README – Trust Management Simulator

**INTRODUCTION:**

This trust management simulator is extended on the basis of TM/RM Simulator and used in the performance evaluation of trust management algorithms. It consists of several crucial components: (1) A 'TraceGenerator' that outputs trace files describing a network initialization and transactions; (2) A 'TraceSimulator' that takes a trace and trust algorithm as input, simulates the network run, and outputs statistics about how the trust manager performed; (3) Four attack models are defined and the corresponding feedback-rating strategies are also set as reported in EigenTrust; (4) several trust models are included, e.g. NonTrust, EigenTrust and M2MTrust.

**TRACE GENERATION:**

Traces are generated by running 'TraceGenerator'. Customization of traces is achieved at the command line. All arguments are optional, as the program uses defaults for those not provided. The output should have a \*.trace extension. Otherwise, the below listing should be self-explanatory:

> NUM\_USERS: (int) # of good/malicious users/nodes in entire network

> NUM\_FILES: (int) # of distinct files in network

> NUM\_TRANS: (int) # of transactions to simulate

> ZIPF: (float) Zipf controlling file popularity

> OUTPUT: (string) Filename for trace output (\*.trace)

> PRE\_TRUSTED: (int) # of 'pre-trusted' users, a subset of 'good'

> USR\_PURE: (int) # of 'purely malicious' users

> USR\_DISG: (int) # of 'disguise malicious' users

> USR\_SYBL: (int) # of 'Sybil attack' users

>

> BAND\_MAX: (int) # of max upload/download connections per user

>> WARMUP: (int) # of warm-up transactions before counting statistics

**TRUST SIMULATION:**

Simulations are performed by running 'TraceSimulator'. The program takes three arguments: (1) the input trace file, (2) the trust mode to use, and (3) an attack model. All arguments are required:

> -input: (string) Input trace file per 'trace\_0.trace' creation (\*.trace)

> -tm: (string) Descriptor of used trust model to simulate

> -strategy: (string) Descriptor of attack mode to simulate

The argument provided to '-tm' must be an algorithm already implemented within the system. Currently, the following algorithms are implemented:

> 'EigenTM': The EigenTrust algorithm .

> 'M2MTrust': Innclud three trust modes: M2MTrust\_UP, M2MTrust\_LRT and M2MTrust.

> NoneTM: Absence of trust management. Randomly select downloading source.

Similarly, the possible arguments to '-strategy' are pre-programmed:

> 'isolated': Attack Model A (Independently Malicious)

> 'collective': Attack Model B (Chain of Malicious Collectives)

> 'disguise': Attack Model C (Malicious Collectives with Camouflage)

> 'spy': Attack Model D (Malicious Spies).

Running the simulator will create an output file. The file will contain statistics about the simulation. The extension of the output file will equal that of the trust-management-algorithm employed. For example, if 'trace\_0.trace's simulated using 'EigenTrust', the output file will be 'trace\_0.EigenTrust'.