Elastic Blockchain: A Solution for Massive Internet Service Abuse

Junhao Li
Cornell University
jl2922@cornell.edu

Abstract
Internet service abuse is a significant threat for service providers, internet users, and even national security. In this short article, I present elastic blockchain, an architecture for solving massive internet abuse. For webmasters, elastic blockchain lowers the cost of service abuse prevention and may even benefit from service abuse. For regular users, it provides a more user-friendly way to prove their goodwill than doing Turing tests. For attackers and water armies, it increases their cost of attacks dramatically, so they are less likely to do that.

Introduction
Internet service abuse is a significant threat for service providers, internet users, and even national security. It consumes lots of computing and network resources, and thus increases the cost of running internet services. It also disturbs and misleads regular users.

There are three parties involved in the massive internet service abuse, the service providers, the internet users, and the attackers. A good abuse prevention solution should create a little cost for the first two parties and an extremely high cost for the third party. Most solutions fail to satisfy all of them. In this short article, I present elastic blockchain, an architecture for solving massive internet abuse that achieves the above criteria based on the proof-of-work mechanism in blockchain and the instant scaling of the serverless architecture.

Architecture
At a high level, elastic blockchain has two main components. One is a blockchain miner at the frontend. The other is a validation service in front of the original backend for verifying the proof-of-works produced by the frontend.

Frontend
The frontend miner produces a proof-of-work for each request. It uses web workers to avoid blocking the user interface and uses GLSL shaders to accelerate the hash calculation. The proof-of-work calculation may or may not be based on the content of the request or the capability of the client hardware. The front-end sends the proof-of-work together with the requests to the elastic validation service. Sometimes, it will be asked to send additional proof-of-works by the validation service.

Backend
The validation service uses a serverless architecture to allow instant scaling during an attack. It validates the proof-of-works which are sent together with the requests and drops the invalid

requests immediately. For the remaining requests, it either lets them pass through to the original backend or requests additional harder proof-of-works from the clients. The decision is based on the request volume or other light heuristics. Since the proof-of-work is easy to verify and the internet uses the SPNP (sending party pays) model, the cost of filtering requests with invalid proof-of-works is low. Since the proof-of-works are computationally hard to produce, the cost of massive attacks is high. Besides, most attacks and water army operations happen in a short period, so the cost can become extremely high due to the additional harder proof-of-work requests. These proof-of-works may also provide webmasters direct monetary gain from the cryptocurrency community. These monetary gains can be used to fund back the validation service, as well as process the abuse requests and build more sophisticated spam detection systems. In the era of machine learning, Turing tests are almost useless, and the best way to deal with machine learning based attacks is to build a more intelligent spam detection system based on machine learning itself and human wisdom. In addition, from the perspective of a real user who is willing to spend time doing Turing tests, a negligible amount of automatic blockchain calculation instead of the Turing tests should be more than welcome.

Conclusion

In conclusion, elastic blockchain benefits both webmasters and regular users, while increasing the cost of organized attacks dramatically. We utilize the instant scaling capability of the serverless architecture and the proof-of-work mechanism to achieve this asymmetric and resilient solution.

References

[1] Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system.
[2] Mirkovic, J., & Reiher, P. (2004). A taxonomy of DDoS attack and DDoS defense mechanisms. ACM SIGCOMM Computer Communication Review, 34(2), 39-53.
[3] Fazli, S., Matthew, S., & Eugene, C. (2018). General-Purpose Computation on GPUs in the Browser Using gpu. js. Computing in Science & Engineering, 20(1), 33.
[4] Hendrickson, S., Sturdevant, S., Harter, T., Venkataramani, V., Arpaci-Dusseau, A. C., & Arpaci-Dusseau, R. H. (2016). Serverless computation with openlambda. In 8th {USENIX} Workshop on Hot Topics in Cloud Computing (HotCloud 16).
[5] Alarifi, A., Alsaleh, M., & Al-Salman, A. (2016). Twitter turing test: Identifying social machines. Information Sciences, 372, 332-346.

Appendix A: Guidelines for Using Elastic BlockChain

For real users, elastic blockchain will incur some additional client-side computation per user request. Users should be notified of such additional computation and be provided the total amount of computation performed. We also recommend that during peacetime, the amount of computation should be kept below 100ms for 99% of the requests.