Since memory allocation problem is ubiquitous in computer science, academic work to determine the efficient and secure allocation mechanism becomes more important these days. In the following we want to present an overview of related work in this area.

As aforementioned, systems with insufficient user data validation can deal with dynamic memory storage. Dewey et al (2015) formulates the “use after free” vulnerability and conditions of this attack.

Further, Qiang Zeng et al (2019) classifies different attack types on such systems. The paper above notices how programs can be patched to gain required defense against “use after free” type. However, the method they are mentioned requires 64 bits of metadata for every memory allocation call and 4Kb of guard pages. In this work we propose the algorithm with less memory usage.

As mentioned in [2], a method introduced by Masmano et al (2004) is the most effective from the time complexity perspective. At the same the additional memory usage is required due to the hash map. This paper introduces the time and memory balanced protocol of allocation.

In terms of safety and efficiency at the same time, Beichen Liu et al (2019) have introduced so-called “SlimGuard” allocator that is designed to be secure and effective. The authors have compared memory and time usage of SlimGuard and different state-of-the art memory management algorithms. Similar to this, we introduce lightweight allocator which performance still needs to be tested and compared with other.

// ADD SOME TEXT//

As can be seen in the literature review above, state-of-the art memory management allocators are lack of either time/memory efficiency or attack protection. Only experimental methods try to approach the optimal state in both directions. In our study we have collected different ideas and proposed time and memory balanced allocator.