



WebRTC

Mateusz Karłowski

Agenda

1. Wprowadzenie
2. Typy translacji NAT
3. Mechanizmy nawiązywania połączenia
4. SDP i Sygnalizacja
5. Demo

Wprowadzenie

Co to jest WebRTC?

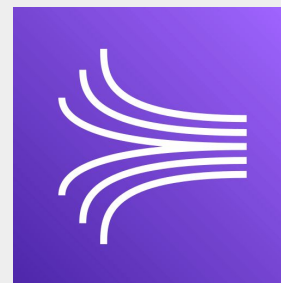
- Mechanizm przesyłu danych w **czasie rzeczywistym**
- Oparty na połączeniach **peer2peer**
- Wbudowany w **nowoczesne przeglądarki**
- Zautomatyzowana kompresja, synchronizacja i buforowanie
- Szerokie wsparcie dla **formatów multimedialnych**



Historia

- **2010** – Google przejmuje firmę Global IP Solutions (GIPS)
- **2011** – Google ogłasza projekt WebRTC i publikuje go jako open-source
- **2013** – Wsparcie WebRTC w Chrome i Firefox
- **2015** – Wzrost popularności w aplikacjach do rozmów wideo
- **2021** – WebRTC zostaje zatwierdzone jako standard przez W3C i IETF

Zastosowanie

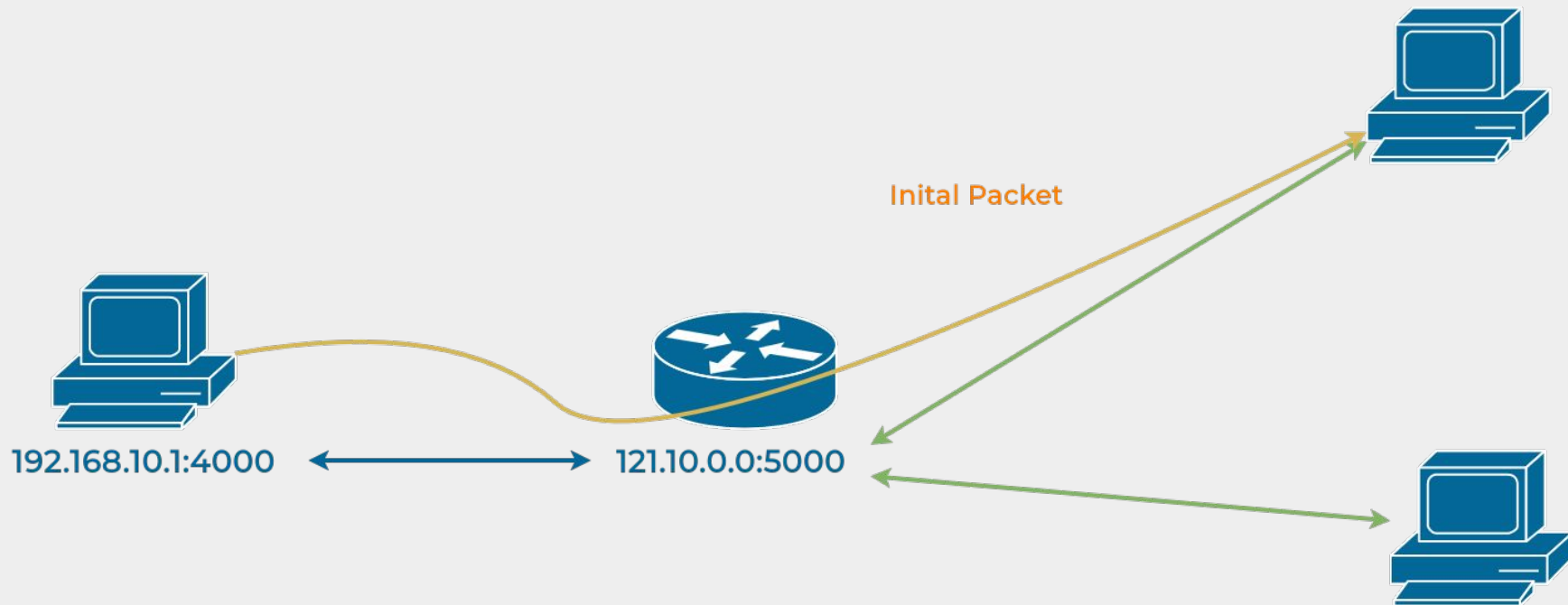


Typy translacji NAT

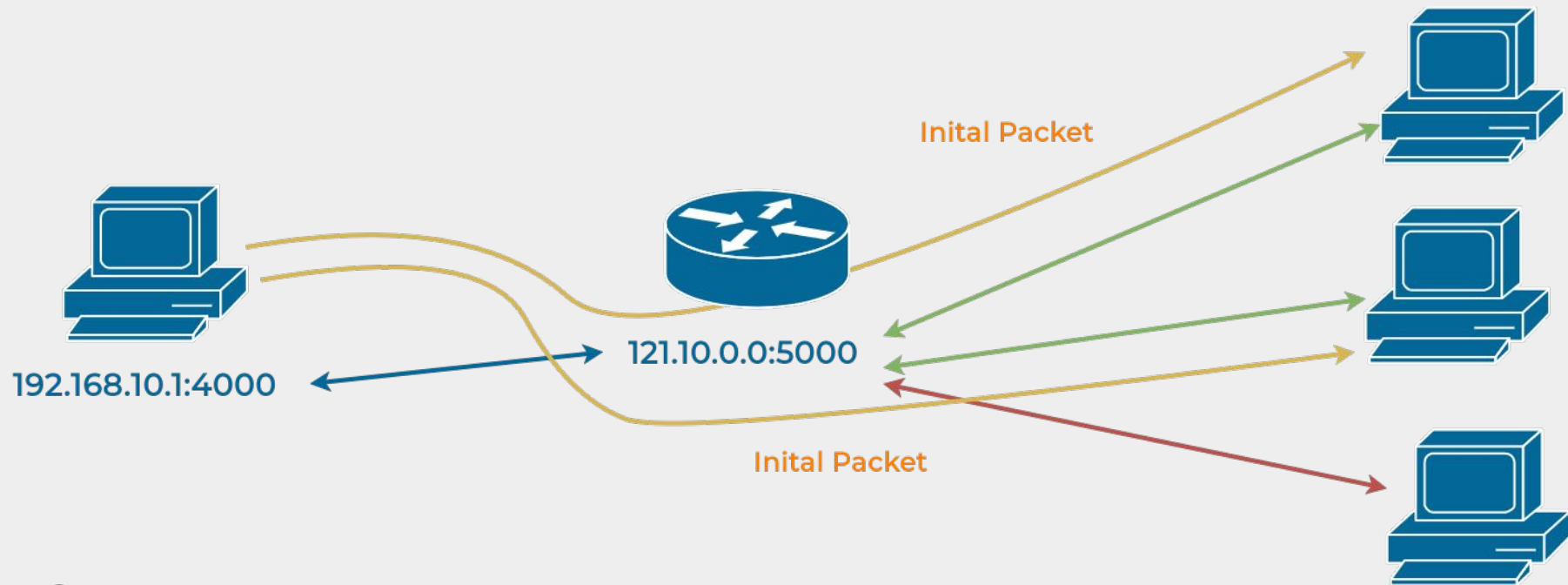
NAT

- Liczba adresów IPv4 nie zaspokaja potrzeb współczesnego Internetu
- Rozwiązanie: **Network Address Translation (NAT)**
- Pozwala wielu urządzeniom korzystać z jednego publicznego adresu
- Mapowanie adresów prywatnych na publiczne
- Realizowane na urządzeniach brzegowych

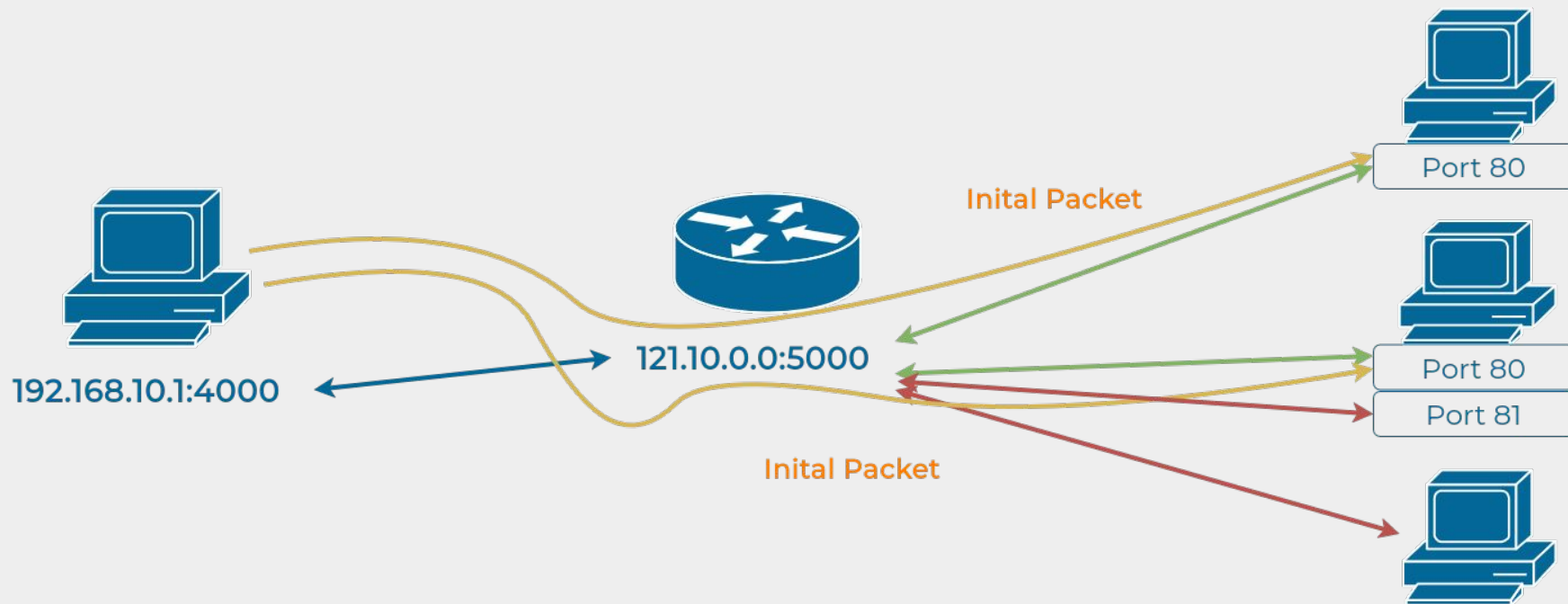
Full Cone NAT



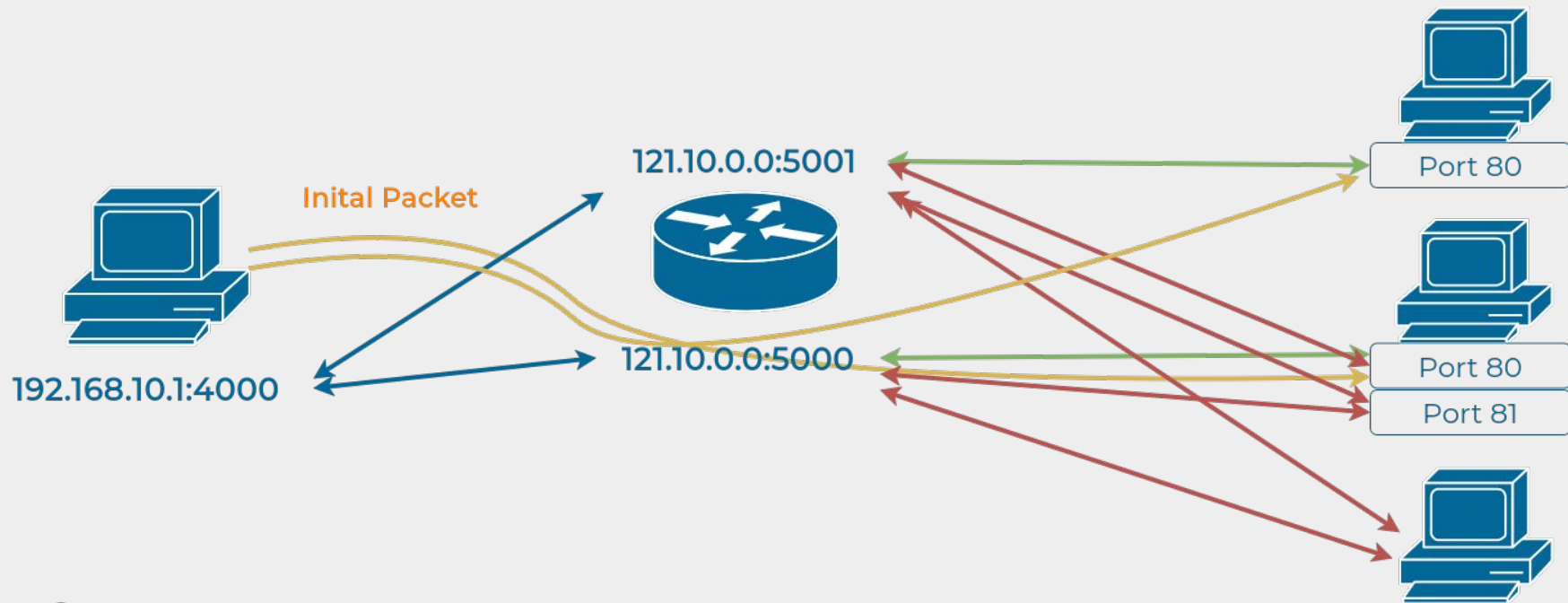
Restricted Cone NAT



Port Restricted Cone NAT



Symmetric NAT



Mechanizmy nawiązywania połączenia

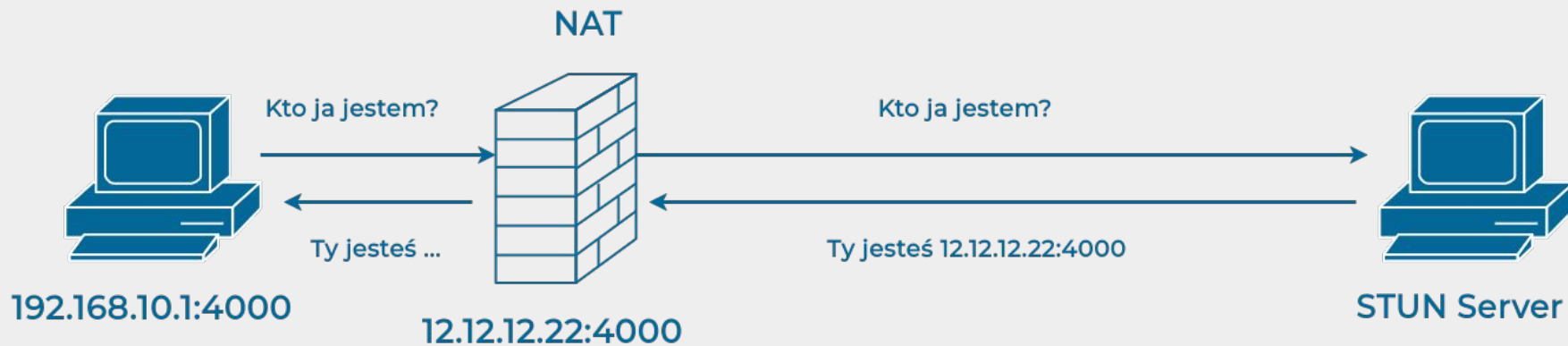
Problematyka

- Chcemy osiągnąć połączenie peer-to-peer oparte na UDP
- NAT – ukrywa prawdziwe adresy IP i porty
- Połączenia przychodzące są często blokowane przez firewall
- Jak to rozwiązać?

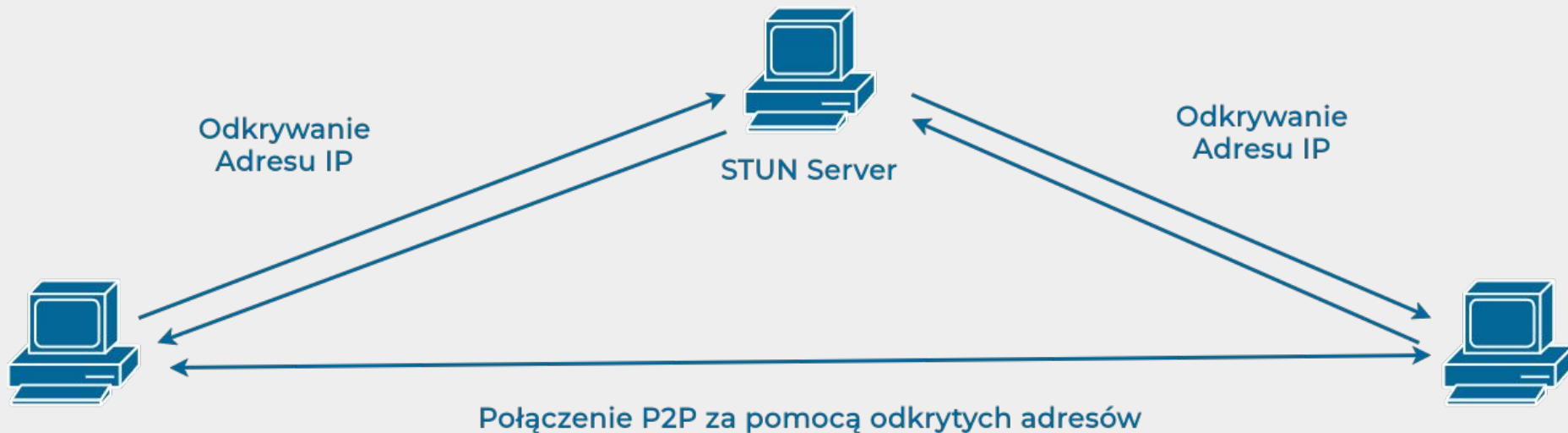
STUN

- Session Traversal Utilities for NAT
- Protokół do odkrywania **publicznego adresu IP i portu** klienta za NAT-em
- Opracowany jako rozwiązanie dla tzw. Permissive NAT:
 - Full Cone NAT, Restricted Cone NAT, Port Restricted Cone NAT
- Działa w architekturze klient serwer
- Klient wysyła zapytanie do serwera STUN (UDP)
- Serwer STUN odpowiada, z jakiego adresu IP i portu widzi zapytanie

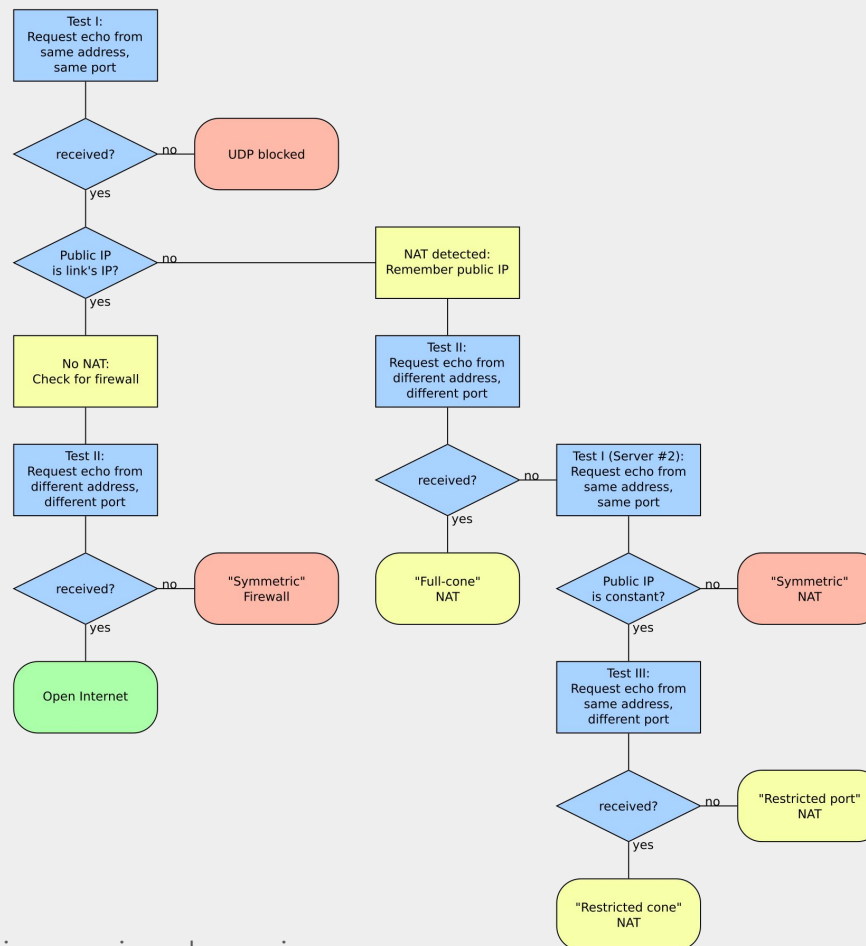
STUN



STUN



STUN



TURN

- Traversal Using Relays around NAT
- W przypadku **Symmetric NAT** po jednej ze stron, połączenie P2P nie jest możliwe
- Niezbędne jest użycie serwera **TURN** jako pośrednika transmisji
- TURN wiąże się z większym opóźnieniem
- Utrzymanie serwera TURN jest kosztowne

TURN



ICE

- Zbiera kandydatów (adresy IP i porty) do nawiązania połączenia.
- Używa serwerów **STUN** i **TURN**
- Testuje dostępność tras (sprawdza, które połączenia są możliwe).
- Wybiera najlepszą trasę połączenia (optymalizuje łączność).

SDP i Sygnalizacja

SDP

- Session Description Protocol
- Format wymiany informacji o sesji
- Zawiera dane o:
 - Typach mediów (audio, wideo),
 - Kodekach
 - Adresach IP i portach do komunikacji.

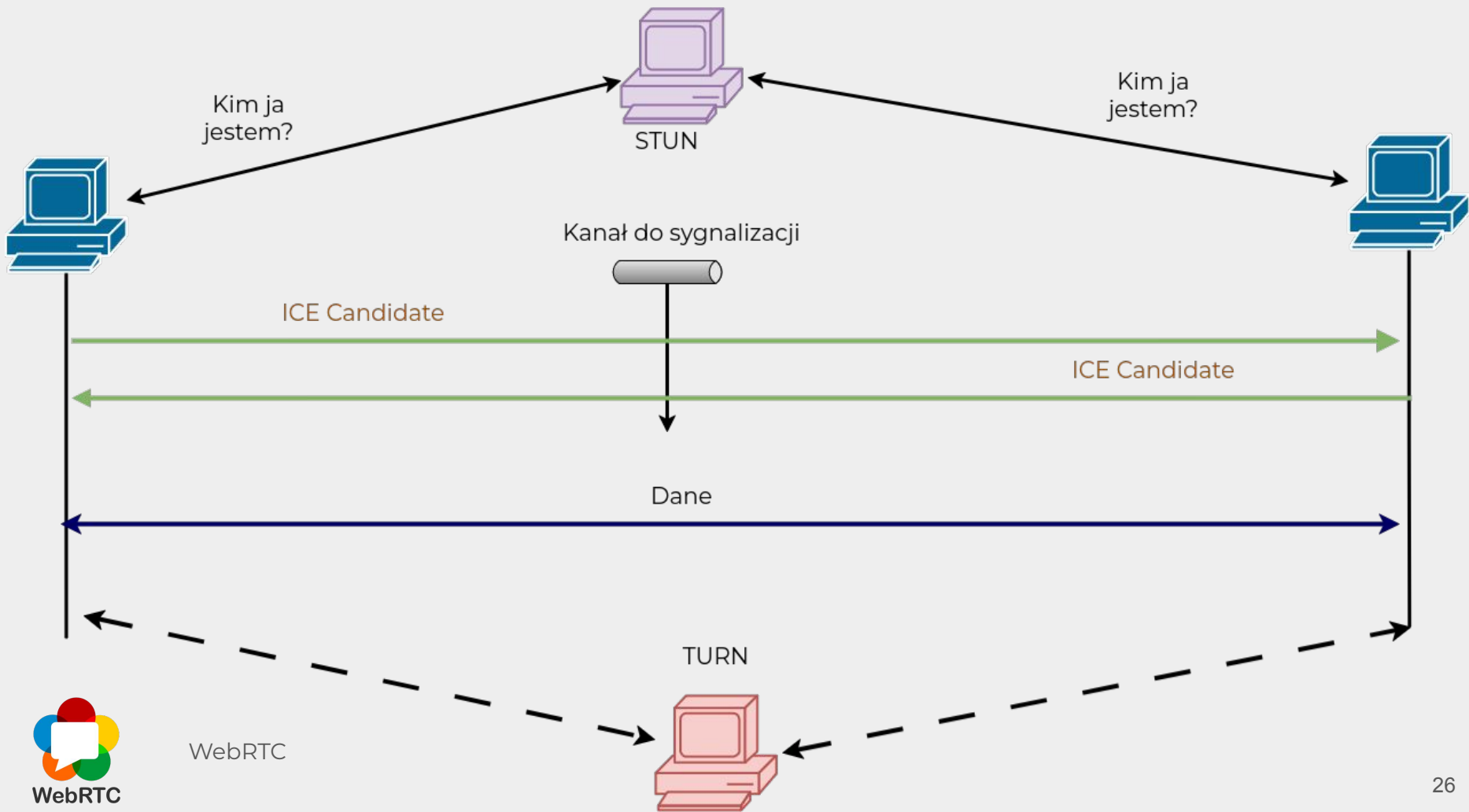
SDP - przykład

```
v=0
o=- 123456 123458 IN IP4 10.0.1.2
s=My sample flow
i=4 channels: c1, c2, c3, c4
t=0 0
a=recvonly
m=audio 5004 RTP/AVP 98
c=IN IP4 239.69.11.44/32
a=rtpmap:98 L24/48000/4
a=ptime:1
a=ts-refclk:ptp=IEEE1588-2008:00-11-22-FF-FE-33-44-55:0
a=mediaclock:direct=0
```


Sygnalizacja

- Sygnalizacja to proces wymiany informacji między klientami w celu nawiązania połączenia.
 - Wymiana SDP
 - Wymiana kandydatów ICE
- Sygnalizacja nie jest częścią standardu WebRTC.
- Powszechnie w tym celu używa się protokołu HTTP.





Demo

Socket Stream

Table 1



WebRTC Sample

Keywords: *depression, mood, anxiety, self-esteem, self-efficacy, self-esteem, self-efficacy*



Dziękuję za uwagę