Computer Security Symposium 2015
21 - 23 October 2015

Revisiting Block Withholding Attack in Bitcoin Cryptocurrency

Samiran Bag† Sushmita Ruj‡ Kouichi Sakurai†

†Department of Informatics
Kyushu University
{bag@inf,sakurai@csce}.kyushu-u.ac.jp

‡R. C. Bose Center for Cryptology and Security
Indian Statistical Institute
sush@isical.ac.in

Abstract This paper deals with a variant of block withholding attack. Here, we analyze the strategies of a selfish Bitcoin miner who in connivance with one pool attacks some other pool and receives incentive from the former mining pool for attacking the latter. We have named this attack as 'sponsored block withholding attack'. We have included detailed quantitative analysis of the monetary incentive that a selfish miner could earn by adopting this strategy under different scenarios. We prove that if some conditions are met the attacker can maximize her gain using some attacking strategies by utilizing her computing power wisely. We also show that an attacker may use this strategy for attacking both the pools for earning higher amount of incentives.

1 Introduction

Bitcoin is a decentralized online cryptocurrency where users can make anonymous payments by signing a transaction using her secret key. Bitcoin was first proposed by an anonymous person who identified himself as 'Satoshi Nakamoto' [4]. Mining Bitcoin requires finding a solution to the proof of work puzzle based on Back's hashcash [1]. Mining Bitcoin is requires massive amount of computational resources. Hence, multiple Bitcoin miners collaborate in constructing Bitcoin blocks by joining their computing power to simultaneously attempt to find the solution to the proof of work puzzle. This is called pooled mining. Once, a mining pool wins the mining game, all its constituent members share the block reward among themselves. Pooled mining has got both benefits as well as shortcomings. The advantage of pooled mining is that it increases the computational work force invested for mining Bitcoin. The probability of win of a small miner is very less if she mines independently. But if she joins a pool the chance of winning increases. On the other hand, this kind of pooled mining makes the pool vulnerable to selfish mining attacks. Details of this attack is provided in section 4.

2 Contribution

In this paper we study a variant of block withholding (BWH) attack. In BWH attack,

the attacker uses part of her computing power to mine solo and she uses the rest of her computing power to mine in a pool dishonestly. She pretends to mine honestly by submitting her shares regularly to the pool. Whenever by chance, she finds a valid solution of the proof of work, she opts to withhold the solution by not submitting it to the pool. Thus, she makes herself entitled to the share of the reward in case the pool wins but never actually does something fruitful for computing a proof of work for the pool. Luu et al. [3] performed detailed study of BWH attack and showed the gain of the attacker under different scenarios. They proved that this attack could indeed increase the gain of the attacker. They derived various expressions for the attackers gain under different settings. The intuition behind the attacker's incentive earned through BWH attack is that she uses part of her computing power in reducing the probability of win of a pool, thereby increasing her own chance of winning. In this work we propose what we call a "sponsored block withholding attack". We observe that by carrying out a BWH attack on a victim pool the attacker indirectly increases the probability of win of another pool (as well as of her own). Hence, she can conspire with some other pool to spend a fraction of her computing power to attack one pool. In such case, she may be rewarded by the colluding pool and the amount of reward should be proportionate to the increase of gain of the colluding pool that the attacker causes by attacking the victim pool. As such, the expected gain of the attacker will be higher than what was calculated by Luu et al. in [3]. In this paper we study the amount gain that an attacker can earn from "sponsored block withholding attack".

3 Organization

The rest of the paper is organizes as following: in section 4 we do a background study of our work. In section 5 we define the 'gain' of the miner and give the reader an idea about mining in Bitcoin network. In section 6 we describe the attack model and our assumptions in this study. In section 7, we analyze the incentive of the attacker who attacks a single pool only. In section 8, we study the incentive of the attacker who victimizes both the pools. We conclude the paper in section 9.

4 Related Work

We shall discuss these topics for our background study:

Block Withholding attack: BWH attack [5] are well known and are discussed in Bitcoin forums. In this attack rogue miners try to increase their gain by reducing the probability of win of other miners. As discussed before, many miners collude to form a mining pool in order to sum up their mining powers for yielding a massive computing powerhouse. In such a mining pool every miner needs to regularly submit a proof of work to the pool administrator. The miners are required to find a nonce such that the hash of the block header comes to be below a difficulty level Z'. This Z' is higher than the difficulty of the Bitcoin network(Z). Hence, computing such a nonce by brute force search is much easier than finding a solution for the Bitcoin network i.e. finding a nonce such that the hash value is less than Z.

In [2] Courtois and Bahack showed that a rogue miner may act dishonestly for her personal gain and put the reputation of Bitcoin in danger by indirectly using the computing power of honest miners for her own gain and thus depriving them from the incentive they

deserve. In [3] Luu et al. gave a quantitative analysis of the amount of incentive a miner may gain by carrying out block withholding attack on a victim pool. They showed that the gain of the rogue miner who uses Block withholding attack comes from the fact that she reduces the probability of win of the victim pool, thus increasing her own chance of winning the Bitcoin mining game. It was reported that on June 13, 2014, a large scale BWH attack was conducted against the mining pool Eligius [2] causing a loss of 300BTC at the expense of honest miners.

5 Preliminaries

5.1 Definitions

Definition We use the same definition of gain used in [3]. They defined the gain of the miner to be the fraction of computing power actually used by a miner for mining with respect to the actual computing power used by the entire network for mining. Note, that if some computing power is used for attacking, it is not counted as an actual mining computing power. The actual computing power belongs to those miners who either mine honestly in a pool or mine solo.

5.2 Bitcoin Mining

In Bitcoin, in order to solve the proof of work puzzle, the miners need to find a nonce such that the SHA-256 hash of the nonce along with the transaction merkle root and some other parameters related to the Bitcoin network happens to fall below a ceiling Z. This value Z determines the difficulty level of solving the mining puzzle and is updated on a regular basis in order to keep the time required for finding a solution nearly equal to 10 minutes. Solving this puzzle requires massive amount of computation. So, multiple miners collude to form a mining pool to amalgamate their computing resources to form a big computing powerhouse called mining pools. These pools are administrated by special miners. In these pools all the constituent miners strive to solve the mining puzzle. Miners regularly submit proofs convincing the pool administrator that they are indeed spending their resources to find a solution. Generally the pool administrator sets a difficulty level $Z' > Z$. So, miners who can find appropriate nonces such that the hash value falls below Z' submit the solution to the pool. So, the miners have a higher probability of solving this puzzle. This serves as a check for the pool administrator that the miners are indeed expending their computing power for mining in pool P. The number of solutions a particular miner submits to a pool is directly proportional to the computing power it expends for mining. Hence, the pool administrator may pay an incentive to a miner proportional to the number of solution it submitted to the pool in case the pool wins the mining game. Every miner of a pool attempts to find a proof of work on a transaction set that contains a coinbase transaction which separates this transaction set from the transaction sets of other mining pools or solo miners. This coinbase transaction causes the pool administrator to claim the mining reward in case the pool wins the mining game. Thus, in case a miner belonging to some pool P can find a solution of the puzzle, it will have only two choices, either to submit it to the pool administrator or to conceal the fact that it has indeed found a solution. The miner can not submit the solution to the Bitcoin network either directly or through some other pool.

6 Basic Attack Strategy

We now describe the attack in detail. We assume the computing power of the entire Bit-

coin network be 1. Let $\mathcal{A}$ be an attacker whose computing power is α. Also, let $cp(\mathcal{A})$ be the computing power of $\mathcal{A}$. She divides her computing power into two parts i.e. $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$, where $cp(\mathcal{A}) = cp(\mathcal{A}_1) + cp(\mathcal{A}_2)$. $cp(\mathcal{A}_1) = \alpha\beta$ and $cp(\mathcal{A}_2 = \alpha(1-\beta))$. Thus, the computing power of The attacker uses $\mathcal{A}_2$ for private mining and uses $\mathcal{A}_1$ for attacking the pool P. $\mathcal{A}_1$ mines in pool P and submits its 'shares' regularly to the administrator of the pool P of which $\mathcal{A}_1$ is a member. Whenever by chance $\mathcal{A}_1$ finds a valid solution of the proof of work, she does not submit it to the pool P. Neither can she directly submit the same to the Bitcoin network [3]. She instead submits it to the pool P' which is a rival of pool P. P' has a secret pact with $\mathcal{A}$. Whenever $\mathcal{A}$ withholds a block from submitting to P, the pool P' offers some rewards to $\mathcal{A}$. It is easy for P' to check the validity of a block submitted by the attacker assuming that P' knows the set of transactions the pool P is working on including the transaction merkle root making it possible for P' to verify the proof of work that the attacker withheld from pool P. Thus the gain of the attacker increases as she has got a sponsor for launching attack on P. The reward that P' gives to the attacker $\mathcal{A}_1$ is proportional to the expected amount of gain, P' makes from $\mathcal{A}$'s attack on P. Thus, both P' and $\mathcal{A}$ gains at the expense of the computational resources of honest miners of P. The expected gain of the attacker is necessarily higher than normal BWH attack where she does not fetch any incentive from rival pools for carrying out this attack.

7 Analysis of the incentive earned by the attacker who attacks a single pool

First we consider a situation when there is a single pool and an attacker. The computing power of the attacker is α. She uses β fraction of her computing power to attack the pool P having computing power p. The attacker uses rest of her computing power to mine solo. The computing power of the entire Bitcoin network is 1. Luu et al. calculated the expected gain of such attacker in [3]. They found that in this scenario, the expected gain of the attacker will be $G = \frac{\alpha^2\beta+\alpha-\alpha^2-\alpha^2\beta^2}{(1-\alpha\beta)(1-\alpha+\alpha\beta)}$. Now, Luu et al had proved that the gain of the attacker takes a peak at $\beta = 0.5$. In addition to their result we show that the gain of the attacker is an increasing function on β when $0 \leq \beta < 0.5$ and a decreasing function when $0.5 < \beta < 1$.

Lemma 1 *The gain of an attacker for the above model is an increasing function when β in the range $(0, .5)$ and is a decreasing function when β is in the range $(0.5, 1)$.*

Proof $G = \frac{\alpha^2\beta+\alpha-\alpha^2-\alpha^2\beta^2}{(1-\alpha\beta)(1-\alpha+\alpha\beta)}$. Taking partial derivative with respect to β,

$\frac{\partial G}{\partial \beta} = -\frac{(2\alpha^4-4\alpha^3+2\alpha^2)\beta-(\alpha^4-2\alpha^3+\alpha^2)}{\alpha^4\beta^4-2\alpha^4\beta^3+(\alpha^4+2\alpha^3-2\alpha^2)\beta^2+(2\alpha^2-2\alpha^3)\beta+\alpha^2-2\alpha+1}$.
At $\beta = 0$, $\frac{\partial G}{\partial \beta}|_{\beta=0} = \frac{((\alpha^4-2\alpha^3+\alpha^2)}{\alpha^2-2\alpha+1} = \alpha^2 > 0$. Hence, G is an increasing function at $\beta = 0$. Also, $\frac{\partial G}{\partial \beta}|_{\beta=0.5} = 0$. Now, the denominator of $\frac{\partial G}{\partial \beta}$ is $Den = 1+\alpha^2-2\alpha+\alpha^2(\alpha^2\beta^4-2\alpha^2\beta^3+(\alpha^2+2\alpha-2)\beta^2+(22\alpha)\beta) > 0$, since α is a small fraction. So, $\frac{\partial G}{\partial \beta}_{0.5<\beta<1} < 0$. Hence, the result. ■

Now, we shall focus on discussing the main issue of the paper i.e. the study of the gain of an attacker who uses the withheld block to earn more incentive from other selfish pools. Let us consider a situation when there are two mining pools P and P' in the entire network and a single attacker $\mathcal{A}$. The computing power of the attacker is α. The attacker($\mathcal{A}$) uses β fraction of its computing strength for mining in the pool P and uses the rest of $1-\beta$ fraction of its computing power for mining solo. The computing power of pool P and P' is p and p' respectively. The attacker $\mathcal{A}$ mines in P with $\alpha\beta$ computing power and it submits its shares

to the pool only when the share does not correspond to a valid solution of the actual proof of work. In case the attacker $\mathcal{A}$ indeed finds a valid block, it does not submit it to the pool. Neither can it submit the same to the Bitcoin network and claim the reward. Instead it submits this to the administrator of the pool P'. The pool administrator gives her some incentive for launching attack on the pool P and thus increasing the probability of win of the pool P'. Let γ be the fraction of the incentive given to the attacker $\mathcal{A}$ by the pool P' for launching block withholding attack on P. We assume that the network has no other player i.e

$$\alpha(1-\beta)+p+p'=1. \quad (1)$$

Lemma 2 *The gain of the pool P' when the attacker launches block withholding attack on the pool P is given by $\frac{\alpha\beta p'}{1-\alpha\beta}$.*

Proof If the attacker does not use her computing power to launch BWH attack on pool P, the computing work force of the entire network would have been 1. Then the expected gain of P' would be p'. But when the attacker $\mathcal{A}$ uses $\alpha\beta$ fraction of her computing power for attacking P, the effective computing power of the network goes down to $1-\alpha\beta$ and hence the gain of P' increases to $\frac{p'}{1-\alpha\beta}$. Thus, the increase of gain of P' is $\frac{p'}{1-\alpha\beta}-p'=\frac{\alpha\beta p'}{1-\alpha\beta}$. ■

Let us now calculate the expected incentive of $\mathcal{A}$. Since, $\mathcal{A}_0$ uses $\alpha\beta$ of its computing power to attack P, the active computing power of the network is $1-\alpha\beta$. Here we assume that all other miners of the pool P and all miners of P' are honest. The gain of $\mathcal{A}$ is threefold. Firstly, the gain of $\mathcal{A}$ from mining privately is equal to $G_1=\frac{\alpha(1-\beta)}{1-\alpha\beta}$. The expected share of incentive obtained for mining in P is $G_2=\frac{p-\alpha\beta}{1-\alpha\beta}\frac{\alpha\beta}{p}$. Again, the incentive obtained from P' for carrying out attack on a competitor pool (P) is given by $G_3=\frac{\alpha\beta p'\gamma}{1-\alpha\beta}$. So, the total gain of the attacker is $G=G_1+G_2+G_3=\frac{\alpha(p-\alpha\beta^2+p'\beta\gamma)}{p(1-\alpha\beta)}$.

Lemma 3 *The incentive of the attacker increases if she attacks the bigger pool and receives incentive from the smaller one.*

Proof The gain of $\mathcal{A}$ is $G=\frac{\alpha(p-\alpha\beta^2+p'\beta\gamma)}{p(1-\alpha\beta)}$. Replacing the value of p' with respect to equation 1, $G=\frac{\alpha(p-\alpha\beta^2-(1-p-\alpha+\alpha\beta)\beta\gamma)}{p(1-\alpha\beta)}=\frac{\alpha(1+\beta\gamma)}{1-\alpha\beta}-\frac{\alpha(\alpha\beta^2+\beta\gamma-\alpha\beta\gamma+\alpha\beta^2\gamma)}{p(1-\alpha\beta)}=$
$\frac{\alpha(1+\beta\gamma)}{1-\alpha\beta}-\frac{\alpha\{\alpha\beta^2+\beta\gamma(1-\alpha)+\alpha\beta^2\gamma\}}{p(1-\alpha\beta)}$. Hence, G increases as p increases. Hence, the result. ■

Note that when the attacker only used BWH attack without having a secret agreement with pool P', the expected gain it could have obtained was $G'=G_1+G_2=\frac{\alpha(p-\alpha\beta^2)}{p(1-\alpha\beta)}$. Hence, the ratio of gain is $\Delta G^O_{BWH}=\frac{G}{G'}-1=\frac{p'\beta\gamma}{\alpha(p-\alpha\beta^2)}$. Since, $\alpha\beta<p, \alpha\beta^2<\alpha\beta<p$. So, $\Delta G^O_{BWH}>0$. If the attacker was an honest miner, the incentive she would have obtained through mining honestly is $G_h=\alpha$. So the ratio of the incentive she obtains now by third party backed block withholding attack is $\Delta G^O_h=\frac{G_3}{G_h}-1=\frac{\alpha\beta p+p'\beta\gamma-\alpha\beta^2}{p(1-\alpha\beta)}$. Replacing p by $1-p'-\alpha+\alpha\beta$, we get,
$\Delta G^O_h=\frac{G_3}{G_h}-1=\frac{\alpha\beta-\alpha\beta p'-\alpha^2\beta+\alpha^2\beta^2+p'\beta\gamma-\alpha\beta^2}{(1-p'-\alpha+\alpha\beta)(1-\alpha\beta)}$.

Theorem 4 *Let $\beta_0\in(0,1)$ be such that the total gain of the attacker $G|_{\beta=\beta_0}=\max(G|_{\beta=\beta'})$: $\forall\beta'\in[0,1]$. Then β_0 should satisfy,*

$$A\beta_0^2+B\beta_0+C=0$$

where, $A=-\alpha^2(1-\gamma)p', B=4\alpha^2-2\alpha^3-2\alpha^2p'+2\alpha p'-2\alpha, C=\alpha^3-2\alpha^2+2\alpha^2p'-2\alpha p'-\alpha\gamma p'+\alpha+\alpha p'^2-\gamma p'^2+\gamma p'$.

Proof We have proved that in the proposed setting, the total gain of the attacker $\mathcal{A}$ is $G=\frac{\alpha(p-\alpha\beta^2+p'\beta\gamma)}{p(1-\alpha\beta)}$. Replacing p with $1-p'-\alpha+\alpha\beta$, we get $G=\frac{\alpha(1-p'-\alpha+\alpha\beta-\alpha\beta^2+p'\beta\gamma)}{(1-p'-\alpha+\alpha\beta)(1-\alpha\beta)}$. Differentiating with respect to β we get,
$\frac{\partial G}{\partial\beta}=\frac{\alpha(-2\alpha\beta+\gamma p'+\alpha)}{(1-\alpha\beta)(\alpha\beta-p'-\alpha+1)}+\frac{\alpha^2(-\alpha\beta^2+\gamma p'\beta+\alpha\beta-p'-\alpha+1)}{(1-\alpha\beta)^2(\alpha\beta-p'-\alpha+1)}-\frac{\alpha^2(-\alpha\beta^2+\gamma p'\beta+\alpha\beta-p'-\alpha+1)}{(1-\alpha\beta)(\alpha\beta-p'-\alpha+1)^2}$. Simplifying, we get $\frac{\partial G}{\partial\beta}=\frac{Num}{Den}$, where $Num=-2\alpha^4\beta+\alpha^4-2\alpha^3+$

$\alpha^3\gamma p'\beta^2 - \alpha^3 p'\beta^2 + 2\alpha^3 p' + 4\alpha^3\beta - 2\alpha^3 p\beta - 2\alpha^2 p' - \alpha^2\gamma p' + \alpha^2 + \alpha^2 p'^2 + 2\alpha^2 p'\beta - 2\alpha^2\beta - \alpha\gamma p'^2 + \alpha\gamma p'$ and $Den = (1-\alpha\beta)^2(1-p'-a+\alpha\beta)^2$. We can rewrite the numerator Num as $-\alpha^3(1-\gamma)p'\beta^2 + (4\alpha^3 - 2\alpha^4 - 2\alpha^3 p' + 2\alpha^2 p' - 2\alpha^2)\beta + (\alpha^4 - 2\alpha^3 + 2\alpha^3 p' - 2\alpha^2 p' - \alpha^2\gamma p' + \alpha^2 + \alpha^2 p'^2 - \alpha\gamma p'^2 + \alpha\gamma p')$. It is easy to see that $Den \neq 0 \forall \beta \in [0,1]$. If G is to be maximized at $\beta = \beta_0$, $\frac{\partial G}{\partial \beta}|_{\beta=\beta_0}$ should be equal to zero. So, $-\alpha^3(1-\gamma)p'\beta_0^2 + (4\alpha^3 - 2\alpha^4 - 2\alpha^3 p' + 2\alpha^2 p' - 2\alpha^2)\beta_0 + (\alpha^4 - 2\alpha^3 + 2\alpha^3 p' - 2\alpha^2 p' - \alpha^2\gamma p' + \alpha^2 + \alpha^2 p'^2 - \alpha\gamma p'^2 + \alpha\gamma p') = 0$. Since, $\alpha \neq 0$, we can write the equation as $-\alpha^2(1-\gamma)p'\beta^2 + (4\alpha^2 - 2\alpha^3 - 2\alpha^2 p' + 2\alpha p' - 2\alpha)\beta + (\alpha^3 - 2\alpha^2 + 2\alpha^2 p' - 2\alpha p' - \alpha\gamma p' + \alpha + \alpha p'^2 - \gamma p'^2 + \gamma p') = 0$. Hence, the result. ■

Lemma 5 *Let $F(\alpha,\gamma,\beta,p') = A\beta^2 + B\beta + C$. Then $F(\cdot)$ is a decreasing function on β when $\alpha \leq 0.25$.*

Proof $\frac{\partial F}{\partial \beta} = 2A\beta + B$. Now, $A = -\alpha^2(1-\gamma)p' < 0$. $B = 4\alpha^2 - 2\alpha^3 - 2\alpha^2 p' + 2\alpha p' - 2\alpha = -2\alpha^3 - 2\alpha^2 p' - \alpha(1-2p') - \alpha(1-4\alpha)$. Now, P' is the smaller pool and hence, $p' < 0.5$. So, $-\alpha(1-2p') < 0$. Also, $\alpha(1-4\alpha) < 0$. So, $\frac{\partial F}{\partial \beta} < 0$. So, $F(\cdot)$ is a decreasing function on β. ■

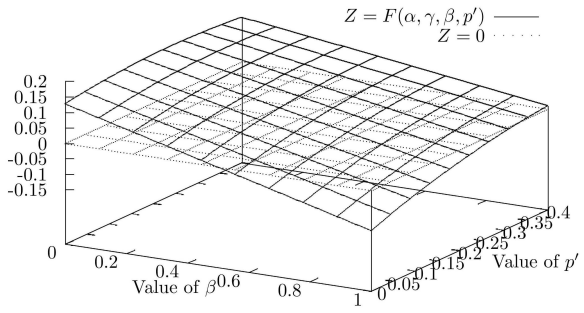

図 1: Graphical depiction of the behavior of $Z = F(\alpha,\gamma,\beta,p')$ when $\alpha = 0.2, \gamma = 0.7$

Figure 1 gives a graphical presentation of the values of $F(\alpha,\gamma,\beta,p') = A\beta^2 + B\beta + C$ where A, B and C are as defines as in Lemma 4. We have plotted the values of $F(\cdot)$ by varying the values of β and p' while keeping α fixed. The figure shows that $F(\cdot)$ takes both positive and negative values when p' is less than a certain limit. Also, we observe that $F(\cdot)$ decreases monotonically, in adherence to the result of Lemma 5. So, we can infer that there exist an β_0 such that $F(\alpha,\gamma,\beta_0,p')$ equals 0 when p' is less than a certain limit. At this point the gain of the attacker $\mathcal{A}$ peaks. Also, since $F(\alpha,\gamma,\beta_0,p') > 0$ for $\beta < \beta_0$, G is an increasing function of β for $\beta < \beta_0$. Similarly as $F(\alpha,\gamma,\beta_0,p') < 0$ for $\beta > \beta_0$, G is a decreasing function of β for $\beta > \beta_0$ when p' is less than the said limit. When p' is beyond the said limit, G is always an increasing function of β, and hence the attacker's best strategy would be to invest all her computing power in attacking the pool P. But when p' is sufficiently low, there is a $\beta_0 \in [0,1]$ that maximizes her gain. This β_0 corresponds to the intersection of the two curves $Z = F(\alpha,\gamma,\beta,p')$ and $Z = 0$ in Figure 1. The value of β_0 can be obtained by solving the quadratic equation $A\beta^2 + B\beta + C = 0$. The value of β_0 that maximizes the gain of the attacker is $\{\frac{-B+4AC}{2A}, \frac{-B-4AC}{2A}\}$. Note that since $F(\cdot)$ is monotonically decreasing, there exists exactly one value of β_0 that maximizes the gain G. This entails either $0 < \frac{-B-4AC}{2A} < 1$ or $0 < \frac{-B-4AC}{2A} < 1$ but not both. Thus, an attacker may calculate the precise value of β that maximizes her gain easily using above equation.

8 Attacking both the pools simultaneously

Let us again consider another situation where the attacker launches this attack on both the pools P and P'. Let us assume like before the computing power of the attacker is α. She splits her computing power into 3 parts; $\mathcal{A}_1, \mathcal{A}_2$, $\mathcal{A}_3$. with computing power equal to $\alpha\beta$, $\alpha\delta$

and $1-\alpha\beta-\alpha\delta$. She uses $\alpha\beta$ of her computing power to attack P and get the incentive from P'. Similarly, she uses $\alpha\delta$ of her computing power to attack P' and get an incentive from P. In such case, her expected gain from private mining will be $G_1 = \frac{\alpha(1-\beta-\delta)}{1-\alpha\beta-\delta\gamma}$. The share of reward she obtains from P is given by $G_2 = \frac{p-\alpha\beta}{1-\alpha(\beta+\delta)}\frac{\alpha\beta}{p}$. The reward she obtains by mining in pool P' is given by $G_3 = \frac{p'-\alpha\delta}{1-\alpha\beta-\alpha\delta}\frac{\alpha\delta}{p'}$. The incentive she gets from P' for attacking P is given by $G_4 = \frac{\alpha\beta p'\gamma}{1-\alpha\beta}$. Similarly, the incentive she gets from the pool P for attacking the pool P' is given by $G_4 = \frac{\alpha\delta p\gamma}{1-\alpha\delta}$. Hence, the total gain of the attacker is given by $G = \sum_{i=1}^{5} G_i = \frac{\alpha-\alpha\beta-\alpha\delta}{1-\alpha\beta-\alpha\delta} + \frac{p\alpha\beta-\alpha^2\beta^2}{p(1-\alpha\beta-\alpha\delta)} + \frac{p'\alpha\delta-\alpha^2\delta^2}{p'(1-\alpha\beta-\alpha\delta)} + \frac{\alpha\beta p'\gamma}{1-\alpha\beta} + \frac{\alpha\delta p\gamma}{1-\alpha\delta}$.

8.1 When the computing power of the pools are constant

Here, we study the gain of the attacker who attacks both the pools P and P' with the assumption that the computing power of P and P' are constant. That is when a miner invests some of her computing power in any of the pools, the pool administrator shuts down some of its miners and maintains the computing power of the system at the same level as before. Alternately, when some miner exits the pool, the administrator uses its back up computing resources to compensate the loss. Thus, the computing powers of P and P' remain the same always. The rationale behind this is to keep the overhead on the pool administrator constant. As we have discussed before, pool members regularly submit shares to the pool administrator which are then verified by the administrator. Thus, the pool administrator checks that all its members are indeed investing their declared amount of computing power for mining in the said pool and any deviation from that would be reflected in the number of shares a member submits to the administrator or in the frequency of submission. Hence, the pool administrator needs to deal with a high number of shares which inevitably requires computation. The pool administrator may have limited infrastructure to handle only a limited number of shares and in order to keep the number of shares constant it may require to limit the computing power of the entire pool. Thus it may put some of the computing resources to sleep in order to keep the mining power constant. In Lemma 6, we prove existence of optimal attack strategy of a miner who uses a fixed amount of its computing power to attack both the pools.

Theorem 6 *Let $\mathcal{A}$ be an attacker who uses e fraction of her computing power in attacking both the pools P and P' i.e. $\beta+\delta = e$. Then there exists a unique value $\beta_0 \in (0, e)$ such that the attacker's gain G is maximized for $\beta = \beta_0$ given the following condition:*

$$\gamma p'^2 + \frac{2\alpha e}{1-\alpha e} > \frac{\gamma p p'}{(1-\alpha e)^2}.$$

Proof The total gain of the attacker $G = \frac{\alpha-\alpha\beta-\alpha\delta}{1-\alpha\beta-\alpha\delta} + \frac{p\alpha\beta-\alpha^2\beta^2}{p(1-\alpha\beta-\alpha\delta)} + \frac{p'\alpha\delta-\alpha^2\delta^2}{p'(1-\alpha\beta-\alpha\delta)} + \frac{\alpha\beta p'\gamma}{1-\alpha\beta} + \frac{\alpha\delta p\gamma}{1-\alpha\delta}$. Replacing δ by $e-\alpha$ we get, $G = \frac{\alpha(1-e)}{1-\alpha e} + \frac{p\alpha\beta-\alpha^2\beta^2}{p(1-\alpha e)} + \frac{p'\alpha(e-\beta)-\alpha^2(e-\beta)^2}{p'(1-\alpha e)} + \frac{\alpha\beta p'\gamma}{1-\alpha\beta} + \frac{\alpha(e-\beta)p\gamma}{1-\alpha(e-\beta)}$. Taking partial derivative w.r.t. β, $\frac{\partial G}{\partial\beta} = \frac{p\alpha-2\alpha^2\beta}{p(1-\alpha e)} + \frac{-p'\alpha+2\alpha^2(e-\beta)}{p'(1-\alpha e)} + \frac{\alpha\gamma p'}{(1-\alpha\beta)^2} - \frac{\alpha\gamma p}{(1-\alpha(e-\beta))^2}$

$\left.\frac{\partial G}{\partial\beta}\right|_{\beta=0} = \frac{2\alpha^2 e}{p'(1-\alpha e)} + \alpha\gamma p' - \frac{\alpha\gamma p}{(1-\alpha e)^2}$. $\left.\frac{\partial G}{\partial\beta}\right|_{\beta=e} = -\frac{2\alpha^2 e}{p(1-\alpha e)} + \frac{\alpha\gamma p'}{(1-\alpha e)^2} - \alpha\gamma p$. Now, since, $\gamma p'^2 + \frac{2\alpha e}{1-\alpha e} - \frac{\gamma}{(1-\alpha e)^2}pp' > 0$,

$\alpha\gamma p' + \frac{2\alpha^2 e}{p'(1-\alpha e)} - \frac{\alpha\gamma p}{(1-\alpha e)^2} > 0$.

Since, P is the bigger pool, $p > p'$. So, $\gamma p^2 + \frac{2\alpha e}{1-\alpha e} - \frac{\gamma}{(1-\alpha e)^2}pp' > 0$.

Hence, $\alpha\gamma p + \frac{2\alpha^2 e}{p(1-\alpha e)} - \frac{\alpha\gamma p'}{(1-\alpha e)^2} > 0$.

From equation 8.1 and 8.1, we get $\left.\frac{\partial G}{\partial\beta}\right|_{\beta=0} > 0$ and $\left.\frac{\partial G}{\partial\beta}\right|_{\beta=e} < 0$. So, there must be a $\beta_0 \in (0, e)$ such that G is maximum at $\beta = \beta_0$. Now, what is left to be proven is the fact that there is a unique $\beta_0 \in (0, e)$ such that $\left.\frac{\partial G}{\partial\beta}\right|_{\beta=\beta_0} = 0$.

Now, $\frac{\partial G}{\partial \beta} = \frac{p\alpha - 2\alpha^2\beta}{p(1-\alpha e)} + \frac{-p'\alpha + 2\alpha^2(e-\beta)}{p'(1-\alpha e)} + \frac{\alpha\gamma p'}{(1-\alpha\beta)^2} - \frac{\alpha\gamma p}{(1-\alpha(e-\beta))^2}$.
So, $\frac{\partial^2 G}{\partial \beta^2} = -\frac{2\alpha^2}{p(1-\alpha e)} - \frac{2\alpha^2}{p'(1-\alpha e)} + \frac{2\alpha^2\gamma p'}{(1-\alpha\beta)^3} - \frac{2\alpha^2\gamma p}{(1-\alpha(e-\beta))^3}$. Now, $1-\alpha\beta > p > p'$. So, $(1-\alpha\beta)^2 > pp' > \gamma pp'$. Also $(1-\alpha\beta) > (1-\alpha e)$. Hence, $(1-\alpha\beta)^3 > \gamma pp'(1-\alpha e)$. So, $\frac{2\alpha^2}{p(1-\alpha e)} > \frac{2\alpha^2\gamma p'}{(1-\alpha\beta)^3}$. From this we conclude that $\frac{\partial^2 G}{\partial \beta^2} < 0$. Hence, $\frac{\partial G}{\partial \beta}$ is monotonically decreasing function on $\beta \in (0, e)$. So, there cannot be more than one $\beta_0 \in (0, e)$ such that $\frac{\partial G}{\partial \beta}\big|_{\beta=\beta_0} = 0$. ■

In order to find β_0 that maximizes the gain G of the attacker one can solve $\frac{\partial G}{\partial \beta} = 0$. So, $\frac{p\alpha - 2\alpha^2\beta_0}{p(1-\alpha e)} + \frac{-p'\alpha + 2\alpha^2(e-\beta_0)}{p'(1-\alpha e)} + \frac{\alpha\gamma p'}{(1-\alpha\beta_0)^2} - \frac{\alpha\gamma p}{(1-\alpha(e-\beta_0))^2}$

Lemma 6 proves that when the attacker has a fixed amount of power invested for carrying out sponsored BWH attack on both the pool, she can split her power intelligently for attacking both the pool such that her expected gain from mining is optimized. In other words if the attacker does private mining with a fixed fraction of her mining power and uses the rest of her power in attacking the pools P and P', she could split her attacking power such that her gain from attacking both the pools is maximized.

9 Conclusion

In this paper we showed how a selfish miner could earn some extra incentive for launching BWH attack on a mining pool. This extra incentive comes from some other like minded mining pool that wants to benefit from this BWH attack. The latter mining pool shares a part of the incentive it indirectly earns from the attack. We quantitatively measured the the total gain that a BWH attacker could expect by following different attacking strategies. We also showed some interesting results that an attacker may use for increasing her total gain. We showed that an intelligent attacker can use this strategy to attack both the pool for higher gain. In future, analyzing the 'sponsored BWH attack' in a Bitcoin network having many pools could be an interesting research topic.

Acknowledgement

Concerning this work, the authors were partially supported by JSPS and DST under the Japan-India Science Cooperative Program of research project named:"Computational Aspects of Mathematical Design and Analysis of Secure Communication Systems Based on Cryptographic Primitives". The third author is partially supported by JSPS Grants-in-Aid for Scientific Research KAKEN-15H02711.

参考文献

[1] Adam Back. Hashcash - a denial of service counter-measure. Technical report, August 2002. (implementation released in mar 1997).

[2] Nicolas T Courtois and Lear Bahack. On subversive miner strategies and block withholding attack in bitcoin digital currency. *arXiv preprint arXiv:1402.1718*, 2014.

[3] Loi Luu, Ratul Saha, Inian Parameshwaran, Prateek Saxena, and Aquinas Hobor. On power splitting games in distributed computation: The case of bitcoin pooled mining. Cryptology ePrint Archive, Report 2015/155, 2015. `http://eprint.iacr.org/`.

[4] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system. *Consulted*, 1(2012):28, 2008.

[5] Meni Rosenfeld. Analysis of bitcoin pooled mining reward systems. *CoRR*, abs/1112.4980, 2011.