2019 IEEE International Congress on Big Data (BigData Congress)

Context-Aware Enforcement of Privacy Policies in Edge Computing

Clemens Lachner
Distributed Systems Group
TU Wien
Vienna, Austria
c.lachner@dsg.tuwien.ac.at

Thomas Rausch
Distributed Systems Group
TU Wien
Vienna, Austria
t.rausch@dsg.tuwien.ac.at

Schahram Dustdar
Distributed Systems Group
TU Wien
Vienna, Austria
dustdar@dsg.tuwien.ac.at

***Abstract*—Privacy is a fundamental concern that confronts systems dealing with sensitive data. The lack of robust solutions for defining and enforcing privacy measures continues to hinder the general acceptance and adoption of these systems. Edge computing has been recognized as a key enabler for privacy enhanced applications, and has opened new opportunities. In this paper, we propose a novel privacy model based on context-aware edge computing. Our model leverages the context of data to make decisions about how these data need to be processed and managed to achieve privacy. Based on a scenario from the eHealth domain, we show how our generalized model can be used to implement and enact complex domain-specific privacy policies. We illustrate our approach by constructing real world use cases involving a mobile Electronic Health Record that interacts with, and in different environments.**

***Keywords*-privacy; edge computing; e-health; context awareness;**

I. Introduction

Cloud computing and the continued centralization of computation and data management has caused growing concern about data privacy [1]. Releasing data to centralized services is especially problematic for systems that handle sensitive data, such as patient data in eHealth systems, as this loss of control can hinder data management workflows from complying to privacy policies [2] or security provisions such as HIPPA (Health Insurance Portability and Accountability Act). Edge computing has been recognized as a key technology for enabling privacy-aware Internet of Things (IoT) applications. However, the complexity inherent to edge computing architectures makes it extremely difficult for application developers to implement mechanisms that can guarantee privacy policy compliance, especially in complex domains with high amounts of stakeholders, such as eHealth. Current research falls short of providing concrete frameworks and solutions for modeling privacy constraints and enacting data processing rules to meet privacy requirements [3]. Data confidentiality, data integrity and data privacy are the key concepts to meet those requirements. To avoid information leakage, strict access policies must ensure the confidentiality and integrity of private data, as well as handling data locality. This is commonly realized by defining roles according to different stakeholders of a system, typically enforced by Role Based Access Control (RBAC) techniques [4]. However, further data privacy agreements (e.g., data exchange between stakeholders), that should also be adequately defined, established and implemented via privacy policies, introduce additional architectural, conceptual, and performance related challenges. In this paper, we present a novel model for defining and enacting privacy policies based on context-aware edge computing. By leveraging context-awareness of edge computers, we enable runtime decision making for applications on how to enforce privacy policies during data workflows. Compared to existing approaches, which are mostly tailored to a specific use case or domain [5], [6], [7], [8], our model focuses on context-awareness and corresponding actions edge devices have to take. We thereby enable flexibility in implementation, and aid system architects or developers in designing and building decentralized systems that can make use of privacy-sensitive data, while complying to complex privacy policies of a given domain. As part of our privacy model, we define privacy levels to incorporate a finer formal description granularity. We demonstrate our approach based on a scenario from the eHealth domain, where privacy is considered a critical requirement. The remainder of this paper is structured as follows. In Section II we present a motivational scenario from the eHealth domain. Section III gives an overview of related work. In Section IV we present our privacy model, and how context-awareness is factored into this model. In Section V we discuss the enforcement of privacy policies by our model based on context-aware edge computing. Finally, Section VI concludes the paper and gives an outlook on future work.

II. Motivational Scenario

The National Committee for Vital and Health Statistics (NCVHS), a key advisory committee to the US Department of Health and Human Services, defines privacy in the context of health information as *"an individual's right to control the acquisition, uses, or disclosures of his or her identifiable health data"* [9]. Hence, granting patients control over their medical records, even if the data are owned by another party (as is common in electronic health records) is fundamental for enabling privacy. The

978-1-7281-2772-9/19/$31.00 ©2019 IEEE
DOI 10.1109/BigDataCongress.2019.00014

concept of mHealth [10] can facilitate this by leveraging mobile platforms to monitor, process, and store medical data in so called mobile electronic health records (mEHR). However, this decentralization increases complexity of data management workflows, in particular when privacy policies need to be enacted by the system automatically at runtime. We consider a typical scenario from the medical domain, where a patient is going into a hospital seeking medical consultation and is subsequently examined by a physician. The patient has a mEHR as an application installed on their smartphone. This record comprises personal data fields such as name, address, or gender; and medical data fields arranged in sections such as diagnoses, prescriptions or therapies. Multiple other stakeholders are also involved in further steps in this scenario. For example, consider a pharmacist who hands out prescribed medication, or a biotechnological specialist (who is, e.g., consulted for specialized artificial implants). These stakeholders can also be other machines such as a drug dispensers or smart medical imaging devices. Furthermore, in this scenario, diagnoses can also be done remotely, which requires a system to be interconnected across different networks. These examples all require the sharing or processing of sensitive data in different contexts between different stakeholders. A system that supports this type of complex scenario requires not only standard role-based access control mechanisms to ensure privacy, but has to *react and adapt* to different changes in environmental context. Such reactions are operations or processes that ensure privacy of data is preserved and carried out in a way that conforms to a corresponding policy.

III. Related Work

Securing the integrity and sharing of information is a critical requirement in complex distributed systems that deal with sensitive data. Data encryption is a common tool to hinder an unintended user to infer information of stored or transferred data. An overview of different well researched encryption approaches and techniques is presented in [11]. To protect data from alteration, modification or deletion in a distributed system several data integrity strategies have been developed and established. These include Provable Data Possession (PDP) [12], Proof of Retrievability (PoR) [13] and Third Party Auditing (TPA) [14]. Nowadays, interconnected edge devices like smart phones, sensors, radio-frequency identification (RFID) tags, or smart home devices are producing a huge amount of data. As these devices become more and more integrated into our daily life, they significantly affect and change our way of living, social behavior and life style [3]. These data are then used to generate context-aware information (e.g., tracking the commute duration from a person's home to their work location) [15]. In this paper we focus on data privacy per se, which deals with safeguarding personal information. For instance, patients private data, diagnoses or therapy plans may be misused by, e.g., insurance companies to adapt rates, and must therefore be well protected. Enforcing privacy in software systems has mostly been addressed by incorporating RBAC techniques. Standards like XACML [16] can be used to define policies and handle access of data but do not take different environmental contexts into account nor describe how and where these policies should be enabled. Furthermore, most of those mechanisms are based on a centralized architecture with a complex constellation of roles and sensitive data is often duplicated or distributed. Our model describes how such well established RBAC approaches could be extended, by incorporating edge computing techniques, to facilitate the development of decentralized privacy preserving systems.

IV. Privacy Model in Edge Computing

The privacy model we propose describes the circumstances under which an entity is allowed to access specific parts of sensitive data. Defining a consistent model enables a coherent definition of policies that ensure privacy of data, also handling data locality. Our privacy model combines and correlates certain levels of privacy (e.g., visibility constraints on specific data sets) with a given context. The determination of a specific context is best handled closest to the according environment. Therefore, edge computing is well suited for this task, where every edge device is exposed to a certain and specific context.

A. Privacy Policies

A common way to enable and enforce privacy in edge computing is to define policies that specify the handling of sensitive data [17], [18]. One use case may be the need for policies to correctly handle the collection, exchange and disclosure of patient data (e.g., in medical consultation scenarios). Retaining data quality and accessibility required for medical processing while respecting privacy aspects of all involved persons is one of the key challenges when defining privacy policies in eHealth. A simple concrete implementation of such a policy could describe which data fields of a given set of personal information should be persisted and which data fields should be handled transiently. The downside of such policies is that they are often tailored to specific use cases and therefore lack flexibility and generality. In this paper we distinguish between two different types of policies. First, there are policies which define the data persistence modalities of private data, and second, policies which describe how and where the data is modified (e.g., anonymized or encrypted) for transport and inspection or further computation. Therefore, we distinguish between logical and physical privacy boundaries. While logical boundaries deal with role-based constraints, legal constraints, etc., physical boundaries comprise location-based constraints, network constraints, or other involved devices.

After policies are defined, only a subset of those need to

be deployed and stored on relevant edge devices, based on the assumption that not every policy is applicable in every context.

B. Privacy Levels

To incorporate more granularity into policies, we suggest to define different levels of privacy. Regarding our example scenario, we divide privacy related data into (i) domain specific data like anamnesis data, medical diagnoses, pharmaceutical prescriptions, etc., and (ii) personal data like full name, address, and other data of personal domain. An example of a privacy policy in the eHealth domain could force a device or system to decide between visibility levels of patient data. One level could define that all data is visible (e.g., for personal usage), while a physician or pharmacist should only be allowed to access specific data sections, like open prescriptions or medical diagnosis. If a physician seeks consultation, another level could state that all personal information is invisible and only medical data is visible. At last, the most restrictive level would be that all data is invisible to anybody, e.g., being encrypted for transmission. Our proposed model presupposes that for every different level contextual information is incorporated. Furthermore, these levels should be defined at a more granular level depending on given system dependencies like domain specific, political, legal or architectural constraints and operate either on holistic data sets, data sections or single data fields.

C. Context-Aware Decision Making

To achieve a higher degree of flexibility in implementing privacy policies we propose a context-aware decision making process to dynamically adapt to changing environment situations at the edge. Context in computer science can be interpreted in many different ways. In the focus of this paper, we use the term *context* as environmental information recognizable by edge devices. This could be information about the network and its topology, connected devices, spatial information, proximity, location or time. A context-aware system is able to interpret changes in the environment and react to them in a predefined manner. One way of telling the system how to react to certain context changes is by defining previously mentioned policies and enhance them with contextual parameters. Regarding privacy, such context-aware systems are on the one hand able to anticipate potential risks and provide recommendations to, e.g., a user which actions to take, and on the other hand automate certain adaptations in data management and processing. However, in real world scenarios such systems are not provided with holistic environment information all the time, and sometimes they have to make decisions based on incomplete data. This has to be taken into account when implementing privacy policies following either a conservative strategy, i.e., disable controls or hide sensitive data on a user interface, or an optimistic decision making strategy like allowing read access

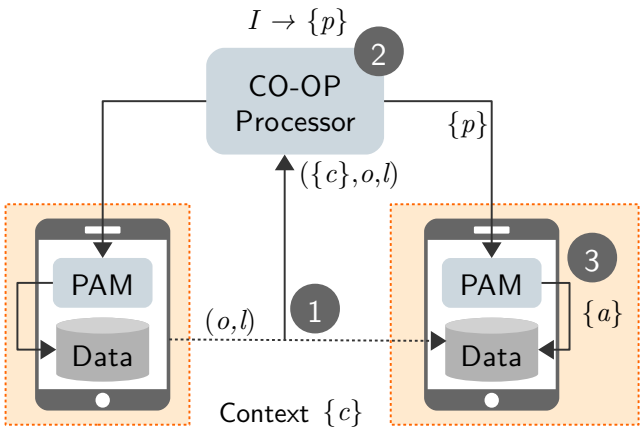

Figure 1. Interaction of model components and edge devices operating in different contexts

on partly obfuscated data. A more sophisticated approach could be to implement machine learning techniques, that automatically infer future decisions based on historical information.

V. Context-Aware Policy Enforcement on the Edge

By sticking to our motivating example, we illustrate how context-aware privacy policies are defined and describe the corresponding decision processes that enable/enforce those given a certain context using our model. The model comprises three essential encapsulated parts that work together in a coherent way but can also be separately implemented (e.g., using existing standards) if need be. Part one describes the process how to define context-aware privacy policies for a certain system and how to describe them using our privacy model. The second part defines an inference mechanism to obtain one or more policies (depending on the policy definitions granularity grade) and the third part describes a mechanism that determines the resulting actions after one or more policies were identified.
Figure 1 illustrates the general principle of our model and how the distinct components work together in three basic steps. First, a device requests access to (private) data on another device. Context, Operation and Privacy Levels are transmitted to a Context Operations (CO-OP) Processor. Second, the CO-OP Processor returns inferred policies to the Policy Application Manager (PAM) implemented on each device. Third, the PAM triggers certain actions performed on corresponding data records to enforce the policies returned by the CO-OP Processor. Each of those elements will be discussed in detail in the following.

A. Policy Definition

Policies should be defined at an early stage during the system design phase, and comprise fine-grained domains-

Policy	Description
p_1	A physician (c) may read (o) the personal section (l) of an mEHR.
$p_{1.1}$	A physician (c) may not persist (o) personal data (l) from the mEHR.
$p_{1.2}$	A physician (c) may create, read, update or delete (o) the diagnoses and prescription part (l) of the mEHR.
$p_{1.3}$	A pharmacist (c) may read (o) the prescription section (l) of an mEHR.
p_2	Drug dispensers (c) may read (o) the prescription section (l) of the mEHR only if the user is in close proximity.
p_3	Data access (o) on the medical section (l) is allowed only after explicit user permission if a connection is established across different networks (c).
p_4	Data access (o) on the medical section (l) is allowed only if the involved devices (c_1) are located in the same geofence (c_2).
p_5	After the creation (c_1) of a prescription section (l) of the mEHR, there is a predefined time window (c_2) for a device of type drug dispenser (c_3) to access (o) this section (l).

Table I
EXAMPLE OF DATA PRIVACY POLICIES OF AN EHEALTH APPLICATION BASED ON DOMAIN-SPECIFIC CONTEXT PARAMETERS, DATA MANAGEMENT OPERATIONS, AND PRIVACY LEVELS

specific parameters as it reduces ambiguity in implementation details. The structural elements of our model are i) privacy levels, ii) data management operations, and iii) context. The first thing to ascertain is the characteristic of either logical or physical types of different contexts in the system. A typical example for a logical form of context is the *role* of a user or device. A physical form of context could be the current *location* of a user or the proximity of a device to an edge node. We describe these types of contexts in more detail in Section V-C. After identifying different types of contexts, their manifestations have to be mapped to certain data management operations (like classical CRUD operations, or persistence constraints) that respect predefined privacy levels.

Table V-A shows exemplary privacy policies defined in the context of our motivational scenario. These policies are based on typical use cases in the eHealth domain. In the textual descriptions of the policy, we highlight the structural elements as defined by our model. A policy defines one or more contexts c_i, a privacy level l, and a data operation o.

B. CO-OP Processor

After several privacy policies are defined, an edge device needs to know if and when a certain policy has to be applied. This task is implemented by a Context-Operation Processor (CO-OP Processor) which infers one ore more policies given domain-specific context parameters, privacy levels, and a set of data operations. The corresponding inference function I can be formally expressed as:

$$I : C^n \times O \times L \rightarrow P^m \tag{1}$$

where C is the set of domain-specific context parameters $\{c_1, c_2, ...\}$, and O is a specific operations (such as CRUD or persist), and L is the corresponding privacy level on which should be operated on. The output is one or more policies $p \in P$ that need to be enacted. The CO-OP Processor should be implemented directly on an edge device.

C. Policy Application Manager

The determination of a certain policy by the CO-OP Processor implies specific actions for a device to be executed. The Policy Application Manager takes one or more such policies $p \in P$ as an argument and returns one or more actions $a \in A$ that have to be taken to enact the privacy policy. This function A_f can be formally defined as:

$$A_f : P^n \rightarrow A^m \tag{2}$$

where P is the set of policies determined by I, and A is the set of possible actions.

Returned actions include basic create, read, update or delete (CRUD) operations or persistence tasks as well as more complex actions like triggering another decision process delegated to the CO-OP processor or transferring, decrypting, or further computing data. Formally described, an action operates within the privacy model of a given policy accomplishing the enforcement of its purpose. The PAM should be implemented on a per device basis or on edge nodes, because triggered actions may differ between, e.g., device types. Delegating actions to the corresponding devices is also performed by the PAM. The following scenarios illustrate the practical application of our model by covering several examples of context manifestations, data management tasks the resulting inference of corresponding privacy policies and their practical application.

1) Role Based Context: To perform specific tasks on a mEHR, policies are defined which describe *who* is allowed to perform these data management tasks. Our predefined policy P1.2 state that *a physician* is allowed to create, update or delete diagnoses and prescription parts of the mEHR and P1 that he is only allowed to read personal data fields. At a special point during the consultation process a physician (e.g., via their PC) wants to establish a connection to the patient's smartphone to access the mEHR. He sends a request to the CO-OP processor at the edge, providing the context in form of his *role*, the operation, in our case *read*, as well as the privacy level *personal and diagnoses section* of the mEHR. The CO-OP processor returns policy ID P1, P1.1 and P1.2 which will then be further processed by the Policy Application Manager. Because of P1.1 personal data won't be persisted on the physicians PC, therefore treated as transient data as long as the connection is established (if not stated otherwise by a policy). On the other hand the mEHR of the patient registers a connection request being made by *a physician* and therefore also requests policies from the corresponding edge node. It then applies the corresponding policy which allows the connection to be established and the physician to access the patients data. On the contrary,

if a pharmacist wanted to do the same, no defined policy would allow him to do so (P1.3 only allows read access on the prescription section). This RBAC strategy is the foundation of our model. Exhaustive research have been conducted on this topic and could therefore facilitate the base implementation of our model.

2) Proximity Related Context: Sticking to our motivating example, we assume the physician prescribed the patient a certain medicament. The patient then connects his mEHR to a special drug dispenser [19]. The patients mEHR application recognizes the drug dispenser as a *special device type* and infers a corresponding policy (P2) which allows any device of this certain type to read the prescription data part of the mEHR. Furthermore, the device running the mEHR has to be in close proximity to the drug dispenser. While the a specific device could also be modeled as a role, proximity has to be *sensed* and processed by an edge device near by. If both constraints are satisfied, the drug dispenser then reads the prescription related data fields and hands out the patient his medicine.

3) Network Related Context: Another privacy policy (P3) defines that incoming connections are only handled automatically if the connecting device (e.g., physicians PC) is located in the same network as the device with the mEHR application installed. As long as this is the case, resulting actions are based on decisions made by defined policies as described above. However, if for instance the patient is at home and a physician tries to access the mEHR from the hospitals network, an edge node recognizes that the mEHR is not connected to its network and therefore sends a notification on the user stating that someone or something is trying to access his mEHR. Via permission dialog or a similar GUI mechanism the patient is then able to allow or deny the incoming connection and processing of data.

4) Location Based Context: Determining the physical location of the patient's device is a common tasks in mobile computing and especially on smartphones. Certain policies could be enforced based on the patients current location. As an example we extend our use case and refer to policy P4, that enforces the *presence* (e.g., at least inside the hospital) of the patient near a specific device, similar to proximity based context. This could be realized by defining so called geofences, which describe a virtual perimeter based on GPS data [20]. This could become relevant for e.g., management tasks like dynamic bed allocation performed by nursing personnel, if patients are not restricted to stay in their room or hospital permanently.

5) State Based Context: The last example of context-aware decision making uses an inferred *state* based on previous actions that were executed on the patients mEHR. For instance it is reasonable that after the physician prescribes the patient a medicine, the next logical step for the patient would be to get to the drug dispenser and collect his prescribed medication. One of our predefined policies (P5) defines, that after the prescription part of a mEHR is altered there is a (for example 72h) time window for the patients smartphone to establish a connection to a drug dispenser. However, those state based context processing could potentially lead to a system incorporating many exception conditions.

VI. Conclusion and Outlook

The Internet of Things (IoT) with all its edge devices generate, process and store a huge amount of data. A lot of these data include privacy sensitive information and can be used to infer specific user behavior patterns or, in worst case scenarios, compromise a user. Holistic approaches are facing architectural, conceptual, as well as performance related challenges. Therefore, we propose a model leveraging edge computing techniques where sensitive data flow is handled closer to the user, because ensuring privacy is not just a matter of authentication and authorization but a more complex task which should take the environmental context in which data is managed into account. Armed with a variety of powerful sensors that are considered to recognize relevant environmental information, these edge nodes take away workload from traditional centralized, cloud based approaches while also aiding sensible data locality tasks. Our model can aid architects and developers to identify these contexts a system is confronted with. By defining policies at an early point in system design, privacy concerns can be mitigated or even eradicated. Involved Edge Devices must be able to *sense* a certain context and developers must implement the corresponding inference functions to enforce a certain policy. Hence, privacy policies have to be tied to several contexts in form of fine grained definitions. We suggest that this can be achieved by enriching policies based on well established RBAC features with contextual information. Therefore, the enforcement of such policies is not limited to be executed only after specific requests of a device to a corresponding edge node. Edge devices, being aware of the context, could anticipate potential risks beforehand and automate certain adaptions in data management and processing. Part of our future work will be further investigation in the field of context-aware privacy enforcement and prototypical implementations of policy enforcing techniques for different kinds of edge devices.

REFERENCES

[1] W. Shi and S. Dustdar, "The promise of edge computing," *Computer*, vol. 49, no. 5, pp. 78–81, 2016.

[2] N. Dong, H. Jonker, and J. Pang, "Challenges in ehealth: From enabling to enforcing privacy," in *International Symposium on Foundations of Health Informatics Engineering and Systems*. Springer, 2011, pp. 195–206.

[3] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet of Things Journal*, vol. 3, no. 5, pp. 637–646, 2016.

[4] D. Ferraiolo, J. Cugini, and D. R. Kuhn, "Role-based access control (rbac): Features and motivations," in *Proceedings of 11th annual computer security application conference*, 1995, pp. 241–48.

[5] Y. Xiao, X. Shen, B. Sun, and L. Cai, "Security and privacy in rfid and applications in telemedicine," *IEEE communications magazine*, vol. 44, no. 4, pp. 64–72, 2006.

[6] D. Kotz, S. Avancha, and A. Baxi, "A privacy framework for mobile health and home-care systems," in *Proceedings of the first ACM workshop on Security and privacy in medical and home-care systems*. ACM, 2009, pp. 1–12.

[7] S. Chakraborty, K. R. Raghavan, M. P. Johnson, and M. B. Srivastava, "A framework for context-aware privacy of sensor data on mobile systems," in *Proceedings of the 14th Workshop on Mobile Computing Systems and Applications*. ACM, 2013, p. 11.

[8] D. C. Klonoff, "Fog computing and edge computing architectures for processing data from diabetes devices connected to the medical internet of things," 2017.

[9] S. Cohn, "Privacy and confidentiality in the nationwide health information network," National Committee on Vital and Health Statistic http://www.ncvhs.hhs.gov/060622lt.htm, 2006, (Accessed 2018-03-26).

[10] M. Kay, J. Santos, and M. Takane, "mhealth: New horizons for health through mobile technologies," *World Health Organization*, vol. 64, no. 7, pp. 66–71, 2011.

[11] U. Arjun and S. Vinay, "A short review on data security and privacy issues in cloud computing," in *Current Trends in Advanced Computing (ICCTAC), IEEE International Conference on*. IEEE, 2016, pp. 1–5.

[12] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable data possession at untrusted stores," in *Proceedings of the 14th ACM conference on Computer and communications security*. Acm, 2007, pp. 598–609.

[13] A. Juels and B. S. Kaliski Jr, "Pors: Proofs of retrievability for large files," in *Proceedings of the 14th ACM conference on Computer and communications security*. Acm, 2007, pp. 584–597.

[14] C. Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-preserving public auditing for data storage security in cloud computing," in *Infocom, 2010 proceedings ieee*. Ieee, 2010, pp. 1–9.

[15] F. Schaub, B. Könings, and M. Weber, "Context-adaptive privacy: Leveraging context awareness to support privacy decision making," *IEEE Pervasive Computing*, vol. 14, no. 1, pp. 34–43, 2015.

[16] X. T. C. Organization for the Advancement of Structured Information Standards, "Xacml," National Committee on Vital and Health Statistic https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xacml, (Accessed 2018-03-26).

[17] J. Yang, K. Yessenov, and A. Solar-Lezama, "A language for automatically enforcing privacy policies," in *ACM SIGPLAN Notices*, vol. 47, no. 1. ACM, 2012, pp. 85–96.

[18] K. Ravichandran, A. Gavrilovska, and S. Pande, "Pimico: Privacy preservation via migration in collaborative mobile clouds," in *System Sciences (HICSS), 2015 48th Hawaii International Conference on*. IEEE, 2015, pp. 5341–5351.

[19] P. Michel, "Medication dispensing device," Jan. 24 1995, uS Patent 5,383,865.

[20] D. Namiot and M. Sneps-Sneppe, "Geofence and network proximity," in *Internet of Things, Smart Spaces, and Next Generation Networking*. Springer, 2013, pp. 117–127.