Experimental Study of Protocol-independent Redundancy Elimination Algorithms

WOSP/SIPEW 2010

Maxim Martynov

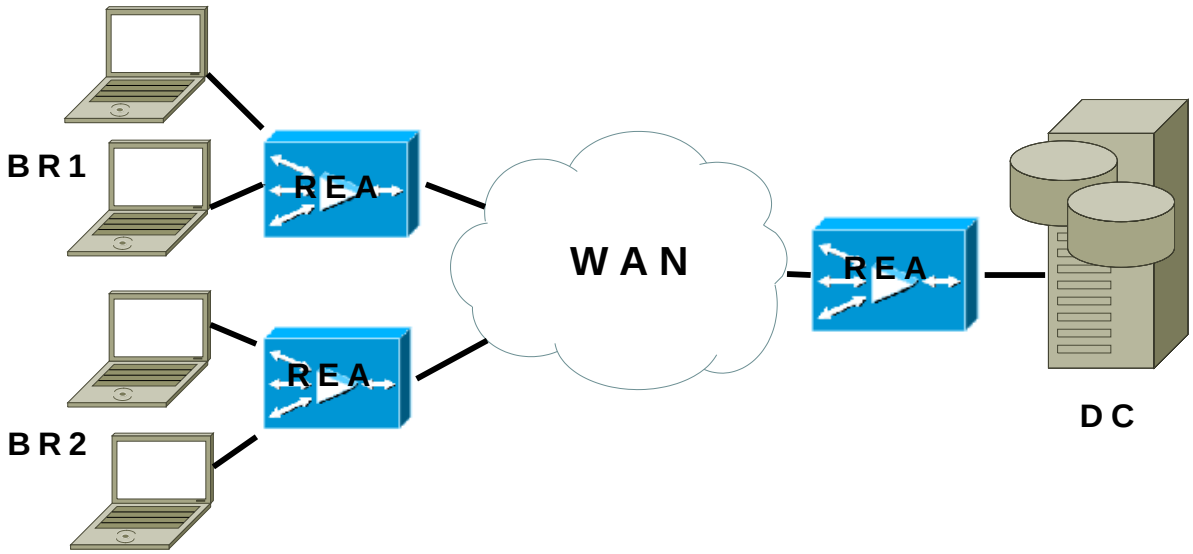
now with Arista Networks, work done while at Cisco Systems

Outline

- Redundancy eliminating (RE) systems overview.
- General architecture.
- Basic alternatives
- Benchmarks and measurements.
- System evaluation methodology.

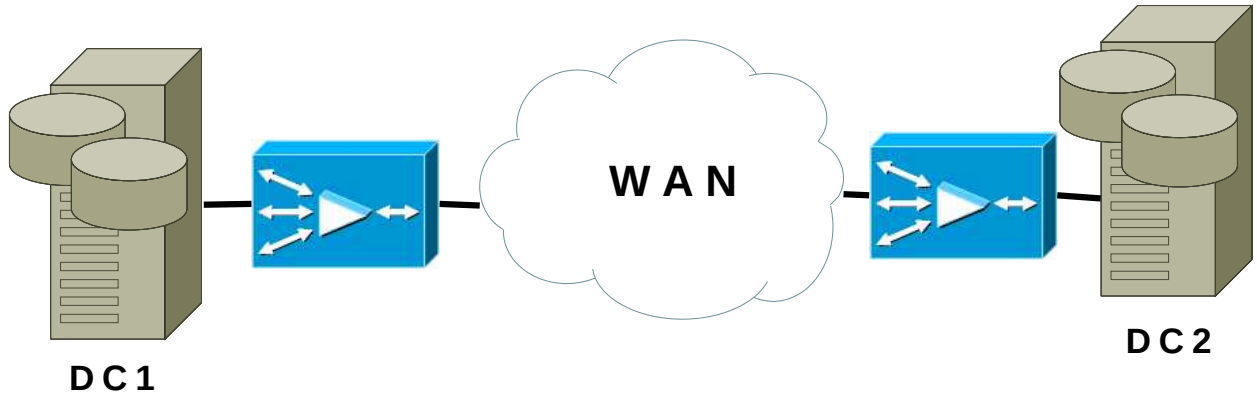
Foundations of RE

- [Manber ’94] - technique to find similarities between large files.
- [Spring & Wetherall ’01] - apply Manber’s technique to network data flow.
- Idea: Find repeating patterns in data flow and replace them with labels.
  - Maintain synchronized data caches.
- Typical deployment:



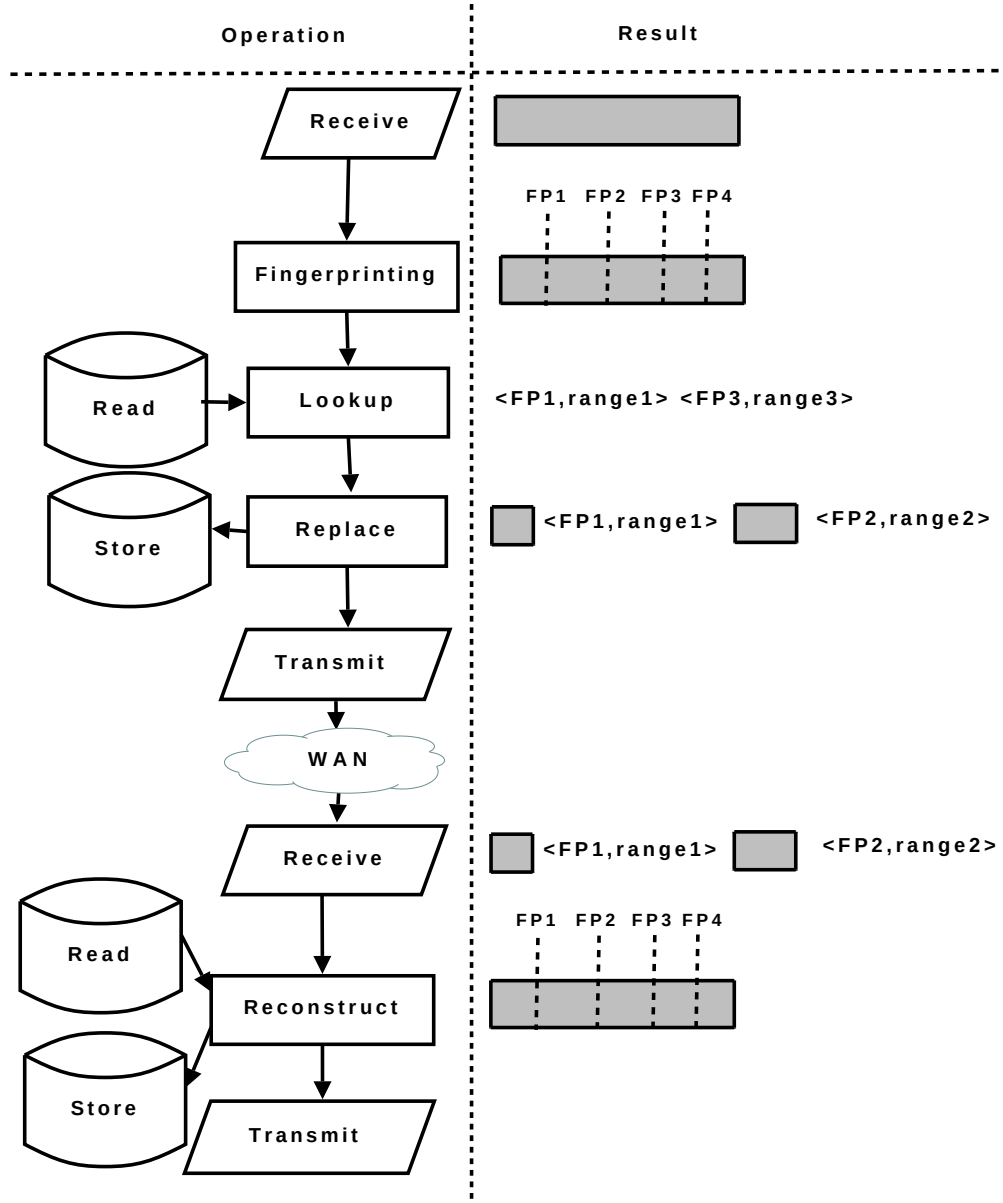
Why performance?

- Currently RE is used to optimize slow links.
- Trends:
  - WAN links are getting faster.
  - Fan-out grows (number of branches increases).
  - Desire to get RE savings everywhere.
- Example of desired deployment:



- RE systems must scale up to higher speeds.

Basic flow


Operation
Result
Receive
FP1 FP2 FP3 FP4
Fingerprinting
Read
Lookup
<FP1,range1> <FP3,range3>
Store
Replace
<FP1,range1>
<FP2,range2>
Transmit
WAN
Receive
<FP1,range1>
<FP2,range2>
Read
FP1 FP2 FP3 FP4
Reconstruct
Store
Transmit

Receiving data

- Goal: obtain data from the network
- Buffers vs. Packets (TCP vs. IP)
- TCP
  - Read and process data in long buffers.
  - Compression upper bound is high.
  - Better storage access locality.
  - Connection termination overhead.
- IP
  - Read and process data in short packets.
  - Faster processing, no connection overhead.
  - Compression upper bound is low.
  - Poor storage access locality.

Fingerprinting/Chunking

- Goal: split data into chunks. Identical chunks imply redundancy.
  - Typically done by Rabin fingerprinting over a sliding window.
- Main parameter of choice: inter-FP distance.
- Short inter-FP distance, fine-grain detection.
  - More redundancies detected.
  - Slower processing.
  - More protocol overhead.
- Long inter-FP distance, coarse-grain detection.
  - Less redundancies detected.
  - Faster processing.
  - Less protocol overhead.

Indexing/Lookup

- Goal: find identical pieces of data in the data stream.
- Done in two phases: fingerprint lookup, data lookup.
  - Find current fingerprint in the cache.
  - Find longest match in the data surrounding the fingerprint.
- Data lookup: going through data around a fingerprint.
- Byte-by-byte comparison
  - Produce more accurate matches.
  - Intensive storage access.
- Checksum comparison.
  - Less storage access.
  - Matches pinned to fingerprint boundaries, less accurate.
  - Checksum calculation overhead.

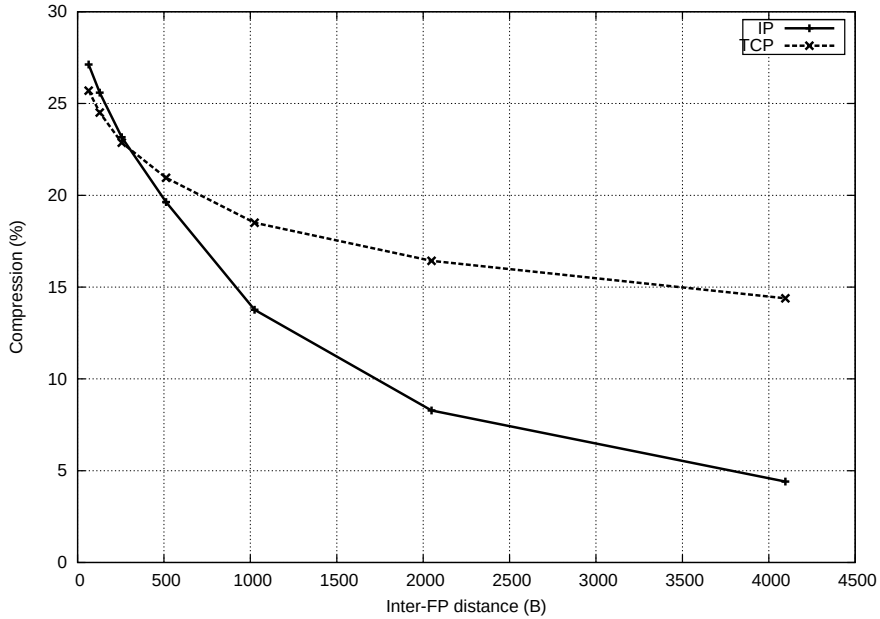
Storing data

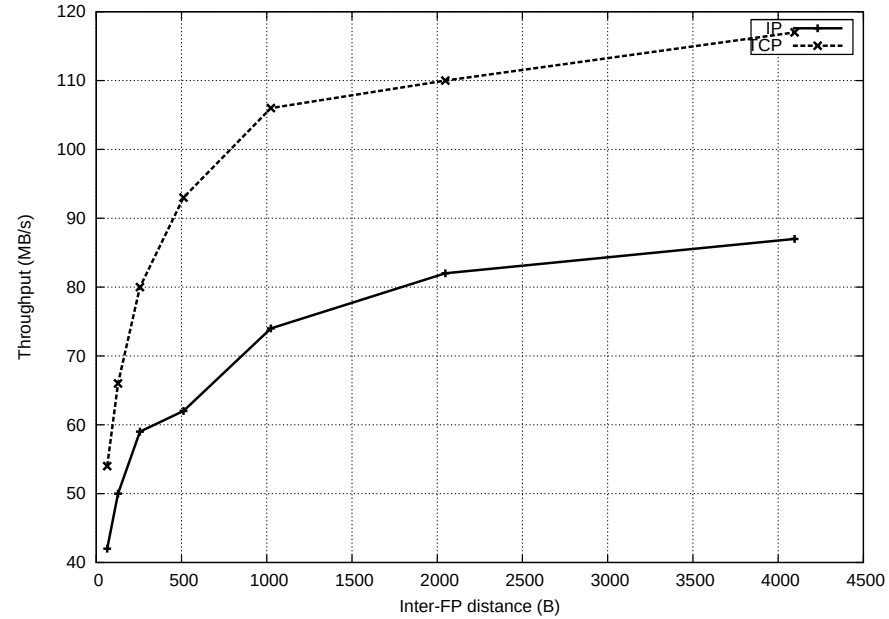
- Goal: store data for possible future retrieval.
- Main choice: What to write?
- Write all incoming data.
  - Write overhead.
  - Storage overhead.
  - Better locality in future.
- Write only new data.
  - Poor locality (sometimes).
  - Less write overhead.
  - Less storage required.

System design choice

- Problem:
  - Too many unknowns.
  - Too many dependencies.
  - No obvious outcome for each parameter choice.
- Approach:
  - Understand most basic trends.
  - Develop a quantitative metrics for system configuration.
- Measurement:
  - System based on a real WAN-Op product.
  - Three synthetic benchmarks to represent special cases.
  - One benchmark with real inter-office data traffic.
  - Run benchmarks data through the system, vary configuration.

Example of RE system behavior





- Benchmark-4: Compression and Throughput vs. inter-FP distance.
  - Including TCP vs. IP comparison.
- More graphs in the paper.

System evaluation

- Benefit function: $\beta(w) = \frac{T(w)-w}{w}$.
  - w – WAN link bandwidth, r – compression ratio, $T(w)$ – effective throughput.
  - $T(w) = \min(T_{max}, \frac{w}{1-r})$
- $\beta(w)$ determines a benefit for a given link.
- As WAN links vary, let $\mathcal{W}$ be a space of link capacities.
- A value of a system for the benchmark i: $\int\limits_{\mathcal{W}_i} \beta_i(w)dw$
- System config score: $S = \sum\limits_i \alpha_i \int\limits_{\mathcal{W}_i} \beta_i(w)dw$
  - α_i – weight of the benchmark i.

System evaluation example

Lookup	Write	Inter-FP distance	TCP or IP	Score
by checksum	new	128	TCP	23.51
by checksum	new	64	TCP	22.81
by checksum	all	128	TCP	22.27
by checksum	new	256	TCP	21.99
by checksum	all	64	TCP	21.36
by checksum	all	256	TCP	20.89
byte-by-byte	new	512	TCP	20.36
byte-by-byte	all	256	TCP	20.26
byte-by-byte	all	512	TCP	19.97
byte-by-byte	new	256	TCP	19.43

Table 1: Rankings and scores of algorithm configurations on Benchmark-4 (top 10)

The End

- Questions?
- maxim@aristanetworks.com

Thank you!