QUANTITATIVE TRUST MANAGEMENT:
QuanTM, Reputation, and PreSTA

Andrew G. West

PRECISE Meeting – 11/18/2009

Report Documentation Page

Form Approved
OMB No. 0704-0188

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE **18 NOV 2009**	2. REPORT TYPE	3. DATES COVERED **00-00-2009 to 00-00-2009**

4. TITLE AND SUBTITLE **Quantitative Trust Management: QuanTM, Reputation, and PreSTA**	5a. CONTRACT NUMBER 5b. GRANT NUMBER 5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER 5e. TASK NUMBER 5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **University of Pennsylvania,School of Engineering and Applied Science,220 South 33rd Street ,Philadelphia,PA,19104-6391**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S) 11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
PRECISE Research Group Presentation, Nov. 2009

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**48**	

TALK OUTLINE

- Introducing QTM
- QuanTM [1] Model
  - TDG: Glue between security & reputation
- Fundamentals of Reputation Management
- PreSTA [3] Reputation Model
  - Partial QTM use-case
  - Applicable for fighting spam, Wikipedia…
- Conclusions

QTM DEFINED

- 'Trust Management' (TM) aspect
  - STATIC delegation of access rights between principals using policy/credentials/conditions
  - Implemented by a Policy-based TM (PTM) language (*i.e.*, KeyNote) and evaluator ('compliance checker')
- 'Quantitative' (Q) aspect
  - DYNAMIC weighting of above delegations, based on reputations of those involved
  - Implemented by a Reputation Management (RTM) algorithm (PreSTA [3], TNA-SL [4], EigenTrust [5])

QTM DEFINED

Policy-Based Trust Mgmt. (PTM)

- Effective for delegated credentials and access enforcement
- Can't handle uncertainty and partial information
- Foundation: Cryptography

Rep-Based Trust Mgmt. (RTM)

- Quantifies trust relationships
- No delegation (non-transferable)
- No enforcement
- Foundation: Aggregation of past behavior via feedback.

QUANTITATIVE TRUST MANAGEMENT (QTM)

- Combine PTM and RTM
- Dynamic interpretation of authorization policies for access control decisions based on upon evolving reputations of the entities involved, and environmental context at evaluation-time [6].

QTM for CPS

- MAIN GOAL
  - Integrating cyber and physical trusts
- ISSUES FORESEEN
  - Authentication/provenance of physical stimuli
  - Environmental uncertainty
- POTENTIAL USE-CASES
  - Voting machines
  - Emergency management

QuanTM Model

Combining TM and RM [1]

BUILDING A TDG

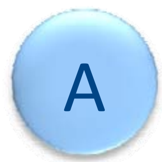
Authorizer: Alice
Licensees: (Bob && Charles)
Conditions:
  operation ==
    “read” -> ALLOW
    “execute” -> MAYBE
    “write” -> DENY
Signature: “rsa-sig:3850...”

Trust Dependency Graph (TDG): Data structure gluing Policy and Reputation based TM.

Above: An example KeyNote credential

BUILDING A TDG

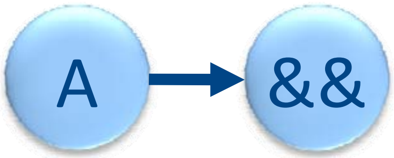
Authorizer: Alice
Licensees: (Bob && Charles)
Conditions:
  operation ==
    “read” -> ALLOW
    “execute” -> MAYBE
    “write” -> DENY
Signature: “rsa-sig:3850...”



Authorizer: The person who is “saying” a particular delegation

BUILDING A TDG

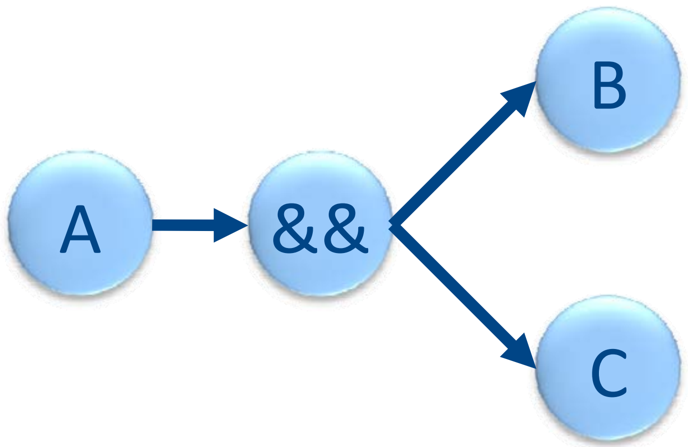
Authorizer: Alice
Licensees: (Bob && Charles)
Conditions:
  operation ==
    “read” -> ALLOW
    “execute” -> MAYBE
    “write” -> DENY
Signature: “rsa-sig:3850...”



Binary Operator: Nature of the delegation. Here, “AND” implies both parties must be present. KeyNote also supports “OR”

BUILDING A TDG

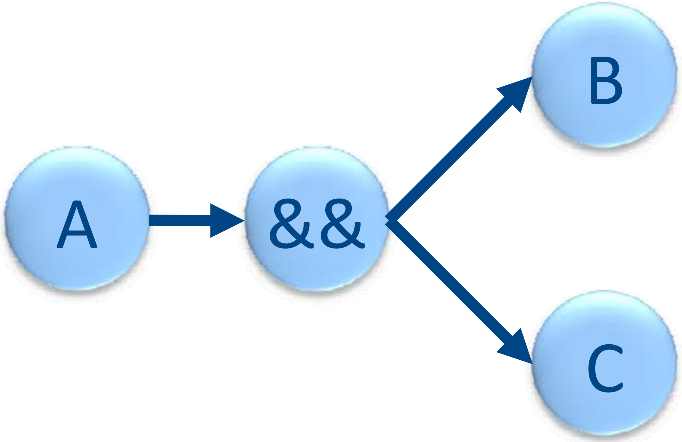
Authorizer: Alice
Licensees: (Bob && Charles)
Conditions:
  operation ==
    “read” -> ALLOW
    “execute” -> MAYBE
    “write” -> DENY
Signature: “rsa-sig:3850...”



Licensees: Those parties the 'Authorizer' is delegating trust to, as constrained by the binary operator

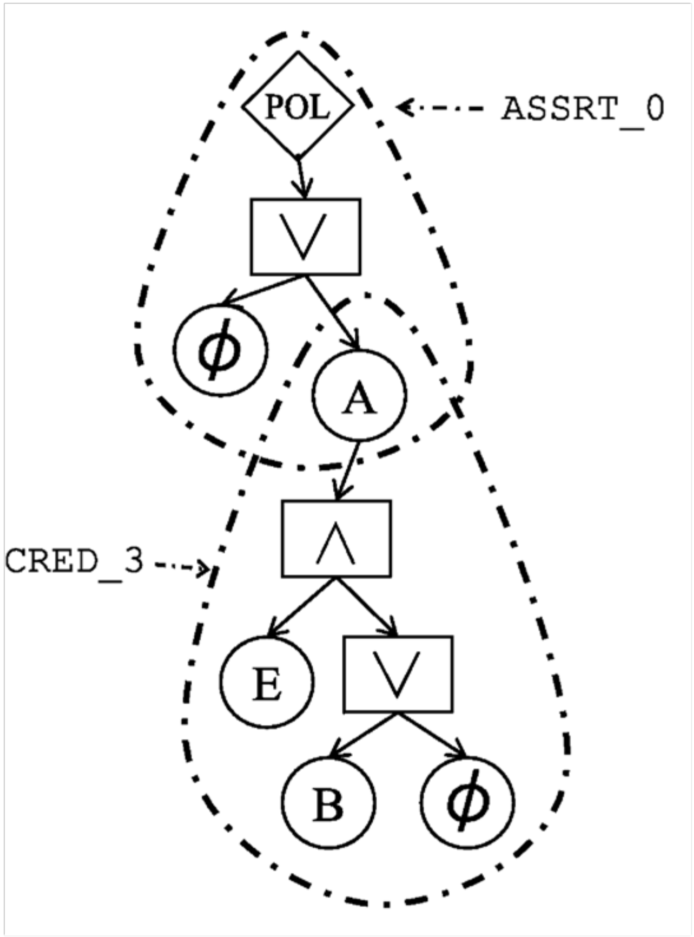
BUILDING A TDG

Authorizer: Alice
Licensees: (Bob && Charles)
Conditions:
  operation ==
    “read” -> ALLOW
    “execute” -> MAYBE
    “write” -> DENY
Signature: “rsa-sig:3850...”

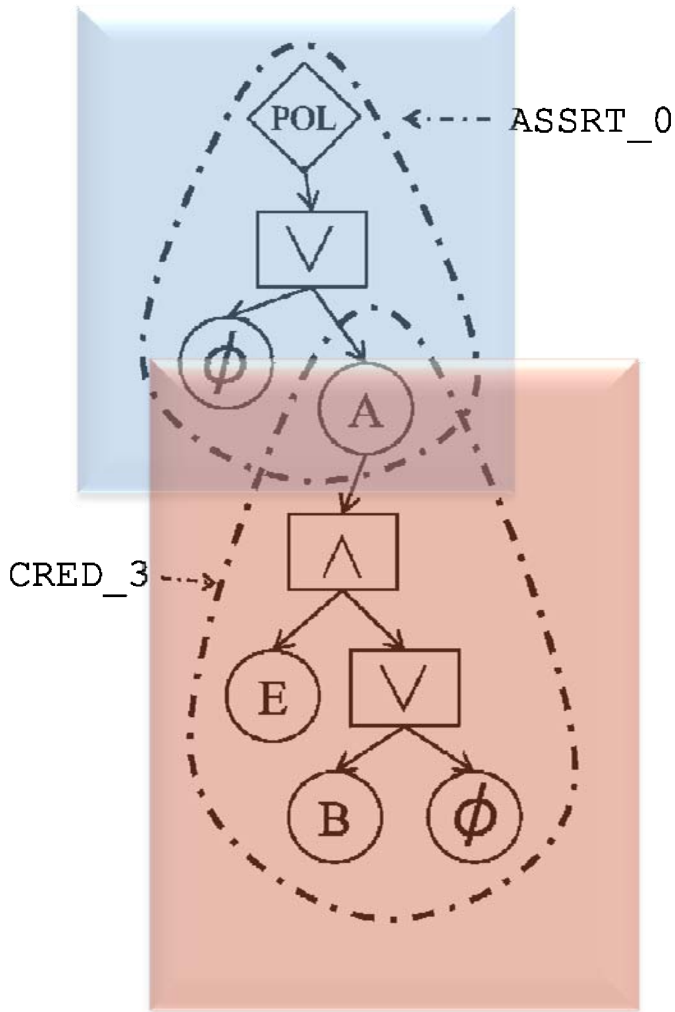


Compliance values: Output of the evaluator. Varies based on evaluation of conditions. Could be a binary YES/NO.

ACTUAL TDG



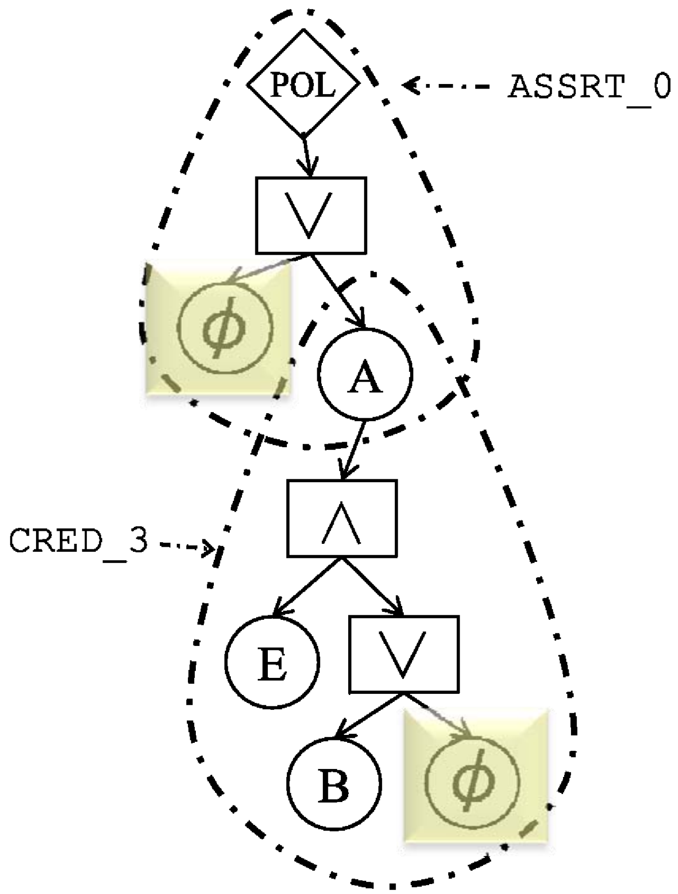
ACTUAL TDG



CREDENTIAL GROUPS:

We divide can divide portions of the graph based on the credentials from which they were derived

ACTUAL TDG



NULL NODES:

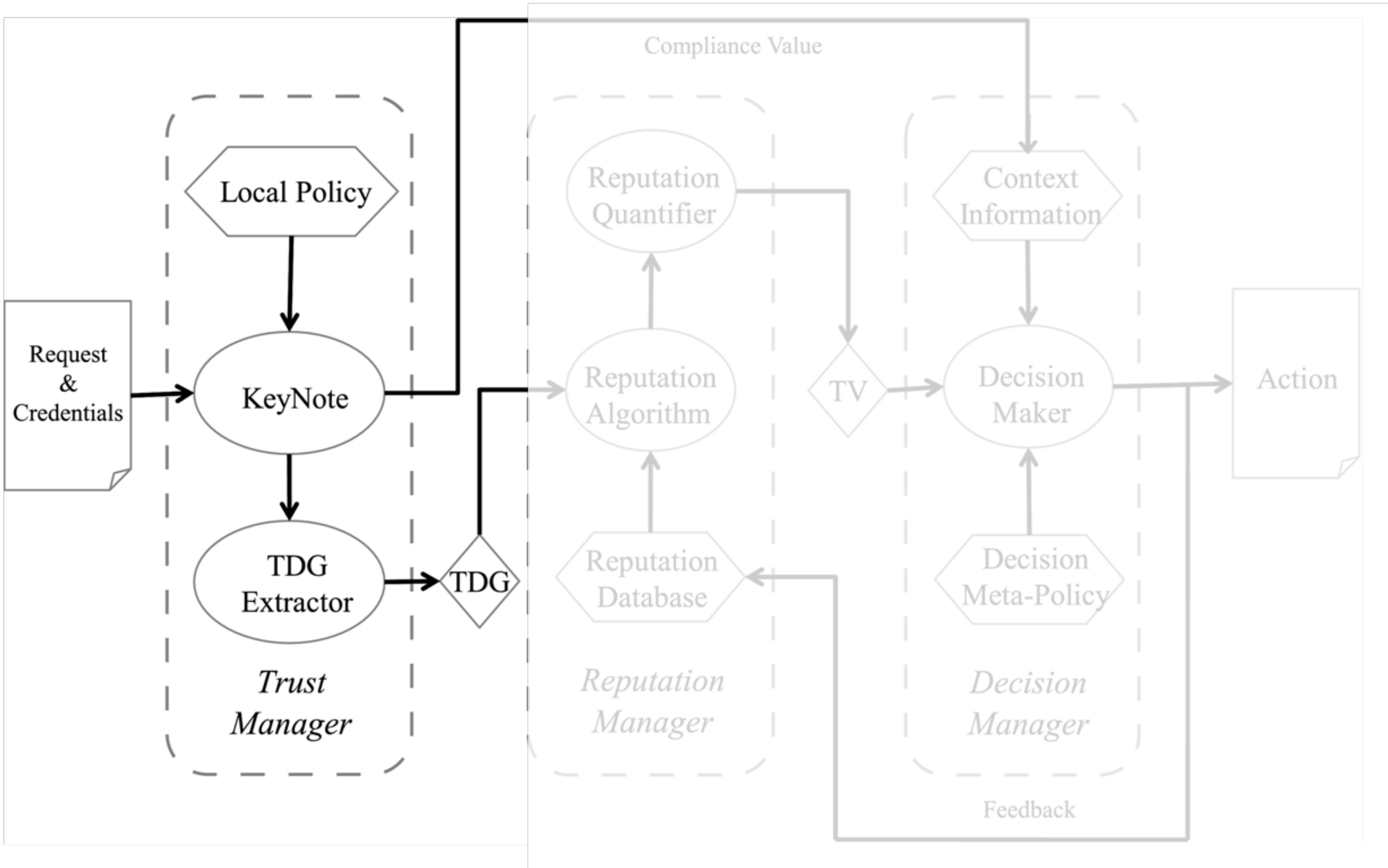
(1) Used to make graph explicitly binary

(2) Overwrite principals mentioned in credentials, but not 'present' in a particular request

TDG QUESTIONS

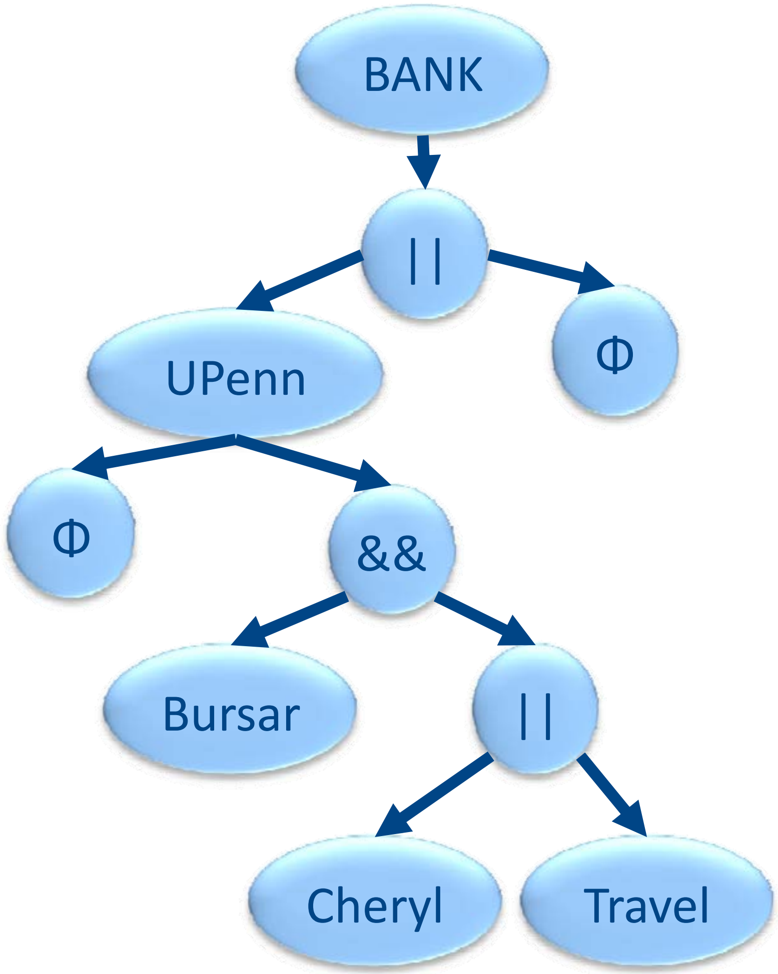
- TDG: Excellent representation of trust dependencies in a KEYNOTE request
  - Other TM languages?
- We would like to have a TDG structure which can encapsulate the features of all/general trust management langs.

QuanTM Arch.



Compliance Value
Local Policy
Request & Credentials
KeyNote
TDG Extractor
TDG
Trust Manager
Reputation Quantifier
Reputation Algorithm
Reputation Database
Reputation Manager
TV
Context Information
Decision Maker
Decision Meta-Policy
Decision Manager
Action
Feedback

USING THE TDG

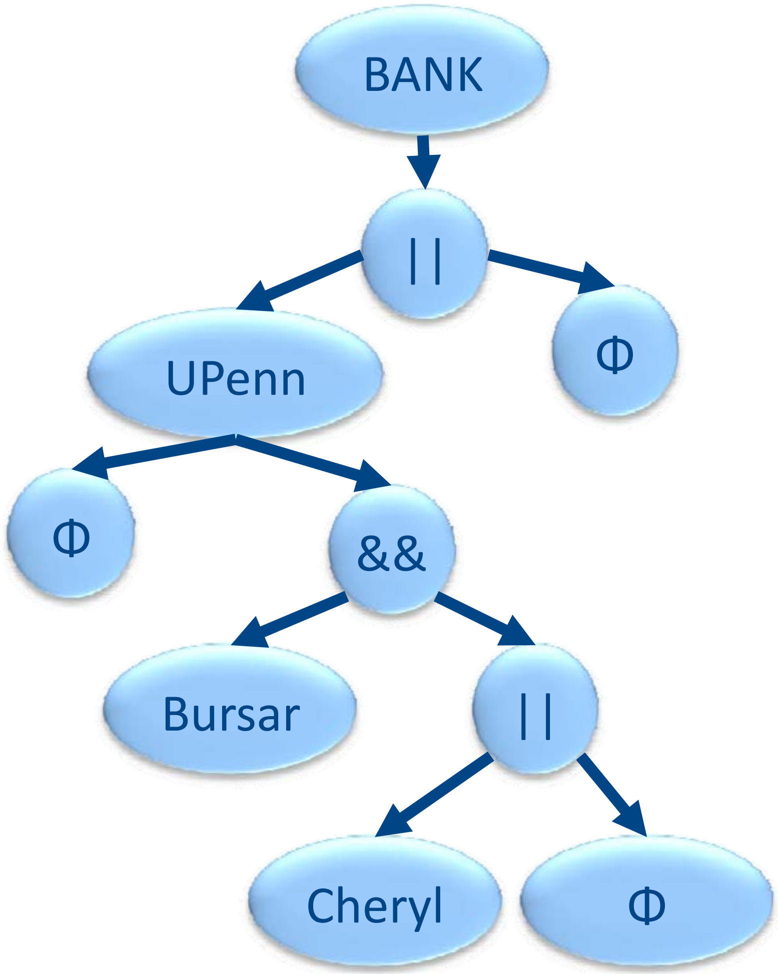


BIG IDEA:

Each graph arc can be weighted with a value speaking to the reputation of connecting parties.

These can be collapsed to produce a single TRUST VALUE for an entire request

USING THE TDG

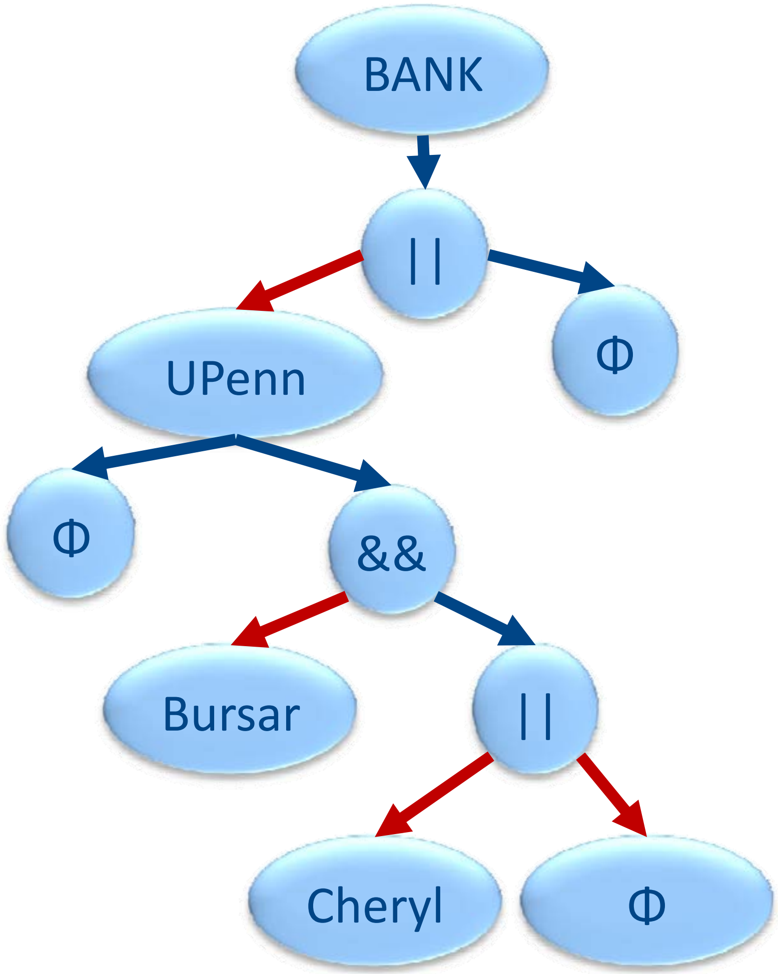


BIG IDEA:

Each graph arc can be weighted with a value speaking to the reputation of connecting parties.

These can be collapsed to produce a single TRUST VALUE for an entire request

USING THE TDG

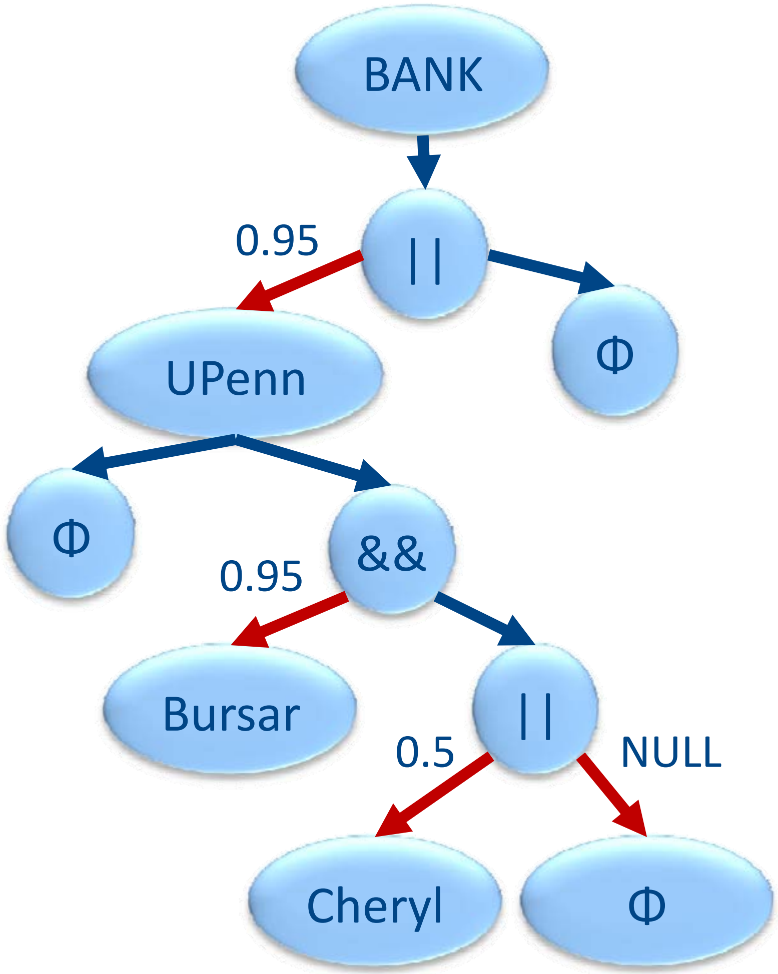


Reputation R_1:

Arcs from operators to principals

Weight with service providers (BANK) reputation valuation of sink principal

USING THE TDG



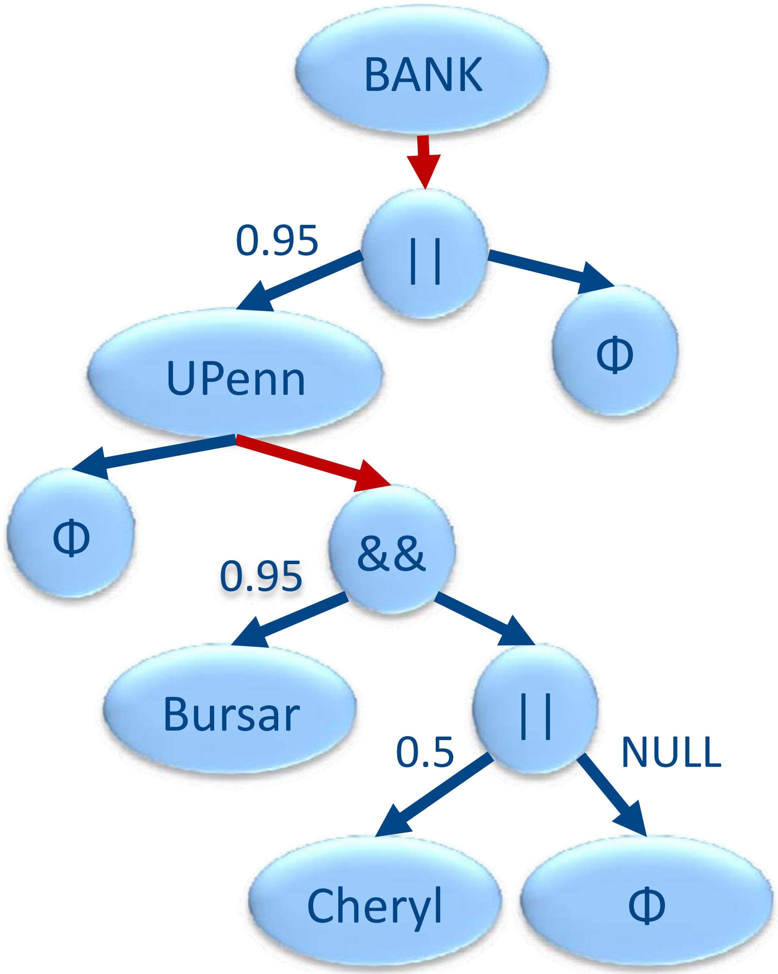
Reputation R_1:

Arcs from operators to principals

Weight with service providers (BANK) reputation valuation of sink principal

* Magic numbers

USING THE TDG

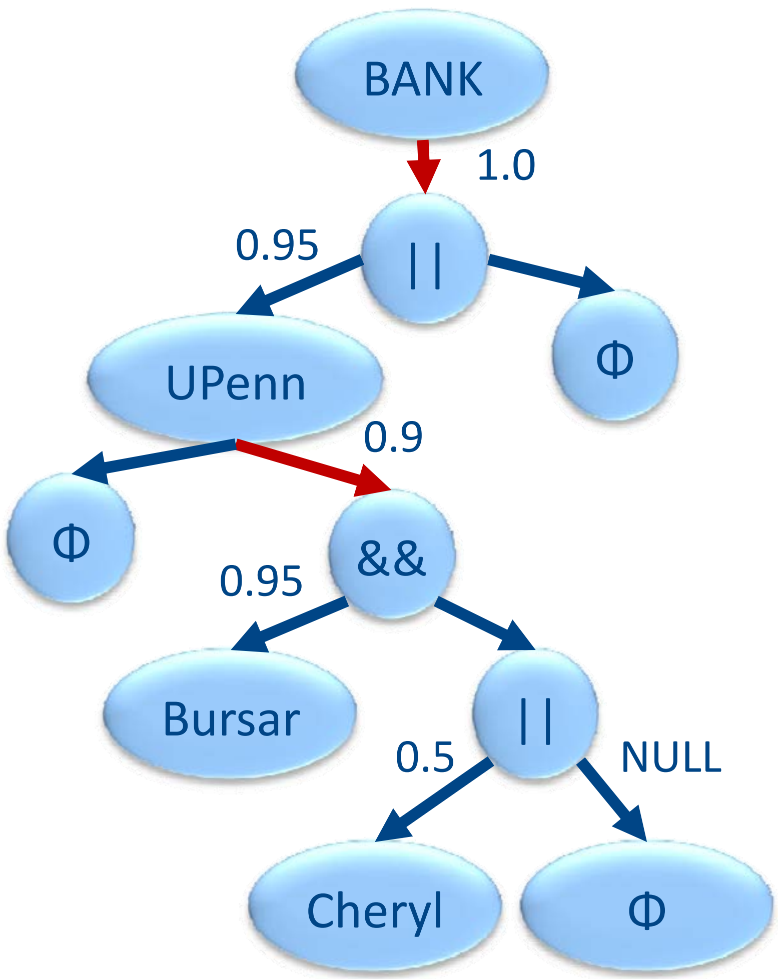


Reputation R_2:

Arcs from principals to operators

Weight with service provider's (BANK) trust in 'the ability of the source principal to delegate'

USING THE TDG



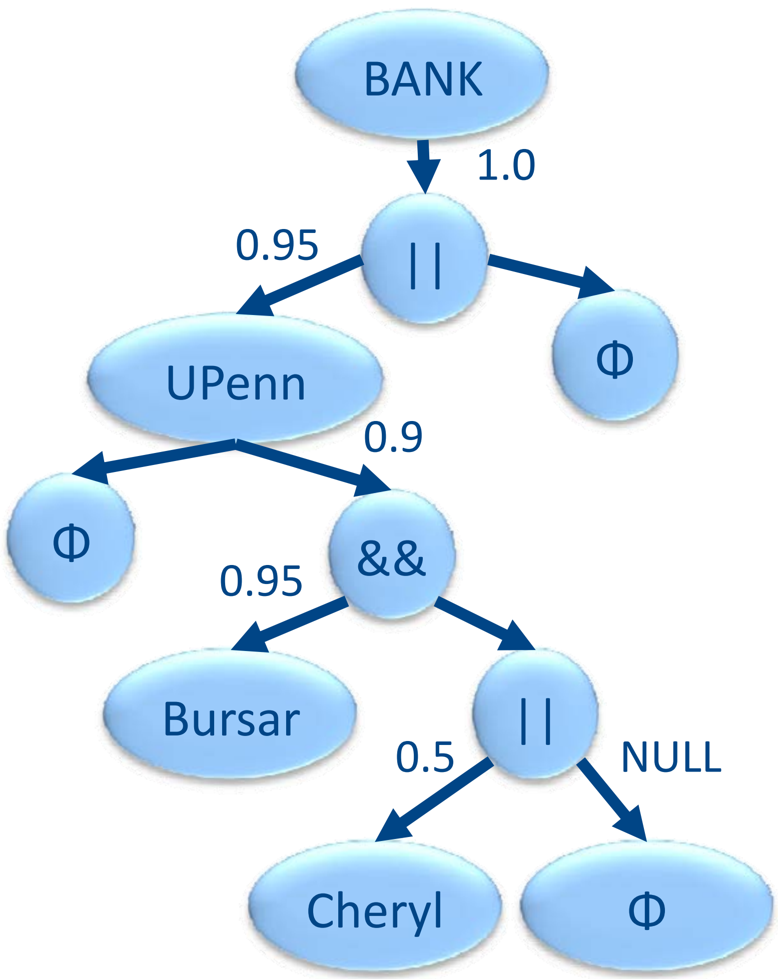
Reputation R_2:

Arcs from principals to operators

Weight with service provider's (BANK) trust in 'the ability of the source principal to delegate'

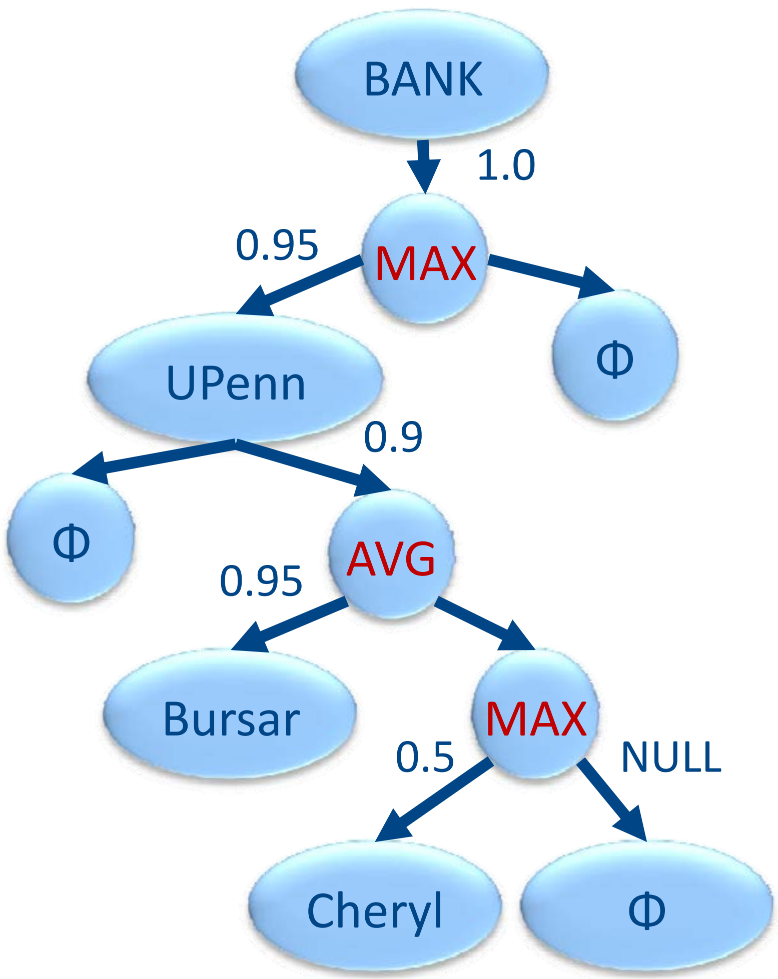
* Mention R_3

USING THE TDG



Graph Collapse:

USING THE TDG

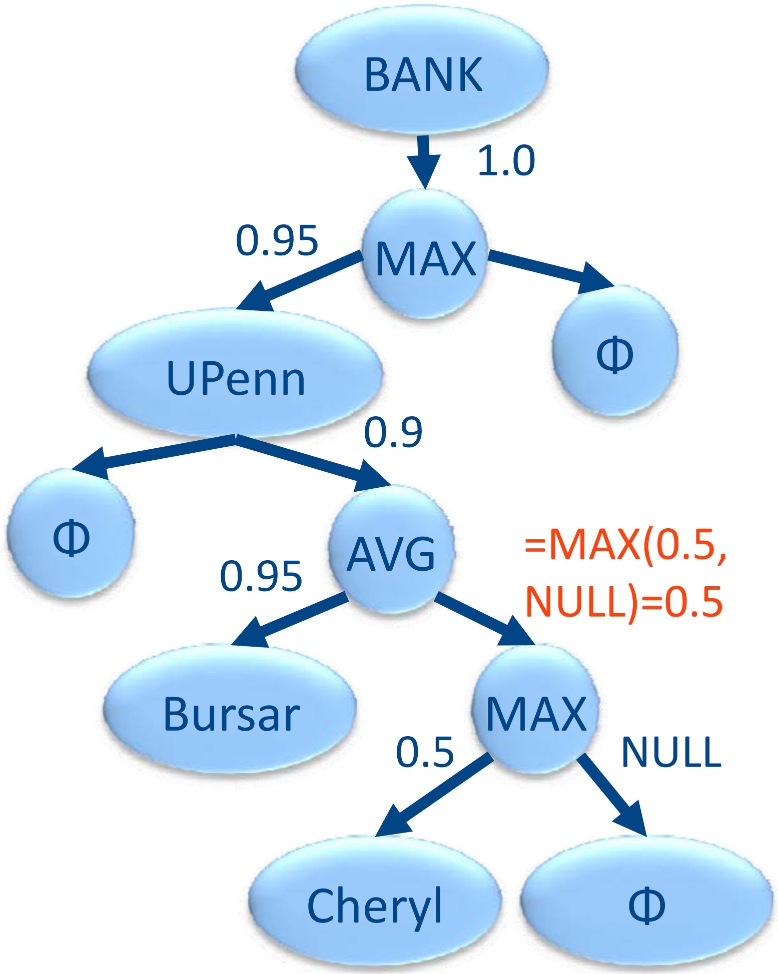


Graph Collapse:

* Swap out binary operators for numeric binary functions

* Start at TDG-bottom, perform functions, pass resulting values up the graph. Transitivity handled by multiply.

USING THE TDG

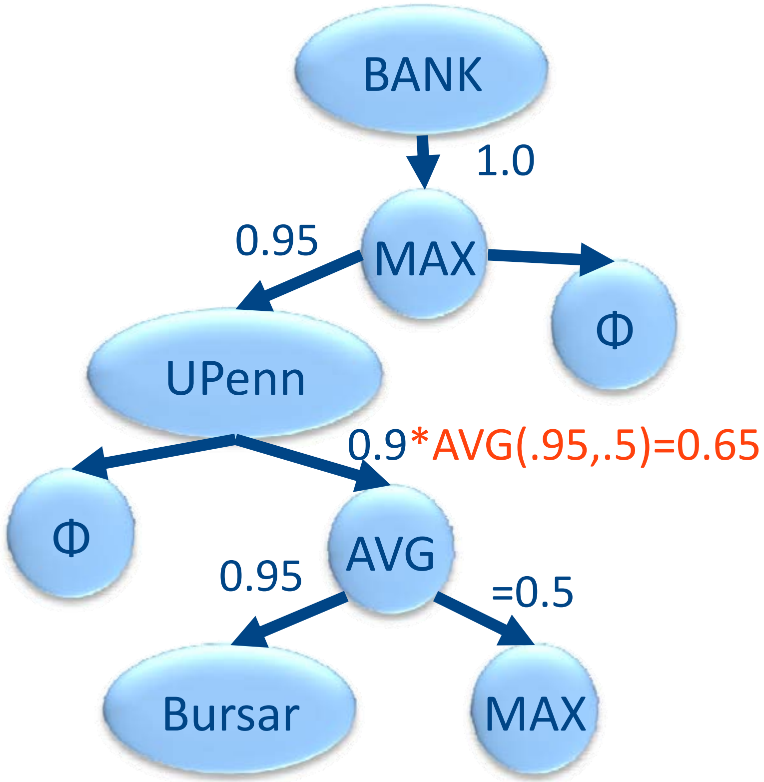


Graph Collapse:

* Swap out binary operators for numeric binary functions

* Start at TDG-bottom, perform functions, pass resulting values up the graph. Transitivity handled by multiply.

USING THE TDG

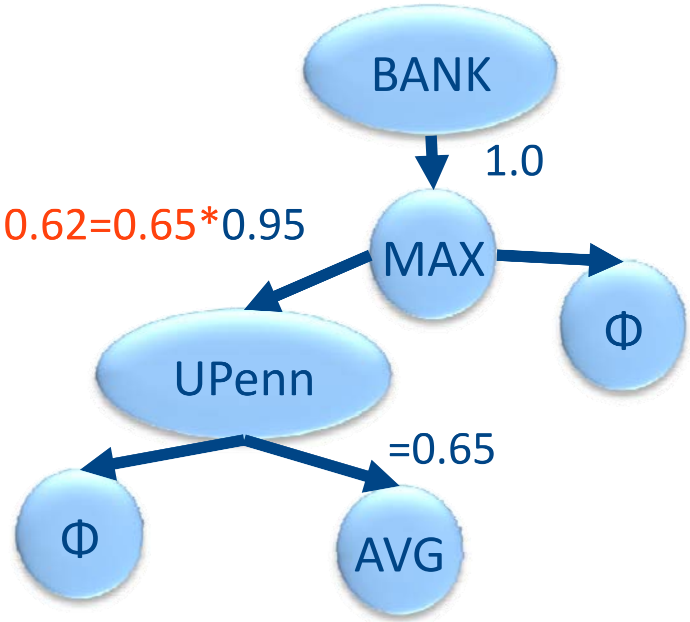


Graph Collapse:

* Swap out binary operators for numeric binary functions

* Start at TDG-bottom, perform functions, pass resulting values up the graph. Transitivity handled by multiply.

USING THE TDG

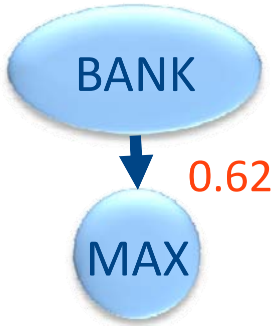


Graph Collapse:

* Swap out binary operators for numeric binary functions

* Start at TDG-bottom, perform functions, pass resulting values up the graph. Transitivity handled by multiply.

USING THE TDG

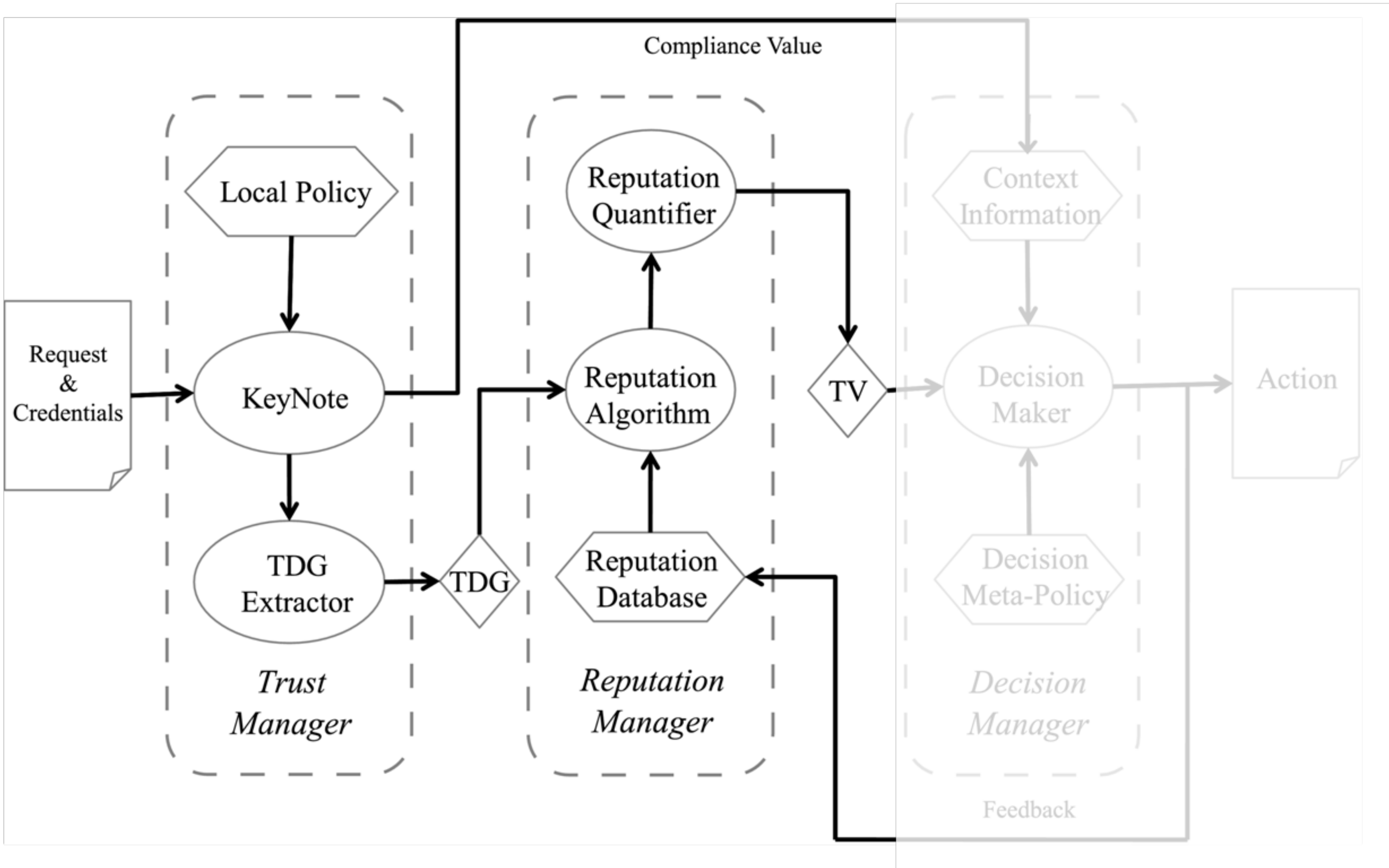


Graph Collapse:

* Swap out binary operators for numeric binary functions

* Start at TDG-bottom, perform functions, pass resulting values up the graph. Transitivity handled by multiply.

QuanTM Arch.

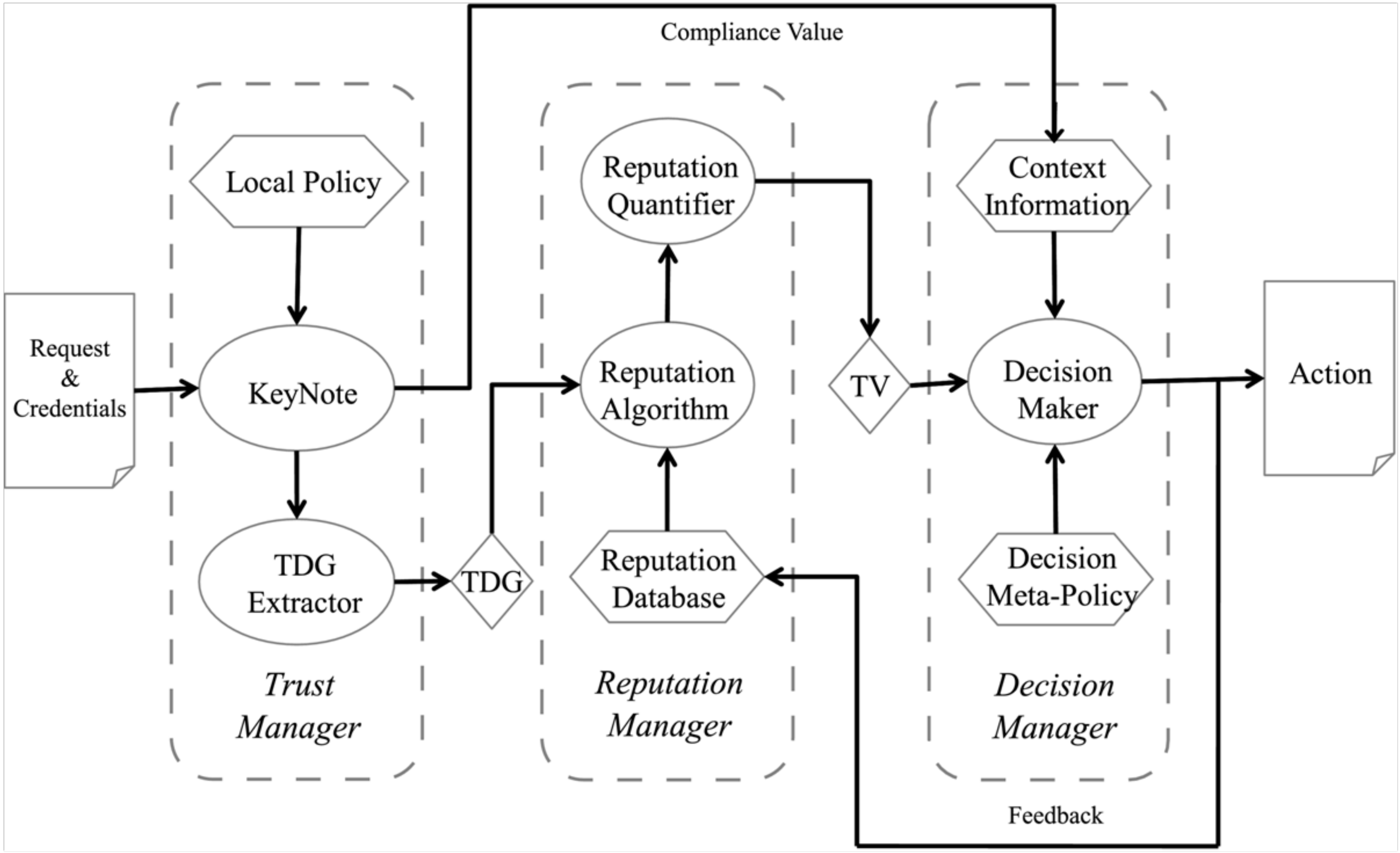


Compliance Value
Request & Credentials
Local Policy
KeyNote
TDG Extractor
TDG
Trust Manager
Reputation Quantifier
Reputation Algorithm
Reputation Database
Reputation Manager
TV
Context Information
Decision Maker
Decision Meta-Policy
Decision Manager
Action
Feedback

DECISION PROCESS

- There was a an action request made…
  - The TM language evaluator outputs some compliance value, *e.g.*, "MAYBE"
  - We generated a TDG, and collapsed it using magic numbers, *e.g.*, "0.62"
- … Combining these two things, and sufficient hand-waving -> binary access decision
  - Cost-benefit analyses

QuanTM Arch.



Compliance Value
Request & Credentials
Local Policy
KeyNote
TDG Extractor
TDG
Trust Manager
Reputation Quantifier
Reputation Algorithm
Reputation Database
Reputation Manager
TV
Context Information
Decision Maker
Decision Meta-Policy
Decision Manager
Action
Feedback

WHAT'S GAINED

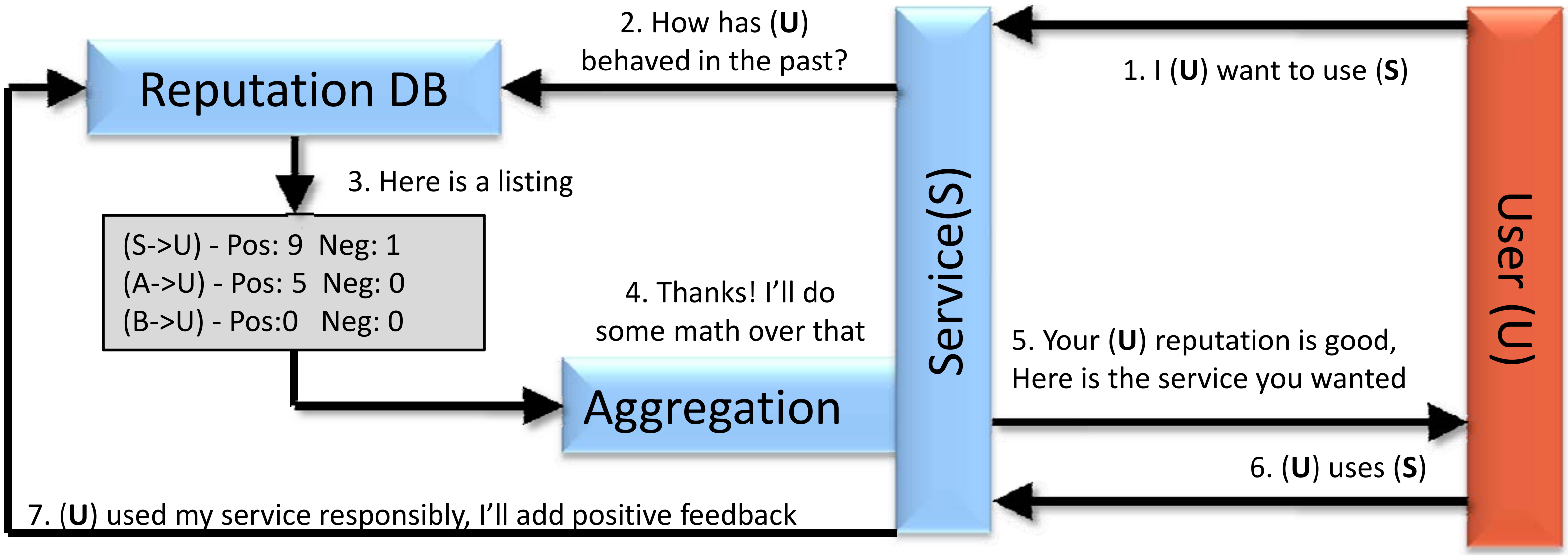
- TM: Revocation difficult - One shouldn't delegate unless they completely trust.
  - QTM: Dynamic revocation using reputation
  - QTM: Safe to delegate in partial trust situations
- TM: Rights can be delegated to principals that service provider knows nothing about
  - QTM: Can check these new principals at the reputation stage
- RM: Lacks enforcement/delegation

Rep. Management

Aggregating Behavioral Feedback

(and testing these strategies [2])

REP MANAGEMENT



- DYNAMIC valuation using (in)direct interaction history between parties
  - Loose interpretation: probability that *A* trusts *B*
  - Informal; produces values in [0,1]
  - Many different logics/systems to aggregate feedback
    - EigenTrust (Garcia-molina) and Subjective-Logic (Jøsang)

EIGENTRUST [5]

- Normalized vector-matrix multiply aggregation towards globally convergent view.
  - Feedbacks viewed in matrix, normalized
  - Pre-trust vector
  - Converge to relative values (t_∞)
  - Elegant and scalable, but normalized, no negative trust

$$A = \begin{bmatrix} \begin{pmatrix} pos:0 \\ neg:0 \end{pmatrix} = 0 & \begin{pmatrix} pos:3 \\ neg:1 \end{pmatrix} = 2 & \begin{pmatrix} pos:3 \\ neg:2 \end{pmatrix} = 1 \\ \begin{pmatrix} pos:9 \\ neg:3 \end{pmatrix} = 6 & \begin{pmatrix} pos:0 \\ neg:0 \end{pmatrix} = 0 & \begin{pmatrix} pos:8 \\ neg:1 \end{pmatrix} = 7 \\ \begin{pmatrix} pos:2 \\ neg:4 \end{pmatrix} = 0 & \begin{pmatrix} pos:5 \\ neg:4 \end{pmatrix} = 1 & \begin{pmatrix} pos:0 \\ neg:0 \end{pmatrix} = 0 \end{bmatrix}$$

$$A' = \begin{bmatrix} {}^0/_6 & {}^2/_3 & {}^1/_8 \\ {}^6/_6 & {}^0/_3 & {}^7/_8 \\ {}^0/_6 & {}^1/_3 & {}^0/_8 \end{bmatrix} \quad p = \begin{bmatrix} {}^1/_3 \\ {}^1/_3 \\ {}^1/_3 \end{bmatrix} \quad t_\infty = \begin{bmatrix} 0.35 \\ 0.49 \\ 0.16 \end{bmatrix}$$

$$t_{k+1} = (0.5 * A'^T * t_k) + 0.5 * p$$

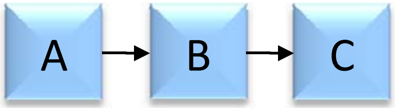
SUBJECTIVE LOGIC [4]

- Trust 4-tuples (belief, disbelief, uncertainty, ...)
- User-centric trust-graph decomposition
- Advantages: Absolute interpretation (beta-PDF), user-centric views, negative trust
- Disadvantages: Scalability, sparse scenarios

Opinion: (b, d, u, a)

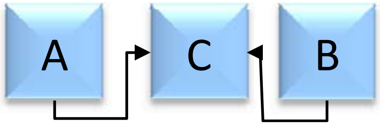
$$
\begin{aligned}
\text{belief} &= (pos/(pos + neg + 2.0)) \\
\text{disbelief} &= (neg/(pos + neg + 2.0)) \\
\text{uncertainty} &= (2.0/(pos + neg + 2.0)) \\
\text{base-rate} &= \begin{cases} 1.0 & \text{if user is pre-trusted} \\ 0.5 & \text{otherwise} \end{cases}
\end{aligned}
$$

Transitivity



$$\omega_C^{A:B} = \omega_B^A \otimes \omega_C^B$$

Average

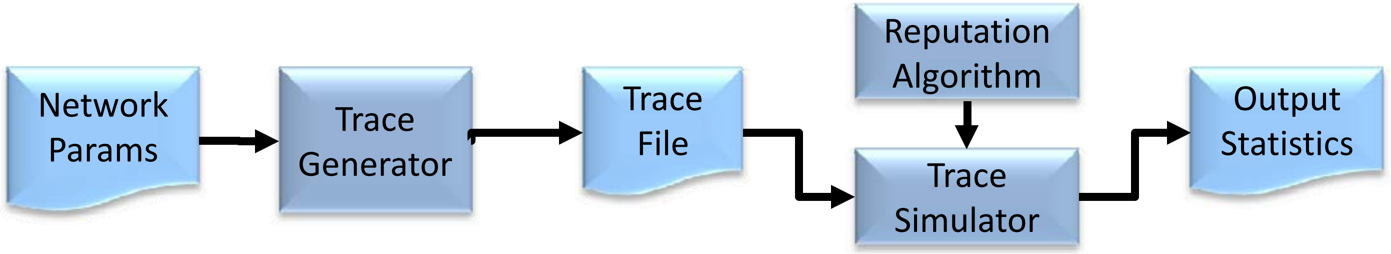


$$\omega_C^{A\diamond B} = \omega_C^A \oplus \omega_C^B$$

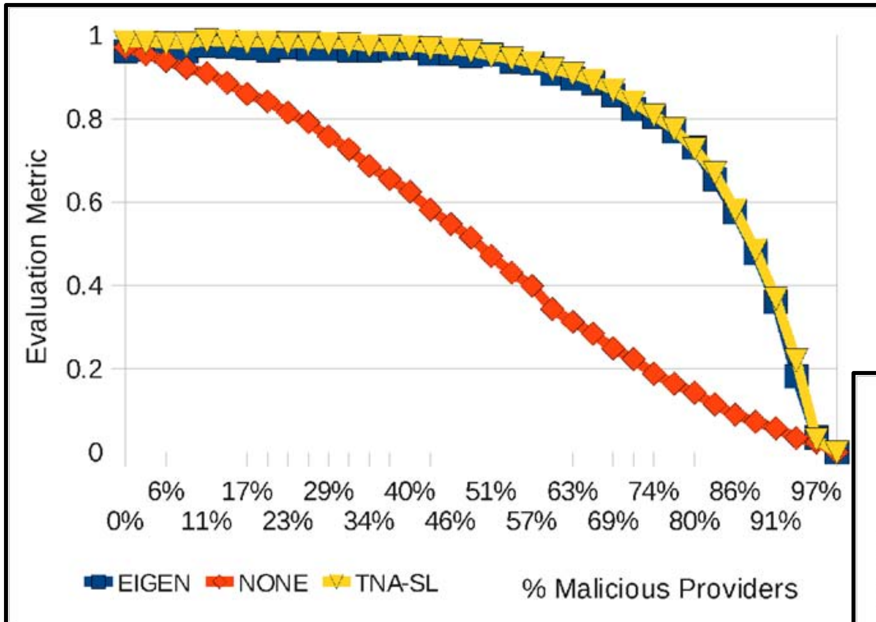
RM SIMULATOR

- How to test effectiveness of RM systems?
- Simulator [2]: File exchange (*i.e.*, P2P network)
  - Good files and corrupt files
  - Behaviors: Clean-up and honesty

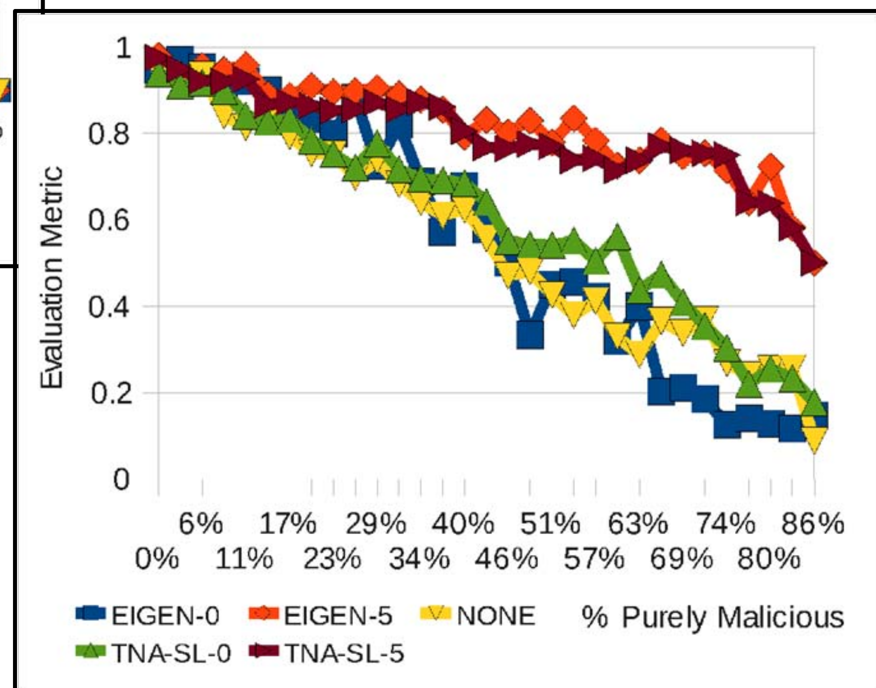
$$Metric = \frac{\#\ \text{valid files received by ‘good’ users}}{\#\ \text{transactions attempted by ‘good’ users.}}$$



SIM [2] RESULTS



(**LEFT**) Under naïve circumstances, all trust algorithms are very effective (a sanity check).



(**RIGHT**) Under complex dishonesty and sparseness, PRE-TRUST becomes very important.

PreSTA Model

(Preventative Spatio-Temporal Aggregation)

Preventing Malicious Behavior (Spam) [3]

PreSTA: BIG IDEA

PROBLEM

SOLUTION

- Traditional punishment mechanisms (*i.e.*, blacklists) are reactive
- PreSTA: Detect malicious users (*i.e.*, spammers) before harm is done

HYPO-THESES:

- Malicious users are spatially clustered (in any dimension)
- Malicious users are likely to repeat bad behaviors (temporal)

GIVEN:

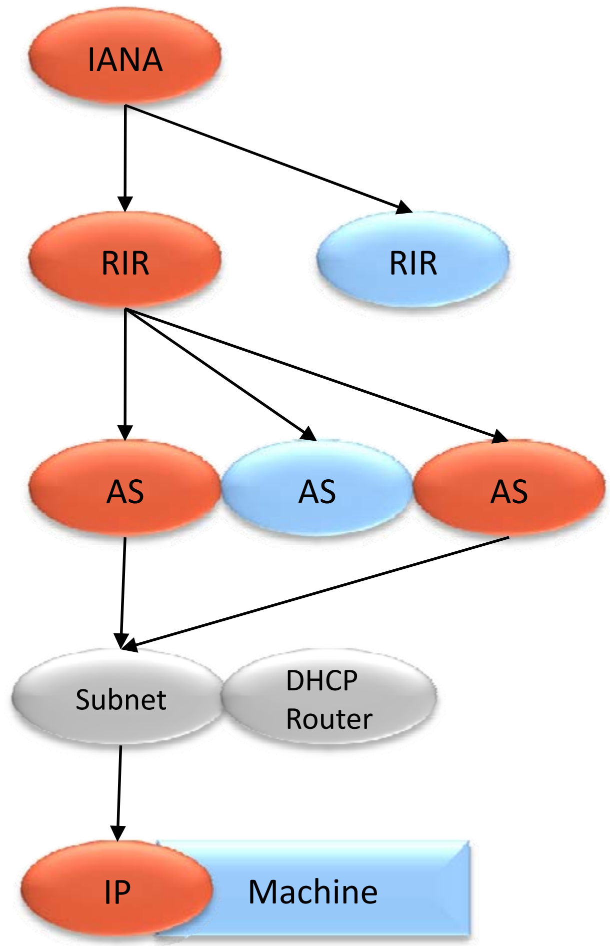
- A historical record of those principals known to be bad, and the timestamp of this observation (feedback)

PRODUCE:

- An extended list of principals who are thought to be bad now, based on their past history, and history of those around them

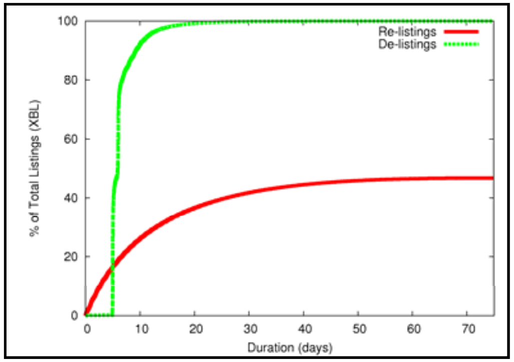
IP DELEGATION

- IP delegation hierarchy extremely similar to TDG
- Exploit this fact:
  - Calculate reputations at varying hierarchy levels
  - Feedback: IP blacklists
  - Combine granularities
- Can more malignants (spammers) be caught?



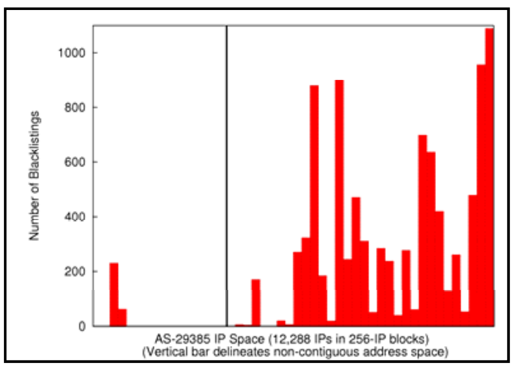
SPATIO-TEMPORAL

TEMPORAL: Bad Guys Repeat Bad Behaviors



- Maximize utilization: re-use
- Predictable blacklist duration
- 25% reappear within 10 days

SPATIAL: Bad Guys Live Together



- Corrupt ISPs: McColo, 3FN
- Geography -> IP space
- Intra-allocation spamming

PreSTA Algorithm

TO CALCULATE REPUTATION FOR ENTITY α:

Old Black List 1

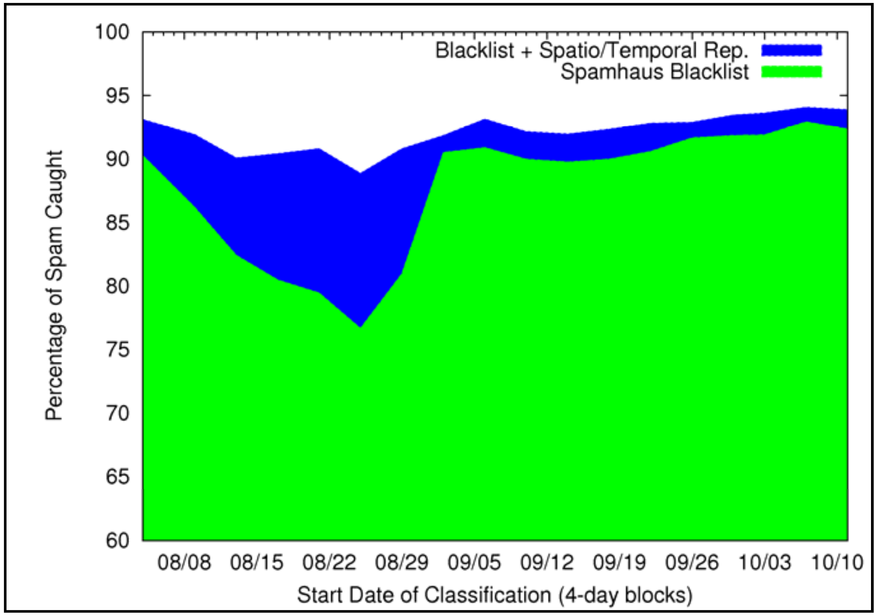
SELECT ROWS MAPPING TO α ----> **BL(α)**

$$raw\_rep(\alpha) = \sum_{i=1}^{i \le |\mathbf{BL(\alpha)}|} \frac{time\_decay(\mathbf{BL(\alpha)}_i)}{magnitude(\alpha)}$$

$$REP(\alpha) = 1.0 - (raw\_rep(\alpha) * \phi^{-1})$$

- time_decay(*): Returns on [0,1], higher weight to more recent events
- magnitude(α): Number of IPs in grouping α
- ϕ: Normalization constant putting REP() on [0,1]

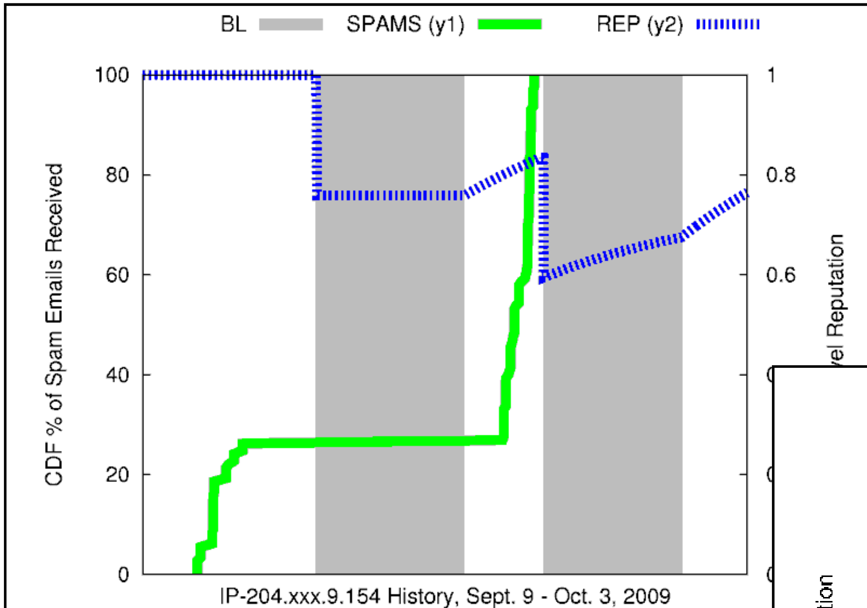
PreSTA Results



Captures up to 50% of mail not caught by traditional blacklists with the same low false-positives

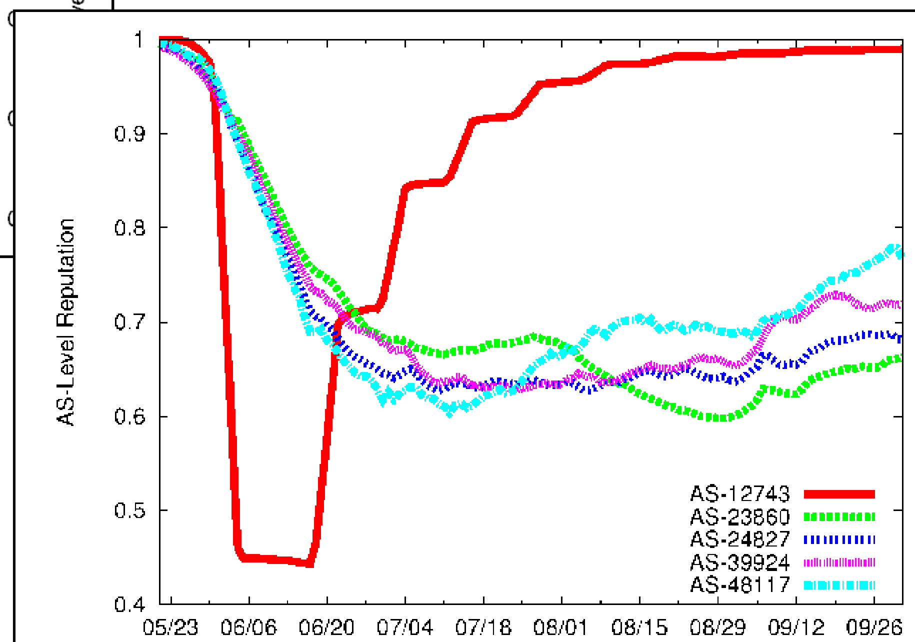
- We capture between 20-50% of spam that gets past current blacklists
  - By design our FP-rate is equivalent to BLs: ~0.4%
- Total blockage remains near constant: 90%
  - Blacklists are reactive, we are predictive. We can cover its slack
  - Cat and mouse. Graph should roll over time

PreSTA Results



(**LEFT**) Temporal (single IP) example where our metric could mitigate spam

(**RIGHT**) Probable botnet attack which our metric could mitigate via both temporal/spatial means



PreSTA: Wikipedia

PURPOSE: Build a blacklist of user-names/IPs based on the probability they will vandalize

TEMPORAL

- Straightforward, vandals are probably repeat offenders
- Registered users have IDs indicating when they joined, are new users more likely to vandalize?

SPATIAL

- Geographical: Based on user location (*i.e.*, Wash. D.C.)
- Topical: A user may vandalize one topic (Rush Limbaugh), while properly editing another (Barack Obama)
- Anonymous users: IP address properties

FEEDBACK

- Certain administrators have rollback (revert) privileges
- Comment: "Reverted edit by X to last edition by Y"

CONCLUDING (ALL)

- Quantitative Trust Management (QTM)
  - Combines Policy-based and Reputation-based TM
- QuanTM [1] framework
  - Theoretical underpinnings of combination
  - TDG as the shared data-structure
  - Partial applications:
    - Simulator [2] for reputation-component
    - PreSTA [3]: Reputation incorporating properties of a hierarchical delegation (as in PTM)

REFERENCES

- [1] - West, A.G. *et al*. QuanTM: A Quantitative Trust Management System. In Proceedings of *EuroSec '09*. Nuremberg, Germany.
- [2] – West, A.G. *et al*. An Evaluation Framework for Reputation Management Systems. Book chapter. To appear in *Trust Modeling and Management in Digital Environments: From Social Concept to System Development* (Zheng Yan, ed.)
- [3] - West, A.G. *et al*. Preventing Malicious Behavior Using Spatio-Temporal Reputation. In submission to *EuroSys '10*. Paris, France.
- [4] – Jøsang, A. et al. Trust Network Analysis with Subject Logic. In *29th Australasian Computer Science Conference, 2006*.
- [5] - Kamvar, S.D. *et al*. The EigenTrust Algorithm for Reputation Management in P2P Systems. In *12th World Wide Web Conference '03*.
- [6] – Blaze, M. *et al*. Dynamic Trust Management. Magazine article. In *IEEE Computer* (Special Issue on Trust Management), 2009.