DOI: 10.15439/2015F282
ACSIS, Vol. 5

Secure service interaction for collaborative business processes in the inter-cloud

Björn Schwarzbach, Michael Glöckner,
Alexander Pirogov, Martin Max Röhling
Leipzig University
Grimmaische Straße 12, 04109 Leipzig, Germany
Email: {schwarzbach, gloeckner, pirogov,
roehling}@wifa.uni-leipzig.de

Bogdan Franczyk
Leipzig University
Grimmaische Straße 12, 04109 Leipzig, Germany
Uniwersytet Ekonomiczny we Wroclawiu
ul. Komandorska 118/120, 53-345 Wroclaw
Email: franczyk@wifa.uni-leipzig.de

***Abstract*—The emergence of a closer relationship between cloud service providers in the cloud computing market is the inevitable consequence of the computing as utility concept. The closer cooperation creates competitive advantages for providers and users of cloud services as well. Capacities and services can be used in a collaborative and flexible way. Despite the numerous potentials of composite cloud services, trust, policy and privacy are the major challenges resulting from the distributed and flexible data handling. The paper derives requirements and solutions in the field of inter-cloud service communication with a special focus on security. The proposed architecture is evaluated with a sample collaborative business process of inter-cloud service interaction.**

I. Introduction

IN 2008 vice president of Gartner Research Thomas J. Bittman published his thoughts on future development of Cloud Computing (CC). In the early monolitic phase, cloud services were built on proprietary architectures of dominant Cloud Service Provider (CSP), such as Google, Salesforce or Microsoft. The second phase vertical supply chain distinguishes itself by the development of first ecosystems of smaller companies within the CC market. New CSPs use proprietary Cloud platforms of dominant providers, i.e. Google App Engine or Microsoft Azure, in order to provide their own services. In the last phase, smaller providers unite to a horizontal federation. That way, the union increases earnings by expanding their capacity and reducing costs through more efficient resource allocation. In parallel, open interoperability standards of service communication in intercloud-environment are developed [1].

The creation of a closer relationship between CSPs on the CC market is a inevitable extension of the computing-as-utility concept, which is about providing computing resources as a service over the Internet. Users of cloud based services may benefit in terms of cost reduction by renting their own distributed, virtualalized IT-infrastructure, improving service robustness and preventing provider dependence by means of interoperability standards [2], [3], [4].

A close cooperation creates certain advantages in competition for providers of cloud services. By using other CSPs' capacities, providers may deliver their products and services even faster and more effective to their clients. Further, making use of virtualization technology reduces costs for a flexible and customizable IT-infrastructure. Its dynamic and smooth scaling has a positive effect on service deployment time. Due to dynamic outsourcing of computational services, power consumption costs for computer centres can be significally reduced [3], [5].

Beside several advantages of collaborative cloud services, cloud specific issues concerning the security of service communication in an intercloud environment still exist. Trust is an essential precondition in order to create an intercloud federation. Without trust, security of cloud-interactions can't be guaranteed. Policy is another issue concerning intercloud interactions. It is essential to have effective control mechanisms so potential policy clashes, which would affect the safety of the whole system, are detected and removed. Identity and data privacy are other challenges to intercloud communication since users of cloud services transfer their personal data to the CSPs. Appropriate tools for access and identity management are essential for data protection [2], [6], [5], [7].

The paper focuses on the creation of an intercloud architecture, which enables secure service communication in collaborative business processes. The second chapter consists of an overview of relevant theoretical concepts while the next chapter introduces and explains a collobarative process of payment transaction. Communication models of services in the Intercloud environment are analyzed in chapter four. Further, chapter five introduces a draft architecture and suggestions for its implementation. Finally, the conclusion completes the paper.

II. Theoretical background

The following chapter deals with the theoretical basis of service communication in the intercloud environment. Special attention is paid to security aspects of service interactions in collaborative business processes. Table I gives a resume of criteria for safe service communication that will be elaborated in this chapter.

The work presented in this paper was funded by the German Federal Ministry of Education and Research under the projects PREsTiGE (BMBF 16KIS0082K) and LSEM (BMBF 03IPT504X).

TABLE I
CRITERIA FOR SECURE COMMUNICATION IN THE INTERCLOUD

Criterion	**Challenge**	**Solution**
Interoperability	Diverse, partly proprietary communication protocols	Broker; standardized interfaces; hybrid approach
Robustness	Single point of failure; workload balancing	Distributed implementation of the IT infrastructure
Optimal service provisioning, service selection, and service allocation	Orchestration of services and dependency resolution in real-time; automated selection of QoS-Criteria	Central platform that fulfills the orchestration in the intercloud; mechanisms for conflict resolution in case of conflicting QoS policies
Access & identity management	Management of multiple accounts of service users and CSPs; Establishment of a secure trust context for service interaction	Outsourcing of credential management to third parties; digital identities; Identity federation with the use of protocols such as Security Assertion Markup Language (SAML), eXtensible Access Control Markup Language (XACML), OpenID, OAuth, WebID, and SSO; secure authentication methods; multi factor authentification
Trust	Establishment of a secure trust context; dynamic determination of trust; de-perimeterization	PKI, XACML and SAML based communication protocols; reputation based, dynamic trust index
Policy	Inconsistency; inefficiency; semantic interoperability; static, predetermined SLAs, that hinder the ad-hoc service interaction	Mechanisms for the combination of policies on cloud federation level; monitoring and conflict resolution; dynamically, automatically, and instantaneously created federation-level agreements
Privacy	data privacy; identity privacy	Encryption; anonymization; pseudonymization

A. Intercloud environment

The term *intercloud* is not standardized in scientific literature. Though terms like *cross-cloud* [5], *multicloud* [7] or *cloud-federation* [2] can be found, but summarizing author-specific definitions do not differ basically.

Our work is based on the definition of the Global Inter-Cloud Technology Forum: "A cloud model that, for the purpose of guaranteeing service quality, such as the performance and availability of each service, allows on-demand reassignment of resources and transfer of workload through an interworking of cloud systems of different cloud providers based on coordination of each consumer's requirements for service quality with each providers SLA and use of standard interfaces." [8]

If not explicitly stated different, our work uses the term *intercloud*. In the authors' opinion intercloud is a better definition for a multicloud environment since it deals with a highly integrated environment where service communication is structured by the use of coordinating instances. Fig. 1 describes a typical intercloud environment with different types of Clouds. Closer cooperation of CSPs enables the usage of different strategies for resource consumption like outsourcing and cloud bursting. Service users are either the end consumers or other CSPs as well.

By combining different CSPs' services, the problem of being dependent on one provider, so called lock in effect, is solved in the intercloud environment. Moreover, flexibility as well as scalability and robustness of the whole system can be improved, because all intercloud CSPs are able to provide identical services. Further, energy can be used more effectively [3], [4].

The heterogeneity of this environment is a special challenge for technical implementation on all levels of intercloud architectures. The cloud-spanning integration of services emphasizes the importance of availability and access speed of services. Interoperability of interclouds is reduced by proprietary interfaces that are needed for service interaction.

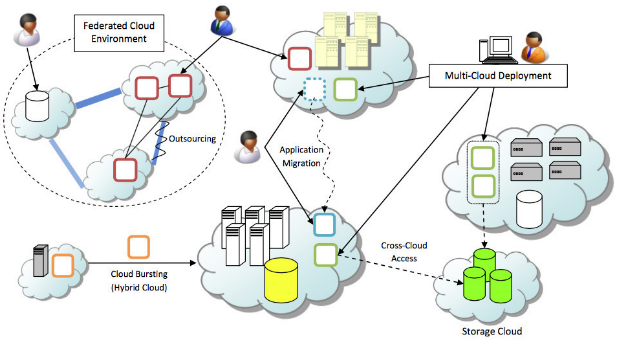

Fig. 1. Cloud interoperability [7]

In addition, the current service delivery model is incompatible with an open and dynamic collaboration, so using customer-specific tools leads the cloud interoperability to a vendor lock-in.

Due to limited access rights of unauthorized users and the bundling of provided services with other resources of the same provider, personalized service customization and cloud-spanning service composition are affected [6], [5], [4].

A secure collaboration of a multitude of CSPs in a heterogeneous environment is only enabled by complex intercloud architectures that meet several requirements. Fig. 2 provides an overview of intercloud architecture challenges. Several ideas of Toosi haven been applied [7]. This paper focuses mainly on the highlighted security relevant aspects.

From a technical point of view, interoperability can only be accomplished through a broker, which is in control of all communication between the CSPs, or via standardized interfaces. It's possible to use a hybrid of both ideas, depending on whether it is economically and technically useful. The

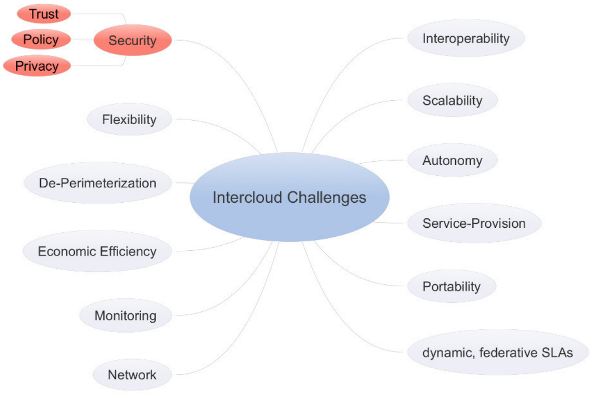

Fig. 2. Challenges of intercloud architectures

last option consists of an adaption of the architecture by all participants, such as the CSPs and the service users. A better option could be the introduction of a central platform, which would reduce the need for adaptation when joining the intercloud federation and which would provide a uniform interoperable vision of service communication.

B. *Cloud based collaborative business process*

A collaborative business process is a dynamic process, run by several involved instances. All steps are conducted locally by interacting work-flow engines, following a common service definition. The engines of the interacting process partners are staying in control of all of their subprocesses [9].

C. *Trust*

Due to de-perimeterization, trust is a very important factor in the intercloud environment. Since resources are outsourced to the CSPs, service users lose control over their personal data and that's why CSPs need to be trustworthy and service consumers need to be equipped with effective control mechanisms [6].

The transitive trust is essential when giving authorization to a third party. First, mechanisms of delegation have to allow a dynamic creation of service level agreements (SLA) in order to deny access to unauthorized user. Second, third parties must not modify service requests arbitrarily. Third, delegation has a time limit so service user representatives may only act in a certain time period legitimately. Another intercloud communication problem is the creation of a trust context between the interacting CSPs. Current intercloud trust models are based on a public-key-infrastructure-system (PKI), which judges the trustworthiness of entities in an absolute way and is therefore not suitable for a dynamic certification. The reputation-based trust complements the trust-context creation with dynamic aspects. When using parameters of reputation, utilizing relatively trustworthy computing resources in the intercloud is given, if they match with practical experience values of participating CSPs. Hence, it is possible to create a real trust federation along with third parties [10], [7].

Federated identity management is very important in the intercloud environment. It is possible to create an identity federation, which will support the Single-Sign-On (SSO) approaches and the management of digital identities by using formalized internet protocols like SAML and XACML specifications as well as open-source ones like OpenID, Oauth and WebID. This results in taking charge from interacting CSPs by outsourcing the credential management [5].

Literature addresses other dynamic and scalable approaches of confidentiality. The friend of a friend approach is interesting, since it provides machine-readable ontologies for object description. Despite of security concerns, the approach is flexible and executable without a centralized database [7].

D. *Policy*

Policies are guidelines for the execution and monitoring of the interactions between the CSPs. Because of its inconsistency and incompatibility, policy-heterogenity causes the main security risk in an intercloud environment. Conflicts arise from a collision of local CSPs' policies and federated policies.

Policy-inconsistencies especially arise from a collision of access guidelines in distributed environments with a multitude of interacting entities. Different policies may be contradictory to each other if they have different effects on entities and their attributes. An exception occurs when two policies affect the same instance differently while being hierarchically linked. A policy-correlation leads to a partial conflict, where two overlapping policies treat one certain object differently, but only one of the two policies allows overlapping.

In an intercloud environment, policies can be collected in a list of guidelines, which could lead to significant reduction of the communication performance of access authorization. Firstly, policy-redundancies may occur in a way that one request is mapped to various policies with identical effects on one and the same object. Secondly, when combining policies, attention has to be paid to semantical and syntactical correctness of the new federated policy [6].

The SLAs complete the IT policies with legal aspects and are essential for policy-compatible intercloud interaction. At the moment, SLAs are limiting the dynamic intercloud communication because they reduce the flexibility of CC business models. More complex SLAs, which possess powerful management and monitoring tools, are essential for proper legal data processing [6], [7].

Another important policy aspect is quality of service (QoS). It helps clients to choose an appropriate service. The mechanisms of service selection have to be able to harmonize various rivaling and maybe excluding QoS objectives [2], [11]. The incorporation of QoS factors makes an intercloud system more flexible, customer-oriented and eventually more attractive to potential user.

E. *Privacy*

Privacy is a strongly by legal restrictions influenced concept, which assures control over information and information flows and restricts access for illegitimate entities. Different laws (i.e. European and American) apply to the term Privacy in different ways, especially in terms of sensibility of personal data and legal precautions. Countries of the EU apply the data minimization concept: no access to personal data unless absolutely necessary. Service users may explicitly prohibit the usage of their personal data for advertising purposes. The USA do not have an equivalent to those legal restrictions [12], [13]. Therefore, it is extremely important to compare and adjust definitions, especially in terms of trans-regional interclouds.

The technical realization of these privacy requirements is a complex issue. It is necessary to distinguish between two basic privacy-strategies: data privacy and identity privacy. Data-privacy strategies consist of altering the content beyond recognition so data cannot be used by third parties easily. Data-perturbation completes original with *noise data* which subsequently becomes unreadable for non-legitimate instances. Unfortunately, the resulting redundancies may cause scalability problems in the intercloud. By using compression methods, cost for communication and request-handling can be cut. Perturbation approaches have to be flexible in order to find a compromise between the user and the privacy guarantee [13], [6].

Encryption by data transformation is one common privacy method. Besides several advantages, this method causes restrictions in the intercloud environment, from a service-communication point of view. First, data-utility aspect plays a more important role than the safety aspect. Second, *data at rest encryption* blocks data indexing and data search. Finally, no efficient methods for operation of data at transit are developed yet [13].

Identity privacy strategies aim at hiding the real identity of interacting instances from unauthorized user. By using anonymity, one CSP can authenticate a user without revealing his true identity. Unlinkability hinders CSPs from identifying users by using a transaction portfolio [14].

III. Collaborative sample process

After providing basic information about intercloud environment in chapter II, an example is introduced in the following section. Building up on the example, chapter V will deal with the implementation of an architecture.

A. Scenario

In the last few years providers like Google and Apple have contributed their solution of mobile payment to the market with Google Wallet and Apple Pay. Despite being two of the most successful companies, support by commerce for these providers is still missing. A better solution consists of retailer and customer are working with a neutral and flexible intermediary, i.e. payment provider.

This example deals with the scenario of electronic payment via smartphone. Mobile payment comprises customer authorization and realization of payment via smartphone. Instead of paying via credit card, the smartphone has to be put on a terminal to initiate the payment. Through near field communication (NFC) two electrical devices, in direct physical proximity, are able to exchange data. In this case, the payment amount is authorized via smartphone. After having put the smartphone on the terminal, the central CSP is contacted and requested for user identification of the smartphone. Afterwards, the user has to authorize the payment via finger print or PIN. When successfully matched, the transaction will be initiated by the central CSP. In order to explain which accounts are involved and which internal processes are initiated in the Central CSP, the process shown in fig. 3 will now be explained in more detail.

B. Coupling of customer's account and payment provider

In this scenario, the CSP acts as an intermediary. The CSP does not have an account and cannot transfer any payments, it only delivers the payment order to the payment provider. Payment providers do not necessarily have to be one and the same nor even similar. Examples of payment providers may be PayPal, Visa or the German website sofortueberweisung.de. Payment provider have to be added to the central CSP in advance. This process is called linking and should ideally only be executed once. While linking, the user is authenticated at the provider and receives two tokens (access and refresh token). The client, in this case the central CSP, uses these tokens to authenticate future orders at the payment provider. After the linking, all payment providers are available via the CSP service and account. If a customer wants to use the payment service of the central CSP, only the authentication to the CSP is needed.

C. Service discovery and execution of the transaction

Fig. 3 provides a detailed overview of the introduced process so all internal steps of the central CSP are visible. After placing the smartphone on the terminal and after the successful authorization, an automatic selection of a payment provider follows, according to rules and settings the customer has set before. Useful rules involve e.g. minimal transaction costs. Afterwards, the payment is initiated at the provider's side.

If the transaction failed, another provider will be chosen according to the rules set by the customer until the transaction is finally successful. This confirmation will be forwarded to the retailer in order to print the receipt. It is important to mention that the BPMN model is only constructed for a positive result. The exception of not finding an appropriate payment provider, due to strict selection criteria or only negative responses, has to be considered in future work.

D. Evaluation

A big advantage in the introduced scenario is that the central CSP is provider neutral. If the CSP integrates new providers, customers and retailers will benefit. Further, the CSP is able to provide its orchestrated service as a SaaS in the intercloud federation.

IV. Service Communication Models in heterogeneous Intercloud Ecosystems

The selected communication models of services provide a comprehensive conceptual view on the intercloud communica-

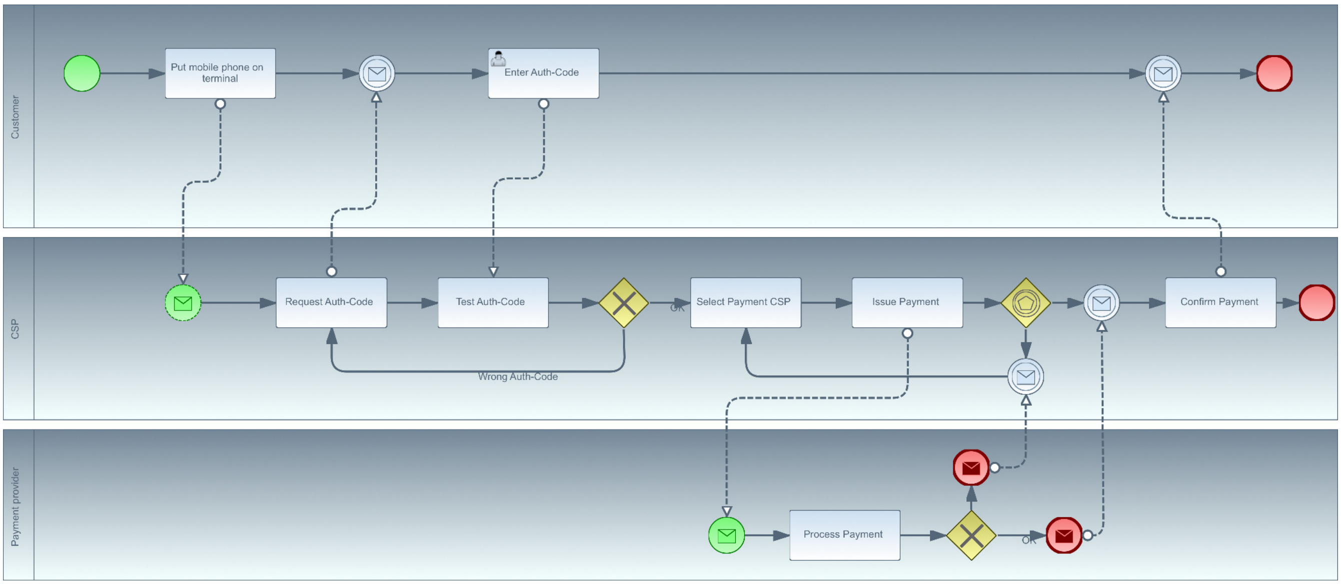

Fig. 3. Collaborative sample process

tion and provide a basis for our proposed architecture. Since some intercloud models, e.g. Demchenko and Makkes [15], [16], [17] and Lloret [4], are well known but focus on special areas of CC, e.g. Demchenko and Makkes focus on IaaS, we present only those models that are relevant for our work.

A. Intercloud Domain-based Trust Model by Bernstein

The intercloud roots (IR), intercloud exchanges (IE), intercloud gateways (IG), and cloud computing resource Catalogs (CCRC) provide the basis of the domain-based trust model proposed by Bernstein [10]. Fig. 4 provides a comprehensive view of all components as well as the corresponding technologies and protocols.

The IR handle the broker functions to operate the service communication. They are structured hierarchical and self reproducing like nodes in peer-to-peer networks. IRs act as security trust service providers, handle the namespace, dynamic naming of the intercloud, and host the distributed CCRCs [19], [18].

The communication and collaboration of the heterogeneous intercloud environment is supported by the IEs, which utilize the information of the CCRC to provide an optimal computation resource matching. IEs act as the trust agents of one domain by collecting information of the confidentiality of other domains and providing the trust level of a domain while initiating the interaction between the domains [10].

The IGs provide authentication mechanisms and standards and protocols for interoperability. Another responsibility of the IGs is to check for availability of resources, the state of interactions and to transform the parameters of the communication from one cloud to another [10].

The CCRC provides a holistic and abstract view on computing resources in the intercloud federation. It enables the resource adjustment between individual CSPs on the basis of selected preferences and constraints. The catalog stores information on available resources, interoperability standards and guidelines, and SLAs [18], [10].

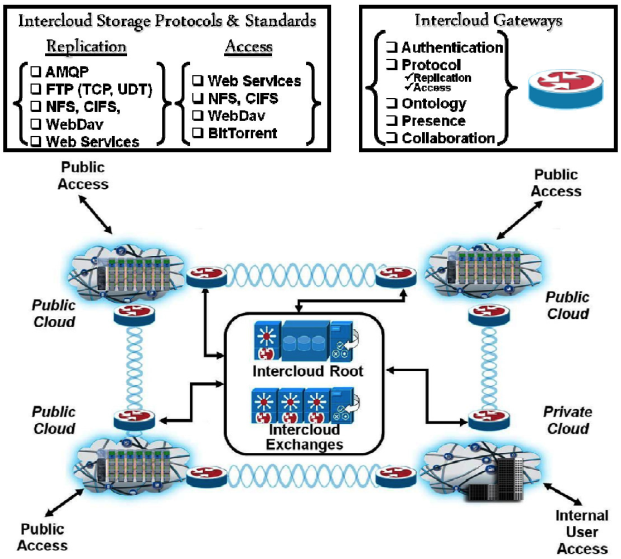

Fig. 4. Intercloud domain-based trust model by Bernstein [18]

The trust management of the intercloud consists of two components, the PKI provides services for the use of trust certificates and trust chains. The fixed-term certificates for short-term transaction are issued by IEs, the long-term transactions are certified by IRs. The PKI is of limited suitability for the certification of processes in the intercloud, since it classifies entities either as trustworthy or not [10].

CSPs are differentiated into confidentiality domains based on the dynamic, time-dependent trust index. The trust index

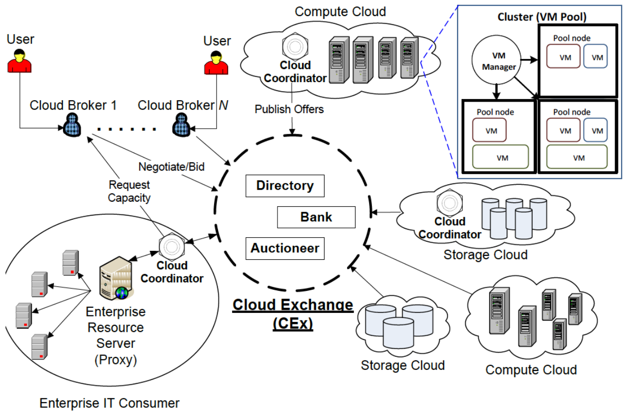

Fig. 5. Utility-oriented federation of clouds by Buyya [2]

comprises information on the CSP and its reputation. The application of the trust index enables the consumption of relatively trustworthy resources that may be located in another domain with a lower trust index [10].

B. Utility-oriented Federation of Clouds by Buyya

Buyya proposed an intercloud architecture that focuses with economically efficient resource matching in the cloud market. The major components of the model are cloud coordinator (CCr), cloud broker (CB), and cloud exchange (CEx) as shown in fig. 5. The CCr, which is integrated into the CSPs' infrastructure, manages domain-specific clouds. It provides a programming, management, and installation environment for applications in the federation of clouds. The CCr acts as a proxy for communication between the participating CSPs and manages all corresponding processes for information exchange [2].

The CCr's service matching process is deeply influenced by economic aspects, e.g. energy consumption, heat output, and node utilization of virtualized cloud environments. It is also influenced by the QoS optimization problem that the service consumer may have selected multiple conflicting QoS objectives. The CCr resolves such conflicts by applying heterogeneous optimization algorithms [2].

The CEx manages all service requests and offers of the cloud federation and supports a SLA based matching process. Further it supports the exchange of information on the current state of allocated resources between the individual CSPs. Given that all participants provide standardized interfaces a directory service enables the CE to lookup desired service offers or requests. The trading process is managed by a dynamic bidding based service, which provides a trustful auctioneer and takes care of offer updates in the cloud federation. The payment management is implemented by an autonomous banking entity, which enforces the agreements on financial transactions of the global CExs. To ease the handling of financial transactions the integration of cloud based accounting systems with existing online payment systems, e.g. PayPal, is taken into consideration [2].

Finally, CB identifies appropriate cloud services on behalf of the users and agrees on the QoS based resource reservation with the CCrs. If a cloud is not able to process an incoming request locally, CB creates a new query comprising QoS information and forwards this query to the CE. Essential conditions for the successful completion of the matching process are: feasibility of QoS targets specified by the user and the equal resource distribution to the individual nodes [2].

C. Cross-Cloud Federation by Celesti

Celesti proposes a process oriented ad-hoc cross-cloud federation, which comprises three phases: in the discovery phase a cloud looks for available resources of other clouds. Afterwards the most appropriate CSP is selected during the matching phase. Finally the authentication phase comprises establishment of a trust context for secure communication of the interacting clouds. The model distinguishes two types of clouds: the home cloud requests compute resources that are offered by the foreign cloud [5].

Fig. 7 provides an overview of the entities and components involved in creating the intercloud federation. In order to form a cross-cloud the internal architecture of the participating cloud must be converted to the following 3 layer structure, first: virtual machine manager (VMM), virtual infrastructure manager (VIM), and cross-cloud federation manager (CCFM) [5].

VIM is a dynamic orchestrator for virtual environments and enables the creation, installation, and management of virtual environments regardless of the underlaying technology. If the home cloud is not able to instantiate another virtual machine due to a lack of additional resources, it forwards the request to the cloud federation. Subsequently the VIM selects a suitable foreign cloud. To securely share and transfer resources with each other the VIMs of the individual clouds create a trust context with each other [5].

The CCFM orchestrates the three phases of the cross-cloud federation. The discovery agent supports the discovery process of the dynamic intercloud environment by publishing and updating the information on offered cloud services on behalf of the individual cloud [5].

The match-making agent makes authorization decisions based on policies and performs compatibility testing of communication policies of interacting clouds. In case of incompatible policy languages the match-making agent applies some transformation algorithms [5].

The authentication agent provides credential management to the cross-cloud federation and coordinates the creation of a security context for the cross-cloud communication using identity providers. By applying SSO after authenticating once with the system the services can be consumed without additional security checks. Using digital identities, which are provided by external identity providers, home clouds can connect to any foreign cloud [5].

The security context in cross-cloud federation is created in two layers: on authentication agent layer and on VIM layer.

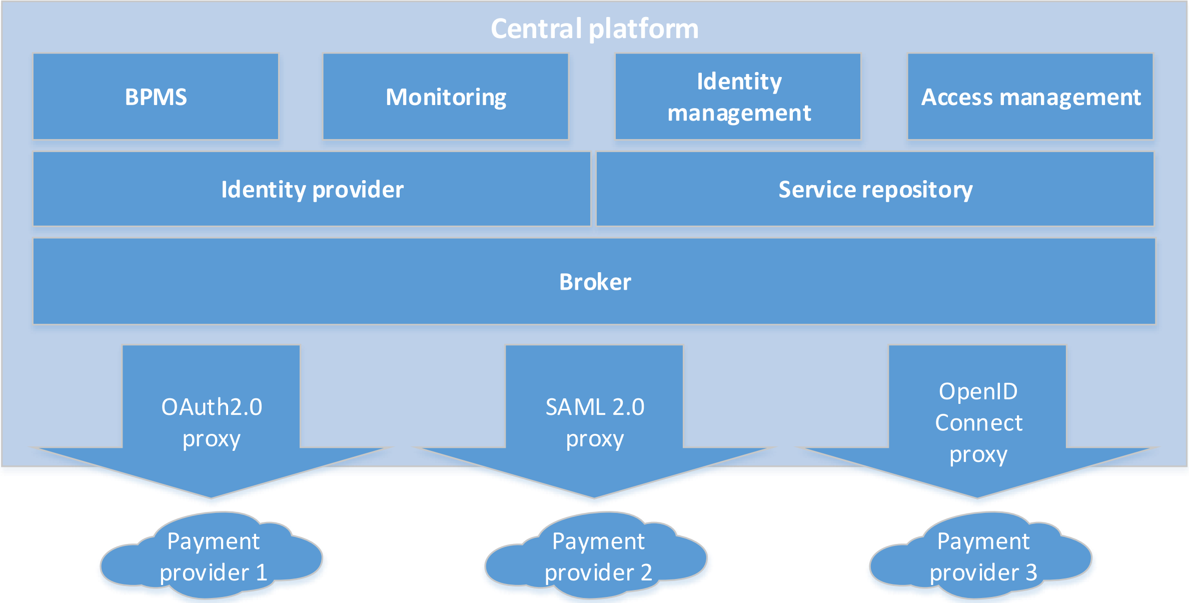

Fig. 6. Intercloud architecture for collaborative business processes

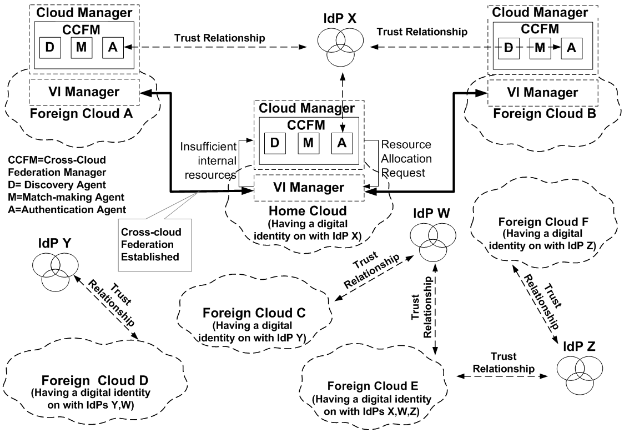

Fig. 7. Cross-Cloud federation by Celesti [5]

The latter act on a lower level compared to the CCFM and facilitate cross-cloud resource provisioning [5].

D. Proxy-based Computing Cloud Framework by Singhal

Singhal proposes another way of tackling the secure communication problem in intercloud environments. In this model the proxy is the central component, which establishes secure communication without predetermined agreements or standardized interfaces and specification [6].

A proxy is an intermediary for traffic between interacting clouds and provides a confidential and trustworthy computing platform, which protects the user data and cloud applications from malicious attacks. The strategic proxy deployment is crucial for efficiency. Thus, in the proxy selection criteria such as latency and forecasted load are taken into account [6].

Since the proxies act on behalf of the users or CSPs, a mechanism for secure delegation is necessary. This mechanism includes dynamic SLA creation, prohibit the proxy access on the processed data, as well as protection against malicious proxy interaction, e.g. modification of transactions. Further a proxy is only allowed to act on behalf of an entity as long as its task is not completed. Such secure delegation can be realized by warrant-based-proxy signatures, PKI or OAuth protocol. Proxies monitor and eliminate conflicts based on policy heterogeneity while processing the service request to avoid potential security risks [6].

A cloud service can be accessed by multiple proxies, these proxies can be assigned different, specific roles. They operate automatically and need no further interaction with the commissioned instance. This interaction is realized by allowing proxies to communicate with other proxies and to initiate necessary service request by themselves. Hence, secure collaboration of multiple CSPs without predetermined agreements is possible [6].

The proxy based model features five types of intercloud architectures: cloud-hosted proxy, proxy as a service, peer-to-peer proxy, on-premise proxy, and hybrid proxy infrastructure. The individual scenarios differ in the implementation strategy of proxy instances.

In the cloud-hosted proxy scenario proxies are integrated into the infrastructure of the individual CSPs. This approach is advantageous from the CSPs point of view, because the CSPs maintain control over the proxies and can adjust the proxies to specific needs [6].

The proxy-as-a-service scenario proposes an autonomous proxy cloud, which provides proxies to the CSPs. Users can create an account either on the proxy cloud or the CSP's cloud. In both cases resource requests are handled by the proxy [6].

The peer-to-peer-proxy scenario organizes the proxies in a P2P-manner. The proxies are managed collectively by the CSPs or proxy providers or autonomously by the proxy nodes [6].

In the on-premise scenario proxies are part of the client's infrastructure. Users can submit their request either via a proxy or directly to the CSP. If the provider is not able to provide the requested resource this will be forwarded to the proxy that will continue the service discovery process with other CSPs [6].

If some or all of the previous variants are combined this is called hybrid-proxy infrastructure. The proxy selection may depend on the type of the requested service and infrastructural peculiarities of the CSP. This approach provides the greatest flexibility in term of intercloud creation [6].

V. Architecture for secure intercloud communication

In this section we propose an architecture for secure intercloud communication in terms of trust, policy, and privacy and evaluate a prototype of this architecture with the sample process of chapter III.

A. Intercloud architecture for the sample process

Basically the collaborative communication of the services is organized by a central platform, which applies concepts of the work shown in the previous section. To enable the communication with the payment providers of the sample process, we propose to use proxies to bind the payment services to the platform. Such a modular handling is necessary as there is no common standard for authentication in CC. Fig. 6 shows the main components of the architecture for secure collaborative business processes. OAuth 2.0, SAML 2.0, and OpenID Connect are the selected proxies that we have implemented as a proof of concept because these are the most widely used authentication standards in CC.

The components and a short description of their tasks is shown in table II. The connection of services through proxies enables flexible scaling of the platform. Another advantage is that the maintenance of the communication infrastructure remains in the CSPs' responsibility. In addition to the integration of the payment process into existing service ecosystems the payment process can be provided as a separate service that can be used by multiple CSPs. The evaluation environment assumes the integration of a payment terminal at the merchants, but there are far more scenarios possible. Since the service is part of payments it requires very high security standards. Hence, sophisticated security components need to be incorporated into the platform. The identity and access management system ensures it is clearly decidable and accountable who is authorized for performing certain actions with the resources, e.g. who is allowed to issue a payment. Such a system needs to authenticate with other system. Hence, protocols for authentication and authorization need to be supported. The most widely used protocols for authentication and authorization are shown in table III.

TABLE II
Core components of the architecture

Kernkomponente	**Funktion**
BPMS	Business process management system to model, instanciate, and collaborative business process on the platform
Monitoring	Monitoring of the whole system
Identity Management	Secure and connect third party identities to the local identities; management of local identities
Access Management	Access policy enforcement and policy conflict handling
Service Repository	List of available services and their descriptions
Broker	Service selection and QoS
Proxy	Invocation of services

The prototype we have implemented and evaluated for our research implements OpenID Connect. There are two reasons. On the one hand OpenID Connect is a protocol for authentication and authorization that is very new but already adopted or will be adopted in the very near future by plenty of the big CSPs, e.g. Google, Microsoft, and PayPal. On the other hand there are two kinds of protocols: centralized and federated ones. Since federated ones are more complex in terms of communication flow and the prototype should be recognized as a proof of concept we chose the federated OpenID Connect protocol.

When a business process is instantiated by the BPMS and the BPMS needs to perform a task of the business process, the broker selects an appropriate service of the service repository. The broker looks up the type of the service's invocation pattern and instantiates the right type of proxy. Then the proxy performs the actual service call and sends the data back to the broker. Prior to sending the user data to the proxy the broker applies privacy rules. This behavior is discussed in [20].

The proxies are composed of four internal components: communicator, privacy guard, data adapter, and service adapter. A service request first passes the communicator that ensures that all communication to the outside of the proxy is encrypted and secured in compliance with the policies. The privacy guard is applies the privacy policies to the payload of the service request by hiding, anonymization, and pseudonymization of data. The privacy compliant payload is then forwarded to the data adapter, which transforms the data from the internal data scheme to the payment provider's data scheme. This is necessary since every payment provider uses another set of parameters to issue the payment. Finally, the transformed payload is sent to the payment provider's service by the service adapter. After the procession of the service request the data goes the same way back threw all components of the proxy. While the data flows threw service adapter, data adapter, privacy guard, and communicator all changes applied to the payload are reverted. This approach ensures a maximal security and privacy level while consuming the service.

B. Evaluation of the architecture

The proposed architecture ensures secure service interaction in collaborative intercloud scenarios with a maximal flexibility in terms of the participating cloud authorization protocols. Table IV shows the implementation of the requirements that have been put on the system.

VI. Summary and outlook

This paper addressed the question how security challenges for service communication that arise due to the adoption of Intercloud environments can be addressed. To illustrate the identified challenges a sample business process has been introduced. This collaborative process that is operated by a central CSP is able to handle payment requests of merchants with a maximum flexibility in terms of the selected payment provider. To identify the optimal architecture multiple approaches of Intercloud service interaction have been reviewed. Based on the outcome of the reviews an architecture for secure service invocation in collaborative business processes has been developed. The architecture is based on proxies that are managed by a central broker engine.

During evaluation phase it turned out that this architecture is able to cope with the most challenges, e.g. interoperability, flexibility, scalability, and auditing. We also identified some limitations of the proposed architecture. First, due to the central management of the proxies it is necessary that the CSP provides enough resources to create and operate enough proxies, even when there are big peaks of requests. The centralized infrastructure also exposes the platform to the risk of outages if a component of the platform has a failure. To reduce this risk we implemented fault tolerance mechanisms into central parts of the platform on virtual machine level.

To ensure a maximal security of all components additional research is necessary. A first step has been made in [20], in this paper the communication flow and structure of the broker and proxy layer is described in detail. Another important field for additional research is the handling of access control to ensure data privacy throughout the whole business process.

References

[1] T. Bittman. (2008) The evolution of the cloud computing market. [Online]. Available: http://blogs.gartner.com/thomas_bittman/2008/11/03/the-evolution-of-the-cloud-computing-market/

[2] R. Buyya, R. Ranjan, and R. N. Calheiros, "Intercloud: Utility-oriented federation of cloud computing environments for scaling of application services," in *Algorithms and architectures for parallel processing*. Springer, 2010, pp. 13–31.

[3] N. Grozev and R. Buyya, "Inter-cloud architectures and application brokering: taxonomy and survey," *Software: Practice and Experience*, vol. 44, no. 3, pp. 369–390, 2014.

[4] J. Lloret, M. Garcia, J. Tomas, and J. J. Rodrigues, "Architecture and protocol for intercloud communication," *Information Sciences*, vol. 258, pp. 434–451, 2014.

[5] A. Celesti, F. Tusa, M. Villari, and A. Puliafito, "How to enhance cloud architectures to enable cross-federation," in *Cloud Computing (CLOUD), 2010 IEEE 3rd International Conference on*, 2010, pp. 337–345.

[6] M. Singhal, S. Chandrasekhar, Tingjian Ge, R. Sandhu, R. Krishnan, Gail-Joon Ahn, and E. Bertino, "Collaboration in multicloud computing environments: Framework and security issues," *Computer*, vol. 46, no. 2, pp. 76–84, 2013.

[7] A. N. Toosi, R. N. Calheiros, and R. Buyya, "Interconnected cloud computing environments: Challenges, taxonomy, and survey," *ACM Computing Surveys (CSUR)*, vol. 47, no. 1, p. 7, 2014.

[8] GICTF, "Use cases and functional requirements for inter-cloud computing," 2010.

[9] Q. Chen and M. Hsu, "Inter-enterprise collaborative business process management," in *Data Engineering, 2001. Proceedings. 17th International Conference on*, 2001, pp. 253–260.

[10] D. Bernstein, D. Vij, and S. Diamond, "An intercloud cloud computing economy-technology, governance, and market blueprints," in *SRII Global Conference (SRII), 2011 Annual*, 2011, pp. 293–299.

[11] S. Ran, "A model for web services discovery with qos," *ACM Sigecom exchanges*, vol. 4, no. 1, pp. 1–10, 2003.

[12] B. Krumay and M. C. Oetzel, "Security and privacy in companies: State-of-the-art and qualitative analysis," in *Availability, Reliability and Security (ARES), 2011 Sixth International Conference on*, 2011, pp. 313–320.

[13] M. Mowbray, S. Pearson, and Y. Shen, "Enhancing privacy in cloud computing via policy-based obfuscation," *The Journal of Supercomputing*, vol. 61, no. 2, pp. 267–291, 2012.

[14] K. YIN and H. WANG, "A mutual authentication protocol providing security and privacy protection for intercloud environments," *Journal of Computational Information Systems*, vol. 10, no. 21, pp. 9087–9093, 2014.

[15] Y. Demchenko, C. Ngo, M. Makkes, R. Stgrijkers, and C. d. Laat, "Defining inter-cloud architecture for interoperability and integration," in *CLOUD COMPUTING 2012, The Third International Conference on Cloud Computing, GRIDs, and Virtualization*, 2012, pp. 174–180.

[16] Y. Demchenko, C. Ngo, C. d. Laat, M. X. Makkes, and R. Strijkers, "Intercloud architecture framework for heterogeneous multi-provider cloud based infrastructure services provisioning," *International Journal of Next-Generation Computing*, vol. 4, no. 2, 2013.

[17] M. X. Makkes, C. Ngo, Y. Demchenko, R. Stijkers, R. Meijer, and C. d. Laat, "Defining intercloud federation framework for multi-provider cloud services integration," in *CLOUD COMPUTING 2013, The Fourth International Conference on Cloud Computing, GRIDs, and Virtualization*, 2013, pp. 185–190.

[18] D. Bernstein and D. Vij, "Simple storage replication protocol (ssrp) for intercloud," in *EMERGING 2010, The Second International Conference on Emerging Network Intelligence*, 2010, pp. 30–37.

[19] ——, "Intercloud security considerations," in *Cloud Computing Technology and Science (CloudCom), 2010 IEEE Second International Conference on*, 2010, pp. 537–544.

[20] B. Schwarzbach, A. Pirogov, A. Schier, and B. Franczyk, "Inter-cloud architecture for privacy-preserving collaborative bpaas," Shanghai, 2015.

TABLE III
EVALUATION OF AUTHORIZATION AND AUTHENTICATION PROTOCOLS

Criterion	SAML 2.0	OAuth 2.0	OpenID Connect	CAS
Exchange format	XML	URL parameter / JSON	URL-Parameter / JSON	URL-Parameter
Transfer protocol	HTTP, SOAP	HTTP	HTTP	HTTP
Token size	$5.7kB$	$77B$	$121B$	$35B$
IdP discovery	x		x	
SSO	x	x	x	x
Strengths	Encryption	Wide spread	Authentication & authorization	

TABLE IV
EVALUATION OF THE ARCHITECTURE

Requirement	Description	Implementation
Interoperability	Communication has to be independent of the used authentication and authorization protocol	By the use of proxies the actual authentication and authorization protocol is been hidden from the other components. The broker instantiates the right proxy for the service provider.
Scalability	Flexible expandability	By adding new proxies the system can easily be expanded for new service providers.
Reliability	High availability of the payment service	In case on payment provider is not available the broker select the next suitable service provider. Hence, only if all suitable payment providers are unavailable the whole service is not working any more.
Optimal service provisioning	Service discovery, separation of combined services	The platform is divided into the components central platform, broker, service repository, and proxies
Access & identity management	Secure management of multiple identities; secure trust context	The identity management is done by the identity management system that is part of the central platform. This ensures no information is transferred to third parties unnecessarily. Additional firewall policies to protect the central platform are applied. Necessary communication on identities is realized by proxies, which implement secure authentication and authorization protocols.
Trust	Trust context	Externally by the use of proxies. Our sample process uses OpenID Connect and its PKI
Policy	Policy resolution	Monitoring
Privacy	Data & identity privacy	All communication, internal and external, is encrypted. The broker applies privacy control on the data.