Pong Revised: Network-Based Competitions Through Secure Socket Services

Noah T. Jennings, Destiny D. Hale, Jared D. Williams, Michael J. Lively-Scholz, Dr. Malik Gladden
School of Cybersecurity, Old Dominion University, Norfolk, VA, 23529

Abstract

This paper demonstrates sPONG and cPONG, a pair of Secure Socket Service (SSS) applications that arrange an extended version of the original Pong game within a local area network (LAN). We aim to provide a safe, thrilling, locally hosted, and educational multiplayer experience that can be quickly replicated in modern Capture The Flag (CTF) events. For this, we followed a client-server model, creating a distributed system dedicated to securing connections over a network and providing seamless multiplayer competition. sPONG, our SSS application, deploys an investigative desktop application that readily provides visual insights into LAN properties, audited events, and manipulated TCP/UDP protocols. Also, cPONG, the sister SSS application, employs a simple user interface that guides user interaction with local Secure Socket Services spun by sPONG. Our analysis has not only shown the effectiveness of our SSSs, but it also exposed key implementation details of modern security strategies. By pursuing transparency, documented source code and network communications highlight the inner workings of security concepts like Load Balancing, Zero Trust, and Proxy Servers. The environment our SSSs create has also been shown to spark excitement for Cybersecurity and Ethical Hacking techniques, something sought after in today's CTF events.

Keywords: Cybersecurity, Distributed Systems, Secure Socket Services, Local Area Network, Zero Trust, Load Balancing, Proxy Servers, Transparency, Auditing, Capture The Flag

Introduction

Video games serve as valuable educational tools by incorporating key learning strategies, such as gradual scaffolding, immediate feedback, and a low-pressure setting that encourages experimentation [1]. By applying these gamified principles to technical training, security exercises have evolved into immersive competitive formats. Capture the Flag (CTF) competitions have built a solid reputation as an effective and successful approach for training cybersecurity professionals [2]. Building on these concepts, we can utilize the understanding of gaming to repurpose these frameworks as dynamic environments for a Game Based Learning (GBL) experience. We introduce sPONG and cPONG, a pair of Secure Socket Service (SSS) applications that transform the classic Pong game into a secure, Local Area Network (LAN)-based, multiplayer educational tool tailored towards modern CTF events. Our system provides real-time visual insights into network protocols and audited events while demonstrating the practical implementation of core security strategies like Zero Trust, Load Balancing, and Proxy Servers. Ultimately, these applications bridge the gap between competitive gaming and cybersecurity education, offering a safe, locally hosted environment designed to promote interest in ethical hacking.

Methodology

To study our revised Pong environment, our sister applications rely on a local area network that is depicted in ***Figure 1.*** This topology allows the applications to wirelessly distribute their workload, sustain the custom environment, and provide real-time performance metrics. Machines hosting sPONG dedicates local ***ports*** for service hosting, secure network communications, speedy requests, and statistical analyses. sPONG users access the user interface in ***Figure 2*** to browse events, examine socket service conditions, and perform investigations into system integrity. cPONG users are also given a simple UI, shown in ***Figure 3***, to sense broadcasted services, utilize the available SSSs, and configure or join a game of Pong.

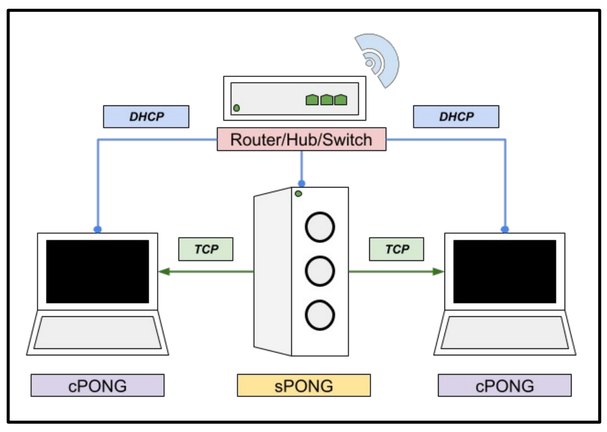


Figure 1. Pong Environment Network Composition.

Secure Socket Services (SSS)

Our implementation uses ***Python***'s native ***asynchronous TCP client-server architecture*** and ***event loop framework*** to provide Secure Socket Services across the network. ***Figure 1a*** shows the critical dependencies required by the sisters to serve and interact with custom, network-wide services. The current configuration of our SSSs enforces request filtering, enables data ownership, encourages information privacy, and mitigates cyber risks.

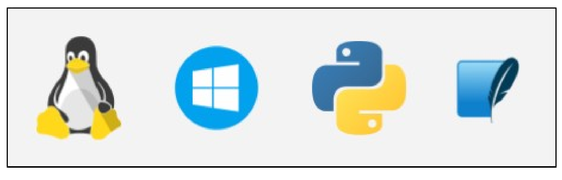


Figure 1a. Pong Environment Critical Dependencies.

Figure 2. sPONG Investigative UI.

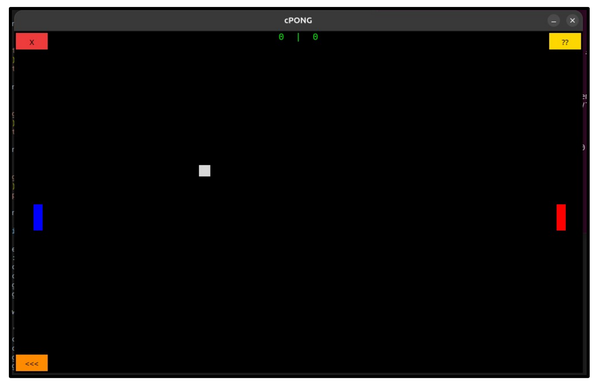
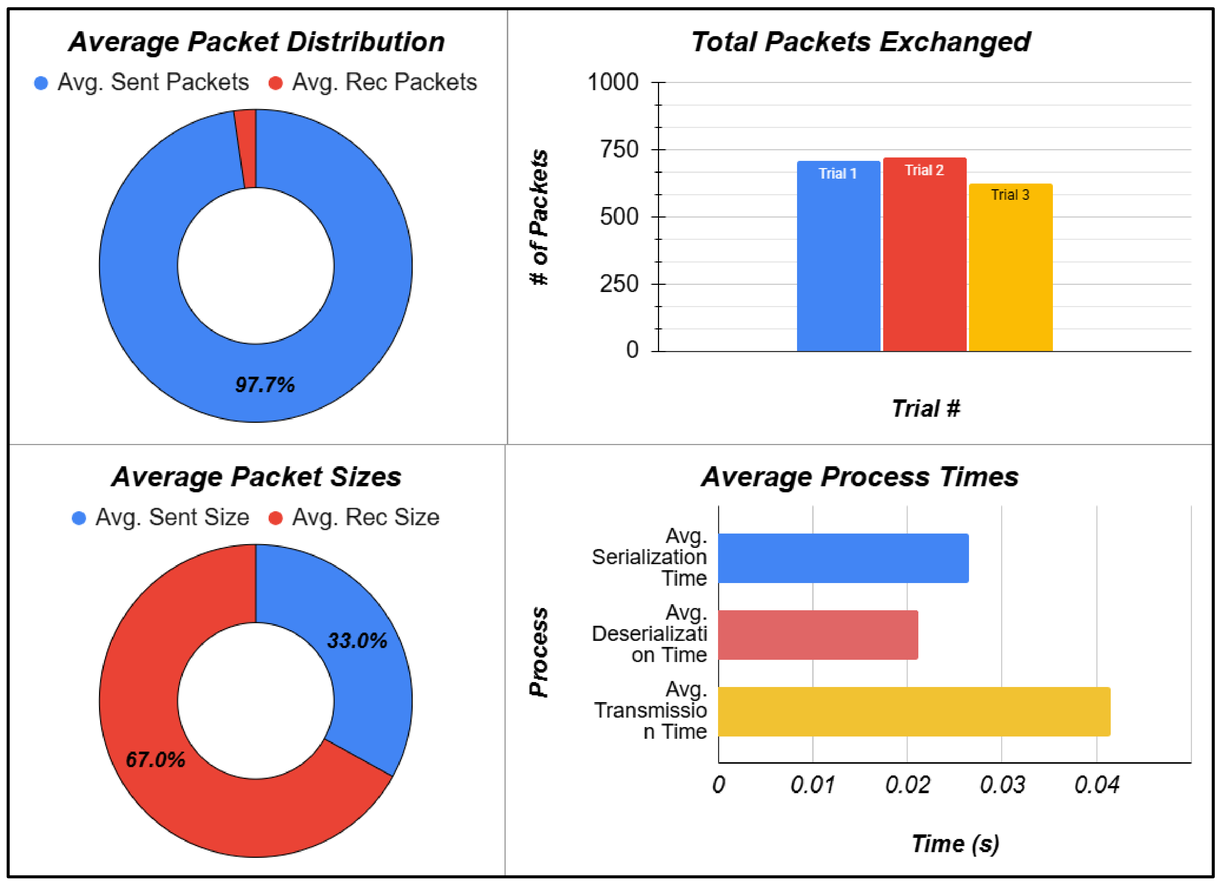


Figure 3. cPONG Play Area.



Figures 4a-4d. Server-Side Packet Distribution, Packet Exchange, Packet Sizes, and Process Times

Results

Our experiment showed the efficacy of combining intensive logging, taxing UIs, and hefty TCP communications. ***Figures 4a-4d*** depict the combined computational efficiency of the environment in terms of ***packets*** exchanged, ***data*** sizes, and ***process*** times. Results show that our SSSs can arrange a latent environment where data is exchanged in less than ***5 milliseconds*** on average.

Conclusion

Our implementation proved that Python's various frameworks could be scaled to enable mass web-connected competitions. We also find that our SSSs not only provide users with a quality experience but also inner workings can be studied and translated across disciplines. On the other hand, our sister applications lack a quality Zero Trust Architecture (ZTA) as well as a proper memory management system. Without a thorough ZTA, the security profile of the sisters is weakened, putting sPONG services and cPONG clients at risk. Similarly, without properly managing the queue of service requests, we run the risk of overloading sPONG services and ruining user experiences.

Future Work

Beyond ODU's Knowledge and Creativity Expo, we plan to integrate into or host CTFs. We plan to continue developing the sisters to introduce simulated cyber warfare and defense capabilities, such as Denial-of-Service attacks, Database Integrity, and Escalation of Privileges. The next phases of our Pong environment will include AI opponents and iOS distributions. With enhanced Zero Trust, Cryptographic Methods, and Cloud Infrastructure, we hope to connect mobile players across the globe utilizing our SSSs. The Cyber Hive hopes to continue our research into gamifying cybersecurity education, as well as the impact our project has on the culture of cybersecurity. We eventually intend to monetize this project by offering additional in-game cosmetics for players and hosting educational events or competitions surrounding the sister applications. When monetized, we plan to give back to the community by awarding scholarships and funding aspiring cyber professionals.

Acknowledgements

Our work was supported, in part, by The Cyber Hive, Dr. Malik Gladden, Dr. Alvin Holder, Professor Thomas Kennedy, Old Dominion University, and ODU's School of Cybersecurity.

References

1. Estes, A., & Shon, H. (2022, August). Teaching Control Systems With Pong. In 2022 ASEE Annual Conference & Exposition.
2. Lyu, Y., Dotson, L., Draves, N., & Zhang, A. (2026). CTF for education. arXiv preprint arXiv:2601.17543.
3. Miguel, C. J., & Azcarate, M. I. (2026). CTF as a Service: A reproducible and scalable infrastructure for cybersecurity training. arXiv preprint arXiv:2603.22511.
4. O'Neill, M., Heidbrink, S., Whitehead, J., Perdue, T., Dickinson, L., Collett, T., ... & Zappala, D. (2018). The secure socket {API}: {TLS} as an operating system service. In 27th USENIX Security Symposium (USENIX Security 18) (pp. 799-816).
5. Yevseiev, S., Milov, O., Opirskyy, I., Dunaievska, O., Huk, O., Pogorelov, V., ... & Tomashevsky, B. (2022). DEVELOPMENT OF A CONCEPT FOR CYBERSECURITY METRICS CLASSIFICATION. Eastern-European Journal of Enterprise Technologies, 118(4)