanges which change between real world currencies (e.g. USD, EUR, PLN, etc.) and Bitcoins. Exchanges such as Mt.Gox [4] operate web sites where users can exchange Bitcoins. The exchanges are means of cashing in Bitcoins (or buying them to later use Bitcoins to pay for services or goods). Because of the exchanges, there is a link between Bitcoins and other currencies, something not available with previous digital currencies or e-cash. Thanks to the exchanges, users can buy and sell using the pseudonymous Bitcoin system, but at any point can also get real world money. This is especially useful for service and goods providers who need real world money to buy or acquire goods, later to be sold using the digital currency.

The exchange rates between Bitcoins and real world money depend on the user demand and interest. When more users want to buy Bitcoins real world value of a Bitcoin goes up, and when there is less interest the value goes down. This has to be considered when using Bitcoins and is one of the challenges (see Section IV).

There are numerous exchanges such as Mt.Gox [4], BTC-E [5], and others. The exchange volumes and popularity of the exchanges can be traced via various other websites, such as Bitcoin Watch [6]. Moreover, some of the exchanges are becoming a registered payment services provider (PSP) [7]. This allows them to receive an international bank ID and the ability to transfer money from real world banks to the exchange's customers' Bitcoin accounts, and vice versa.

III. Deterring Attacks and Abuses with the BitDeposit System

To deter attacks and abuses in cloud computing applications, we propose an economic measure based on digital payments which use Bitcoins. The operation of our deterrence scheme is depicted in Figure 1 with the stages of operation highlighted and described in detail in the following subsections.

A. Deposit to Service

Before a user can use a service, he or she makes a digital payment to the provider. This is done via the Bitcoin network and does not require interaction with the provider yet. The amount required before service can be used will be specified by the service provider. Each provider can set his or her own level of deposit. Moreover, different users can be required to make different deposits (e.g. a new user will have to make a large deposit, while a returning user will make a lower deposit). The payment is made in the Bitcoin network and is tied to the user's public key, VK.

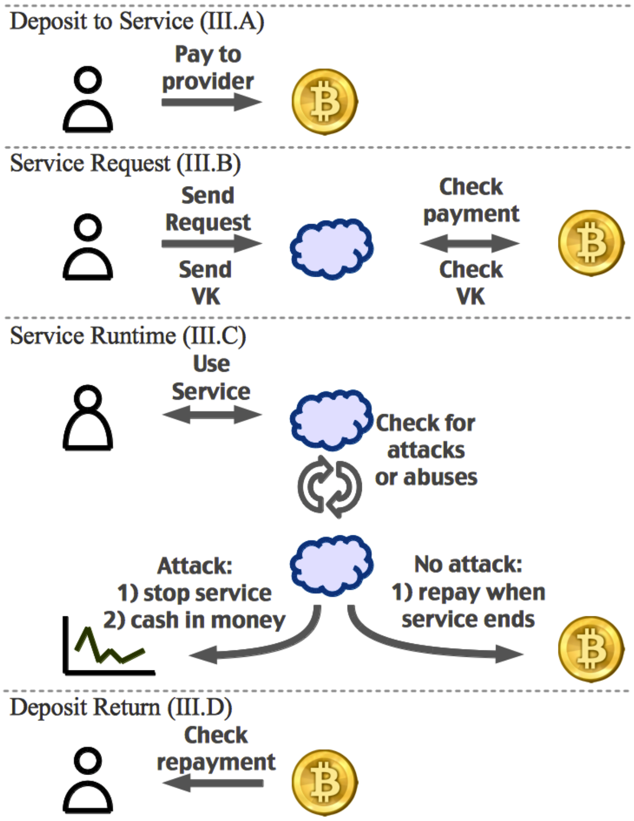

Figure 1. Operation of BitDeposit scheme, each stage is labeled with the section number containing explanation of the corresponding stage.

B. Service Request

Once a payment has been made, the user contacts the service provider to initiate a service. The user provides the public key that was used to make the payment (or the service provider could have cached a key for returning users). The VK will be later used by the service provider to verify messages sent by this user – VK is essentially an identifier that ties the user to the payment.

Upon receiving a request, the service provider needs to make sure the payment, i.e. the deposit, has been made. The provider checks with the public Bitcoin blockchain to see if the transaction of the required amount has been made to the provider. If such transaction has been recorded, the user has made a payment. The provider can use the blockchain to also check that the public key, VK, provided by the user matches the one in the Bitcoin blockchain.

C. *Service Runtime*

Once the user's payment and public key have been verified with the Bitcoin network, the service can be run. During the service runtime, the user interacts with the service (e.g. send e-mails through GMail application) – this is when a potential abuse or attack could occur (e.g. user sends spam e-mail).

The service provider needs to monitor the user's actions and detect any attacks or abuses. This can be done through use of spam e-mail detection. In other scenarios, such as IaaS cloud computing, users are leasing virtual machines (VMs), hypervisor-based virtual machine introspection can be used to detect abusive VMs.

If any attack or abuse is detected, the service is terminated. Moreover, the payment made to the service provider is not repaid back to the malicious user and the provider gets to keep this deposit. The service provider can accumulate the deposits and use the Bitcoin exchanges to turn the digital payments into real world money. The money can be used to pay for computing resources or operational costs, such as electricity, to compensate for the attack damages if any were done (e.g. the wasted computing done when the spam e-mails were sent).

If no attack or abuse was detected, the user is repaid back their amount of the deposit. Because the detection of attack or abuse may not be immediate, e.g. it will take time before spam e-mails sent out will be received and marked as spam, the repayment may be made after some amount of time.

D. *Deposit Return*

The user can monitor the Bitcoin network and the blockchain to see if a transaction back to his or her Bitcoin address has been recorded. The Bitcoin transactions are not placed in the blockchain immediately. The network is set up to generate new Bitcoins about every 10 minutes, at which time the transactions get logged. The user can monitor the network for broadcasts of a transaction and later come back to ensure it was captured in the blockchain.

Repeat users of a service could also agree for the service provider to keep the deposit for future service invocations. Rather than cycle through making deposit, receiving return, making deposit, etc., if the service provider keeps the deposit for a longer period of time, the user can keep coming back without having to make a deposit again.

IV. Challenges

Our proposed BitDeposit system presents numerous challenges which make this area viable for further research.

A. *Deposit Value*

The value of the deposit is a subjective quantity that needs to be determined by the service provider. Large deposits may detract some users, while deposits that are too small will not be effective. One means of determining the deposit value may be to examine the costs of the exploits that are sold on the black market. By examining the costs of different types of exploits, the service providers can estimate which types of software or services are popular for attack and raise the deposits for these types of services accordingly.

B. *Transaction Costs*

Our proposed scheme relies on users and service providers exchanging small amounts of digital money. The BitDeposit scheme actually repays the users less than they paid because of the transaction costs. While the costs are very small, they are non-zero. In the Bitcoin network, however, the transactions costs are actually variable, although set at a fixed percentage through a community consensus. If different transaction types were defined, similar to merchant category codes [8] associated with credit card payments, different levels of transaction costs could be defined for each transaction type. If a certain transaction type occurs often, the cost for the transaction could be lower since overall many more transactions of this type would happen still giving miners their share of profits. Now, however, the transaction participants could lie about the transaction type to get a lower cost.

C. *Bitcoin Exchange Rates and Costs*

One of the key advantages of the proposed deterrence scheme is that the malicious users will lose money while the service providers can cash in the digital payments for real world money. The exchange rates between Bitcoins and real world currencies fluctuate over time. When deploying the scheme, the payment may actually have to vary in the amount of Bitcoins (but be constant in some real world currency) so that the market fluctuations do not affect the users and service providers. As one example, a malicious provider could try to hold the Bitcoin deposits for a longer period of time so that they can sell them when the price is high, re-buy them when the price is low, and thus make extra profit before returning them to the customers.

A related challenge is also the costs of exchanging the digital currency for real world money. The exchanges keep a percentage of the money. Thus any transaction made by the service provider to get real world currency will lose some of the money. The transaction fees should be accounted for when the service providers fix the amount of payments that users should make.

D. *Breaking Pseudonomity*

One of the main advantages of Bitcoin is that it relies on pseudonymous identifiers (the public key of a user) to perform the transactions. The identifiers are not tied to real world information about the users, thus Bitcoins can be used anonymously. However, when the scheme is used with any service that requires or obtains information about the user (e.g. user's friends' e-mail addresses that they use to communicate with) the service learns some information that can be potentially used to break the pseudonomity. This breaks the pseudonomity of the public keys in the Bitcoin network for the user. This can be potentially mitigated by users changing their keys. However, now the service will see a "new" key, that is not associated with a known user and may consequently require a larger deposit, since the provider thinks it is a new unknown user who has no reputation.

E. Key Reuse

A related issue which also concerns the public keys of a user is the key reuse. To be able to tie the transactions to a user, the same key used in the Bitcoin network needs to be used to sign messages later sent to the service provider. Such reuse of the same key for different purposes (one is to make payments, the other is to sign messages) is not optimal.

F. Deposit Returns

An unanswered challenge is also how to make sure the service provider eventually returns the deposit (if there was no attack or abuse). For large service providers, reputation could be used so users would likely know that it is a good provider and that their deposit will be returned. This, however, is still an issue for small or new service providers with no reputation. Similar issues have been faced in online auctions, such as eBay, where new users have to build up reputation before they are likely to sell high-value items. The deposit return challenge in our BitDeposit system could be mitigated using such means.

V. Opportunities

Schemes such as our proposal present opportunities for increasing assurance in cloud computing in a number of areas.

A. Payment Guarantee

Because the deposits are made, and checked, before the service is invoked the provider is guaranteed that they have received money. This could allow service providers to open up their services to a wider audience. Also, service providers could reduce the amount of information they need to collect and verify before the service is run. For example, instead of the burdensome process of filling out various forms on the user side and credit card verification on the provider side, the service can be run immediately once the deposit is verified.

B. Combining Sevice Deposits and Payments

For services that actually require payment, the payment could be subtracted from the deposit. A service provider could set a larger deposit in anticipation that during the service runtime, user will use some paid service. For example, in December 2012 Facebook begun testing a potential service where users can pay a 1 USD fee to send a message to another Facebook users whom they are not friends with [9]. Such an extra message sending fee could be deduced from the deposit if Facebook utilized BitDeposit, precluding users from needing to provide credit card information to Facebook.

C. Attack Damage Mitigation

Many deterrence schemes aim to discourage attacks, but do not do much to mitigate the attack once it has happened. The ability to cash in the payments from abusive or malicious users actually lets the service providers have means to mitigate the damages. Importantly, many attacks will result in real world losses before they are detected (e.g. electricity used to run a server sending spam). The withheld deposits can be used to pay for the attack damages and compensate the service providers and potentially the victims. Interestingly, attack victims could actually receive small payments for having received spam e-mail, for example.

VI. Related Work

Our work focuses on deterring attacks and abuses through an economic approach. Much computer security work has been done deterring and preventing attacks and we only provide a short list of related works. Readers are encouraged to consult publications from venues such as the Workshop on Economics and Information Security or the "Economics and Security Resource Page" run by Ross Anderson [10].

Anti-spam measures have probably received the most attention over the years with numerous papers and books, e.g. [11], available. Deterring abusers in various services, such as to prevent spam e-mail, has been approached through proof-of-work schemes and surges, e.g. [12] presents overviews of different deterrence approaches. With much work on "proof-of-work" schemes, still researchers have argued that they do not work [13] while others present counter arguments that they do work [14]. It is not clear if such an approach is best.

An alternative to proving that a user has done some work before a service can be used, is to collect a nominal payment from the user. The fee could be paid in real world money through credit cards or similar means. The credit card transaction costs [15], however, are impractical for very small payments (many stores for example have a lower limit on credit card transaction values). The alternative is then to use digital currency, which is what we use in our proposal.

Our work uses the popular Bitcoin digital currency [2]. This is by no means the only proposed digital currency [16], [17], [18], [19], but it has been so far very successful at establishing a large user base. The BitDeposit scheme could be used with other proposed digital currencies, although any other currency used should have a large user base, low transaction costs. Also, it is very desirable to have the money exchanges.

VII. Conclusion

We proposed an economic measure to deter attacks and various service abuses in cloud computing applications. By requiring a user to deposit a small amount of digital currency, potential attackers and abusers of cloud computing services are deterred as they will lose the deposit if they behave maliciously. On the other hand, well-behaved users get their deposit back (minus a minimal transaction cost) when done using the service. The Bitcoin deposit value can also be set proportional to security level desired. The deposits use digital currency which is pseudonymous, so making deposit does not reveal any real world information about the user, while there is still guarantee that service provider will be able to cash in the deposited money for real world currency if an abuse later is detected. We believe that a cloud computing service attack or abuse deterrence though use of digital currency, with the option to exchange deposits for real world money opens up a new interesting research direction. Ours is only an initial solution to the problem and many listed challenges and opportunities make this a viable direction for further exploration.

References

[1] T. Garfinkel, M. Rosenblum *et al.*, “A virtual machine introspection based architecture for intrusion detection,” in *Proc. Network and Distributed Systems Security Symposium*, February 2003.

[2] S. Nakamoto, “Bitcoin: A Peer-to-Peer Electronic Cash System,” http://www.bitcoin.org.

[3] S. Barber, X. Boyen, E. Shi, and E. Uzun, “Bitter to better how to make bitcoin a better currency,” in *Financial Cryptography and Data Security*, ser. Lecture Notes in Computer Science, 2012, vol. 7397, pp. 399–414.

[4] “Mt. gox – bitcoin exchange,” https://mtgox.com/.

[5] “Btc exchange,” https://btc-e.com/.

[6] “Bitcoin watch,” http://www.bitcoinwatch.com/.

[7] CBC NEWS, “Bitcoin digital currency site to operate like bank in France,” December 2012, http://www.cbc.ca/news/technology/story/2012/12/07/bitcoins-banking-euro.html.

[8] Visa, “Merchant Category Codes,” 2004, http://usa.visa.com/download/corporate/resources/mcc_booklet.pdf.

[9] “Facebook tests $1 fee for inbox access,” December 2012, http://news.cnet.com/8301-1023_3-57560256-93/facebook-tests-$1-fee-for-inbox-access/.

[10] R. Anderson, “Economics and Security Resource Page,” http://www.cl.cam.ac.uk/~rja14/econsec.html.

[11] G. Schryen, *Anti-spam Measures: Analysis and Design*. Springer-Verlag Berlin Heidelberg, 2007.

[12] P. Heymann, G. Koutrika, and H. Garcia-Molina, “Fighting spam on social web sites: A survey of approaches and future challenges,” *Internet Computing, IEEE*, vol. 11, no. 6, pp. 36 –45, Nov. – Dec. 2007.

[13] B. Laurie and R. Clayton, “Proof-of-Work Proves Not to Work,” in *Proceedings of the Workshop on Economics and Information Security*, ser. WEIS, May 2004.

[14] D. Liu and L. J. Camp, “Abstract and overview proof of work can work,” in *Proceedings of the Workshop on Economics and Information Security*, ser. WEIS, June 2006.

[15] S. Mitchell, “Soaring credit card transaction fees squeeze independent businesses,” May 2009, http://www.ilsr.org/soaring-credit-card-transaction-fees-squeeze-independent-businesses/.

[16] D. Chaum, “Blind signatures for untraceable payments,” in *Proceedings of the International Cryptology Conference*, ser. CRYPTO, August 1982, pp. 199–203.

[17] T. Okamoto, “An efficient divisible electronic cash scheme,” in *Advances in Cryptology: Proceedings of CRYPTO*, 1995, vol. 963, pp. 438–451.

[18] J. Camenisch, S. Hohenberger, and A. Lysyanskaya, “Compact e-cash,” in *Advances in Cryptology: Proceedings of EUROCRYPT*, 2005, vol. 3494, pp. 302–321.

[19] S. Canard and A. Gouget, “Divisible e-cash systems can be truly anonymous,” in *Advances in Cryptology: Proceedings of EUROCRYPT*, 2007, vol. 4515, pp. 482–497.

Jakub Szefer and Ruby B. Lee,
"BitDeposit: Deterring Attacks and Abuses of Cloud Computing Services Through Economic Measures,"
in Proceedings of the Workshop on Assured Cloud Computing (ACC), May 2013.