Some Problems in Sanitizing Network Data

Matt Bishop, Rick Crawford, Bhume Bhumiratana,
Lisa Clark, Karl Levitt
Dept. of Computer Science
University of California at Davis
One Shields Ave.
Davis, CA 95616-8562

Contact Information

Matt Bishop

Department of Computer Science
University of California at Davis
One Shields Ave.
Davis, CA 95616-8562
United States

phone: +1 (530) 752-8060

email: mabishop@ucdavis.edu

Outline

- Problem statement and assumptions
- Network data sanitization
- Addresses and namespaces
  - IP address ranges and properties
  - Implications of finite namespaces
  - Nuances
- TCPsani
- Related work

Problem Statement

- *collector* collects data, gives it to analyst
- Some information in data is confidential
  - Sanitize it to remove enough information that sensitive data cannot be determined
- *adversary* tries to “reverse engineer” data to recover unsanitized information

Assumptions

1. Adversary inserts markers into data as it is generated
2. Adversary has private information that, when combined with sanitized data, allows deduction of unsanitized data
3. Adversary can deduce unsanitized information directly from sanitized data

Our Assumptions

4. Adversary may have access to other sources that, when combined with sanitized data, allows deduction of unsanitized data
   - Privacy policy, threat model account for this
5. Collector makes data equally available to the analyst and adversary
   - Really, a worst case assumption

Why Sanitize?

- Need network traffic
  - Synthesize data
  - Capture, use reference data
  - Capture, sanitize data

Our Focus

- Packet headers
  - Care about MAC, IP, TCP/UDP headers
  - Overwrite application layer data
- Privacy policy: *no IP address may be associated with an individual*

Types of Anonymization

- Pseudo-anonymous
  - All instances of “John” become “Paul”
- Fully anonymous
  - John becomes “Paul”, “George”, “Ringo”
- Do mapping via tables, hash functions, *etc*.

Problems

- IP address ranges
  - Tables vs. hash functions
- Finite name spaces
  - Applying the pigeonhole principle
- Semantics
  - Sanitized data conveys contextual information not apparent from the data itself

IP Address Ranges

- Prevent mapping of 2 IP addresses to same target address
- Hash functions
  - Mutually distrusting collectors: risk dictionary attacks
- Tables
  - Mutually distrusting collectors can share portions of code book

Finite Namespaces

- Pseudo-anonymous mappings are permutations
- Fully anonymous mappings risk overflowing target namespace
  - Risks repetitions of sanitized names
  - “Bob” → any of 3 names
  - “Alice” → any of 10 names (more privacy)
  - Alternate approach: iterate sanitization

UC Davis
Computer Security Laboratory

Semantics of Data

- Medical database
  - Not enough to sanitize names: other data might identify patient (date of birth, ZIP, gender)
  - Value of one field affects others; eg. ratio of height to weight significant
  - Names affect analysis; "Bob" unlikely to be at risk for pregnancy; "Shikigawa" more likely to be lactose-intolerant than "Smith"
- Analysis policy, metric must be explicit

The IP Version

- Packets to, from host all have port 53
  - Host is clearly a DNS server
  - Pseudo-anonymity preserves this
  - Full anonymity does not
- Pseudo-anonymity allows mapping the infrastructure

Moral

- In order to sanitize properly, three models required:
  - Threat model
  - Privacy policy
  - Analysis policy

TCPsani

- Inputs *tcpdump savefile*, sanitizes it, outputs in *savefile* format
  - Can be fed back to *tcpdump*, *etc.*
- Modified version of *tcpdump* that invokes Perl routines to sanitize
  - If you don’t like ours, can write your own …

Supplied Versions

- Pseudo-anonymous mode:
  - Maps IP address byte by byte; IP address prefix determines map
  - Maps can be preconfigured to be shared
- Fully anonymous mode:
  - Map source region *R* to target region *T*
  - New address: pick empty slot in *T*, use it, add it to table

Related Work

- Classless IP address prefixes (*tcpdpriv*, *CryptoPan*)
    - Require trust among different collectors
- Network layer vs. application layer (Paxson, Pang)
- Pseudo-anonymous sanitization in file names and firewall logs (Sobirey, Fischer-Hübner, Rannenberg; Biskup, Flegel; Lundin, Jonsson)

Related Problems

- Privacy, analysis policies constrain *inferences*
- Database query-audit problem
- Cryptographic exchanges: crowds, anonymity set
- Semantics and structure (or lack thereof)
  - Preserving digital signatures of raw data to validate origin, integrity after sanitization

Conclusion

- Need explicit models to know what to do
- Related to several other, classic problems
- But *critical* we understand and make progress on it in order to further research and balance it with privacy needs
- Hard problem!

Concluding Thought

The truth about a man lies first and foremost in what he hides.
—Andre Malraux