Poster: Hades: Practical Partitioning Attack on Cryptocurrencies

Vinay Shetty
Saarland University
s8vishet@stud.uni-saarland.de

Piyush Kumar Sharma
imec-COSIC, KU Leuven
pkumar@esat.kuleuven.be

Devashish Gosain
imec-COSIC, KU Leuven
Max Planck Institute for Informatics
dgosain@esat.kuleuven.be

***Abstract*—Bitcoin is unarguably one of the most widely used cryptocurrency systems and is an excellent realization of blockchain technology. However, Bitcoin is shown to be vulnerable to partitioning attacks. These attacks aim at isolating a Bitcoin node from the rest of the Bitcoin network such that the attacker controls the victim(s) view of the blockchain.**

The latest attack, Erebus, however, requires cooperation from AS-level adversaries (e.g., tier-1 AS), which makes the attack practically daunting. In this work, we demonstrate a new practical partitioning attack, *Hades*, that does not need the cooperation of ASes; instead, it just requires control of a few hundred cloud hosts. We exploit the capability of instantiating multiple Bitcoin Tor hidden services on a single host. This helps to achieve a scale as available to an AS-level adversary. We demonstrate the feasibility of conducting the partitioning attacks through simulations, utilizing attacker-controlled malicious onion addresses.

I. Introduction

Partitioning attacks focus on isolating a Bitcoin node from the rest of the Bitcoin network. Thus the victim node(s) has a different view of the blockchain that the adversary controls. Some of the motivations for performing the partitioning attack are isolating mining power, 51% attack, and double spending attacks [1].

The existing literature proposed multiple approaches to perform partitioning attacks, including but not limited to BGP prefix hijacking [1], [3]. However, the most recent partitioning attack (Erebus [5]), which is stealthy and difficult to be detected by an adversary, assumes cooperation from a large tier-one transit AS. This capability allows an attacker to control thousands of IP addresses allowing them to send large amounts of Bitcoin requests to poison all the connections of a victim node over time. However, it is non-trivial to achieve such cooperation from ASes in practical scenarios.

Thus, in this work, we evaluate the feasibility of a partition attack involving onion addresses. This is because, lately, there has been increased adoption of onion services by the Bitcoin nodes[1]. In our new attack *Hades*, the adversary does not require cooperation from ASes; rather, it requires setting up a few thousand onion services as bitcoin instances. The idea behind this attack is simple: a single physical (or cloud hosting) machine can host hundreds of onion services. Since Bitcoin supports the use of onion services, this translates to running hundreds of Bitcoin instances over a single physical machine. If an attacker controls hundreds of such machines, this means it practically controls thousands of Bitcoin instances without the cooperation of an AS-level adversary. This scale of nodes helps the attacker to perform the partitioning attack following the approach described in the original Erebus paper.

Through simulations, we demonstrate the feasibility of Hades. Our initial results show that it is certainly beneficial for the adversary to consider onion addresses as an attack vector for the partitioning attack. Hades can have severe implications with respect to partitioning attacks as even a normal user with control of a few hundred cloud hosts can successfully launch this attack. This significantly reduces the required resources in comparison to an AS level adversary. Moreover, Hades still maintains the same stealth as the original Erebus attack making it harder to detect.

[1]https://bitnodes.io/dashboard/

II. Background and Related Work

A. Hidden Services

Tor, by default, provides one-way anonymity to the client because the receiver does not know who actually sent the request, but the sender is required to know the details of the destination. Hidden services [4] enable the destination server to hide its IP address from the client (but still host the service). Thus, hidden services provide two-way anonymity. A typical hidden service is associated with an onion address that ends with a `.onion` suffix. Thus hidden services are also widely known as onion services.

B. Partitioning attacks

Partitioning attacks focus on isolating the Bitcoin node from the rest of the Bitcoin network; the victim node(s) has a different view of the blockchain that the adversary controls. The attacks discussed in the literature are:

1) Eclipse attacks: Authors in [2] present off-path attacks where the adversary only controls the end hosts and not the infrastructure between the victim and the rest of the Bitcoin network. Using their approach, the adversary attempts to occupy all the connections of a victim Bitcoin node. The attacks populate the victim node tables with trash IPs (the ones not associated with any Bitcoin node).

2) Partitioning a node with BGP prefix hijacking: In 2017, Apostolaki et al. [1] proposed an attack where the goal of an AS-level adversary was to hijack the most specific BGP prefix that includes the IP address of the victim, thereby hijacking the traffic towards the victim. When the attacker hijacks the traffic, the attacker controls all the network information toward

the victim addresses. The attacker can potentially choose to alter, drop, or delay the messages directed to the victim node. This effectively creates partitions in the Bitcoin network where each partition has a different view of the blockchain.

3) Erebus attack: Due to countermeasures after the Eclipse attack, occupying the victim's connections with the trash IPs is not trivial. The adversary should own numerous IP addresses (in thousands) where actual Bitcoin software is running. Thus, Tran et al., [5] proposed a new variant of partitioning attack (Erebus) in which they consider an *AS-level adversary*.

The attack leverages the capability of an *on-path* AS-level adversary (e.g., tier-1 ISP) to send malicious Bitcoin *Addr* messages containing the IP addresses of non-Bitcoin nodes. The AS can use thousands of such IP addresses to partition the victim node.

III. Attack Feasibility

The partitioning attack involves controlling hundreds of thousands of bitcoin addresses. Thus Erebus even requires the cooperation of an adversary AS for a successful attack. We note that, unlike an IP address node that can host only one Bitcoin address, an onion address node can host hundreds to thousands of Bitcoin addresses on a single host. We leverage this capability to perform our attack. But first, we tested the feasibility of running multiple onion services on a single machine that can eventually serve as different Bitcoin addresses.

We first configured our host to run as an onion addresses bitcoin node. This can be easily achieved by changing some parameters in the bitcoin configuration file. Then we ran 100 hidden services on our VP and probed all the hidden services from one of our other nodes for 19 days to check the hidden service availability. The histogram plot in Figure 1 shows that 73% of all hidden services were available for 90-95% of the duration of the probes and that 92% of the hidden services were available for more than 85% of the duration of the probes. This indicates that multiple hidden services can simultaneously be hosted on a single host machine with high uptime.

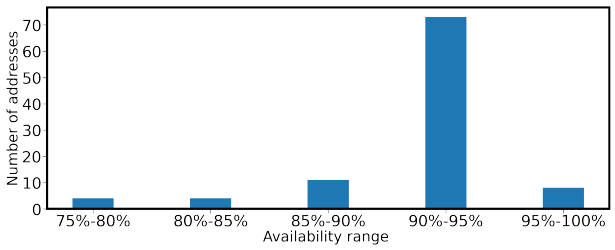


Figure 1: Reachability of 100 Bitcoin onion services on a single cloud host.

IV. Analysis and Results

Modelling the behavior of a Bitcoin node is a non-trivial task as the incoming and outgoing traffic patterns of Bitcoin nodes are variable. So, to control different configurations (e.g., incoming traffic rate), Tran et.al., [5] developed a custom Bitcoin simulator that replicates the IP address management and outgoing connections of a real Bitcoin node. We modify the simulator to incorporate the onion addresses and executed Hades.

To test the effectiveness of the attack, we executed the following attack scenario where the attacker controls 2000 onion and 100 IP addresses.

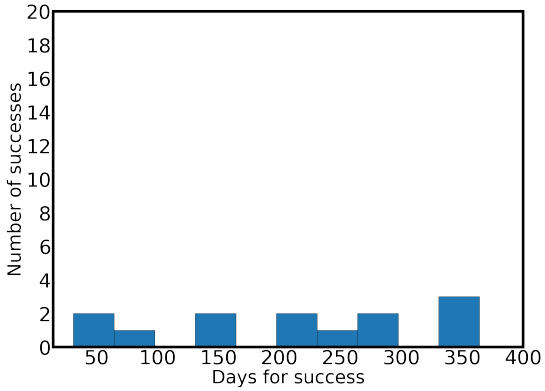


Figure 2: Attack success: Histogram plot where attacker-controlled 2000 onion and 100 IP addresses.

In the simulations, the attacker sends malicious *addr* messages from the attacker-controlled IP address nodes. The malicious *addr* messages contain attacker-controlled onion addresses. We performed 20 simulations, every time randomly selecting the attacker-controlled addresses from the pool of the pre-generated addresses.

In this scenario, the number of days for successful simulations takes anywhere between 20–370 days. A successful simulation means that the adversary occupies all outgoing connections (within 381 days). Otherwise, we consider the simulation to be failed. In this case, the simulation failed seven times. In the future, we plan to simulate more scenarios, e.g., by varying more onion and IP addresses.

V. Conclusion

In this work, we extend the existing partitioning attacks (using only the IP addresses) to incorporate onion addresses as well. Moreover, unlike the existing Erebus attack, our modified Hades attack does not involve cooperation from a tier-1 AS. Using simulations, we show that Hades can successfully isolate a victim node using malicious onion addresses just by controlling 2000 onions and 100 IPs.

References

[1] M. Apostolaki, A. Zohar, and L. Vanbever, "Hijacking Bitcoin: Routing attacks on cryptocurrencies," in *2017 IEEE symposium on security and privacy (SP)*. IEEE, 2017, pp. 375–392.

[2] E. Heilman, A. Kendler, A. Zohar, and S. Goldberg, "Eclipse attacks on Bitcoin's peer-to-peer network network," in *24th USENIX Security Symposium (USENIX Security 15)*, 2015, pp. 129–144.

[3] M. Saad and D. Mohaisen, "Three birds with one stone: Efficient partitioning attacks on interdependent cryptocurrency networks," in *2023 IEEE Symposium on Security and Privacy (SP)*. IEEE Computer Society, 2022, pp. 1404–1418.

[4] Tor, "Tor Onion Services," https://support.torproject.org/onionservices/, 2022.

[5] M. Tran, I. Choi, G. J. Moon, A. V. Vu, and M. S. Kang, "A stealthier partitioning attack against bitcoin peer-to-peer network," in *2020 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2020, pp. 894–909.

Hades: Practical Partitioning Attacks on Cryptocurrencies

Vinay Shetty[1], Piyush Kumar Sharma[2], Devashish Gosain[2,3]
[1] Saarland University, [2] KU Leuven, [3] MPI-INF

Onion addresses and Partitioning attacks

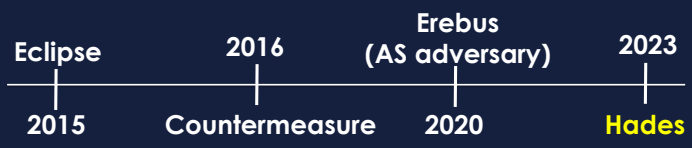
Partitioning attack

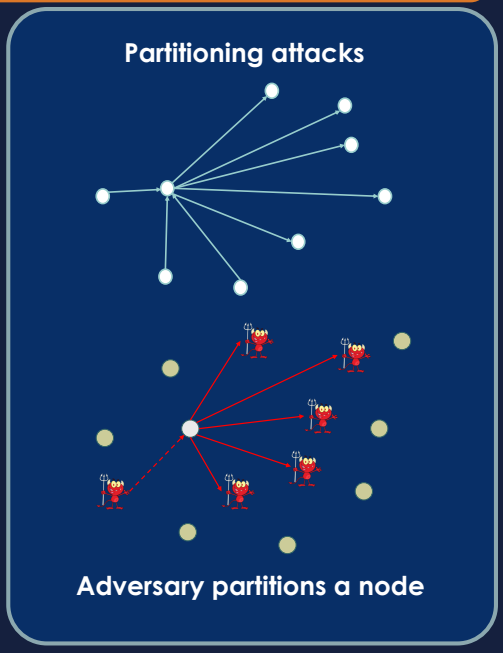
- Occupy all the connections of a victim node
- The victim node has a view of the blockchain controlled by the adversary

Onion Addresses (Hidden Services)

- Assigned to a hidden service only reachable by the Tor network
- Facilitates two-way anonymity for both the sender (end-user) and the receiver (hidden service)

Partitioning Attack Timeline

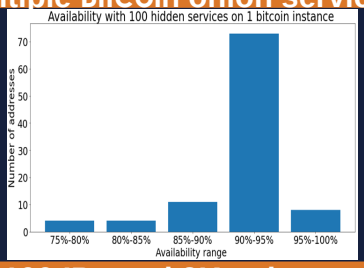




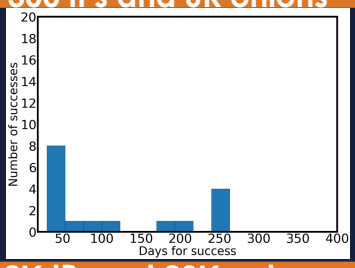
Proposed Attack: Hades

- **Does not require AS level adversary**; instead of using thousands of IP addresses (under the purview of the adversary AS), use onion addresses.
- Create hundreds (or thousands) of Bitcoin onion services on a single host
- Each onion node sends malicious "addr" message to the victim node

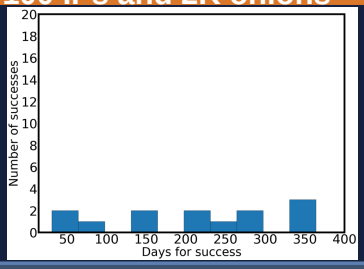
Multiple Bitcoin onion services



300 IPs and 6K onions



100 IPs and 2K onions



2K IPs and 20K onions