The notion of a trusted environment is somewhat fluid. The departure of a trusted staff member with access to sensitive information can become a data breach if the staff member retains access to the data subsequent to termination of the trust relationship. In distributed systems, this can also occur with a break down in a web of trust. Most such incidents publicized in the media involve private information on individuals, i.e. social security numbers, etc Loss of corporate information such as trade secrets, sensitive corporate information, details of contracts, etc or of government information is frequently unreported, as there is no compelling reason to do so in the absence of potential damage to private citizens, and the publicity around such an event may be more damaging than the loss of the data itself.

Although such incidents pose the risk of identity theft or other serious consequences, in most cases there is no lasting damage; either the breach in security is remedied before the information is accessed by unscrupulous people, or the thief is only interested in the hardware stolen, not the data it contains. Never the less, when such incidents become publicly known, it is customary for the offending party to attempt to mitigate damages by providing to the victims subscription to a credit reporting agency, for instance.

CONCLUSION

In a perfect world there would be no need to hand over sensitive data to agents that may unknowingly or maliciously leak it. And even if we had to handover sensitive data, in a perfect world we could watermark each object so that we could trace its origins with absolute certainty. However, in many cases we must indeed work with agents that may not be 100% trusted.

In spite of these difficulties, we have shown it is possible to assess the likelihood that an agent is responsible for a leak, based on the overlap of his data with the leaked data and the data of other agents, and based on the probability that objects can be ‘guessed´ by other means. Our model is relatively simple, but we believe it captures the essential trade-offs. The algorithms we have presented implement a variety of data distribution strategies that can improve the distributor’s chances of identifying a leaker. We have shown that distributing objects judiciously can make a significant difference in identifying guilty agents, especially in cases where there is large overlap in the data that agents must receive. It includes the investigation of agent guilt models that capture leakage scenarios that are not studied in this paper.

REFERENCES

[1] Data Leakage Detection, an IEEE paper by Panagiotis Papadimitriou, Member, IEEE, Hector Garcia-Molina, Member, IEEE NOV-2010.
[2] Watermarking relational databases. In VLDB ’02: Proceedings of the 28th international conference on Very Large Data Bases, By R. Agrawal and J. Kiernan, pages 155–166. VLDB Endowment, 2002.
[3] An algebra for composing access control policies, By P. Bonatti, S. D. C. di Vimercati and P. Samarati, ACM Trans. Inf. Syst. Secur., 5(1):1–35, 2002.
[4] P. Buneman, S. Khanna, and W. C. Tan. Why and where: A characterization of data provenance. In J. V. den Bussche and V. Vianu, editors, Database Theory - ICDT 2001, 8th International Conference, London, UK, January 4-6, 2001, Proceedings, volume 1973 of Lecture Notes in Computer Science, pages 316–330. Springer, 2001.
[5] P. Buneman and W.-C. Tan. Provenance in databases. In SIGMOD ’07: Proceedings of the 2007 ACM SIGMOD international conference on Management of data, pages 1171–1173, New York, NY, USA, 2007. ACM.
[6] Lineage tracing for general data warehouse transformations, By Y. Cui and J. Widom, In The VLDB Journal, pages 471–480, 2001.
[7] Digital music distribution and audio watermarking, by S. Czerwinski, R. Fromm, and T. Hodes.