ICACCT 2018

Secure Data Exchange and Data Leakage Detection in Untrusted Cloud

Denis Ulybyshev, Bharat Bhargava, Aala Alsalem

Computer Science Department, CERIAS

Purdue University, West Lafayette, USA

Outline

- **Problem Statement**
- Related Work
- **Core Design**
- **Evaluation**
- **Contributions**

Problem Statement

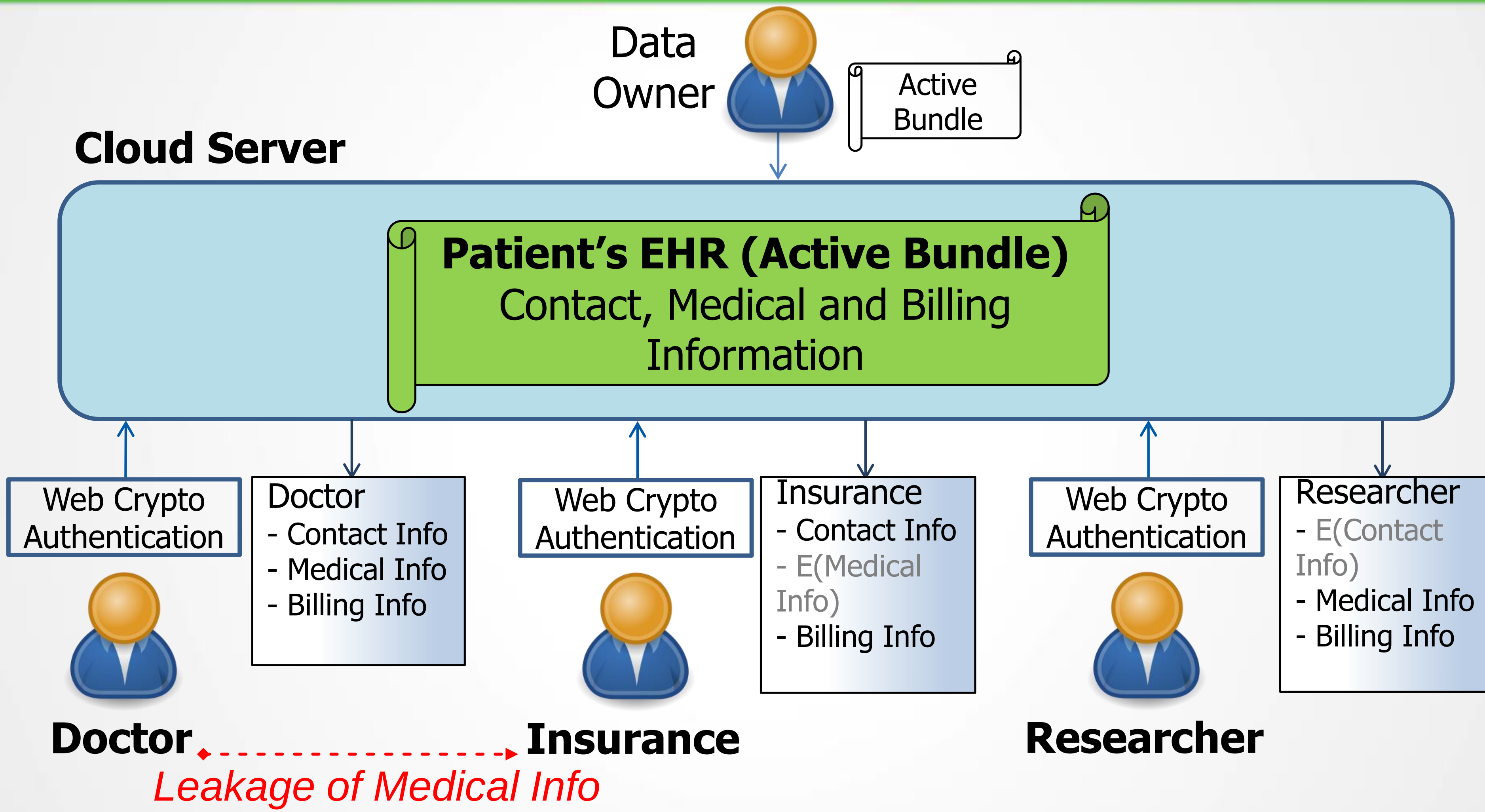
Secure Data Exchange / Leakage Detection

- Authorized service can only access data items for which it is authorized
- Data exchange model must consider context and client's attributes
- Detect data leakages made by insiders to unauthorized services
- Measure data leakage (what got leaked, when, to where, how sensitive was the data)

Recent Data Leakages Examples

Company	Time	Incident Details
Adobe Systems	Oct.2013	150 million accounts of software subscription database got leaked
Anthem	Feb.2015	78.8 million of PII records got leaked
Experian Information Solutions and T-Mobile, USA	Sep.2015	Data (SSN, credit card information) of about 15 million customers who applied for credit got leaked
U.S. Office of Personnel Management: Agency of the U.S. Federal government	Jun.2015	SSN, names, addresses, places of birth of 22 million people got leaked

Problem Statement

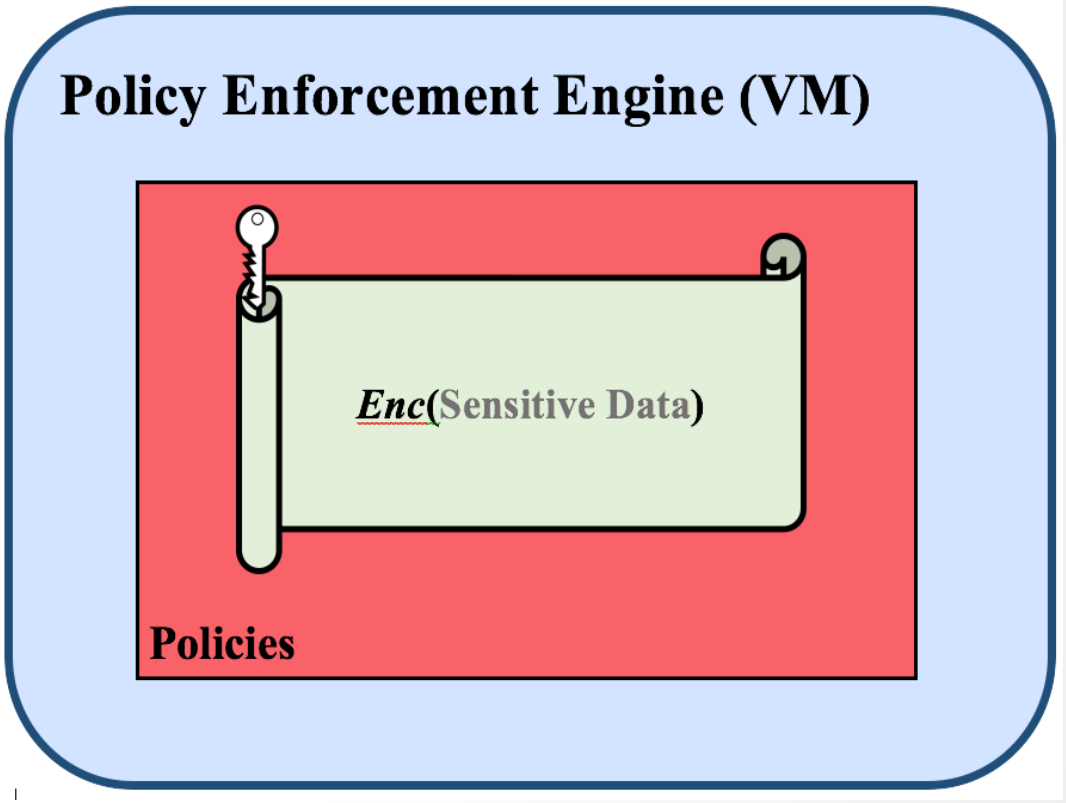


Scenario of EHR Dissemination in Cloud (proposed by Dr. Leon Li, NGC)

AB Core Design

Active Bundle (AB) parts [17], [18]

- *Sensitive data*:
  - Encrypted data items
- *Metadata*: describe AB and its access control policies
  - Policies [21], [22] manage AB interaction with services and hosts

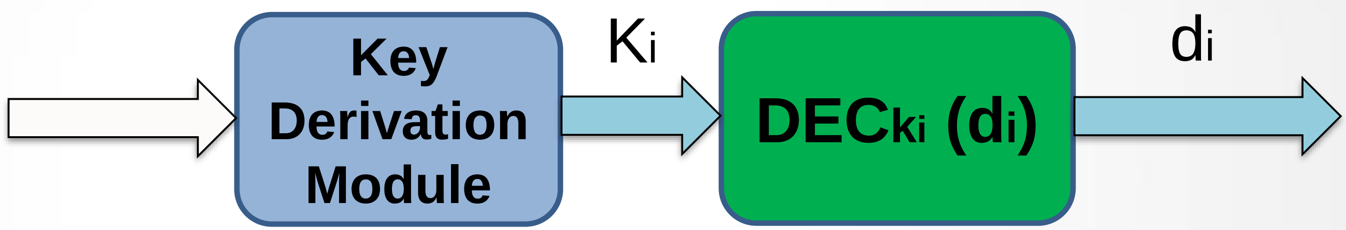


- *Policy Engine* [26]: enforces policies specified in AB
  - Provides tamper-resistance of AB [1]

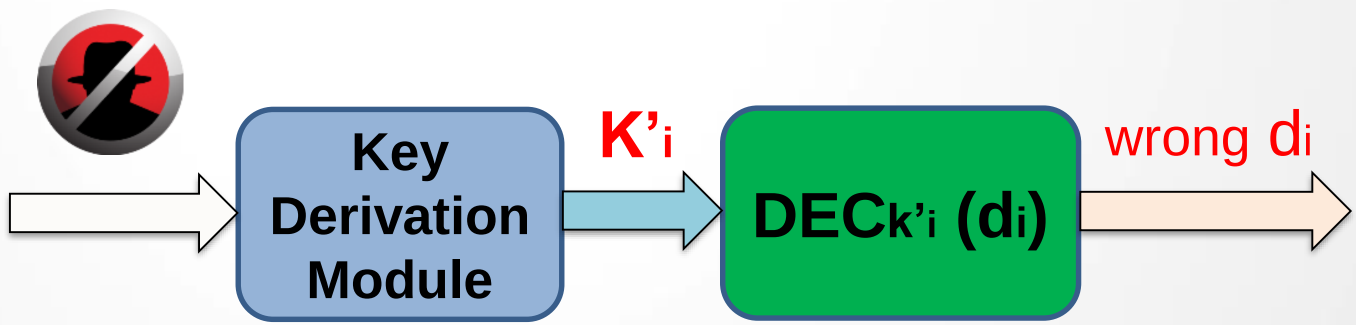
Tamper Resistance of AB

- Key is not stored inside AB [2]
- Separate symmetric key is used for each separate data value
- Ensure protection against tampering attacks

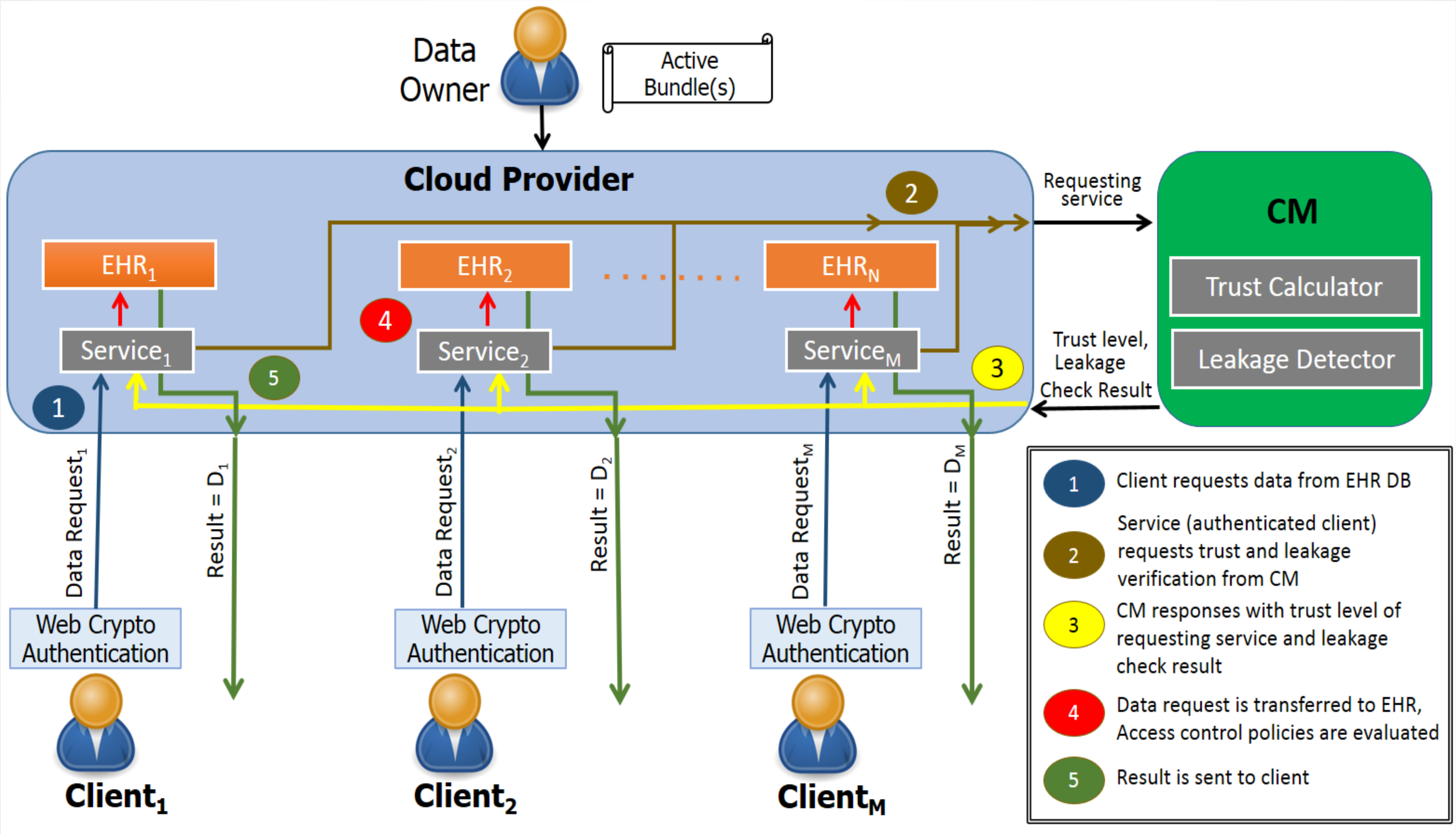
Aggregation{d_i}
(*Execution info;*
Digest(AB Modules);
Resources)



Aggregation{d_i} (
Tampered (
Execution info;
Digest(AB Modules);
Resources ***)***)



Framework Architecture

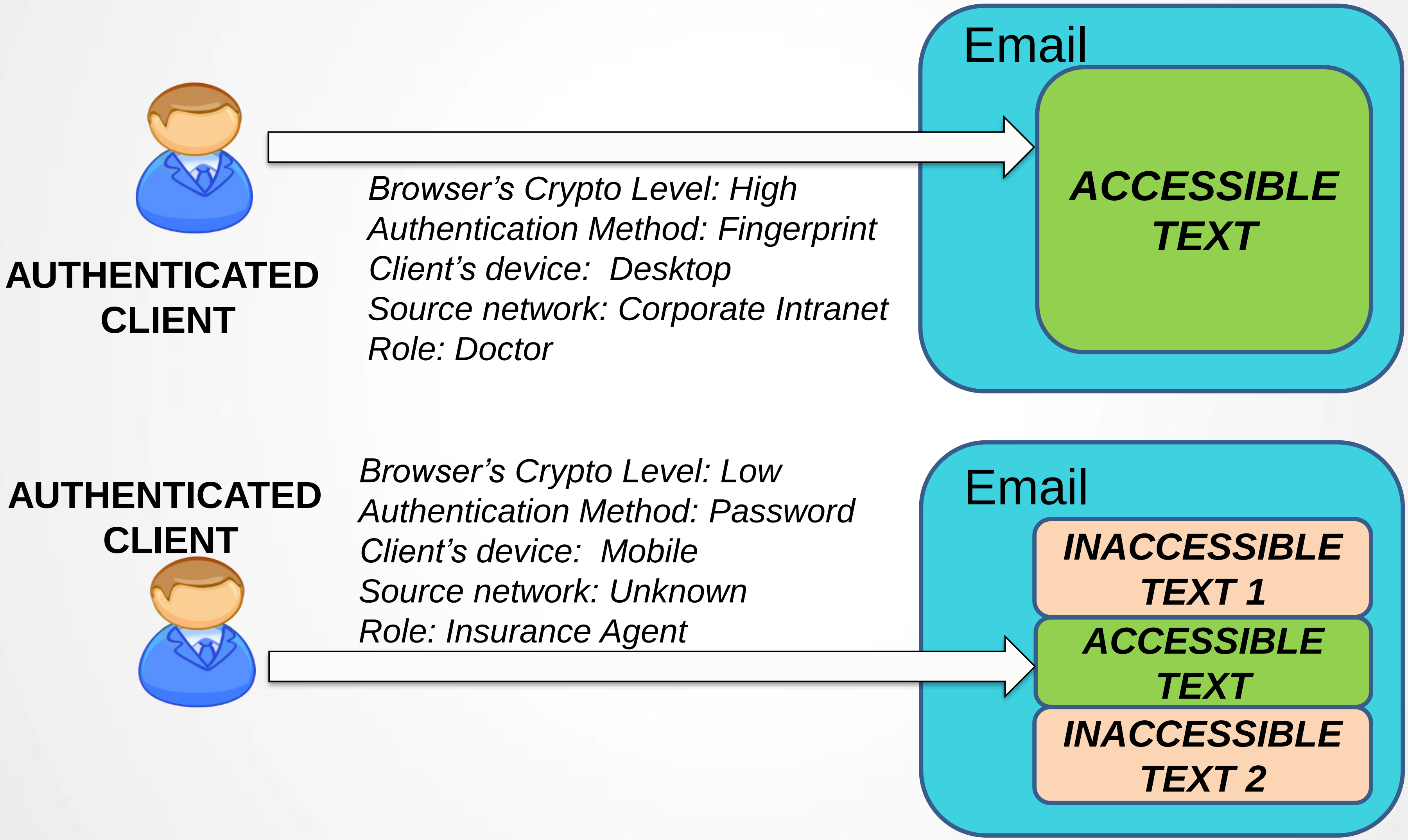


Data dissemination features

Data Dissemination based on [3]:

- Access control policies [27]
- Trust level of a subject (service, user)
- Context (e.g. emergency vs. normal)
- Security level of client’s browser (crypto capabilities) [23], [24]
- Authentication method (password-based, fingerprint etc)
- Source network (secure intranet vs. unknown network)
- Type of client’s device: desktop vs. mobile (detected by Authentication Server)

Attribute and role–based data dissemination

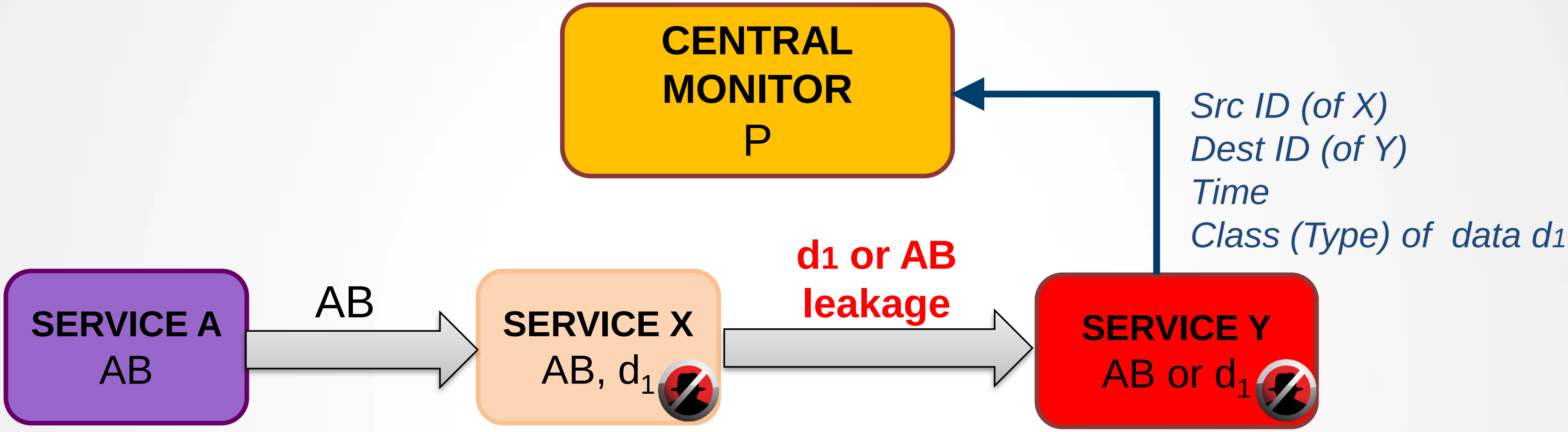


Data leakage detection

How can data get leaked by authorized subject?

- In the form of encrypted data (the whole AB is leaked):
  - Data is protected by AB, but fact of leakage can be detected
  - Detection Phase 1: digital watermark [12] can be checked by web crawler to detect copyright violations
  - Detection Phase 2: based on Obligations: how data is used by authorized party?
    - *Obligations are enforced by Central Monitor (TTP)*
    - *CM checks whether data is supposed to be where they are*

Core Design: Data Leakage Detection



AB contains:

- Enc [Data(D)] = $\{Enc_{k1}(d_1), \ldots, Enc_{kn}(d_n)\}$
- Access Control Policies (P) = $\{p_1, \ldots, p_k\}$

- Service X is authorized to read d_1 from AB
- Service X may leak decrypted d_1 or the entire AB to Y

Core Design: Data Leakage Detection

- When service tries to decrypt AB data, CM is notified about that: "Service Y tries to decrypt d_1 arrived from X"
- If CM is unreachable, decryption terminates
- CM checks against centralized Obligations DB: whether d_1 is supposed to be at Y. If NO then:
  - Blacklist X, Y
  - Reduce their trust level
  - Mark data d_1 as compromised and notify services about it
  - Raise the level of d_1 classification

Plaintext Data Leakage Detection

How can data get leaked by authorized subject?

- In the form of decrypted (raw) data:
  - Data is not protected by AB anymore
  - Detection based on visual / digital watermarks embedded into data

Plaintext Data Leakage Mitigation Methods

- **Layered Approach:** Don't give all the data to the requester at once
  - First give part of data (incomplete, less sensitive)
  - Watch how it is used and monitor trust level of using service
  - If trust level is sufficient – give next portion of data
- **Raise the level of data classification** to prevent leakage repetition
- **Intentional leakage** to create uncertainty and lower data value
- **Use provenance data stored at CM** to identify the list of suspects
- **Monitor network messages**
  - Check whether they contain e.g. credit card number that satisfies specific pattern and can be validated using regular expressions [25]

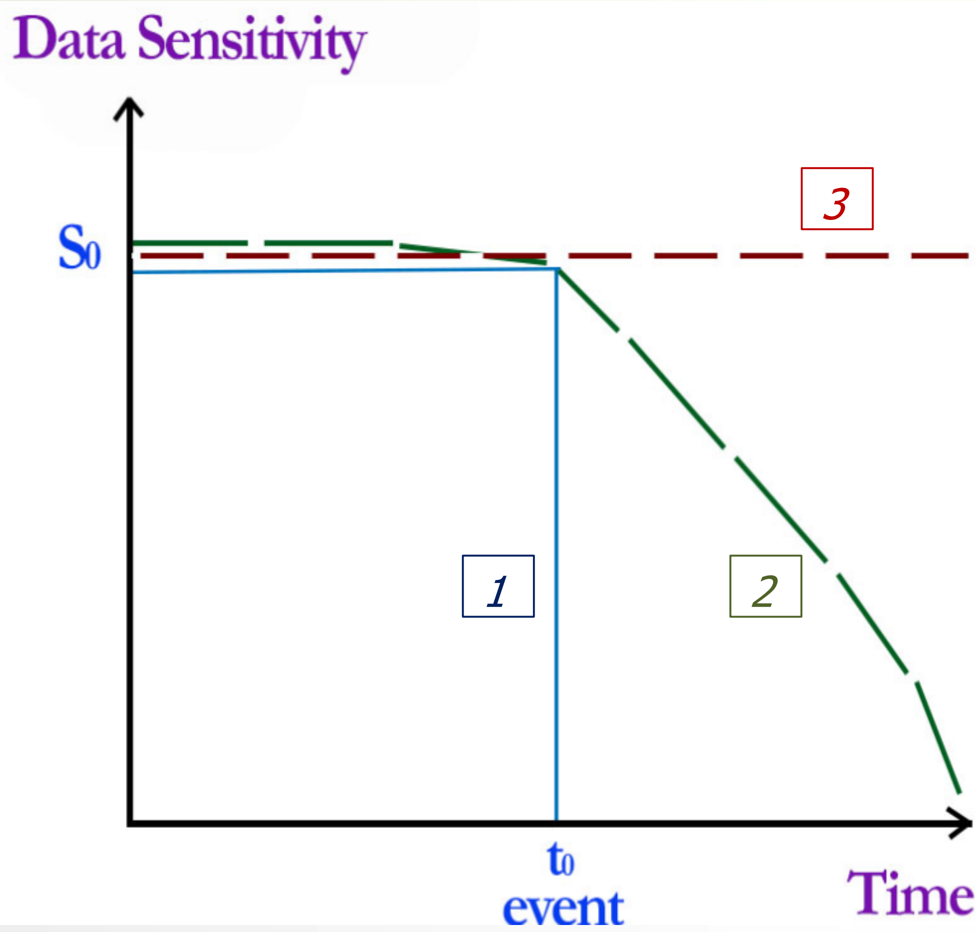
Data Leakage Damage Assessment

- After data leakage is detected damage is assessed based on:
  - To whom was the data leaked (unknown service with low trust level vs. service with high level of trust)
  - Sensitivity (Classification) of leaked data (classified vs. unclassified)
  - When was leaked data received (recent or old data)
  - Can other sensitive data be derived from the leaked data (i.e. diagnosis can be derived from leaked medical prescriptions)

Damage = K(Data is Sensitive) * K (Service is Malicious) * F(t)

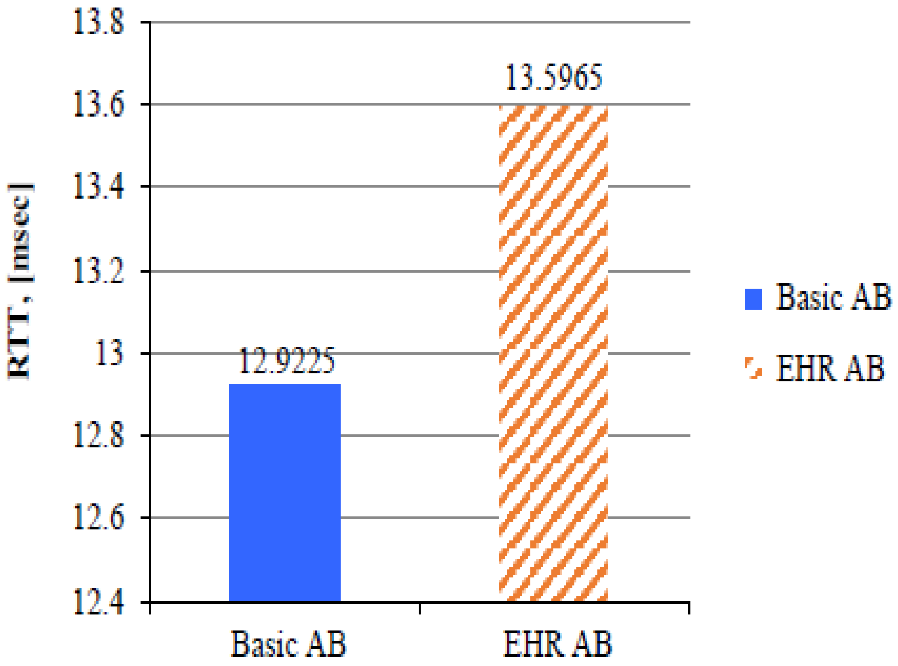
, where F(t) is the data sensitivity function in time

Timing of Leaked Data

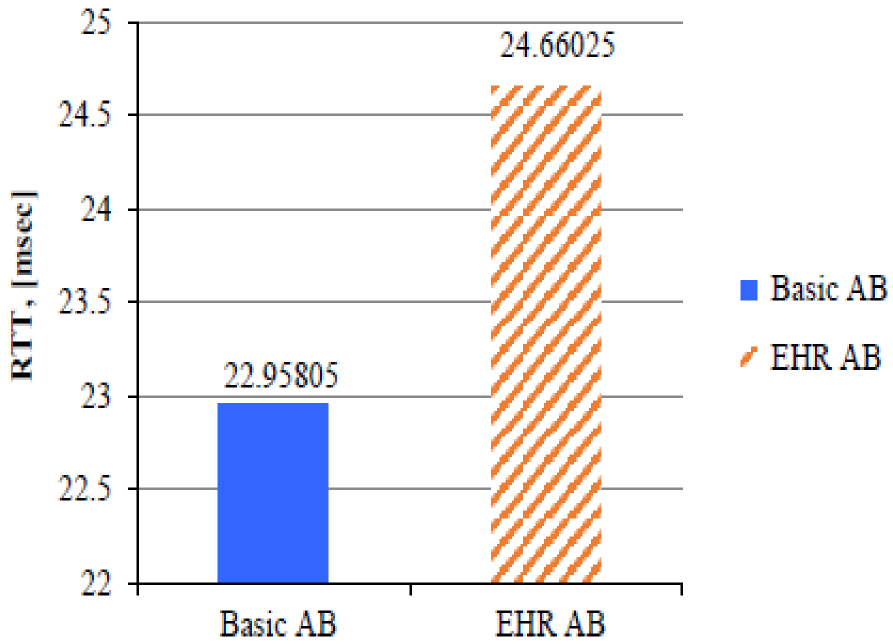


- Data-related event (e.g. final exam) occurs at t_0
- Threat from data being leaked before t_0 is high
- Threat from data being leaked after t_0:
  1) No threat at all
  2) Linearly decreases with time
  3) Remains constant (for highly-sensitive data)

Evaluation

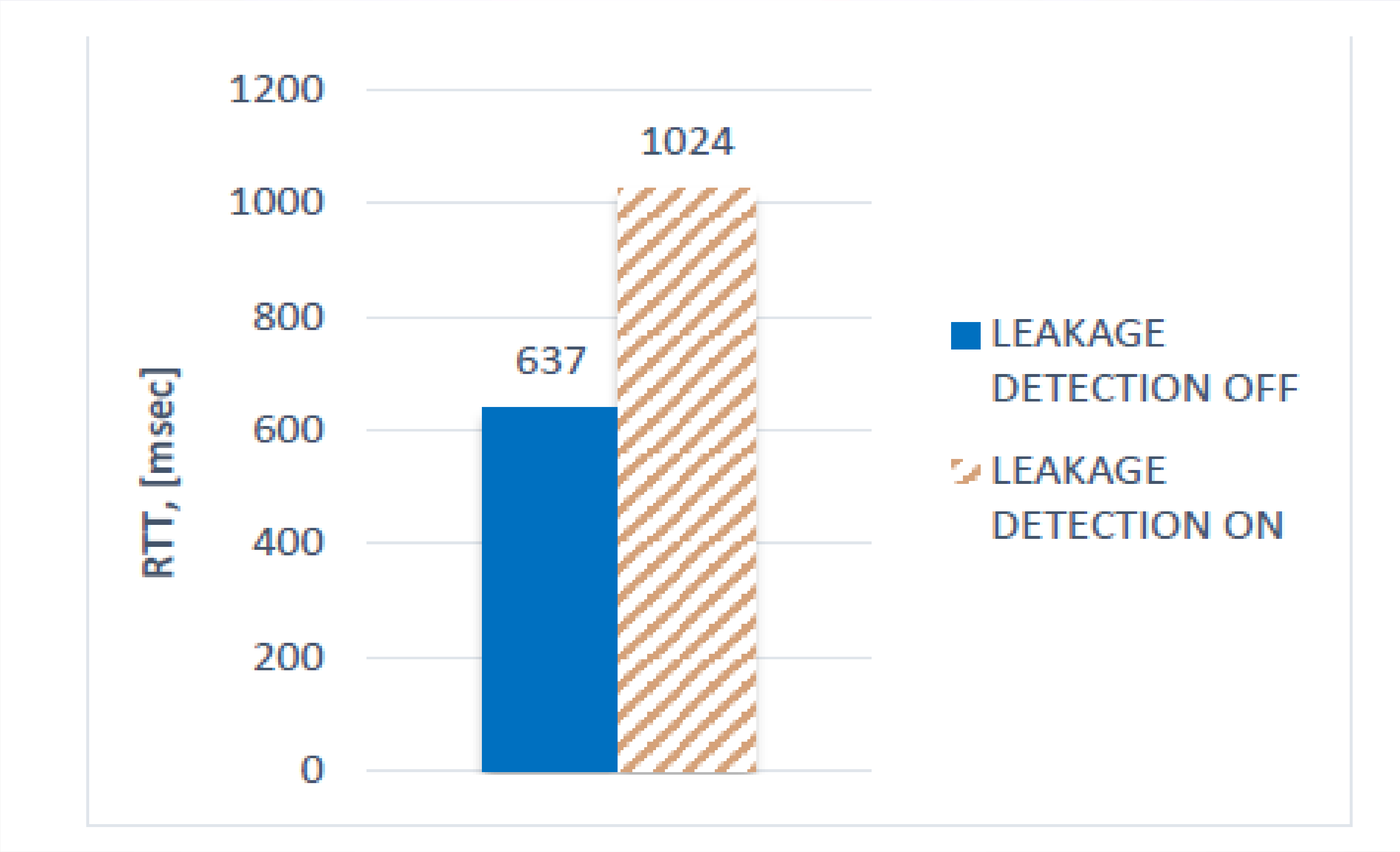


Performance overhead for EHR, hosted locally



Performance overhead for EHR, hosted by Google Cloud

Evaluation



Performance overhead imposed by data leakage detection capabilities

Contributions

Contributes to Data Confidentiality and Integrity

- Dissemination does not require data owner’s availability
- Trust level of subjects is constantly recalculated
- On-the-fly key generation
- Supports data updates for multiple subjects
- Supports attribute-based data dissemination. Attributes include cryptographic capabilities of client’s browser [28]
- Tamper-resistance: data and policies integrity is provided
- Data leakage detection and leakage damage assessment
- Captures data provenance for use in leakage measure and forensics

References

1. R. Ranchal, “Cross-domain data dissemination and policy enforcement,” PhD Thesis, Purdue University, Jun. 2015
2. C. Qu, D. Ulybyshev, B. Bhargava, R. Rohit, and L. Lilien. "Secure Dissemination of Video Data in Vehicle-to-Vehicle Systems.” 6th Intl. Workshop on Dependable Network Computing and Mobile Systems (DNCMS2015), Sep. 2015
3. D. Ulybyshev, B. Bhargava, M. Villarreal-Vasquez, A. Alsalem, L. Li, D. Steiner, J. Kobes, H. Halpin, and R. Ranchal. “Privacy-Preserving Data Dissemination in Untrusted Cloud” , *Submitted for IEEE Cloud 2017, under review*
4. R. Ranchal, D. Ulybyshev, P. Angin, and B. Bhargava. “Policy-based Distributed Data Dissemination,” *CERIAS Security Symposium, April 2015* ***(Best poster award)***
5. D. Ulybyshev, B. Bhargava, L. Li, J. Kobes, D. Steiner, H. Halpin, B. An, M. Villarreal, R. Ranchal. “Authentication of User’s Device and Browser for Data Access in Untrusted Cloud,” *CERIAS Security Symposium, April 2016*.
6. F. Li, B. Luo, P. Liu, D. Lee, and C.-H. Chu, “Enforcing secure and privacy- preserving information brokering in distributed information sharing,” IEEE Transactions on Information Forensics and Security, vol. 8, no. 6, pp. 888–900, 2013.
7. S. Pearson and M. C. Mont, “Sticky policies: an approach for managing privacy across multiple parties,” IEEE Computer, no. 9, pp. 60–68, 2011.
8. S. Calzavara, R. Focardi, N. Grimm, M. Maffei, “Micro-policies for web session security”. *Computer Security Foundations Symp. (CSF), 2016 IEEE 29th* (pp. 179-193), Jun. 2016
9. Anonymus, “Micro-policies for web session security,” 2016, available at https://sites.google.com/site/micropolwebsese, accessed: Mar.2018

References

10. SandMark: A Tool for the Study of Software Protection Algorithms http://cgi.cs.arizona.edu/~sandmark/sandmark.html

11. Y.L. Simmhan, B. Plale and D.A. Gannon, “A survey of data provenance in e-science,” SIGMOD Rec., 34(3):31–36, 2005.

12. Z.J. Xu, Z.Z. Wang and Q.Lu, "Research on Image Watermarking Algorithm based on DCT", 3rd Intl. Conf. on Environmental Science and Information Application Technology, Journal of Procedia Environmental Sciences, vol. 10, pp. 1129-1135, 2011

13. M. Stamp, "Digital Rights Management: The Technology Behind The Hype," Journal of Electronic Commerce Research, Vol. 4, No. 3: 102-112, Aug. 2003.

14. Q. Liu, R. Safavi-Naini and N. Sheppard, “Digital rights management for content distribution”, In Proc. of Australasian Information Security Workshop, pp. 49–58, 2003.

15. “Windows media DRM,” https://en.wikipedia.org/wiki/Windows_Media_DRM , accessed:Mar.2018

16. M. Nickolova, E. Nickolov, “Hardware-based and Software-based Security in Digital Rights Management Solutions”, Intl.Journal "Information Technologies and Knowledge", Vol.2, 2008.

17. L. Ben Othmane and L. Lilien, “Protecting privacy in sensitive data dissemination with active bundles,” 7-th Annual Conf. on Privacy, Security and Trust (PST 2009), Saint John, New Brunswick, Canada, Aug. 2009, pp. 202-213

18. L. Lilien and B. Bhargava, ”A scheme for privacy-preserving data dissemination,” IEEE Trans. on Systems, Man and Cybernetics, Part A: Systems and Humans, vol. 36(3), May 2006, pp. 503-506.

References

19. B. Bhargava, “Secure/resilient systems and data dissemination/provenance,” NGCRC Project Proposal, CERIAS, Purdue University, Mar.2018
20. D. Ulybyshev, B.Bhargava, “Secure dissemination of EHR,” demo video https://www.dropbox.com/s/30scw1srqsmyq6d/BhargavaTeam_DemoVideo_Spring16.wmv?dl=0
21. “Lightweight data-interchange format JSON,” http://json.org/ , accessed: Mar.2018
22. “eXtensible access control markup language (XACML) version 3.0,” http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-os-en.html, accessed: Mar. 2018
23. “W3C Web Cryptography API,”https://www.w3.org/TR/WebCryptoAPI/, accessed: Mar.2018
24. “Web authentication: an API for accessing scoped credentials,” http://www.w3.org/TR/webauthn , accessed: Mar.2018
25. “Finding or verifying credit card numbers,” http://www.regularexpressions.info/creditcard.html , accessed: Mar.2018
26. “WSO2 Balana Implementation,” https://github.com/wso2/balana , accessed: Mar.2018
27. L. Lilien and B. Bhargava, ”A scheme for privacy-preserving data dissemination,” IEEE Trans. on Systems, Man and Cybernetics, Part A: Systems and Humans, vol. 36(3), May 2006, pp. 503-506.
28. D. Ulybyshev, B. Bhargava, M. Villarreal-Vasquez, D. Steiner, L. Li, J. Kobes, H. Halpin, R. Ranchal and A. Oqab-Alsalem. “Privacy-Preserving Data Dissemination in Untrusted Cloud.” IEEE CLOUD 2017, pp. 770-773