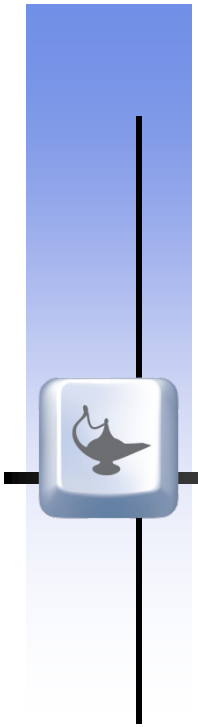


Reliable Client Accounting for P2P-Infrastructure Hybrids

Paarijaat Aditya[†], Ming-Chen Zhao[‡], Yin Lin[*],
Andreas Haeberlen[‡] , Peter Druschel[†], Bruce Maggs[*◊], Bill Wishon[◊]

[†]Max Planck Institute for Software Systems (MPI-SWS)

[‡] University of Pennsylvania

[*]Duke University

[◊] Akamai Technologies

NSDI 2012, San Jose, April 25, 2012

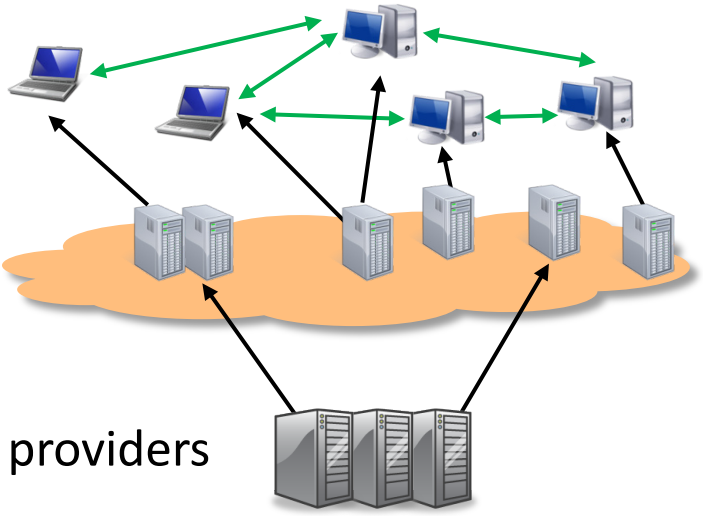
Trends in Content Distribution Networks

- Centralized CDN
  - Clients download from CDN servers, customers pay CDN provider
- New trend: hybrid or peer assisted distribution
  - Clients download from peers and CDN servers
  - Scalability of P2P + reliability & manageability of a centralized system
  - E.g. Akamai NetSession, Velocix P2P Assisted delivery, …

Clients

CDN Servers
(Infrastructure)

Customers – Content providers



Hybrid Systems - Challenges

- Untrusted clients + Infrastructure can't observe P2P communication

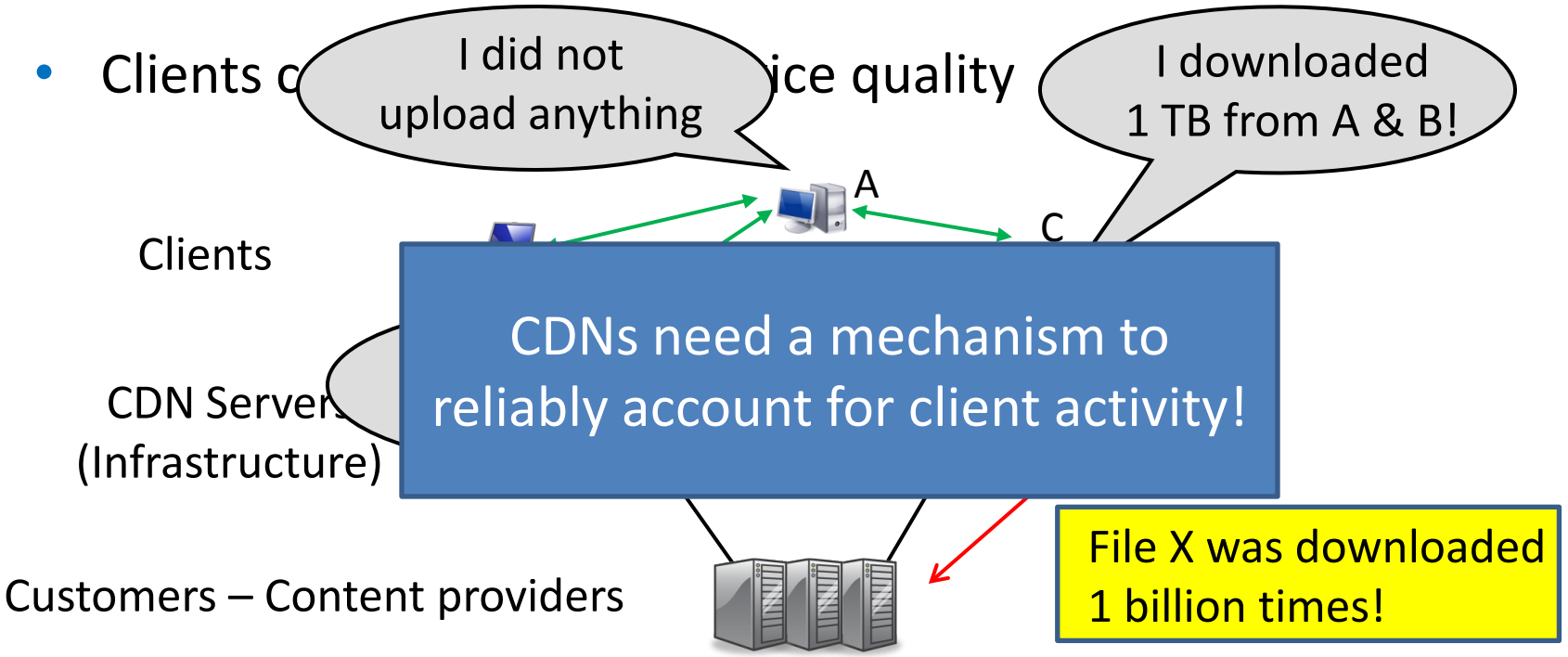
- What could go wrong? In principle clients may
  - Mishandle content: modify, inject or censor content
  - Affect service quality: delay or abort transfers
  - Misreport P2P transfers

What Do CDNs Currently Do

- Infrastructure provides signed metadata
  - Clients can verify content integrity
- Infrastructure as fallback
  - Maintain service quality in case of failed transfers

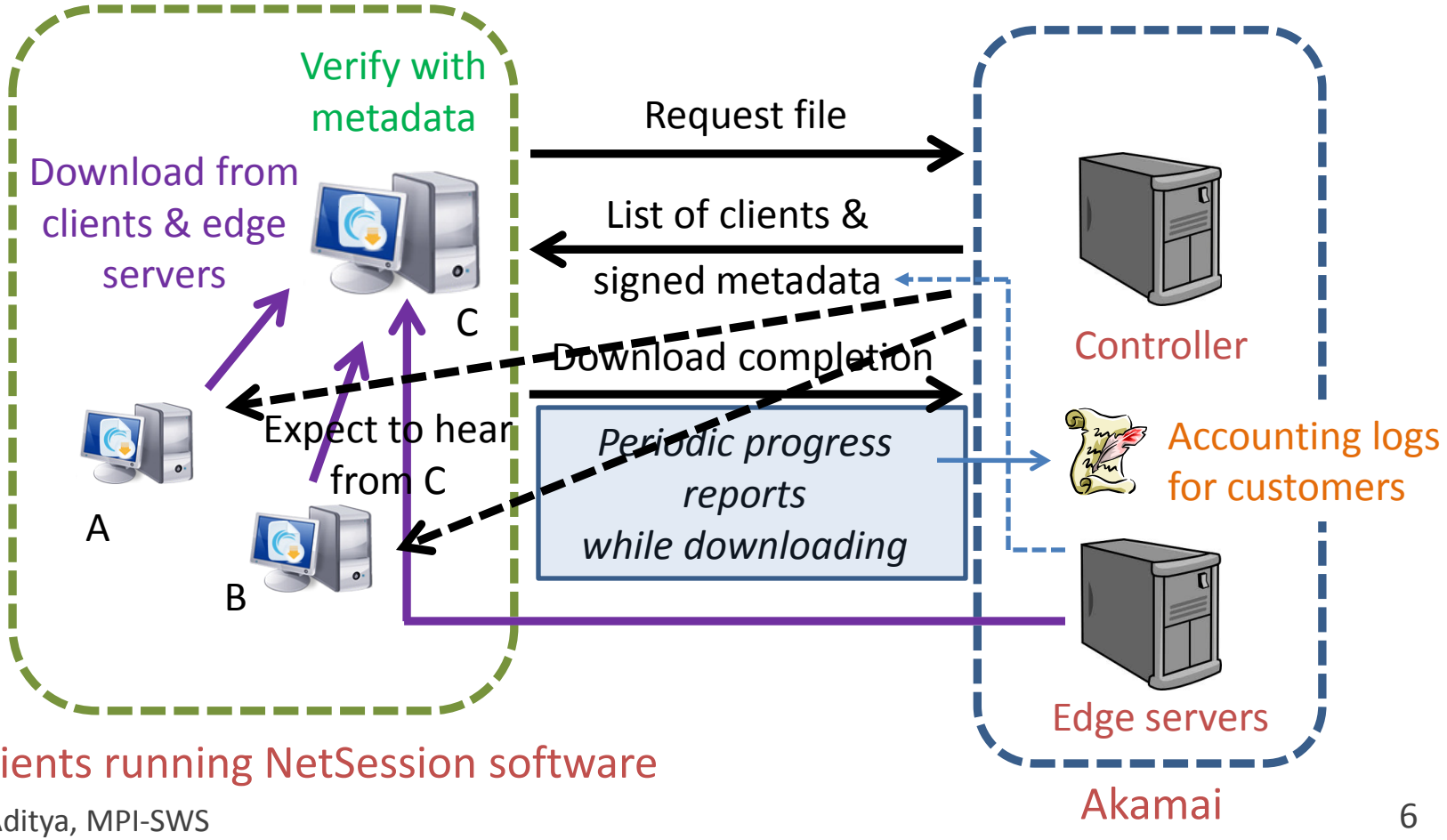
What Could Still Go Wrong?

- Inherent problem: infrastructure can't observe P2P communication
- Clients could still misreport
  - CDN may end up reporting downloads that did not happen
- Clients c ice quality



Akamai NetSession

- Peer assisted CDN operated by Akamai
  - Used for distributing large files – software installers and videos
  - Client software is bundled with customer specific installer

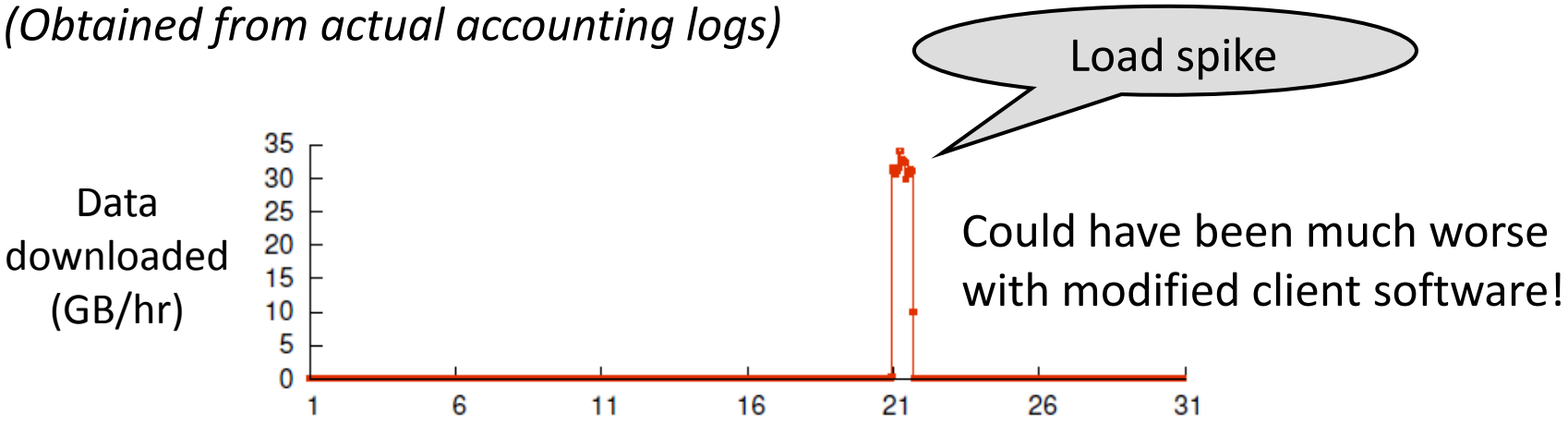


Inflation Attack on NetSession

- Have an unmodified NetSession client report fake downloads
- Performed
  - Targeted

Single client can cause significant accounting inaccuracies!

(Obtained from actual accounting logs)



Outline

- Introduction
  - Hybrid CDNs: clients can misreport
  - Need a way to reliably account for client activities
- Reliable Client Accounting (RCA)
  - Reliably capture client activities
  - Identify misbehaving/suspicious clients
  - Handle misbehavior without affecting service quality
- Evaluation
- Related work & Conclusion

Types of Attacks

- Misbehaving client software

  - Unilateral – deviations from the correct protocol
    - Misreport interactions with honest clients
    - Serve bad content to disrupt quality of service

RCA can detect deterministically

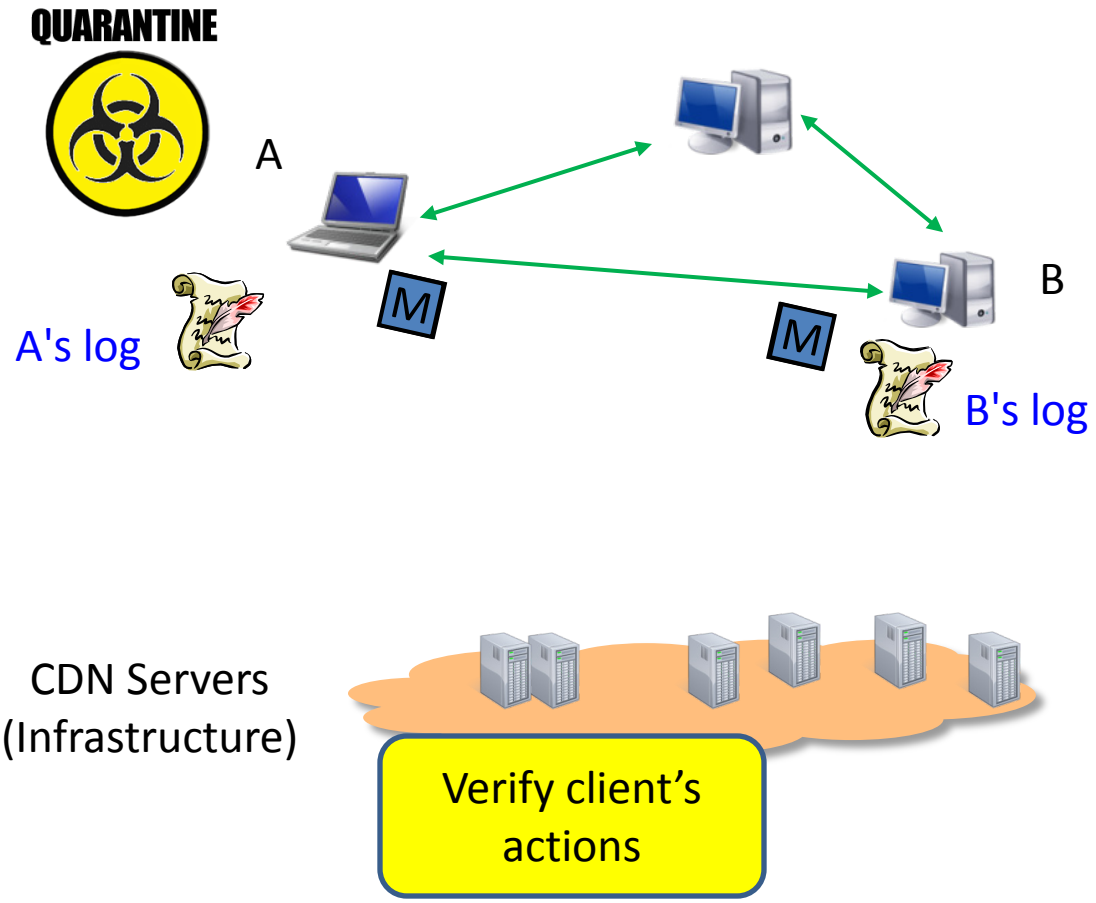
  - Collusion – multiple clients collude to misreport activities
    - Difficult in practice because infrastructure assigns peers

- Suspicious user behavior

  - Repeatedly downloading content to drive up demand
    - Can be amplified by a Sybil attack
  - Not unique to hybrid systems

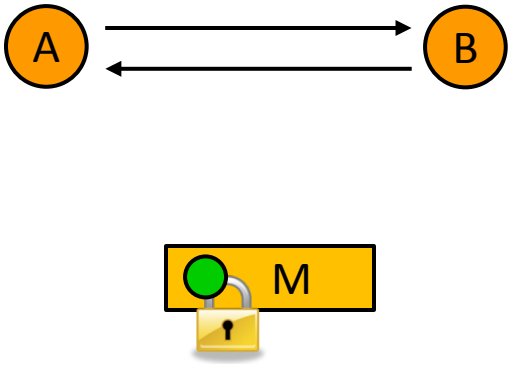
Require statistical checks

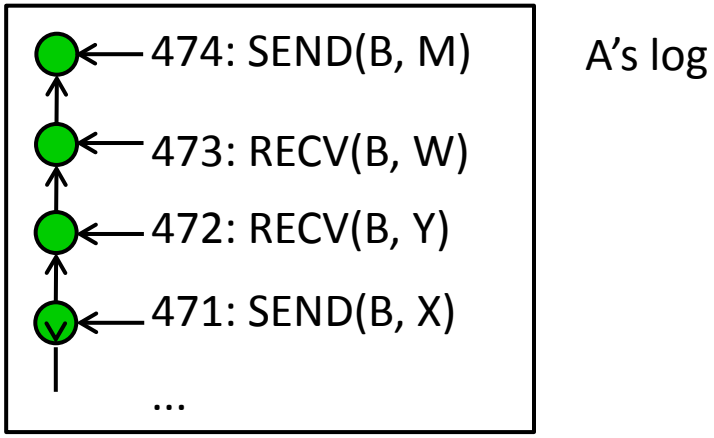
Reliable Client Accounting – Overview



- Clients maintain a tamper evident log of their network activity
- Logs periodically uploaded to infrastructure and verified
- Quarantine clients if suspicious

Reliably Capturing Client Activity





- Tamper evident logging & log consistency checks [PeerReview, SOSP 2007]
- Log entries form a hash chain
- Signed hash (authenticator) is included with every message sent
  - Client commits to its entire event history
- Log hash chains + authenticators are sufficient to
  - Verify whether all clients report a consistent sequence of message exchange
- Clients cannot unilaterally report fake downloads

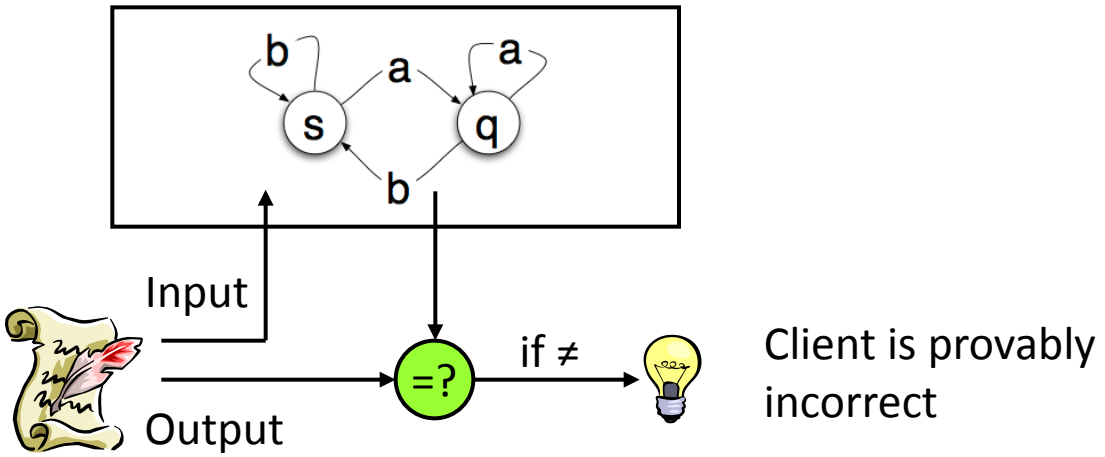
Reducing Processing Overhead on the CDN

- Signature verification overhead α number of authenticators
- Previous implementations
  - Records one authenticator for each message
  - Overhead: O(number of messages sent or received)
- RCA: cumulative authenticators
  - Records only two authenticators for each remote client
  - Overhead:

    O(number of communicating client pairs) << O(number of messages)

Verifying Client Activity

- A consistent log might still be implausible
  - Contact clients not assigned by infrastructure
  - Serve bad content
- Plausibility checking
  - Verify whether the log is consistent with a valid execution of software
- NetSession protocol can be modeled as a simple state machine
  - Manually identified rules a correct client must obey
  - Verify logs against these rules



Types of Attacks

- Misbehaving client software

  - Unilateral – deviations from the correct protocol
    - Misreport interactions with honest clients
    - Serve bad content to disrupt quality of service

RCA can detect deterministically

  - Collusion – multiple clients collude to misreport activities
    - Difficult in practice because infrastructure assigns peers

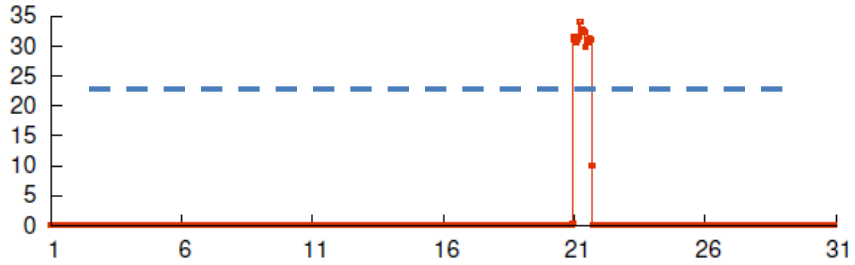
- Suspicious user behavior

  - Repeatedly downloading content to drive up demand
    - Can be amplified by a Sybil attack
  - Not unique to hybrid systems

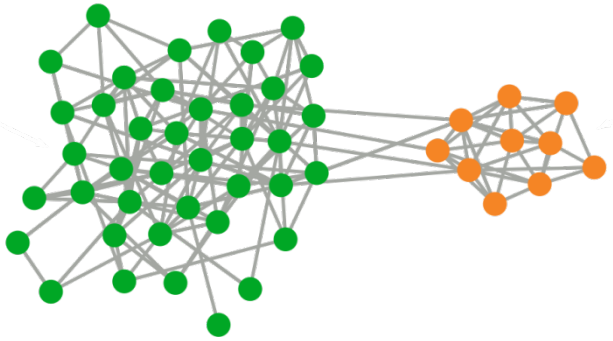
Require statistical checks

Statistical Checks

- Look for anomalous client behavior
- Large amount of prior work
  - Assume the availability of correct information
- RCA provides a sound basis for anomaly detection



Flag clients who download more than a threshold



Analyze communication patterns to identify colluding clients

Handling Malicious/Suspicious Clients

- Blacklist clients
  - False positives – blacklist an innocent client?

- Quarantine clients
  - Not allowed to upload content
  - Can still download from the infrastructure

- Quarantining an innocent client is safe
  - Does not affect service quality of client
  - Slight increase in resource cost to infrastructure

- Enables aggressive anomaly detectors
  - Tamper evident logging: provides accurate information
  - Quarantining: safe way to handle false positives

Outline

- Introduction
  - Hybrid CDNs: clients can misreport
  - Need a way to reliably account for client activities
- Reliable Client Accounting (RCA)
  - Reliably capture client activities
  - Identify misbehaving/suspicious clients
  - Handling misbehavior without affect service quality
- Evaluation
- Related work & Conclusion

Evaluation

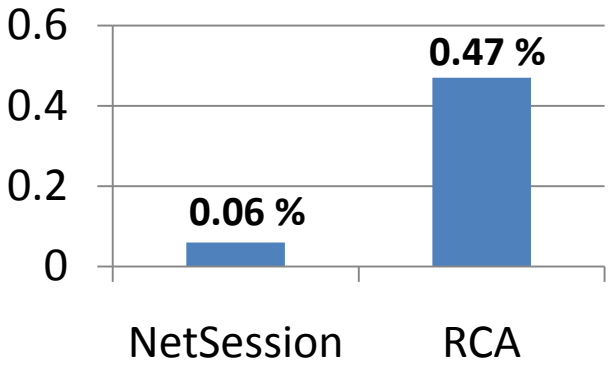
- Implemented a clone of NetSession client & Infrastructure software
- Experiments performed in a network emulation environment
  - Driven by actual client activity traces of Akamai NetSession for Dec 2010
- Experiment:
  - Reproduce clients' download activity over a month
  - 500 randomly selected clients
  - 1 edge server and 1 control plane server

Evaluation - Questions

- Client's Perspective
  - Network overhead
  - CPU overhead
  - Log storage
- CDN's Perspective
  - Log processing overhead
  - Statistical checks
  - Effectiveness

Client's Perspective

- Network overhead (in terms of % of actual content downloaded)



Avg extra client B/W: 192 KB/day (signatures + log upload)

- CPU overhead
  - Maximum additional client CPU usage = 0.5%
- Log Storage (with daily log uploads)
  - On average: 100 KB/day

CDN's Perspective

Projections for a large deployment:

- 100 million clients, downloading 100 PB content/month
- Log Uploads & Log Processing
  - 0.05 PB/month of logs uploads (0.05% of transferred content)
  - 35 machines required to process these logs
- For comparison, NetSession as of Dec 2011
  - has 25 million clients, downloading 0.85 PB/month
  - uses about 10 machines for log processing
- Effectiveness
  - Tried out various attacks. RCA caught them as expected

Related Work

- Misbehavior in P2P systems
  - Maze [Q. Lian et al., 2007]

    Empirically study client misbehavior in p2p file sharing systems
  - Dandelion [M. Sirivianos et al., 2007], Antfarm [Peterson et al., 2009]

    Use cryptographic virtual currency to handle selfish peers
  - ➢ RCA doesn't aim for fairness and considers more general Byzantine behavior
- Anomaly detection
  - An intrusion detection model [D. Denning, 1987]
  - BotGrep [S. Nagaraja et al., 2010]

    Detect BotNets by studying client interactions
  - ➢ RCA enables building complex statistical checks

Conclusion

- Fundamental challenge for P2P-Infrastructure hybrids
  - Infrastructure cannot observe P2P communication
- Demonstrated an inflation attack on the live Akamai NetSession system
- Reliable Client Accounting (RCA)
  - Reliably capture client activity
  - Sound basis for anomaly detection
  - Quarantine: safely handle suspicious clients
- Applied RCA to Akamai NetSession
  - Comprehensive evaluation using actual client traces
  - RCA overhead is reasonable