Canal: Scaling Social Network-Based Sybil Tolerance Schemes

Robert Pohnke

Plan

1. Problem Description
2. Canal mechanism
3. Results
4. Q&A

Reputation systems

- A **reputation system** computes and publishes reputation scores for a set of objects (e.g. service providers, services, goods or entities) within a community or domain, based on a collection of opinions that other entities hold about the objects.

Reputation systems

- Many different variations: recommender systems, collaborative filtering, voting systems
- Examples: eBay, PageRank, YouTube, Digg, CoachSurfing
- But also social media – Facebook, LinkedIn

Sybil attacks

- Multiple pseudonymous identities forged by a malicious user (or a group)
- Used to gain unfair advantage over honest users
- Possible effects: content manipulation, SPAM, fraud transactions, ...

Defense mechanisms:

- Sybil prevention – preventing the attacker from creating Sybil identities (CAPCHA, Document Verification)
- Sybil detection – detecting and removing fake identities;
- Sybil tolerance – designed to limit the impact that a malicious user can have on others

Detection vs. Tolerance

- Although an attacker can create an arbitrary number of Sybil identities in the social network, she cannot establish an arbitrary number of social connections to non-Sybil identities.
- The non-Sybil region of the network is densely connected, meaning random walks in the non-Sybil region quickly reach a stationary distribution

Ostra

- Ostra is targeted at countering unwanted communication (i.e., SPAM). Ostra assumes the existence of a social network, and assigns credit values to the links. When a user wishes to send a message to another user, Ostra locates a path with available credit from the source to the destination.

SumUp

- SumUp is designed to prevent users with multiple identities from manipulating object ratings in content sharing systems like Digg. SumUp assumes the existence of a social network and selects a trusted vote collector.

Bazaar

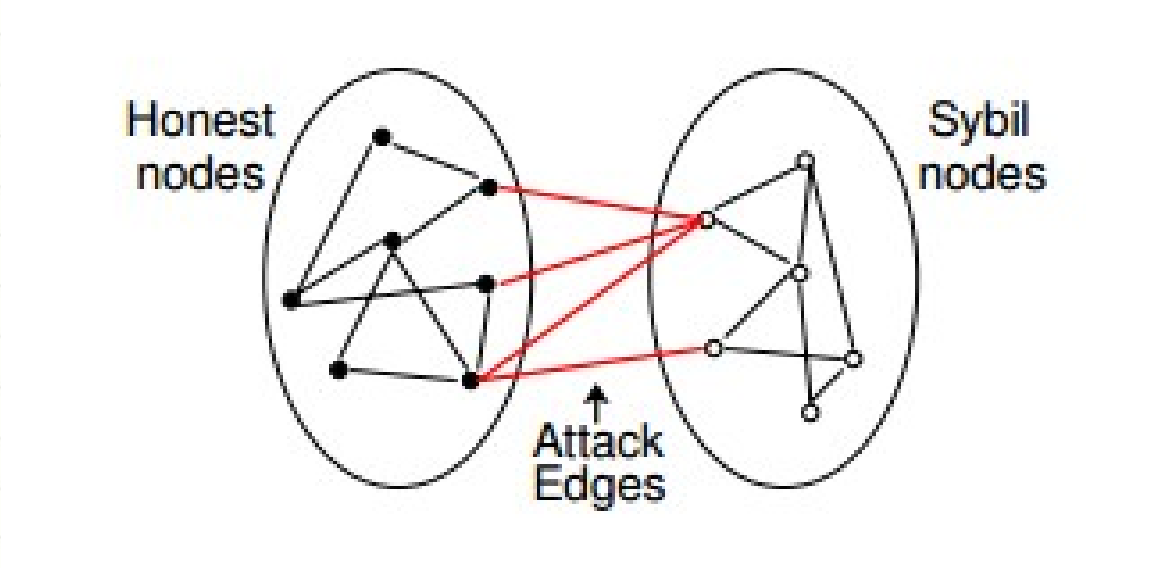
- Bazaar provides stronger user reputations in online marketplaces like eBay. To do so, Bazaar creates a transaction network by linking pairs of identities that have successfully completed a transaction; the weight of each link is the dollar value of the transaction.
- When a new transaction is about to take place, Bazaar compares the value of the new transaction to the max flow between the buyer and seller.

Credit network

- Most popular approach to sybil tolerance
- Based on trust: each user assigns amount of trust to other users, typically credits.
- Every action (i.e. sending a message) has a cost. The action is allowed if a path from A to B exists with enough credits to cover the action cost.
- If the action is not reported as a fraud, the credit is refunded

Why does it work?

- Based on the second assumption:



Problem?

- The most efficient algorithms for the maximum flow problem run in O(V3) or O(V2 log(E))
- The network is dynamic, credit values change rapidly

Canal

Canal is extending the concept of credit networks. It trades off accuracy for speed. It is designed to run alongside an existing Sybil tolerance scheme, providing two services:

- maintaining the state of the credit network ,
- conducting credit payments.

Landmark routing

Simple idea – instead of computing max flow to all users, lets compute the distance to a landmark and stitch a path from A to B via landmark.

Note that credit transfer does not require the path to be the shortest one – we are only interested if it exists.

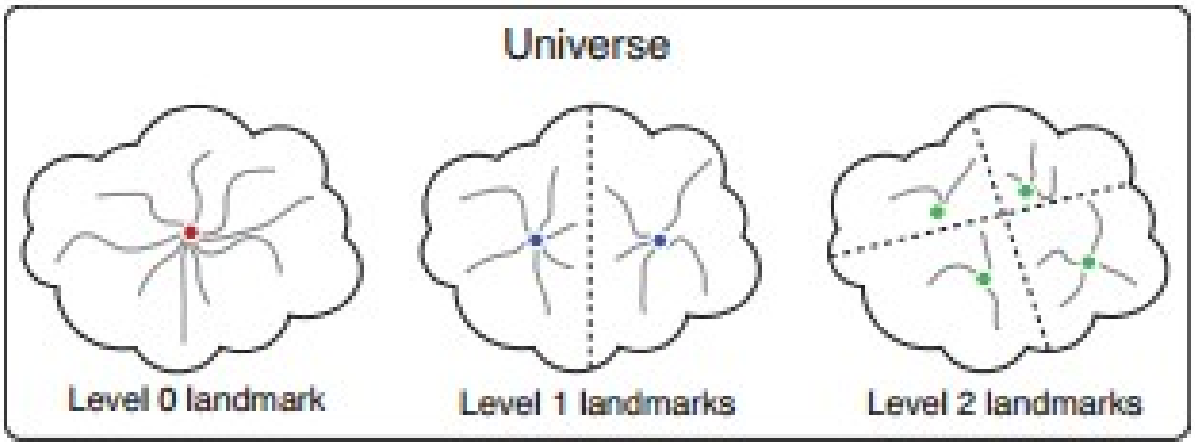
Landmark Universe

- Lets define k-Universe as a network with k levels of landmarks, each of them consisting of 2k elements:

20, 21, 22, ... , 2k (2k+1 -1 in total)

- Every user computes the path to the nearest landmark on each level. Every pair of users is bound to have at least one common landmark.

Landmark Universe



Universe Creator

1. Randomly select k random node sets of the appropriate sizes from the network. Let the selected sets be denoted by V0, V1, V2, ... Vk. These sets contain the new landmarks at each level.
2. For each set Vi, and every node u ∈ V , calculate the shortest path from u to each of the landmark nodes in each set Vi. This is done by having the processes perform BFSs from each landmark in Vi.
3. Finally, using the BFSs, construct the landmark map for level Vi by select the closest landmark node in Vi and the next hop for all nodes.

Path stitcher

1. Scan the k landmark maps and collect the set of common landmarks between *a* and *b*.
2. For each shared landmark, use the next hop in the landmark map to “stitch” together a path via the landmark.
3. Refine the path by eliminating any cycles and performing path short-cutting. To perform short-cutting, we traverse the path up to the landmark node and see if there is a link from any of these nodes to a node lying in the path after the landmark node. If so, we short-circuit the path by using that link to create a shorter path between *a* and *b*.

Updating paths

- The path stitcher process pays as much credit as possible along each path. For each path, the path stitcher process walks the path, obtaining the lock on each link of the path, temporarily lowering the credit available to 0, and then releasing the lock.
- Once the end of the path has been reached, the path stitcher calculates the maximum credit available on the entire path. Next, the appropriate values are restored on the whole path.

Results

Network	Nodes	Links	Avg. degree	Avg. max flow time (s)
Renren [18]	33 K	1.4 M	21.1	0.352
Facebook [38]	63 K	1.6 M	25.7	0.445
YouTube [23]	1.1 M	5.8 M	5.2	2.91
Flickr [22]	1.6 M	30 M	18.8	15.2
Orkut [23]	3.1 M	234 M	76.3	220

Table 1. Statistics of the networks we evaluate Canal on. Also included is the average time for completing a max flow computation.

Renren	Facebook	YouTube	Flickr	Orkut
225	292	4,131	13,296	41,787

Table 2. Time in milliseconds to calculate a level-0 landmark universe with one universe creator process for various datasets.

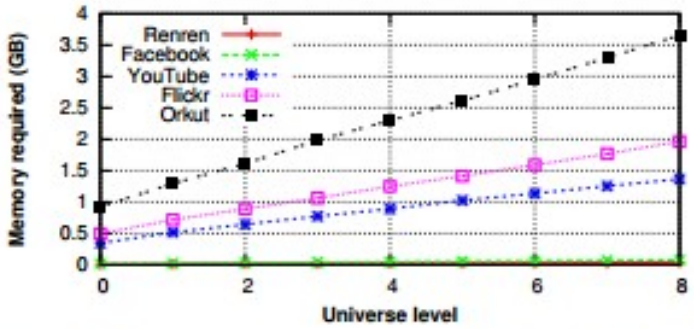


Figure 7. Memory requirements of different universe levels. The memory required increases linearly with the universe level.

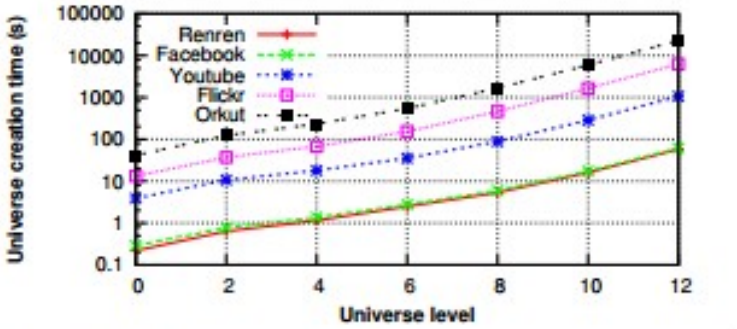


Figure 9. Graph showing the absolute landmark universe creation time as we increase the number of universe levels, for Canal configured with 22 universe creator processes.

Results

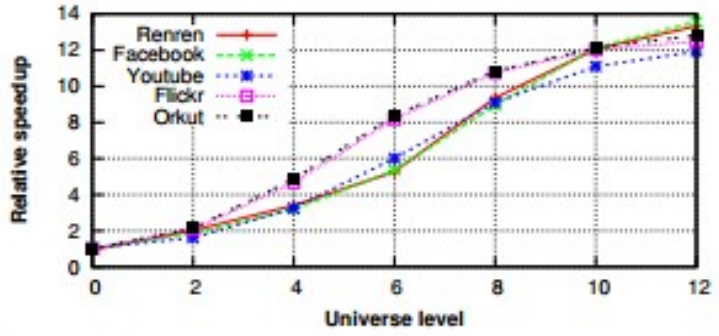


Figure 8. Landmark universe creation time speedup, relative to a single universe creator process, for Canal configured with 22 universe creator processes.

Network	1 credit Median	1 credit 95th P.	5 credits Median	5 credits 95th P.
Renren	0.04	0.09	0.40	0.64
Facebook	0.04	0.13	0.52	0.74
YouTube	0.14	0.59	1.3	2.3
Flickr	0.17	0.51	1.3	2.0
Orkut	0.34	0.83	0.89	1.9

Table 3. Median and 95th percentile time in milliseconds taken by Canal to respond to a `payment` requests pushing a one unit and five units of credit.

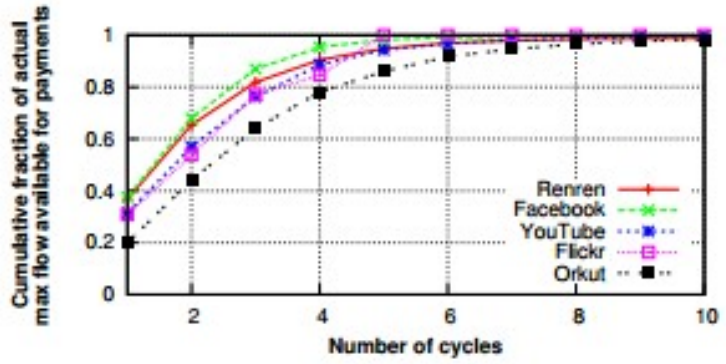


Figure 10. Cumulative fraction of actual max flow that is available for payments in Canal, for increasing cycles of landmark universe creation. We observe that nodes can quickly access all of their available credit.

Category	Nodes	Links
Clothes	1.3 M	5.5 M
Home	1.3 M	4.5 M
Collectables	419 K	1.2 M
Electronics	600 K	1.5 M
Computing	626 K	1.7 M

Table 4. Size statistics of the different categories of risk networks used in evaluating Canal implementation of Bazaar.

Results

Category	Orig. Avg.	Canal Med	Canal 95th P.	Relative Speedup
Clothes	6,290	0.2	3.4	2,329 ×
Home	5,340	0.1	3.4	785 ×
Collectables	1,180	0.08	2.0	1,404 ×
Electronics	1,660	0.09	2.70	1,522 ×
Computing	1,410	0.1	2.56	1,084 ×

Table 5. Time in milliseconds required to process credit network transactions in Bazaar with Canal with 30 level-2 landmark universes. Also included is the original processing time from the Bazaar paper and the relative speedup. We observe speedups between 785-fold and 2,329-fold.

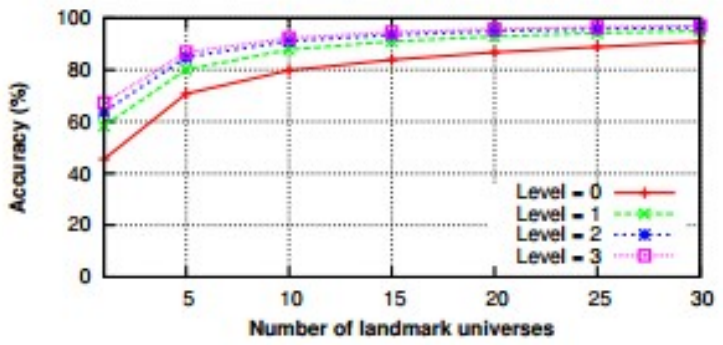


Figure 12. Accuracy of Bazaar with Canal for the Home category, for varying numbers of landmark universes and universe levels. Over 95% accuracy can be achieved with 20 level-3 landmark universes.

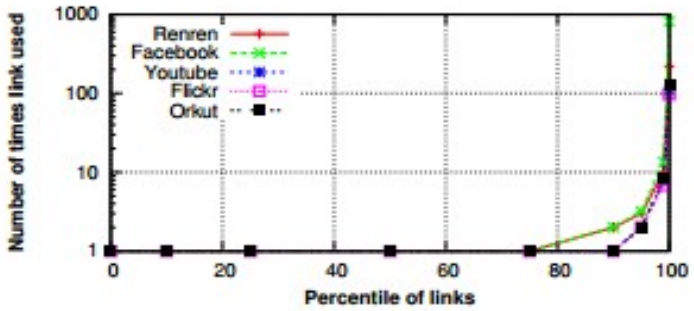


Figure 11. Distribution of the number of times links are used when processing 5,000 random credit payments. For all networks, the 99th percentile links are used fewer than 14 times.

Results

Category	Accuracy
Clothes	94.2%
Home	97.0%
Collectables	97.6%
Electronics	95.4%
Computing	95.9%

Table 6. Accuracy of Bazaar implementation using Canal in each category, relative to the original Bazaar implementation. Canal provides high accuracy for Bazaar, implying that users are rarely impacted by the approximate available credit that Canal finds.

Original Avg.	Canal Median	Canal 95th Percentile	Relative Speedup
35.4	0.05	1.4	186 ×

Table 7. Time in milliseconds required to process a credit network transaction in Ostra in Canal with 30 level-3 landmark universes. Also included is the original processing time from the Ostra paper and the relative speedup.

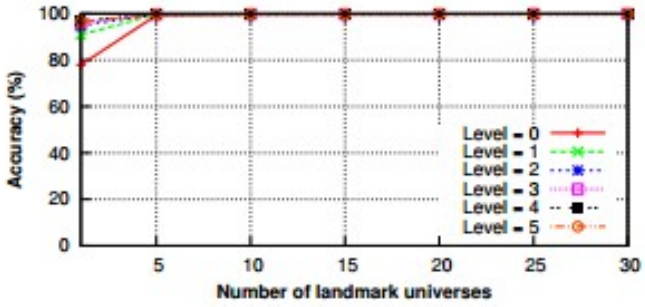


Figure 13. Accuracy of the Ostra implementation using Canal, for varying numbers of landmark universes and universe levels. Over 99% accuracy can be achieved once 5 landmark universes are used.

Thank you for your attention

Questions?