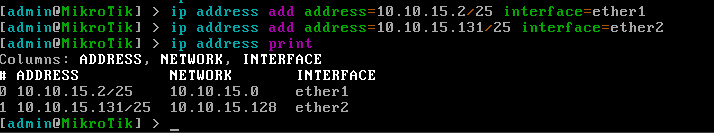
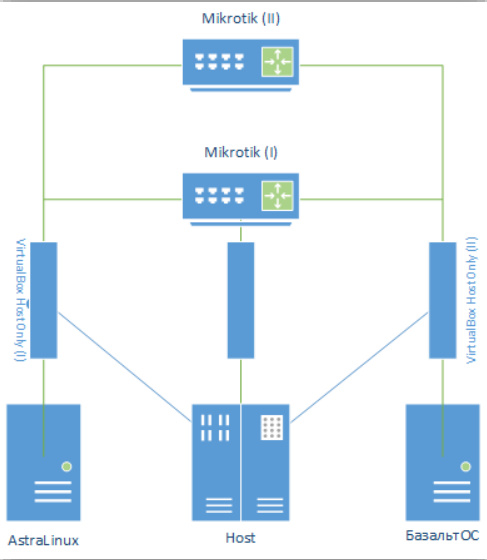
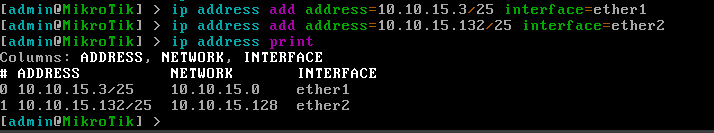
**ПРАКТИЧЕСКОЕ ЗАДАНИЕ № 8**

«Сетевые мосты. Протокол STP»

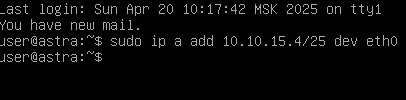
**1.**



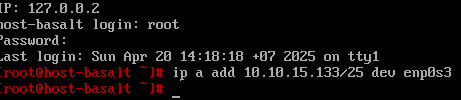
Это кстати микротик1



Это кстати микротик2

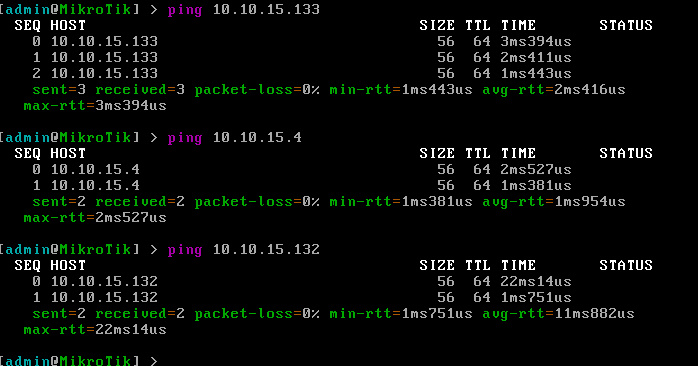


Астра

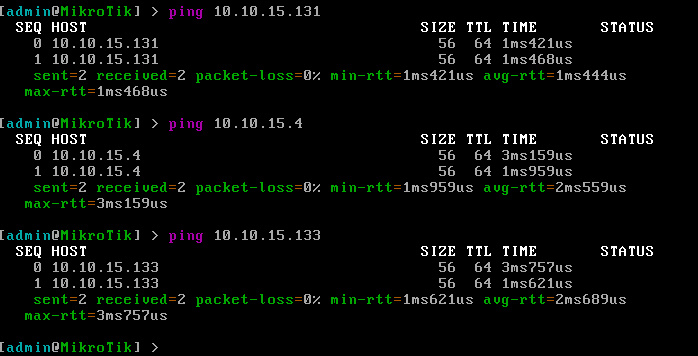


Базальт

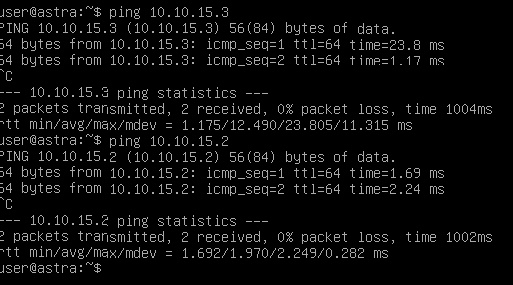
Пинг микротика 1:



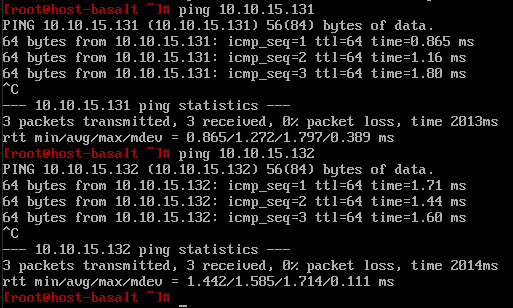
Пинг микротика 2:



С Астра-Линукса пингуются:



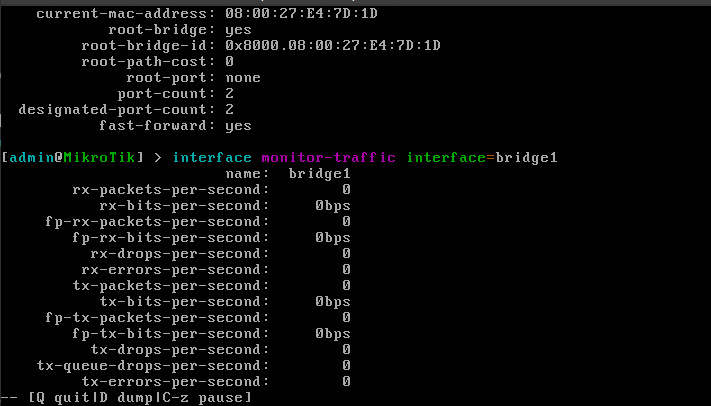
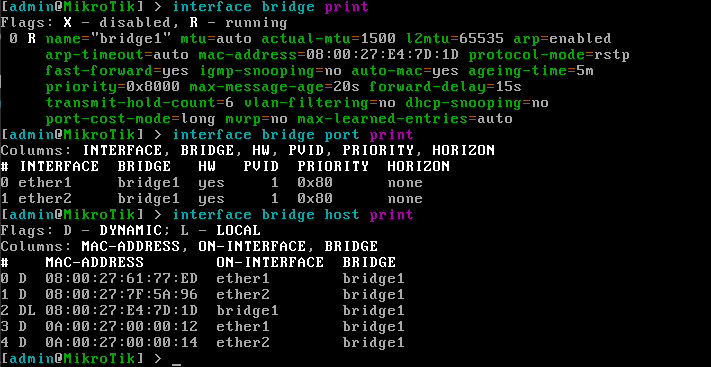
С БазальтОс пингуются:



**2.**

Ну теперь мост для микротик1:





Команды:

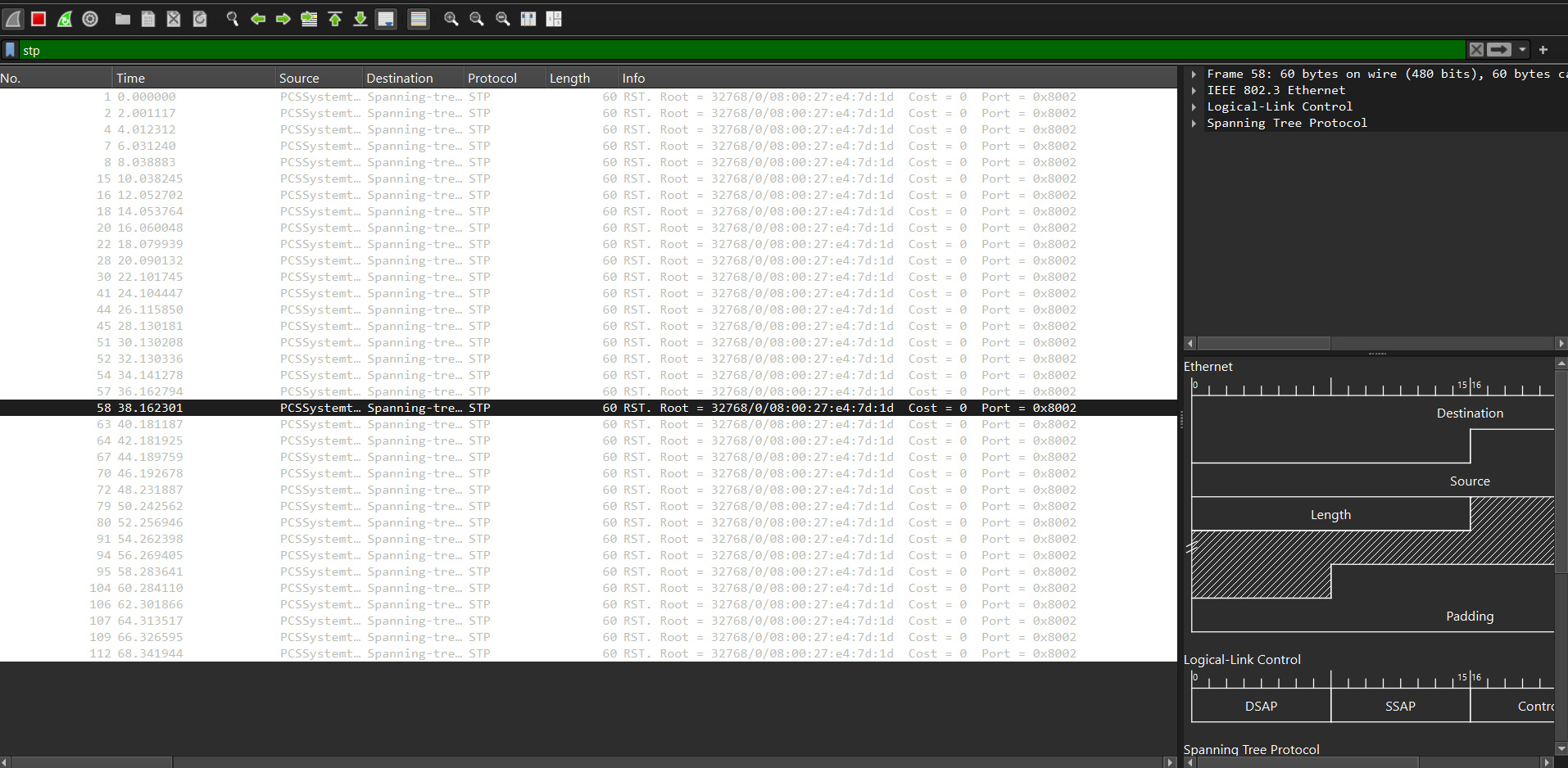
/interface bridge print

/interface bridge port print

/interface bridge host print

/interface bridge monitor bridge1

/interface monitor-traffic interface=bridge1



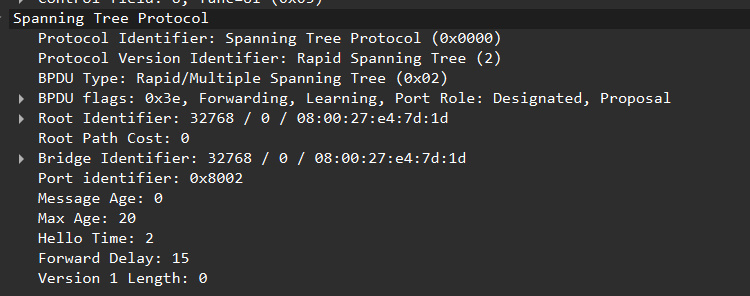
Протокол STP предназначен для предотвращения петель в сети — один из портов на другом устройстве будет переведён в состояние "Blocked", если есть избыточные соединения."

Отправитель: MAC 08:00:27:d7:9b:32 Root = 32768/08:00:27:d7:9b:32—

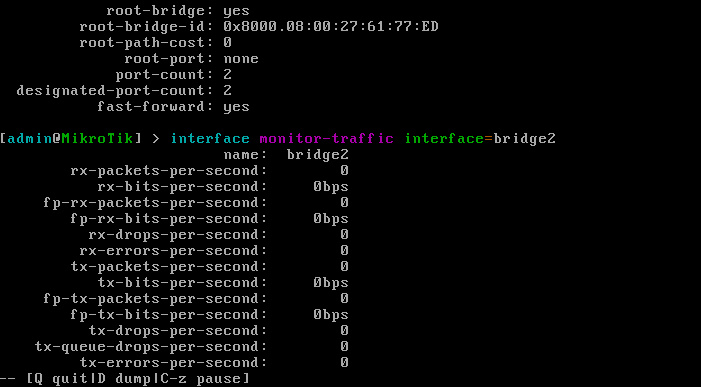
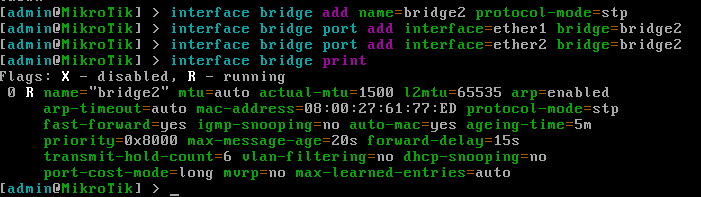
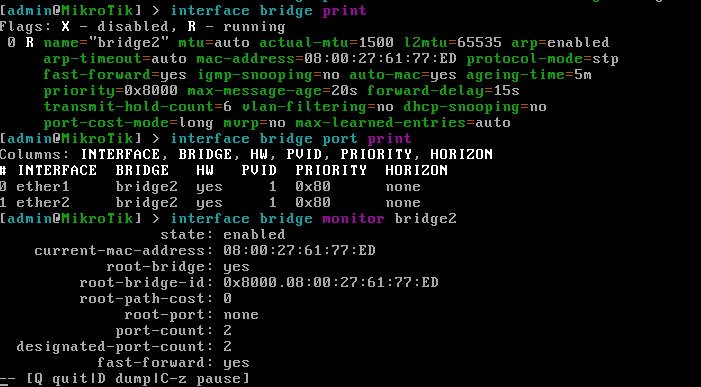
MikroTik1 сам себя считает корневым мостом Port = 0x8002 и т.д. — это ID порта, участвующего в STP

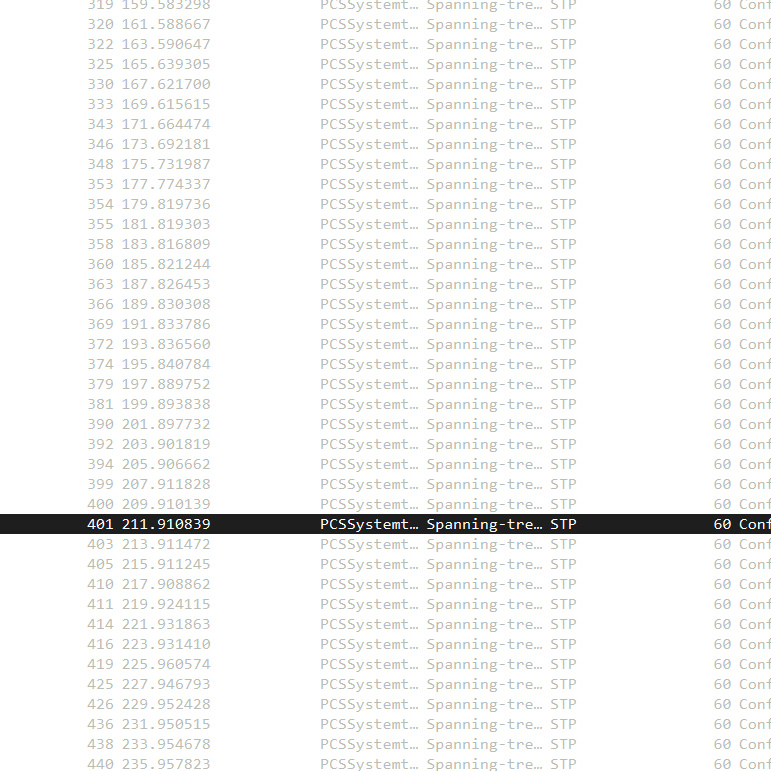
Cost = 0 — путь до корня (самого себя) = 0

STP работает корректно – MikroTik рассылает BPDU-пакеты (Bridge Protocol Data Units)



**4.**





Устройство с MAC-адресом 08:00:27:23:db:18 было выбрано в качестве корневого моста (Root Bridge) с приоритетом 32768.

Корневой мост рассылает BPDU-пакеты (Bridge Protocol Data Units), в которых объявляет себя корневым:

Root = 32768/08:00:27:23:db:18

Порт: 0x8002

Path Cost: 0 (так как это сам корневой мост)

Второй MikroTik (MAC 08:00:27:d7:9b:32) получает эти BPDU и:

Принимает отправителя (08:00:27:23:db:18) как корневой мост.

Анализирует топологию и переводит один из своих портов в состояние Blocked, чтобы предотвратить петли в сети.

Таким образом, STP обеспечил loop-free топологию, выбрав оптимальные пути и заблокировав избыточные соединения.

