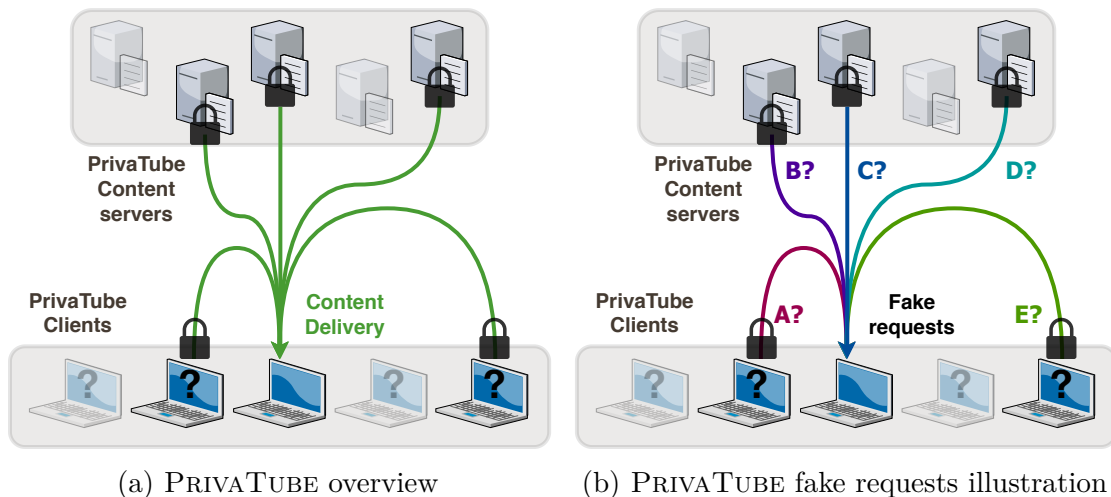


PRIVATUBE

Simon Da Silva

Efficient and scalable video streaming requires a large video delivery platform such as an edge-assisted collaborative Content Delivery Network. However, access histories can reveal critical personal information, and centralized video streaming solutions are notorious for exploiting personal data. Hiding the interests of users from servers and edge-assisting devices is necessary for a new generation of privacy-preserving streaming services.



PRIVATUBE is a *practical* and *privacy-preserving* video streaming system. PRIVATUBE aims to provide strong *privacy* guarantees with unaltered *quality of experience* thanks to edge-assistance (see Figure 1a) through an extension of MS-STREAM. Access histories are masked from the untrusted infrastructure hosting core servers and from other clients by leveraging HTTP proxies in Intel SGX enclaves at both the client and server sides. These proxies encrypt all data and obfuscate access patterns using fake requests that reduce the risk of personal information leaks (see Figure 2). Fake requests are further leveraged to implement proactive provisioning and improve the *quality of experience* (see Figure 1b).

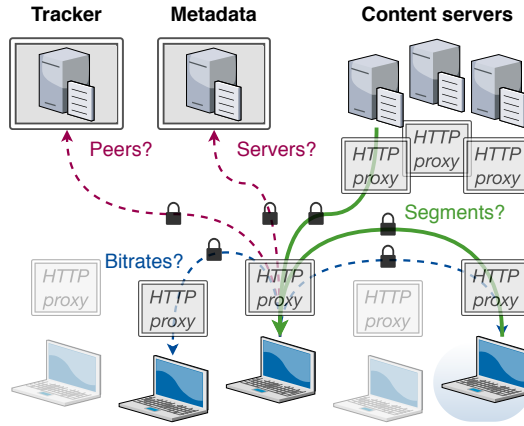


Figure 2: PRIVATUBE architecture

PRIVATUBE preserves the privacy of its users by enforcing δ -unlinkability between specific users and videos, for a chosen value of δ , by introducing fake requests. Fake requests have a cost, that is turned to the system profit by using them for pre-fetching content onto assisting peers and improving availability. This positively impacts the *quality of experience*, in particular for low-popularity videos, and improves PRIVATUBE scalability.

We implemented PRIVATUBE and deployed it on a distributed testbed with up to 14 SGX-enabled servers and clients to evaluate its performance and behavior. We also perform large-scale simulations based on a real-world data set of video access histories. Our results show that PRIVATUBE leverages multiple sources and fake requests to improve QoE, and compares favorably to non-privacy-preserving streaming. Compared to a typical setup, segments download is $2\times$ to $15\times$ faster, video quality 10% to 300% higher, with only a 17% raw server throughput overhead and 40ms slower startup delay for the first video segment.