fog guru

TRAINING THE NEXT GENERATION OF EUROPEAN FOG COMPUTING EXPERTS

Automated application privacy compliance checking in fog environments.

Mozhdeh Farhadi*, Guillaume Pierre**, Daniele Miorandi***

*U-Hopper, Univ Rennes, Inria, CNRS, IRISA
**Univ Rennes, Inria, CNRS, IRISA
***U-Hopper

COMPAS 2021
8 July

The problem

All applications expose a privacy policy.

Do you trust them?

We don’t know if applications handle our personal data as they claim in their privacy policy.

Did you hear about Hikvision cameras security incident in Italy?

@ipvm

@amazon

Hikvision cameras are installed in sensitive locations in Italy:

- Military,
- Airports,
- Government units.

@istockphoto

May 2021

According to the study carried out by the Rai cybersecurity team:

“The cameras are sending data right to China.” [1].

[1] https://www.daily-sun.com/post/553867/Italian-State-News-Investigates-Hikvision.

- Smart assistant devices have access to our personal data:

  e.g.: users' voice.

- What happens if the device sends the user's voice when the user does not consent?
- Privacy violation reports [1,2].

[1]https://www.theguardian.com/technology/2019/apr/11/amazon-staff-listen-to-customers-alexa-recordings-report-says.

[2] https://www.cnbc.com/2019/07/11/google-admits-leaked-private-voice-conversations.html.

A study on the top 20 most popular Android health applications on Google Play showed that *every* studied application failed to comply with at least part of its privacy policy terms[1].

[1] M. Hatamian, « Engineering Privacy in Smartphone Apps: A Technical Guideline Catalog for App Developers », IEEE Access, vol. 8, 2020.

Our approach

- Manually checking the compliance of applications to their privacy policy is error-prone and time consuming.
- To automatically check the compliance of applications to their privacy policy.
- To use the application's host (Cloud or Fog) as a trusting third-party to perform the automatic verification.

Our approach

- The hosting platform has already access to a lot of useful information about the running applications:
  - the network traffic sent or received by the application,
  - the cpu usage of the application,
  - the memory usage of the application.

- We refer to the above properties of the application as *application-signal*.

- We monitor the application-signals unintrusively and utilize them for finding the application's privacy-oriented behaviors.

Our assumptions

- The application provider and the end-user trust the platform.
  - The platform runs the application correctly.
  - The platform will behave according to its privacy policy.
- The application is not actively trying to evade monitoring.
- The platform identifies correctly the privacy violations.

Our approach

- Take machine-readable privacy policy and extract privacy principles from it.
- Monitor application-signals.
- Infer the application's privacy-oriented behavior by analyzing application-signal.
- Compare the inferred application behavior with the privacy principles.

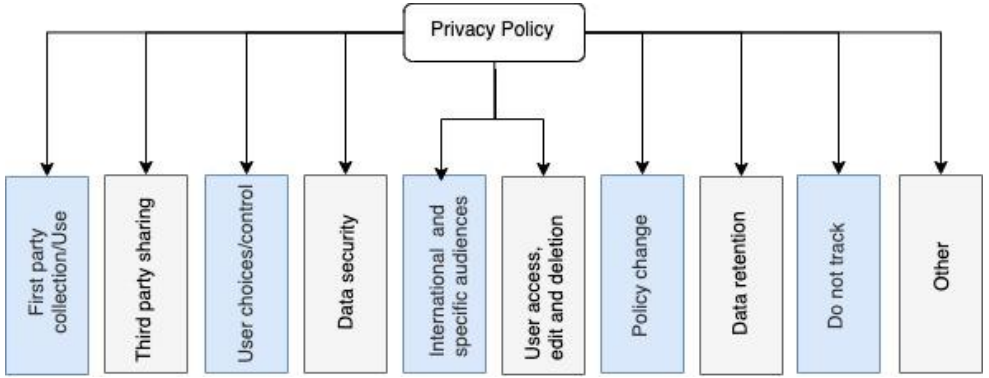
Our approach

- Take machine-readable privacy policy and extract privacy principles from it.
- Monitor application-signals. Network traffic.
- Infer the application's privacy-oriented behavior by analyzing application-signal.
- Compare the inferred application behavior with the privacy principles extracted from the privacy policy.

Related work: Privacy policy interpretation.

A taxonomy of privacy principles from privacy policies text [1].

Polisis extracts Privacy principles with 88.4% accuracy [2].



[1] S. Wilson and et al., “The creation and analysis of a website privacy policy corpus,” in Proc. ACL, 2016.

[2] H. Harkous, K. Fawaz, R. Lebret, F. Schaub, K. G. Shin, and K. Aberer, “Polisis: Automated analysis and presentation of privacy policies using deep learning,” in Proc. USENIX Security, 2018.

Related work: Automatic privacy compliance checking.

- Use of taint-analysis techniques to get the flow of sensitive data in Android [1], [2].
- These approaches are intrusive.
- The approach introduced in [1] has 14% of processing overhead.

[1] W. Enck and et al., “TaintDroid: an information-flow tracking system for realtime privacy monitoring on smartphones,” ACM Transactions on Computer Systems, vol. 32, no. 2, 2014.

[2] M.Hatamian,N.Momen,L.Fritsch,andK.Rannenberg,“A multilateral privacy impact analysis method for Android apps,” in Proc. Annual Privacy Forum, 2019.

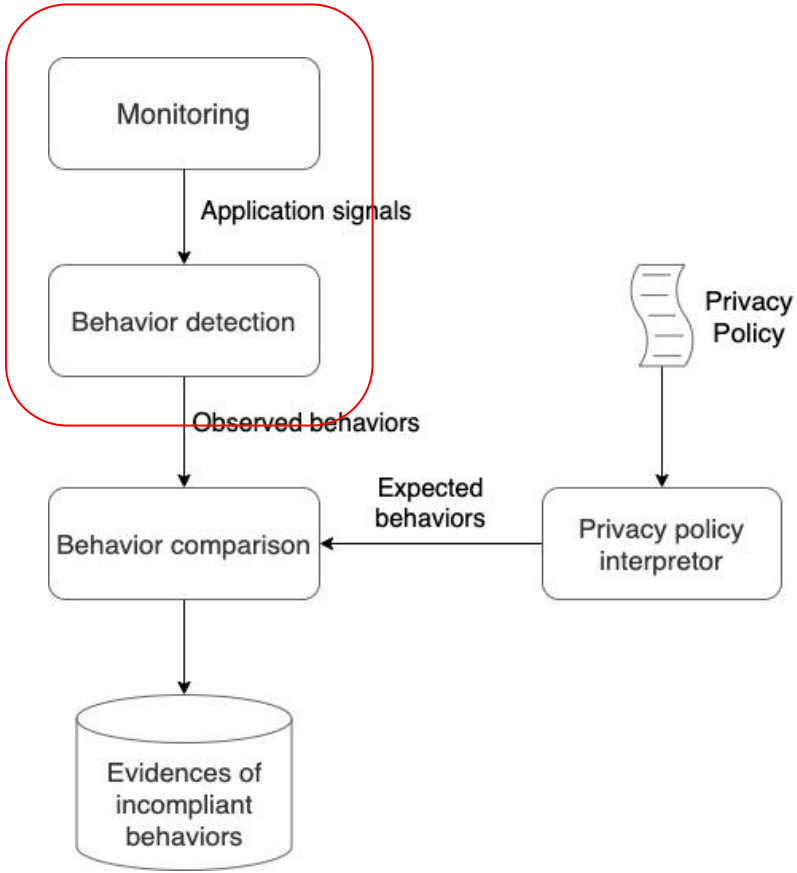
Related work: Network traffic classification.

- Deep packet inspection techniques:
  - Does not work on encrypted network traffic.
- Use of Machine learning for encrypted traffic classification [1,2].
  - The classes are coarse grained for our privacy purpose.

[1] G. Draper-Gil, A. H. Lashkari, M. S. I. Mamun, and A. A. Ghorbani, “Characterization of encrypted and VPN traffic using time-related features,” in Proc. ICISSP, 2016.

[2] M. Lotfollahi, M. J. Siavoshani, R. S. H. Zade, and M. Saberian, “Deep packet: A novel approach for encrypted traffic classification using deep learning,” Soft Computing, vol. 24, no. 3, 2020.

System model



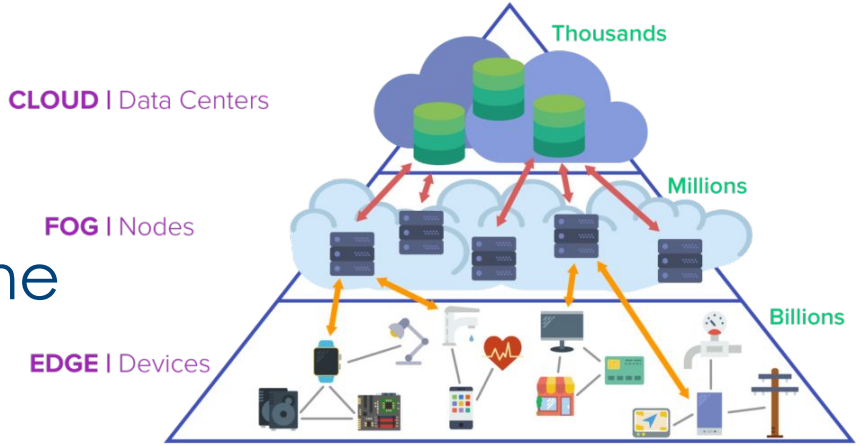
Proof of concept

- An important privacy-oriented behavior is sharing data with third parties.
- GDPR [1] mandates privacy policies to clearly define the third parties and the type of data shared with third-parties.
  The application shares data of type X with third party Y.
- Application signal: The network traffic.
- Environment: Fog computing environment.

[1] EU Parliament, "Regulation (EU) 2016/679 of the European Par- liament," 2016, https://eur- lex.europa.eu/legal- content/EN/TXT/?qid=1590072813969&uri=CELEX:32016R0679.

Fog computing

Enriches cloud with extra resources at the edge of the network [1, 2].

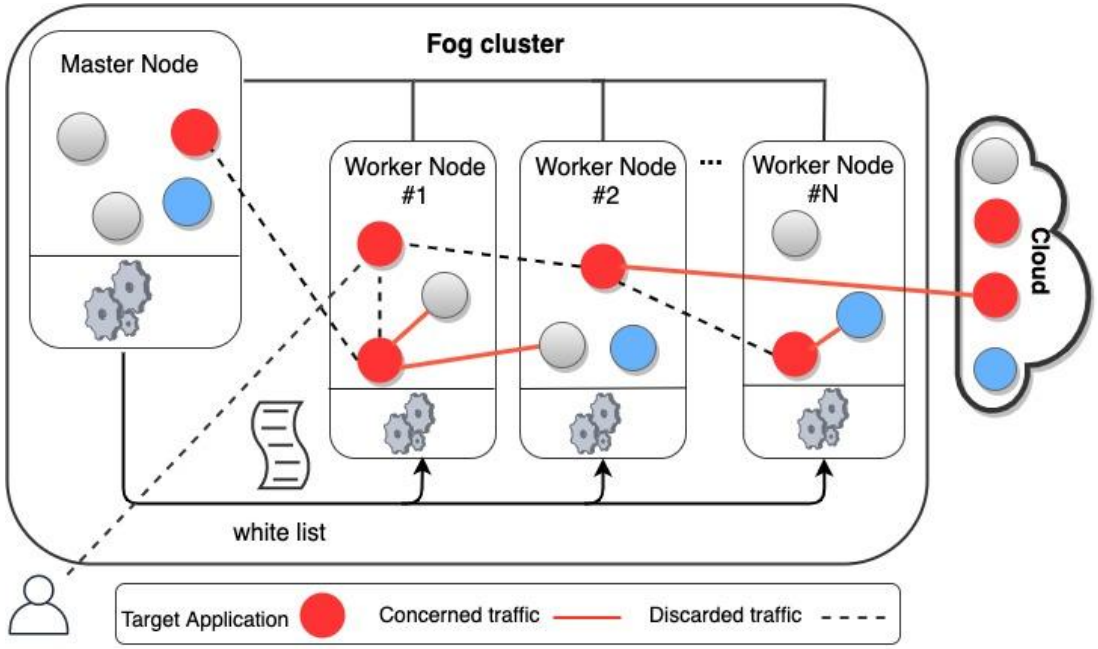


[1] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, « Fog computing and its role in the Internet of Things », in Proc. workshop on Mobile cloud computing, 2012.

[2] S. Pallewatta, V. Kostakos, and R. Buyya, « Microservices-based IoT application placement within heterogeneous and resource constrained fog computing environments », in Proc. IEEE/ACM International Conference on Utility and Cloud Computing, 2019.

Proof of concept

- Kubernetes as the orchestrator of the fog cluster.
- We need to distinguish between the Internal network traffic and the external network traffic of the application.



How to distinguish the type of shared data

- Utilize machine learning (ML) techniques.
- We need to train a classifier to be able to identify type of shared data according to our privacy need.

Created dataset

- Created a dataset by capturing network traffic of applications and labeled the type of captured data.
- We have shared the dataset [1].

Data type	Application Name
Audio	mplayer
	mpg123
	vlc player
Video	streamlinks
	youtube-dl + mplayer
	mpv
File	wget
	curl
	w3m
Other	firefox
	Slack
	Trello

M. Farhadi, “Application specific network traffic with specified activities,” Jul. 2020. [Online]. Available: https://doi.org/10.5281/zenodo.3965834.

The training of the ML model

- Performed 12000 experiments:
  - Decision tree classifier.

Feature Name	Description
prtcl	the most used protocol in the first 10 packets
deltatime_mean_First10	mean value of the delta time between packets in the first 10 packets
deltatime_std_First10	standard deviation of the delta time between packets in the first 10 packets
pcklength_mean_First10	mean value of the packets length for the first 10 packets
pcklength_std_First10	standard deviation of the packets length for the first 10 packets
deltatime_mean_Next200	mean value of the delta time between packets in the next 200 packets
deltatime_std_Next200	standard deviation of the delta time between packets in the next 200 packets
pcklength_mean_Next200	mean value of the packets length for the next 200 packets
pcklength_std_Next200	standard deviation of the packets length for the next 200 packets

Evaluations

- When the system has seen samples of the application:
  - 86% F1-score.

- When the system has not seen samples of the application:
  - 84% F1-score.

Conclusion

- Showed the feasibility of automatic privacy compliance checking.
- Utilized unintrusive and application-agnostic monitoring in the platform layer.
- Reached to F1 of %86 for identifying the type of shared data with third-parties.

Thank you

mojde.farhadi@gmail.com