**Privacy-preserving Crowd-sourced Statistical Data Publishing with An Untrusted Server**

**Abstract:**

 The continuous publication of aggregate statistics over crowd-sourced data to the public has enabled many data mining applications (e.g., real-time trafﬁc analysis). Existing systems usually rely on a trusted server to aggregate the spatio-temporal crowd-sourced data and then apply differential privacy mechanism to perturb the aggregate statistics before publishing to provide strong privacy guarantee. However, the privacy of users will be exposed once the server is hacked or cannot be trusted. In this paper, we study the problem of real-time crowd-sourced statistical data publishing with strong privacy protection under an untrusted server. We propose a novel distributed agent-based privacy-preserving framework, called DADP, that introduces a new level of multiple agents between the users and the untrusted server. Instead of directly uploading the check-in information to the untrusted server, a user can randomly select one agent and upload the check-in information to it with the anonymous connection technology. Each agent aggregates the received crowd-sourced data and perturbs the aggregated statistics locally with Laplace mechanism. The perturbed statistics from all the agents are further combined together to form the entire perturbed statistics for publication. In particular, we propose a distributed budget allocation mechanism and an agent-based dynamic grouping mechanism to realize global w-event -differential privacy in a distributed way. We prove that DADP can provide w-event -differential privacy for real-time crowd-sourced statistical data publishing under the untrusted server. Extensive experiments on real-world datasets demonstrate the effectiveness of DADP.

**SYSTEM REQUIREMENTS:**

**HARDWARE REQUIREMENTS:**

* System : Pentium Dual Core.
* Hard Disk : 120 GB.
* Monitor : 15’’ LED
* Input Devices : Keyboard, Mouse
* Ram : 1 GB

**SOFTWARE REQUIREMENTS:**

* Operating system : Windows XP/UBUNTU.
* Implementation : NS2
* NS2 Version : 2.28
* Front End : OTCL (Object Oriented Tool Command  Language)
* Tool : Cygwin (To simulate in Windows OS)