Sécurisation des réseaux de canaux de paiement sans fil avec des fenêtres de temps de verrouillage réduites

Gabriel Antonio Fontes Rebello, Maria Potop-Butucaru, Marcelo Dias de Amorim, Otto Carlos Muniz Bandeira Duarte

Sécurisation des réseaux de canaux de paiement sans fil avec des fenêtres de temps de verrouillage réduites

Gabriel Antonio F. Rebello[1,2], Maria Potop-Butucaru[2],
Marcelo Dias de Amorim[2] et Otto Carlos M. B. Duarte[1]

[1] *Universidade Federal do Rio de Janeiro, GTA/COPPE, Brazil*
[2] *Sorbonne Université, CNRS, LIP6, F-75005 Paris, France*

Les réseaux de canaux de paiement (Payment Channel Networks - PCN) améliorent l'impact des crypto-monnaies en fournissant une solution rapide et sans consensus. Cependant, les PCN s'appuient souvent sur des nœuds puissants avec une haute disponibilité, une grande capacité de stockage et une forte puissance de calcul, ce qui entrave leur adoption dans les environnements mobiles. Dans cet article, nous considérons une architecture PCN qui étend les fonctionnalités des PCN traditionnels aux dispositifs sans fil à ressources limitées. Nous abordons le problème du vol de jetons, une vulnérabilité critique des PCN sans fil, et proposons une contre-mesure basée sur des fenêtres temporelles réduites qui verrouillent les jetons lorsqu'un utilisateur se déconnecte. Nous évaluons notre proposition en utilisant des données réelles du PCN Lightning de Bitcoin et des réseaux mobiles 3G/4G. Les résultats montrent que la contre-mesure est plus efficace lorsque les appareils présentent une haute disponibilité et qu'il existe un compromis sécurité-efficacité lorsque la connectivité des dispositifs est faible.

Mots-clefs : blockchain, payment channel networks, wireless

1 Introduction

Payment-channel networks offer a scalable and efficient off-chain solution to improve the performance of blockchain-based systems. To open a payment channel, two users sign and publish a funding transaction that transfers a fixed amount of tokens to a joint address in the blockchain. The users can then continuously rebalance the funds of the address by sending private signed commitment transactions to each other, and close the channel by publishing the latest transaction whenever needed. Parties transact directly without paying blockchain fees and waiting for consensus, which enables micro-transactions in real-time, unburdens the consensus protocol, and effectively narrows the gap between cryptocurrencies and everyday-life needs.

Despite its advantages, payment-channel networks present open challenges for resource-constrained wireless devices such as mobile phones, smart objects, and sensors. Current PCNs rely on high availability, large storage capacity, and strong computational power [EMA21, PD16, blE20]. Such implementations are unsuited for wireless devices with limited resources and intermittent connectivity patterns that already account for over half of all the traffic on the Internet [GWB+21]. New security challenges appear in resource-constrained environments, such as how to secure payments with lossy connections and limited access to the blockchain. Addressing such challenges and implementing a PCN for payments via mobile devices is fundamental to allow the mass adoption of blockchain technology.

In this paper, we focus on the security of *wireless payment-channel networks* (WPCN) by: (i) formulating and analyzing the *token theft problem*, a vulnerability that is present in all PCNs but becomes critical in wireless environments; (ii) proposing an efficient countermeasure to the problem based on lock time windows that does not require modifications to current PCN implementations; and (iii) analyzing the efficacy of our approach using real data from Bitcoin's Lightning Network and 3G/4G mobile broadband connections [BEK14, EZB17].

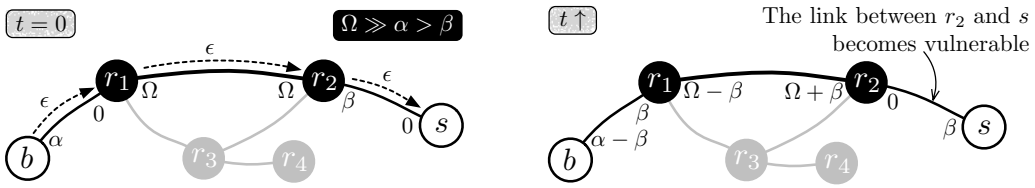

FIGURE 1: An example of the token theft vulnerability. On the left, a continuous amount of ϵ token flows from buyer b to seller s until the channel between r_2 and s is depleted. Then, on the right, s becomes highly vulnerable if it disconnects before closing the channel because r_2 has nothing to lose by closing the channel with a previous balance.

2 The Token Theft Problem in Resource-constrained Environments

Traditional PCNs like Lightning [PD16] and Raiden [blE20] assume that any node that transacts in the network remains online while the channel is open. This mitigates the *token theft problem*, in which one party publishes an old transaction to recover her/his sent tokens as soon as the other party disconnects. The system punishes malicious nodes by allowing the victim to spend all tokens in the channel if she/he recovers during a dispute period. Hence, it is only worth it to attempt the attack if the malicious node can guarantee that the other party will not verify the blockchain until the time window expires. This is critical in an environment with resource-constrained devices since nodes can disconnect for long periods or even indefinitely. We formulate the problem and demonstrate how imbalanced channels in resource-constrained environments enhance the probability of malicious behavior.

Let two nodes, b and s, represent resource-constrained devices from a buyer and a seller, respectively, and be connected to entry nodes r_1 and r_2 via unidirectional payment channels, as shown in Figure 1. Each payment channel $u \leftrightarrow v$ has a balance $bal_{u \leftrightarrow v}(t) = (bal_u(t), bal_v(t))$, where $bal_u(t)$ and $bal_v(t)$ are the balances of nodes u and v at time t, respectively. For edge payment channels between buyers and entry nodes, e.g. $b \leftrightarrow r_1$, we assume an initial balance of $bal_{b \leftrightarrow r_1}(0) = (\alpha, 0)$, where α is an amount of tokens that buyer b reserves for payments in the channel. Conversely, the initial balance of edge payment channels between sellers and entry nodes, e.g. $r_2 \leftrightarrow s$ is $bal_{r_2 \leftrightarrow s}(0) = (\beta, 0)$ where β is the amount of tokens the entry node r_2 reserves for routing payments to the seller s.

Once payment $\texttt{pay}(\langle b, s\rangle, \{r_1, r_2\}, \epsilon)$ of ϵ tokens occurs from b to s in this configuration, r_2 and s sign a commitment transaction $Tx(1)$ containing the new balance of channel $bal_{r_2 \leftrightarrow s}(1) = (\beta - \epsilon, \epsilon)$. If s disconnects at this moment, r_2 can close the channel with the operation $\texttt{closeChannel}(\langle r_2, s\rangle, Tx(0))$ and recover ϵ tokens. This is risky because r_2 would lose $\beta - \epsilon$ tokens if s recovers before the dispute period expires. However, as s receives more payments, the balance in $r_2 \leftrightarrow s$ will converge to $bal_{r_2 \leftrightarrow s}(t) = (0, \beta)$. Once this happens, r_2 has nothing to lose by closing the channel with a previous transaction even if s recovers on time. Thus, acting maliciously is the optimal strategy for any rational entry node once a border payment channel to a seller is depleted, and the seller is prone to token theft even in the absence of actual malicious nodes. Malicious nodes may also decide to attack intermediary cases once the victim disconnects if they expect a good risk-benefit ratio.

3 Defining a Minimum Time Window

We propose a simple statistical approach to the token theft problem: discover a lock time window W that minimizes the chance of attacks. The lock time window of a payment channel is a common security mechanism that locks the tokens of the closing party for some time to prevent token theft. The larger W, the more time victims have to recover and the more secure the channel becomes. However, increasing W too much can create bottlenecks in routing and punish honest nodes. The value of W is a trade-off between security and efficiency, and our goal is to minimize W while guaranteeing a minimum level of security in a wireless environment. Since most payment-channel networks already adopt lock time windows as a security measure [PD16, blE20, RMSKG17], we believe this solution is the easiest to implement and possibly the most impactful.

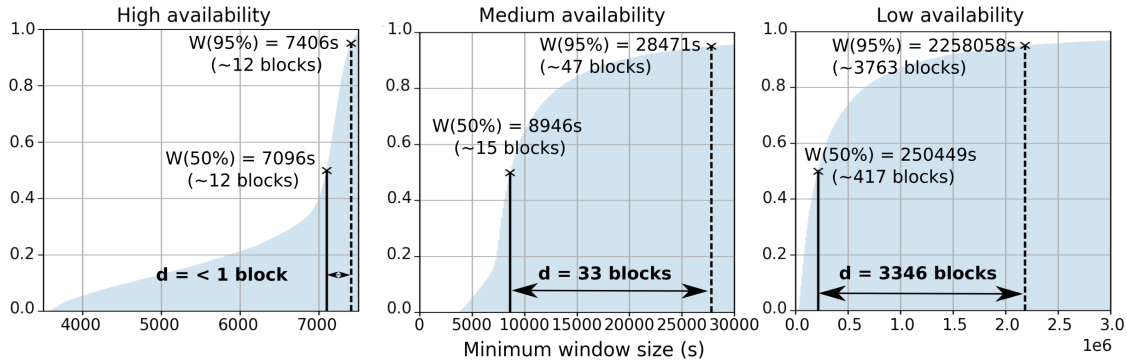

FIGURE 2: Lock time window sizes for all levels of availability. When the availability is high, the distance d between the 50% and 95% thresholds remains below one block time, which indicates a small window is safe for most users. For medium and low availability, the distance increases significantly and forces the user to select a time window that better fits her/his security and delay needs.

We use four parameters to estimate W given some light node s: (i) T_{off}, a random variable that models the time s remains disconnected from the system, which can occur due to device failure or packet loss; (ii) D_{det}, a random variable that models the delay for s to detect the attack; (iii) D_{pun}, a random variable that models the delay for s to punish malicious behavior after detecting it; (iv) Δ, a random variable that estimates the relative bias of each payment channel in the network. Each parameter composes the equation of W:

$$W = (T_{\text{off}} + D_{\text{det}} + D_{\text{pun}})(1 + \Delta). \tag{1}$$

The rationale behind our definition is that the lock time window W must be at least $T_{\text{off}} + D_{\text{det}} + D_{\text{pun}}$, otherwise the victim cannot recover and punish the attacker on time. Then, we proportionally increase the window by a factor of Δ to improve the security of biased channels, which we expect to be more vulnerable. Note that because T_{off}, D_{det}, D_{pun}, and Δ are random variables, W is also a random variable. The actual lock time value to be selected by a user depends on what level of security she/he wishes to adopt for her/his use case. Users who invest heavily in the channel should select higher thresholds to avoid great losses and users who are willing to risk can select smaller thresholds to provide token liquidity.

4 Prototype Analysis and Results

We create three scenarios based on real availability measurements of mobile broadband devices to estimate the distribution of T_{off}, D_{det}, and D_{pun}. For the high-availability case, we use the downtime and packet loss distributions of MBB connections as measured by Elmokashfi *et al.* [EZB17]. In their work, more than 90% of connections use 4G technology and the average downtime of a connection is 86.4s per day. The paper from Baltrunas *et al.* [BEK14] serves as reference for the medium-availability case. The work measures the availability of mobile broadband connections that use 3G as the default technology and shows that the downtime can last for a few hours every day. Lastly, we simulate a low-availability scenario by shifting the medium-availability downtime distribution to the right by the average distance between the high-availability and medium-availability downtime distributions. This yields an average downtime of about one week. By simulating three roughly symmetrical scenarios based on real data, we can predict how different levels of availability impact the minimum lock time window. This could be extended to real-world device data of any kind. We extract the values of Δ from LNChannels [†], an open-source tool that offers a data set of Bitcoin's Lightning Network. We download the channel balances from all closed channels since the beginning of the network and calculate the normalized bias Δ_i of each channel.

Finally, we define thresholds that correspond to subjective levels of security. A user that adopts $W(p)$ assumes p probability of recovering on time and $(1 - p)$ probability of being attacked successfully. We

†. Available at https://ln.fiatjaf.com/

use $W(50\%)$ as a reference for an unsafe threshold and $W(95\%)$ for a safe threshold, and measure the trade-off by calculating the distance d between the two thresholds. Short distances mean no significant gain from adopting a smaller window, while long distances mean the user should carefully select the value of W according to her/his needs. Figure 2 depicts the cumulative density functions for the minimum window sizes of all scenarios. The thresholds $W(p)$ are equivalent to the percentiles of the distribution of W.

In the high-availability scenario of Figure 2, 4G connectivity allows devices to be safe from attacks even with short time windows. The distance of less than one block between $W(50\%)$ and $W(95\%)$ demonstrates that increasing W to a secure level generates no significant delay, so devices with good connectivity should adopt the safest W possible. The result also confirms our assumption that good connectivity profiles present in most traditional PCNs can effectively mitigate token theft.

The trade-off between security and efficiency becomes significant in the medium-availability scenario. The distance of 33 blocks between $W(50\%)$ and $W(95\%)$ corresponds to an increase of 5.5 hours in return delays for the party that closes the channel. T_{off} becomes the dominant parameter in Equation 1. The results indicate that a user with 3G connectivity should define minimum lock time windows of at least a few hours to reduce the probability of attacks; otherwise, attackers with better connectivity can easily exploit them.

The low-availability scenario demonstrates that users with low connectivity should either select W values in the range of days to weeks or use the main blockchain to transact. Delays in such order of magnitude may be economically worthwhile if the transactions fees in the blockchain are too expensive for the user. However, more than 550 transactions could be published within the distance of 3346 blocks, which indicates the time window W may not be the most efficient countermeasure for devices that remain offline for long periods. Instead, we should adopt W with other security features, such as heavier punishment for attackers.

5 Conclusion

We analyzed the impact of the token theft problem in wireless environments. Our main findings show that the problem is not exclusive to wireless PCNs and that our solution may work with traditional PCNs as well. A countermeasure based on minimum lock time windows is efficient when the devices present high to medium availability. For devices with low availability, the minimum lock time window becomes so significant that it may be better to publish the transactions directly in the blockchain. In future works, we will investigate other types of countermeasures to token theft, such as time-varying lock time windows.

References

[BEK14] Dziugas Baltrunas, Ahmed Elmokashfi, and Amund Kvalbein. Measuring the reliability of mobile broadband networks. In *14th ACM IMC*, pages 45–58, 2014.

[blE20] brainbot labs Est. The Raiden network: Fast, cheap, scalable token transfers for ethereum, 2020. Last access: 12 October 2021.

[EMA21] Enes Erdin, Suat Mercan, and Kemal Akkaya. An evaluation of cryptocurrency payment channel networks and their privacy implications. *arXiv preprint arXiv:2102.02659*, 2021.

[EZB17] Ahmed Elmokashfi, Dong Zhou, and Džiugas Baltrünas. Adding the next nine: An investigation of mobile broadband networks availability. In *ACM MobiCom*, pages 88–100, 2017.

[GWB+21] Stefan Geissler, Florian Wamser, Wolfgang Bauer, Michael Krolikowski, Steffen Gebert, and Tobias Hoßfeld. Signaling traffic in internet-of-things mobile networks. In *2021 IFIP/IEEE IM*, pages 452–458, 2021.

[PD16] Joseph Poon and Thaddeus Dryja. The bitcoin lightning network: Scalable off-chain instant payments, 2016. Last access: 12 October 2021.

[RMSKG17] Stefanie Roos, Pedro Moreno-Sanchez, Aniket Kate, and Ian Goldberg. Settling payments fast and private: Efficient decentralized routing for path-based transactions. *arXiv preprint arXiv:1709.05748*, 2017.