SRI International

Composing Cross-Domain Solutions

Ashish Gehani and **Gabriela F. Ciocarlie**
Computer Science Laboratory, SRI International

Dec 3rd, 2012

Motivation – The Big Picture

- Cross-domain solutions (CDSs) are integral components of the U.S. Defense Department's global information grid (GIG)
- CDSs provide assured information sharing, **BUT**

- CDSs have **limitations**
  - Not particularly suitable for net-centric operations
  - Exhibit large deployment times which cannot cope with stringent requirements

Agile CDS Vision

- Decomposing the problem into sub-problems that are more tractable
- Then integrating the component solutions

- The building blocks for achieving this:
  - Formal specifications for generic downgrading engines
  - Formal languages for data sanitization rules
  - Filters for specific data types
  - Attribute-based access control

- Using pre-certified commercial off-the- shelf (COTS) CDS facilitates rapid deployment in the field
  - The availability of COTS devices depends on their timely evaluation

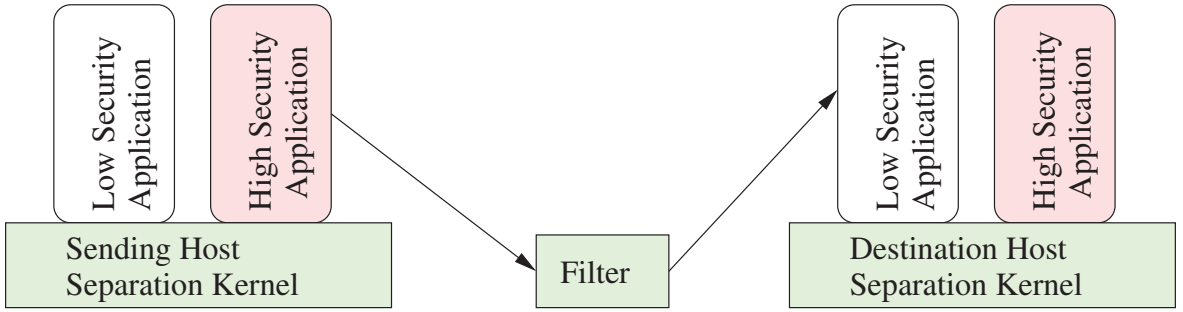
Problem to Solve

Speed-up CDS Evaluation

Data Downgrading

- High-assurance CDSs are instrumental for information sharing across security domains

- A system's security-critical components are decomposed into modules that can each be completely verified (MILS)



- Downgraders need to cope with multiple types of data, requiring transformation and sanitization mechanisms to allow the information flow

Decomposing Sanitization

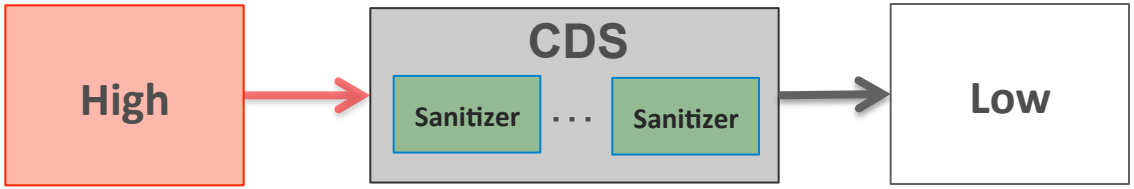
- Address the problem of downgrading data that has components with multiple classification levels by leveraging
  - The nature of the data being downgraded
  - The available trusted computing infrastructure

  to decompose the downgrading functionality to the point that each module can economically be formally specified and have its operational behavior verified

- Make complex data sanitization practical
  - Inspired by multiple independent levels of security (MILS)

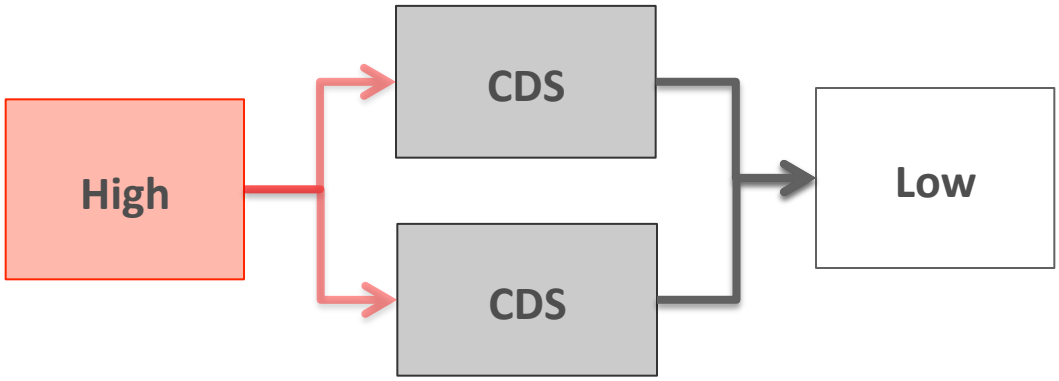
Architectures for Composing Cross Domain Solutions

- Intra-CDS



- Serial CDS

- Parallel CDS

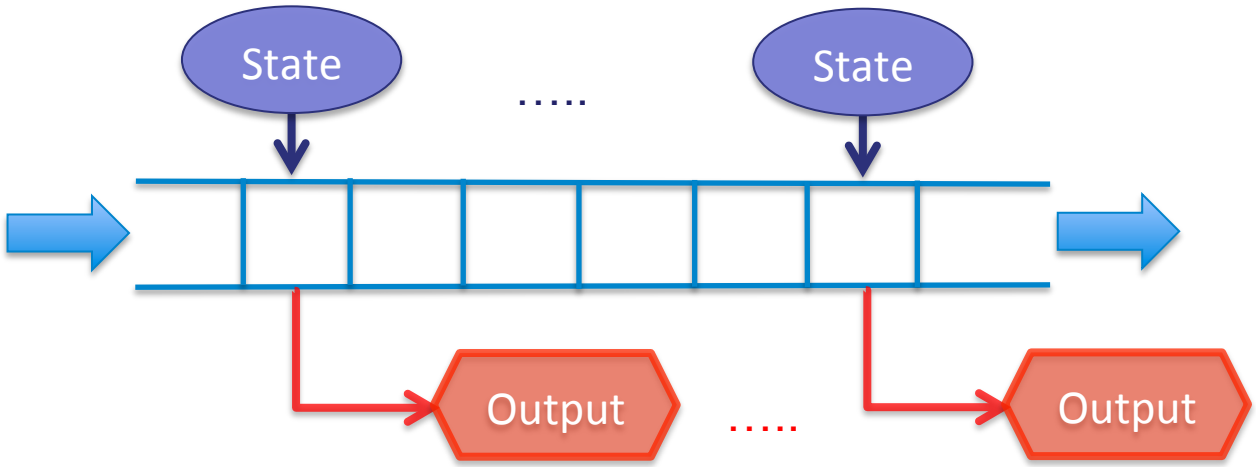


Sanitization Algorithms

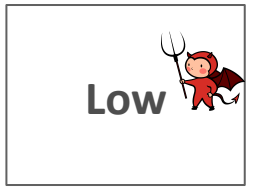
- Previously studied in the context of publishing privacy-sensitive data
  - To preserve the privacy of individual record owners, a downgrader sanitizes information derived from such databases
- Different methods have been approached
  - Perturbing the query inputs and outputs, and restricting the number of queries
  - Suppression that removes records from the sanitized output
  - Randomization that adds noise to perturb the data, and multi-views that provide sanitization through diverse perspectives.
- For CDS, the data may never have been observed previously
  - Recent research on **streaming differential privacy** provides a framework for designing sanitization algorithms appropriate for a CDS

Sanitizers / CDS Characteristics

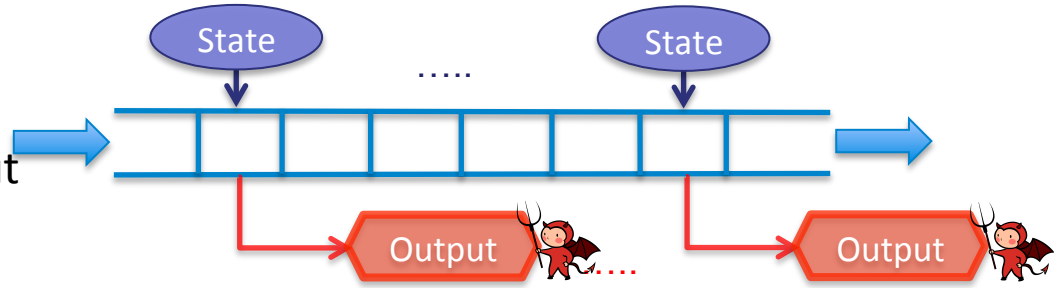
- Operate on a stream of items
- Inspect each item and update internal state
- Produce an output either for each item or at the end of the stream



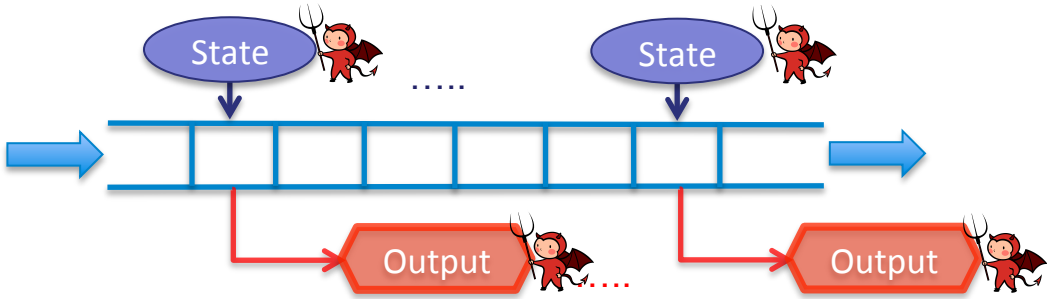
Differential Privacy for Data Streams



- **Privacy against continual output observation**
  - The adversary examines the output **all the time**



- **Pan-Privacy**
  - The adversary examines also the internal state (intrusion)
    - Announced (subpoena)
    - Unannounced
      - **Once? Several times? All the time?**



Characterizing Leakage of Information

- Leakage may depend on auxiliary information available externally, but never observed by the downgrader

- **User-level X-adjacency**
  - Data streams S and S ′ are **X -adjacent** if they differ only in the presence or absence of **any number** of occurrences of a single item $x \in X$

  S = axbxcxdxxxex

  S' = abcdxe

- **Event-level X-adjacency**
  - Data streams S and S ′ are **X -adjacent** if the number of instances of one item replaced by another is **at most 1**

  S = abcdexfg

  S' = abcdeyfg

Differential Privacy Against Continual Observation

- **Assumption**: The sanitizer / CDS is **trusted**
- *A* - algorithm working on a stream of data
- *A* is **ε-differentially private** against continual observation if for all
  - adjacent data streams *S* and *S'* (user or event level)
  - outputs $\sigma_1\ \sigma_2 \ldots\ \sigma_t$

$$e^{-\epsilon} \leq \frac{Pr[A(S) = \sigma_1\sigma_2 \ldots \sigma_t]}{Pr[A(S') = \sigma_1\sigma_2 \ldots \sigma_t]} \leq e^{\epsilon}$$

Pan-Privacy ("inside and out")

- **Assumption**: The sanitizer / CDS is **not trusted**

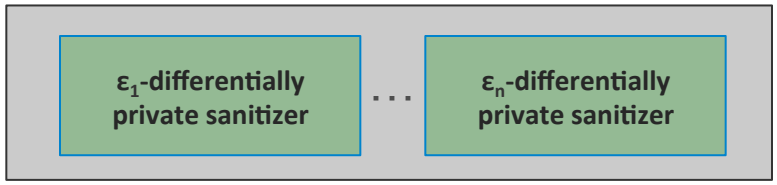
- *A* - algorithm working on a stream of data
- I - the set of internal states of the algorithm
- σ - the set of possible output sequences

- *A* mapping stream items to *I*×*σ* is (**ε-differentially) pan-private** (against a single unannounced intrusion) if for all
  - adjacent data streams *S* and *S'* (user or event level)
  - $I' \subseteq I$ and $\sigma' \subseteq \sigma$

$$e^{-\epsilon} \leq \frac{Pr[A(S) \in (I', \sigma')]}{Pr[A(S') \in (I', \sigma')]} \leq e^{\epsilon}$$

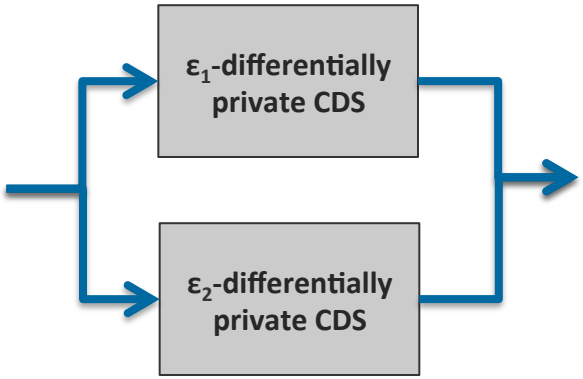
Composable Sanitization

- **Theorem** The composition of an ε_1-differentially private mechanism and an ε_2-differentially private mechanism is **at worst $(\varepsilon_1 + \varepsilon_2)$**-differentially private



$\sum \varepsilon_i$ - differentially private CDS

$(\varepsilon_1 + \varepsilon_2)$-differentially private composition



$(\varepsilon_1 + \varepsilon_2)$-differentially private composition

Conclusions

- High-assurance systems with multiple security levels use data filters to facilitate the safe flow of information
- As the content and context of the data increases in complexity, the cost and time to certify CDS is growing rapidly
- Downgrading functionality should be decomposed to the point where each filter provides a streaming differential privacy guarantee and its certification is economically viable
- The resulting filters can be combined to provide equivalent functionality to that provided by monolithic downgraders

Acknowledgments

- This material is based upon work supported by the National Science Foundation under Grant IIS-1116414. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

Thank You

Headquarters: Silicon Valley

SRI International
333 Ravenswood Avenue
Menlo Park, CA 94025-3493
650.859.2000

Washington, D.C.

SRI International
1100 Wilson Blvd., Suite 2800
Arlington, VA 22209-3915
703.524.2053

Princeton, New Jersey

SRI International Sarnoff
201 Washington Road
Princeton, NJ 08540
609.734.2553

Additional U.S. and international locations

www.sri.com