

Congestion-based Accounting with re-ECN

5th GI/ITG KuVS Workshop on "Future Internet"


Mirja Kühlewind
mirja.kuehlewind@ikr.uni-stuttgart.de
9th June 2010

Universität Stuttgart
Institut für Kommunikationsnetze
und Rechnersysteme (IKR)
Prof. Dr.-Ing. Andreas Kirstädter

Wolfram Lautenschläger
wolfram.lautenschlaeger@alcatel-lucent.com

Michael Scharf
michael.scharf@alcatel-lucent.com

Alcatel-Lucent Bell Labs
Stuttgart, Germany



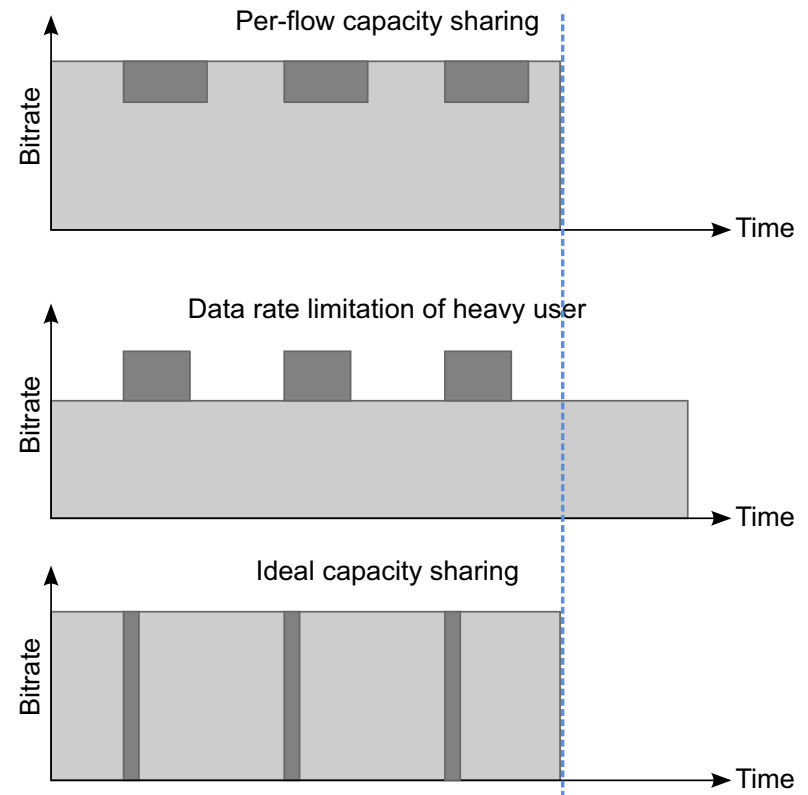
Outline

- Introduction
- Overview of the re-ECN Protocol
- Congestion-based Traffic Engineering
- Congestion Accounting Architecture
- Conclusion and Outlook

Introduction

Capacity Sharing in the Internet

1. Resource allocation is managed through TCP Congestion Control
 - Per-Flow Fairness (but not per user and not over time)
 - Only volunteer support of the end-system
 - New services have different requirements
2. Amount of traffic in the Internet is increasing vs. decreasing profit/bit
 - A minority of heavy users allocate a large share of the bandwidth capacity
 - ISPs try to mitigate by data rate/traffic volume limitations or Deep Packet Inspection



Emerging Challenges

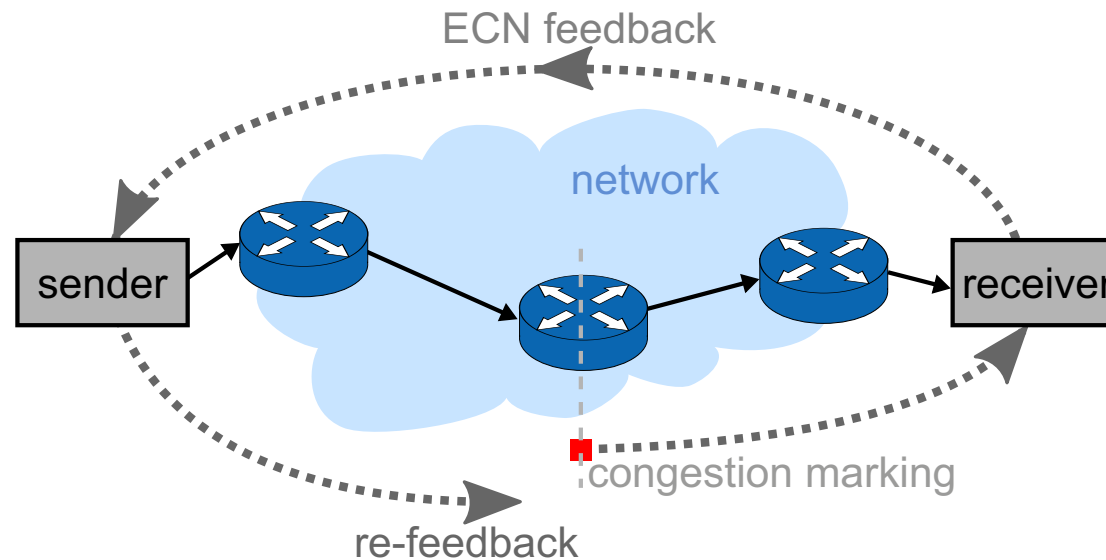
- Policing is only necessary if the available network resources are exhausted
- ISPs need to know the congestion level in their network
- re-ECN will expose the expected congestion on a network path at network ingress
- Congestion Exposure by re-ECN is currently a hot (and controversial) topic in the IETF

Overview of the re-ECN Protocol

A TCP/IP signaling mechanism for "Congestion Exposure"

Principle Protocol Mechanism

- Receive congestion information from receiver (Explicit Congestion Notification – ECN)
- Re-insert congestion information (= estimation about expected congestion level)

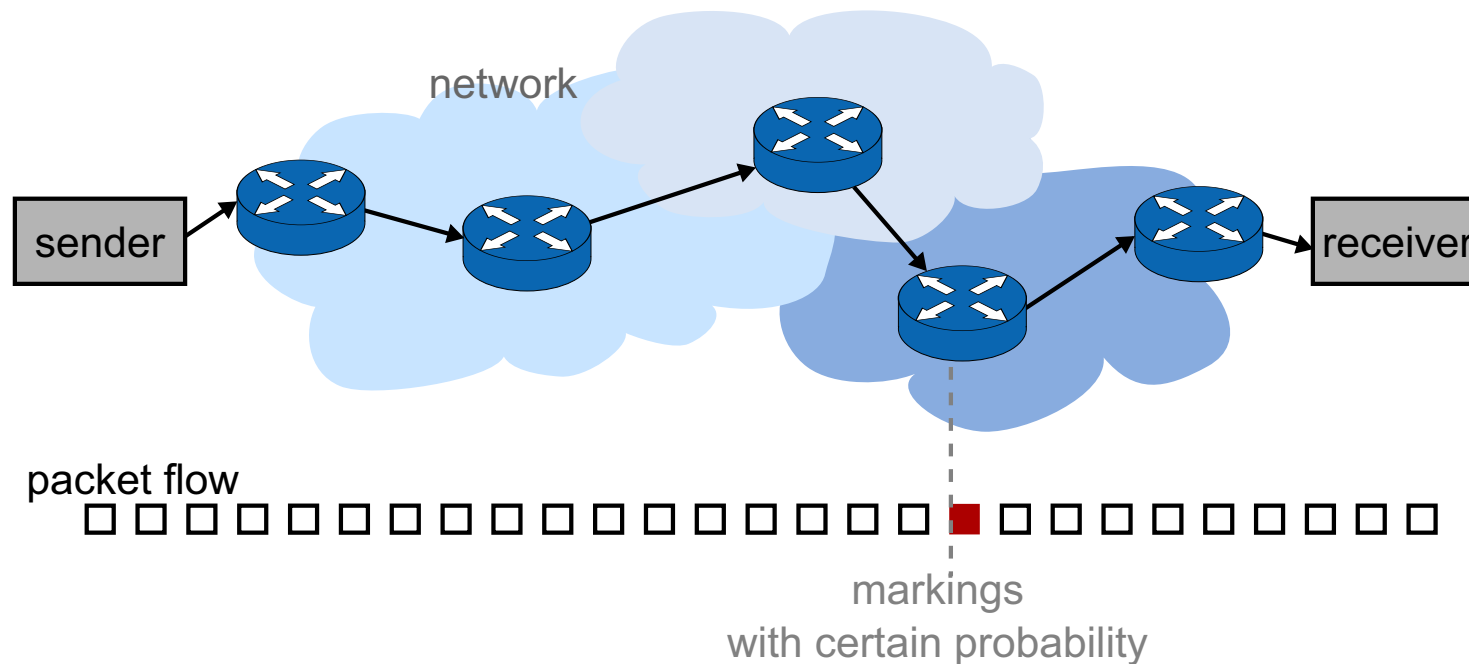


Goal of Congestion Exposure

- Reveal upstream/downstream congestion along the path (to any intermediate node)
- Make senders **accountable** for the congestion they cause in the network

The ECN Protocol

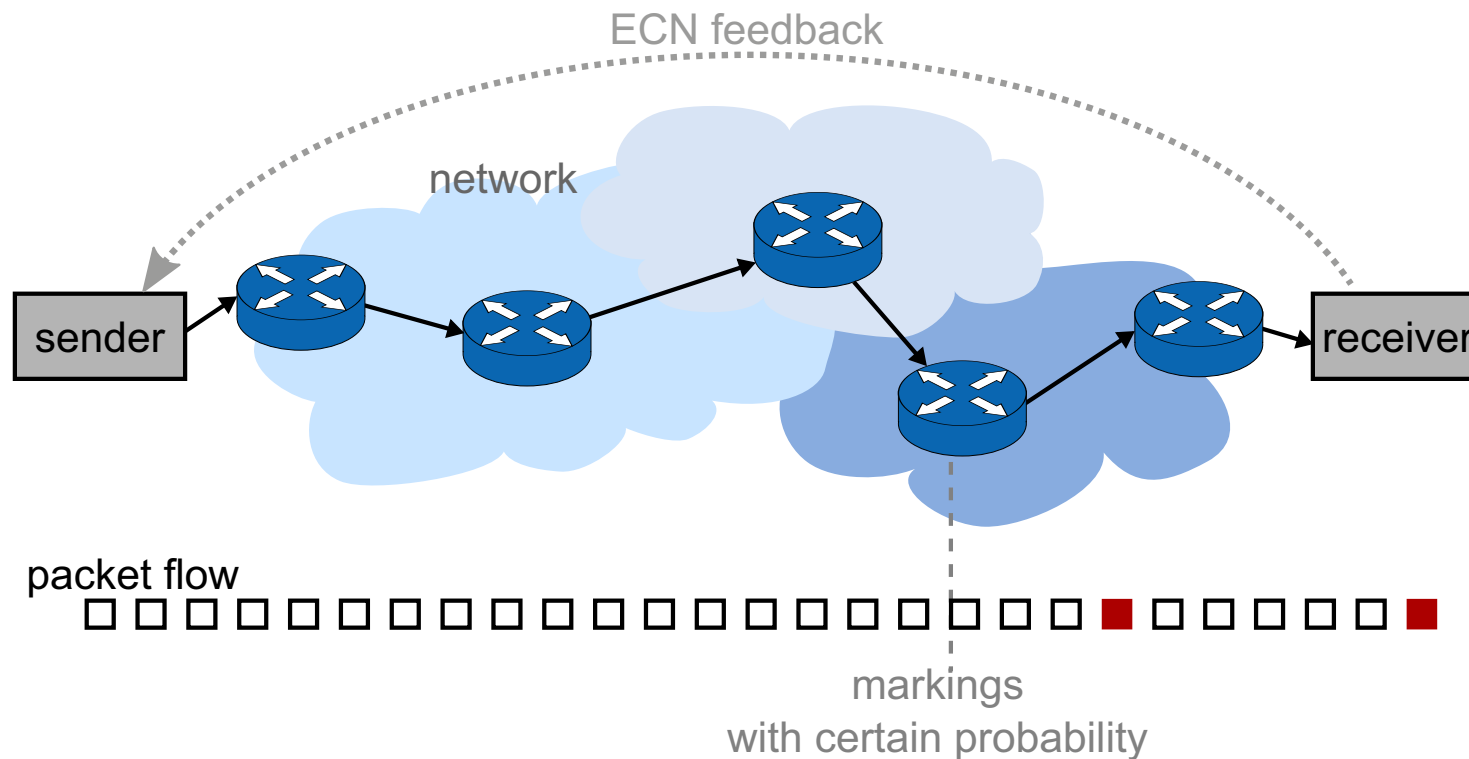
ECN (Explicit Congestion Notification)



1. Routers mark packets (instead of dropping them – Random Early Detection (RED), IP flag)

The ECN Protocol

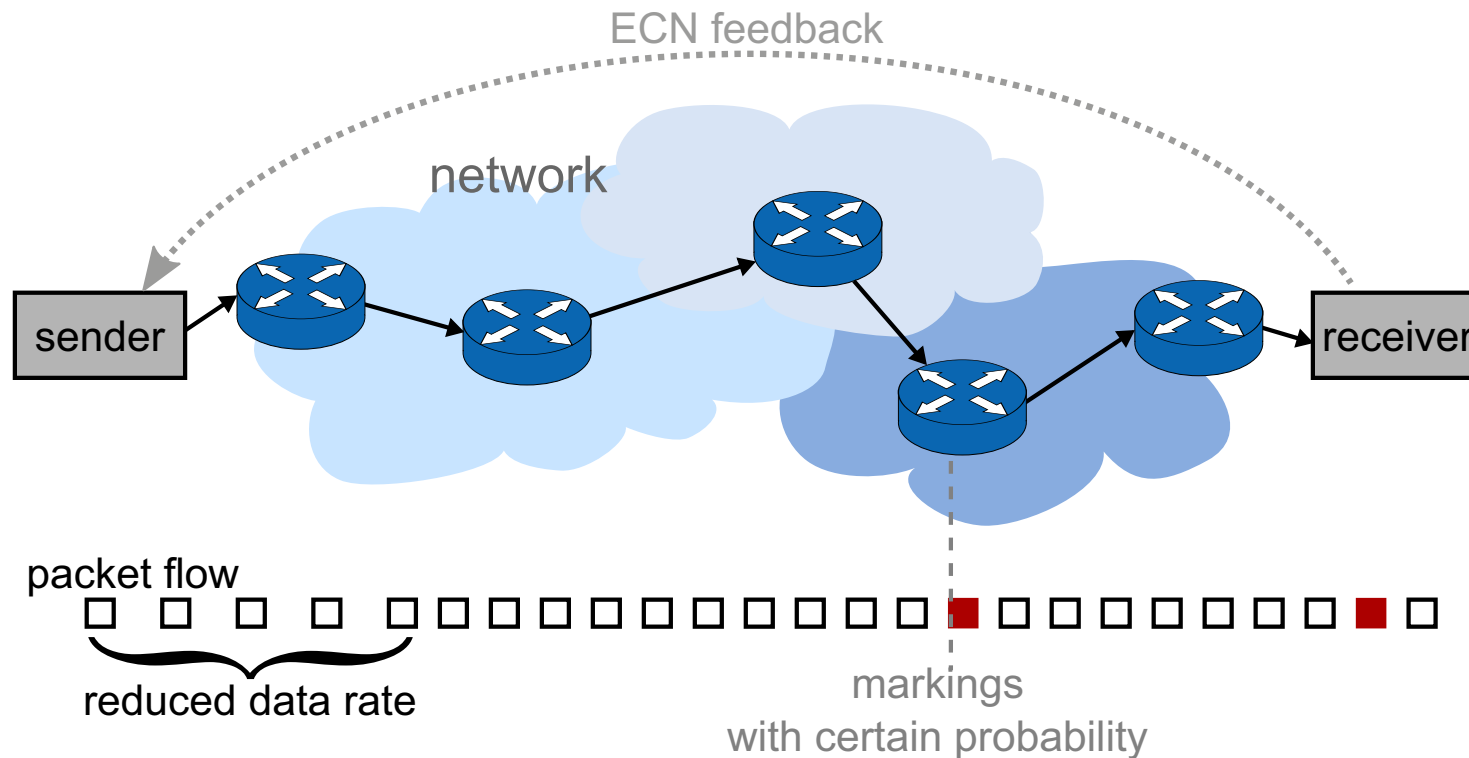
ECN (Explicit Congestion Notification)



1. Routers mark packets (instead of dropping them – Random Early Detection, IP flag)
2. ECN receiver feeds congestion announcement back to the sender (TCP ACK)

The ECN Protocol

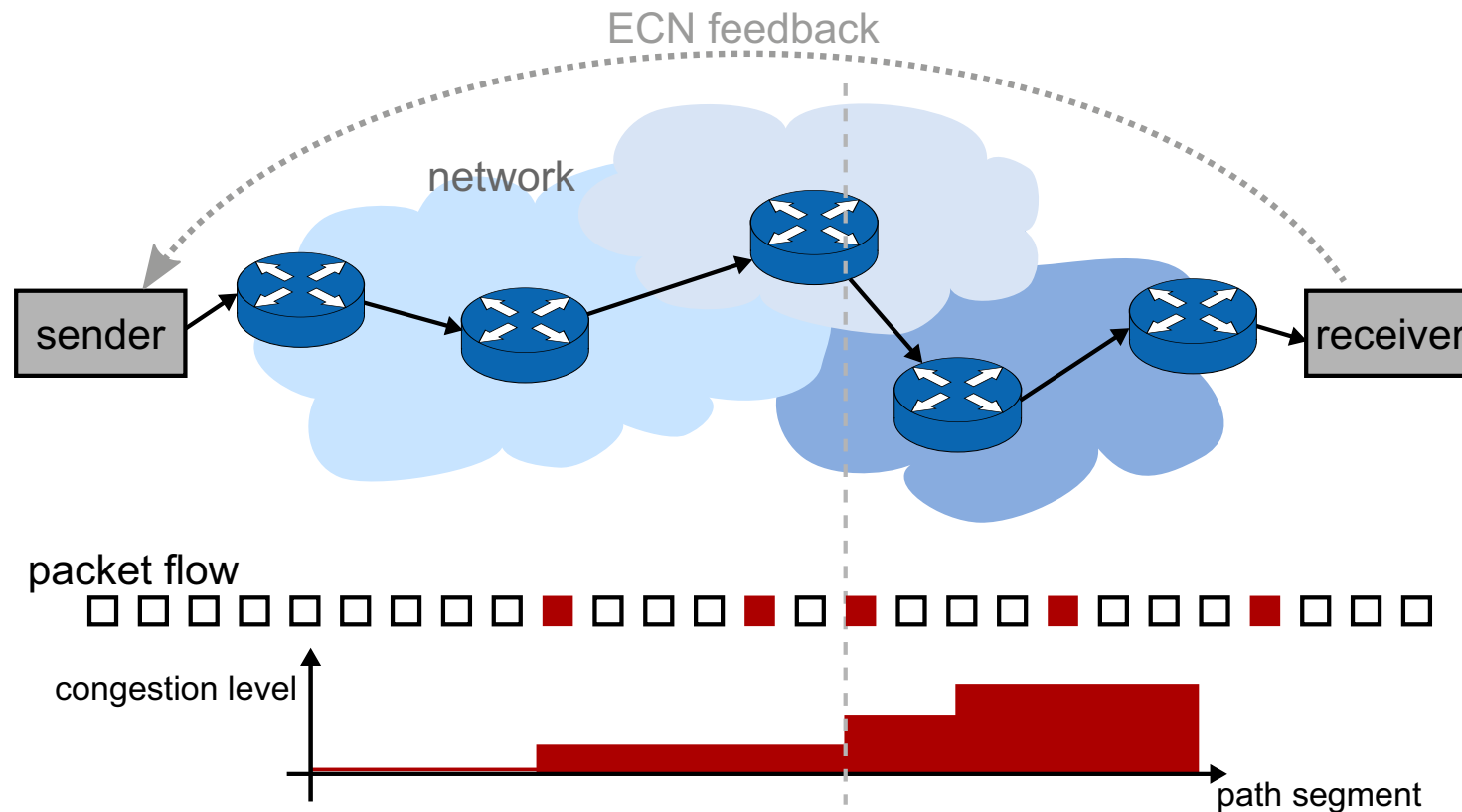
ECN (Explicit Congestion Notification)



1. Routers mark packets (instead of dropping them – Random Early Detection, IP flag)
2. ECN receiver feeds congestion announcement back to the sender (TCP ACK)
3. ECN sender reduces congestion windows (as on loss – TCP Congestion Control)

The re-ECN Protocol

re-ECN (re-insert ECN) – expose expected whole path congestion to network elements

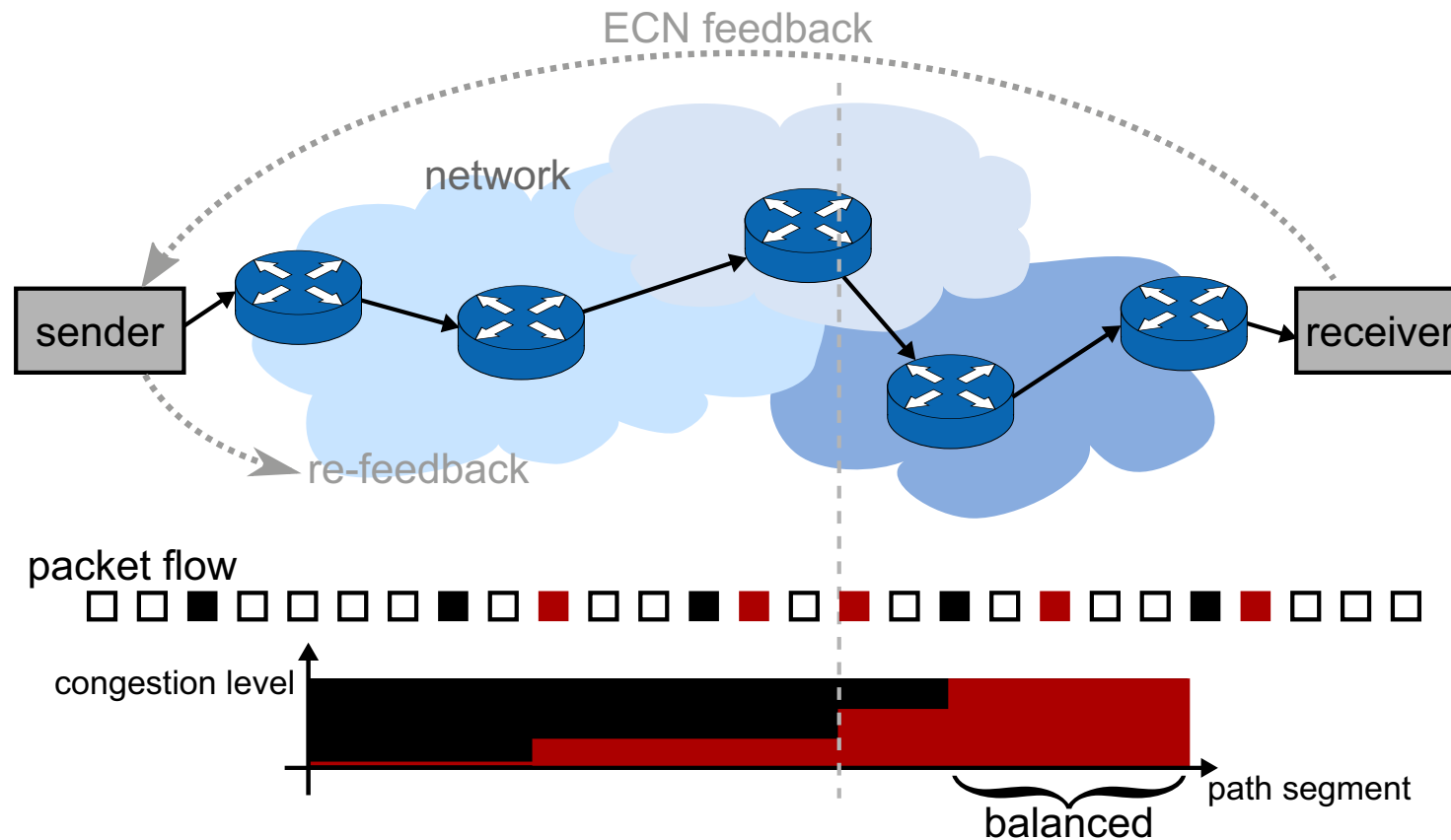


re-ECN sender marks a packet for every congestion announcement from the receiver

- Fraction of red marks ("congested") gives level of congestion experienced so far

The re-ECN Protocol

re-ECN (re-insert ECN) – expose expected whole path congestion to network elements

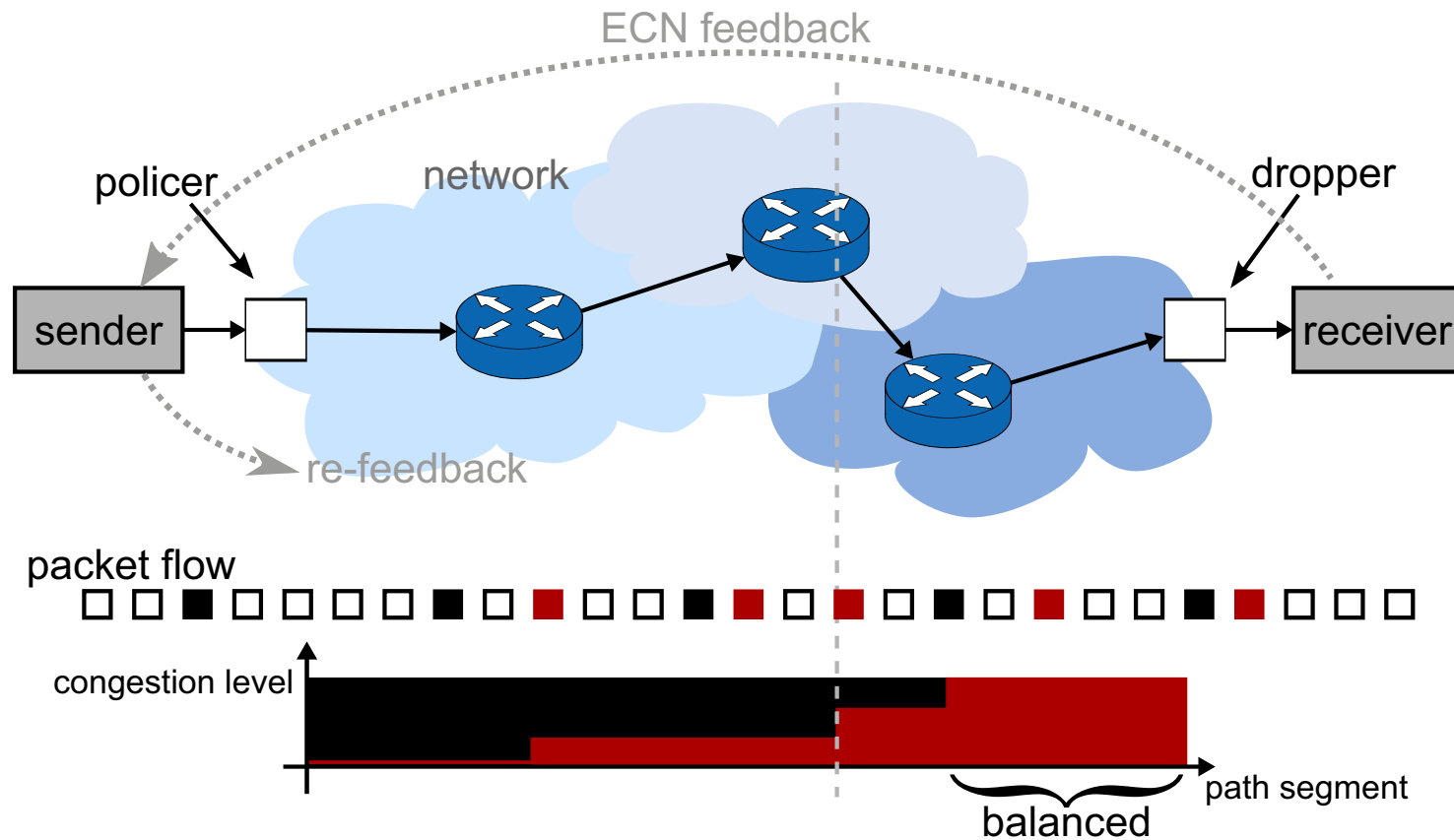


re-ECN sender marks a packet for every congestion announcement from the receiver

- Fraction of red marks ("congested") gives level of congestion experienced so far
- Fraction of black marks ("congestion expected") give the whole path congestion

The re-ECN Framework

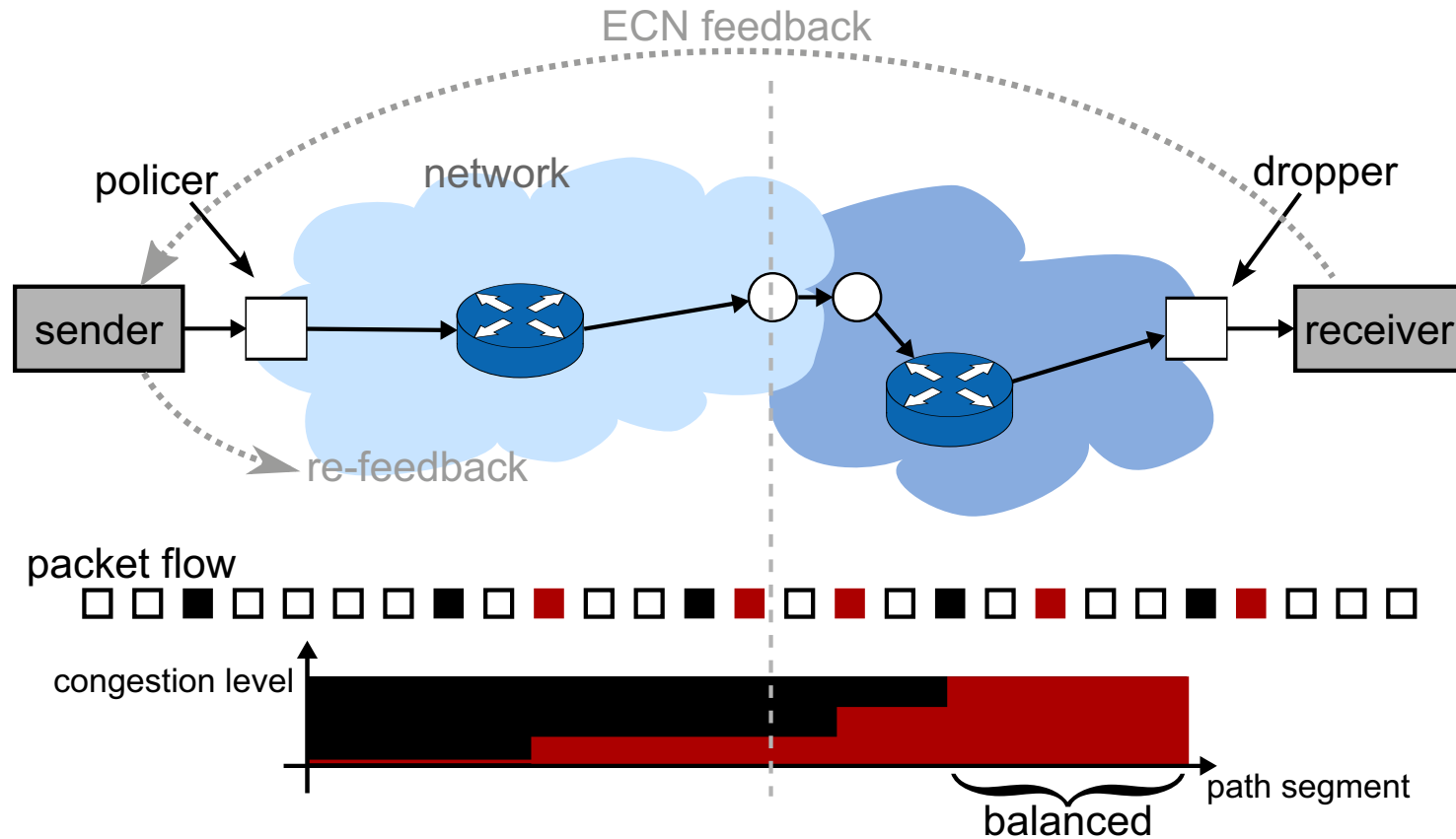
Congestion Accounting – Make senders accountable for the congestion they cause



- Dropper** Penalizing of permanent negative flows (prevent senders from cheating)
- Policer** Implementation of retail contract (e.g. limitation of congestion volume/pos. marked packets, variable congestion charge)

The re-ECN Framework

Inter-Domain Business – Charge sending networks transitively along the path according to the congestion they cause downstream



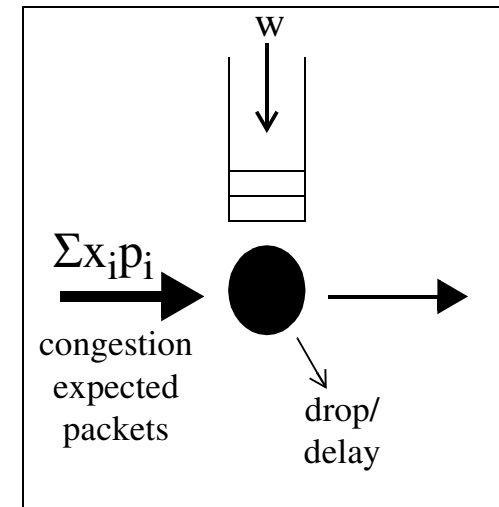
Inter-Domain Policer Wholesale charging for border crossing congestion marks (per link)
Re-routing according to path cost

Congestion-based Traffic Engineering

Token-Bucket Policer

Per-costumer limitation of congestion credits in the ingress node

- Token Rate
 - Provides continuous refill of congestion credits
 - Ensures a minimum access rate
- Bucket Size
 - Determines the max. number of credits that can be stored
 - Saves congestion credits to balance congestion peaks
- Parameterization
 - Might vary for costumers with different contracts, e.g. usage-dependent or flat-rate
 - Might be influenced by agreements between ISPs or e.g. numbers of users in a domain (traffic patterns and/or online times)



Goal

Motivate/control end-system to make appropriate and fair congestion control

→ create an incentives to not cause more congestion than needed while sending data

Congestion-based Traffic Engineering

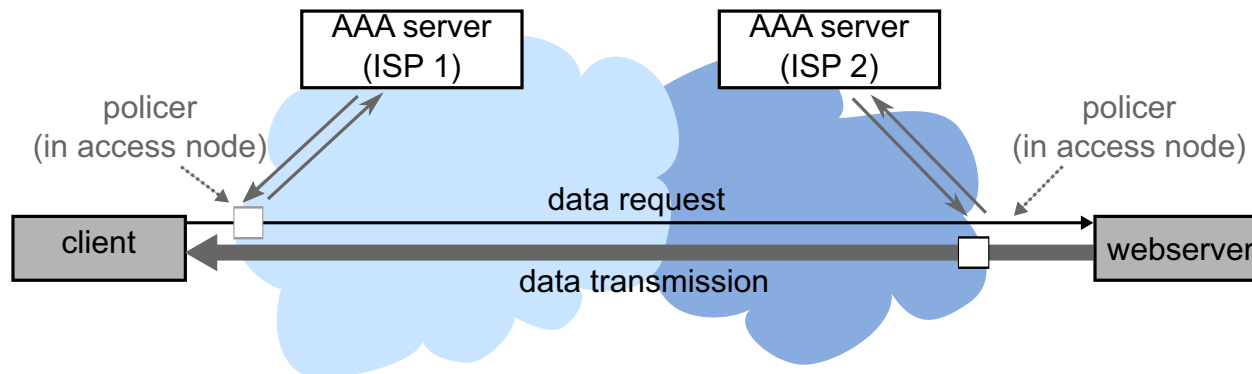
Open Issues

- **Policer design:** Algorithms other than token bucket
 - ISPs may want to use own algorithms, are they compatible?
 - Partial deployment: Not all ISPs on a path may use/support re-ECN
- **Robustness:** Internet resource management must work in all corner cases
 - Congestion is a very volatile metric, difficult to quantify and to explain for congestion management in the end-system
 - Interactions between policers and end-to-end congestion control? Stability?
- **Acceptability:** Not all users and ISPs have an incentive for congestion-based traffic engineering
 - Different to today's business models and SLAs
 - Overall framework has economical requirements, e. g., market competition
- **Unidirectional policers:** Unsuitable for client-server traffic (focus of the rest of this talk)

Congestion Accounting Architecture

Scenario

- Client wants to download data from a webserver
- Policer can only limit congestion volume for a sender's upload traffic
- Receiver is transmission initiator when downloading from a webserver
→ How to react on congestion at sender site?

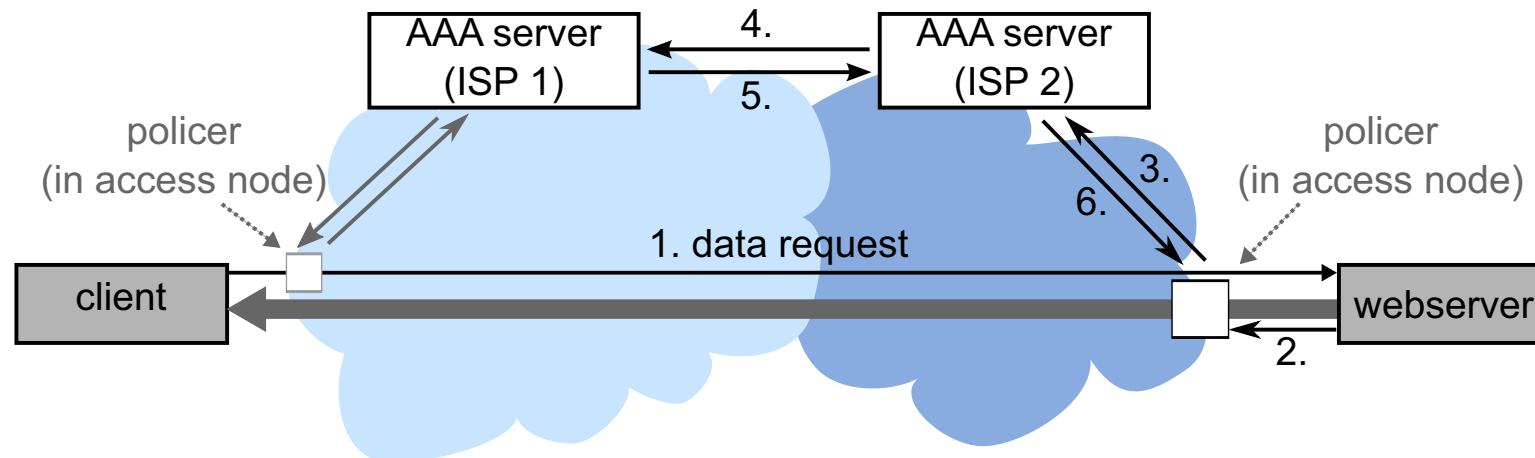


Token Transfer Architecture

- Inform the server about the user-defined importance of a data download, e.g. to conserve a certain sending rate in congestion situations
1. Give the permission to use one's congestions tokens to the server policer (online)
 2. Transfer congestion tokens to the server policer (offline)
 3. Increase amount of congestion token at server side (e.g. advertising)

Congestion Accounting Architecture

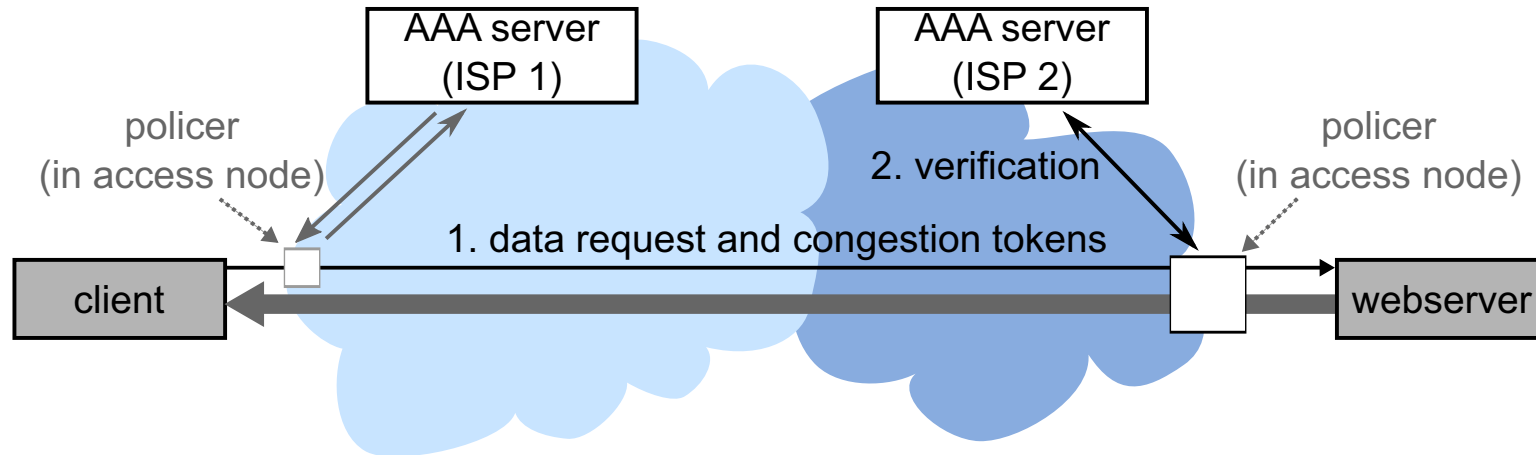
Online Interaction



1. Client sends data request including an assertion that grants the right to make use of the clients congestion credits
 2. The webserver forwards the assertion to the co-located policer
 3. The server-side policer requests congestion credits via its home AAA server from the client's AAA server
 4. The client's AAA server verifies the authenticity/integrity and checks the user's account
 5. The server-side policer will receive a congestion token with a certain amount of credit points as reply (reload is possible depending on the client's refill rate)
- The webserver application will adapt its sending rate to the number of available credit points and to the period until the next reload as well as depending on the congestion level

Congestion Accounting Architecture

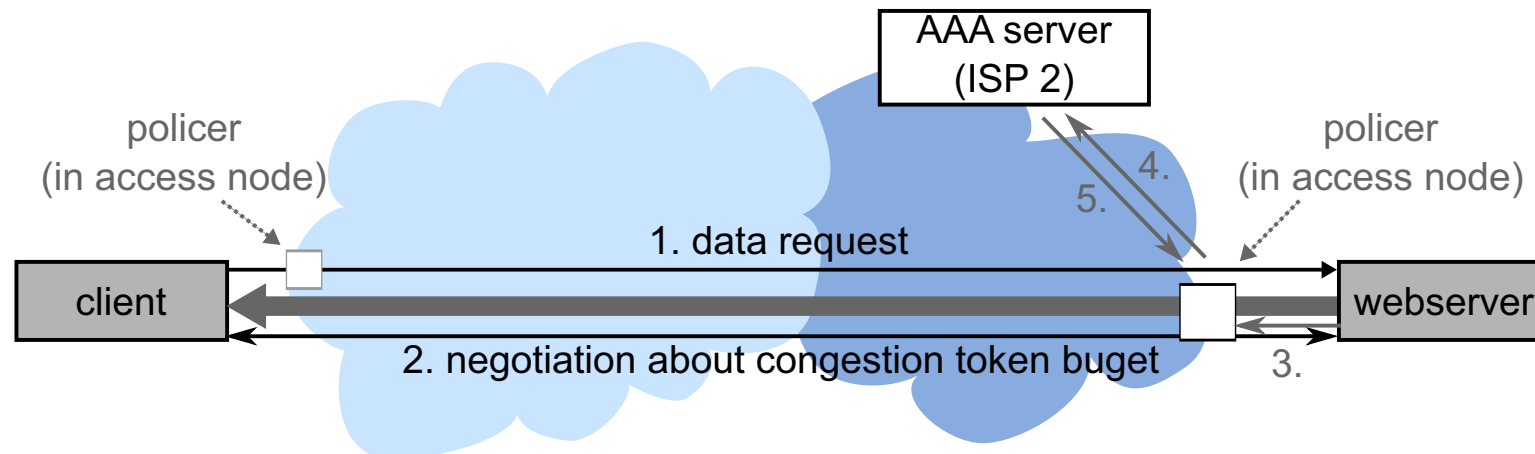
Offline Interaction



1. Client sends data request including congestion tokens signed by the client's AAA server
2. The server-side policer forwards the message for verification and accounting to its home AAA server
3. The webserver will interpret the congestion token message as well to adapt its sending rate to the number of available credit points and depending on the congestion level (reload needs to be requested by the webserver)

Congestion Accounting Architecture

Out-of-Band Interaction



1. Client sends data request
2. Client and webserver negotiate about the user-defined importance of the transmission (e.g. clients is willing to watch commercials, visit online shop, disclose personal data)
3. Webserver will spend its own congestion credits to conserve the negotiated sending rate
4. Webserver might be able to increase its token rate in high congestion situation to fulfill the negotiated requirements

Comparison of Token-Transfer Approaches

Architecture	Online Interaction	Offline Interaction	Out-of-band Interaction
Basic function	ISP2 retrieves tokens from ISP1	Client retrieves tokens from ISP1 and transfers them to ISP2	Server transfers tokens to ISP2; end-user "pays" by other means (e. g., advertisement, e-commerce)
Interaction between ISPs servers	In worst case, message exchange for each client-server transaction	Static security association to verify credentials in tokens	None
End user involvement	Little Must signal its preference to sender	Large Client may be involved in token transfer	Little/None Server can handle this automatically based on user profile or similar
Requirements	Inter-domain AAA interface	Secure transfer of tokens along a path with signatures	Interface to ISP to increase token bucket size

→ Well-known accounting architectures could be used congestion accountability as well

→ Congestion credit transfer is a new paradigm that requires further studies

Conclusion and Outlook

Summary

- **re-ECN:** Exposure of expected congestion on a network path (to network elements)
 - **Policer:** Limitation of congestion volume per end-system/costumer
 - **Congestion Accounting Architecture:** Token Transfer in download scenarios
- Congestion accountability could be a new, disruptive paradigm for the **Future Internet**

Open Issues

- Policer and dropper design
- Congestion management in the end-system
- Congestion Metric: Is RED the right metric for congestion charging?
→ Decouple congestion charging from operative TCP congestion management?
- Probabilistic signaling: Is the probabilistic congestion signaling fast enough?
→ Short lived application streams cannot adapt to it on time
- Economic implications (Inter-domain charging and Inter-domain routing)