Different applications that implement custom allocation mechanism might lack enough secure or time/memory efficiency. In this paper we have presented common allocation approaches and implemented new memory management algorithm that is balanced in both requirements. This can protect programs vulnerable to attacks like “use after free”.

In the future we could develop a virtual laboratory for testing memory allocation mechanisms. This could help to measure the security and efficiency level of the current, existing, and future algorithms and compare them. Last but not least proposed algorithm could be reconstructed from multithreaded perspective in order to support a wider spectrum of applications.