

# SDN

Gwarancja Jakości QoS dla gier komputerowych / streamingu wideo.

• • •

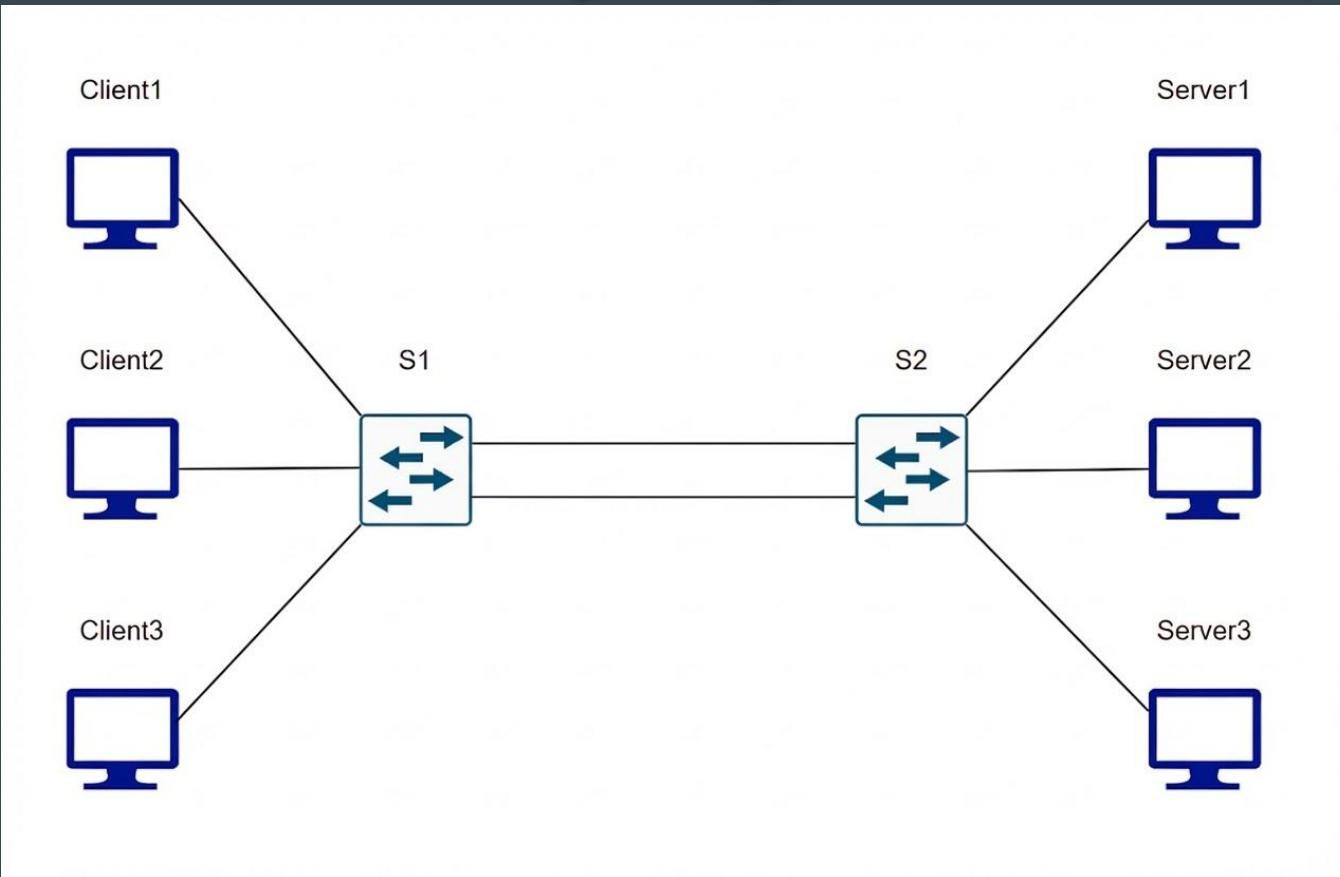
Paweł Oleszczak, Sebastian Gawlak, Gabriel Kraj, Kacper Janicki

# Cel Projektu

**Cel:** Optymalizacja parametrów sieciowych dla usług czasu rzeczywistego:

- > **Minimalizacja opóźnień i jittera w komunikacji z serwerami gier.**
- > **Gwarancja pasma**
- > **Redukcja strat pakietów przy dużym obciążeniu łącza.**

# Topologia



# Generacja Ruchu

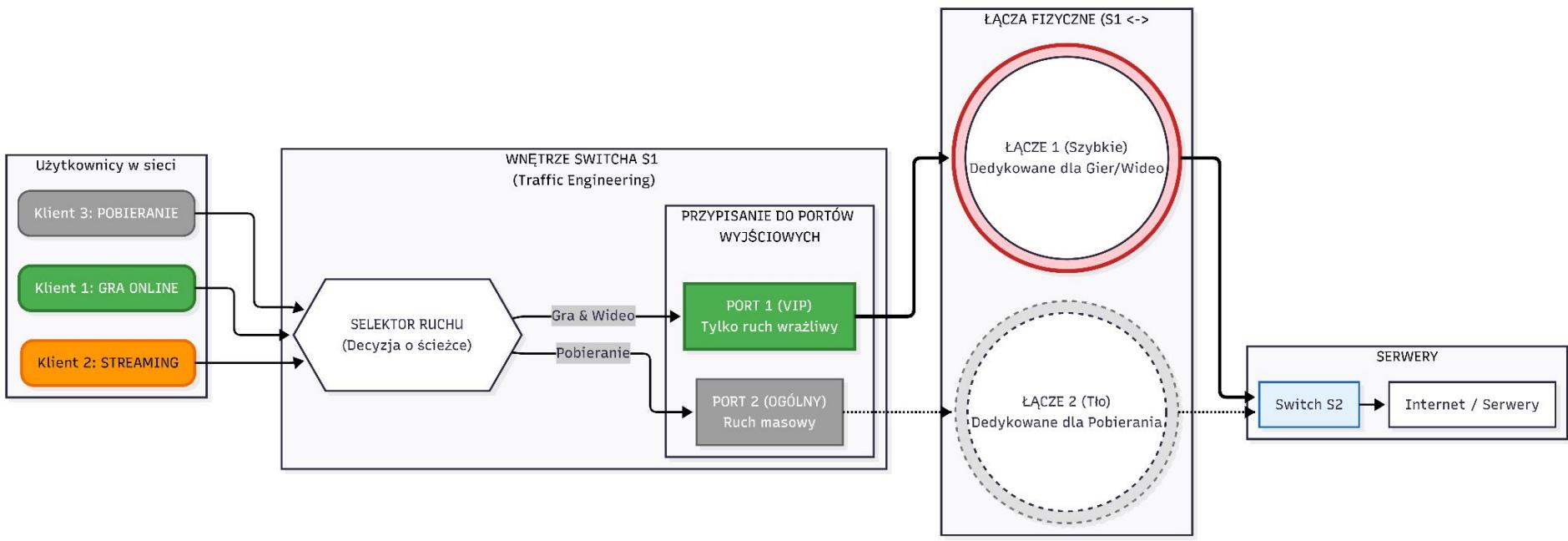
## Automatyczny Generator Ruchu (Python + iperf)

Emulacja trzech typów usług:

- > **Gry (UDP):** Małe pakiety, stały strumień 500 Kbps.
- > **Streaming (UDP):** Wysokie obciążenie pasma 20 Mbps.
- > **Pobieranie (TCP):** Ruch tła działający w trybie *Best Effort (dolne łącze)*.

Zapewniono obsługę parametrów `--type` oraz `--ip`, co pozwala na ręczne wymuszanie scenariuszy lub losową symulację użytkowników. Generator pozwala na weryfikację skuteczności algorytmów QoS i mechanizmu dławienia pasma w czasie rzeczywistym.

# Mapowanie ruchu



# Algorytm Zarządzania Ruchem

## Klasyfikacja Strumieni:

Rozpoznawanie ruchu w czasie rzeczywistym na podstawie portów docelowych:

- > Port 5001: **Gra** (Priorytet)
- > Port 5002: **Wideo** (Dynamiczne pasmo)
- > Port 5003: **Tło** (Best Effort)

## Detekcja Aktywności Gry:

- > System monitoruje czas od ostatniego pakietu gry.
- > Stan „Gra aktywna” utrzymuje się przez 6 sekund od ostatniej aktywności.

## Dynamiczna Reakcja QoS:

- > Gdy wykryto grę: Ruch wideo jest natychmiast kierowany do kolejki 2.
- > Gdy gry nie ma: Ruch wideo powraca do kolejki domyślnej z pełnym pasmem.

## Zarządzanie Przepływami:

- > Wykorzystanie akcji **OFActionEnqueue** do precyzyjnego kierowania ruchu do kolejek sprzętowych.
- > Usuwanie starych wpisów w celu natychmiastowego wymuszenia nowej polityki QoS.

Dziękujemy za uwagę!