|  |  |  |
| --- | --- | --- |
| logogreen | ASIA-PACIFIC TELECOMMUNITY | **Document No.:**  **ASTAP-31/TMP-13** |
| **The 31th APT Standardization Program Forum (ASTAP-31)** |
| 11–15 June 2019, Tokyo, Japan | 13 JUNE 2019 |

EG IS

**4rd REVISED Draft OF FRAMEWORK OF CLOUD ACCESS SECURITY BROKER**

**FOR CLOUD SERVICE SECURITY**

EG IS agreed to establish a new work item on Framework of cloud access security broker for cloud service security in ASTAP-28. The scope was agreed and this document is a draft text for further discussion until now.

**Attachement 1:** Guideline for FRAMEWORK FO CLOUD ACCESS SECURITY BROKER FOR CLOUD SERVICE SECURITY.

**Attachment 1**

**Guideline for**

**FRAMEWORK OF CLOUD ACCESS SECURITY BROKER**

**FOR CLOUD SERVICE SECURITY**

*Editor’s Note: The current content is for 4-tier CASB and it will be updated to satisfy the scope of report in the further discussion.*

*Editor’s Note 2: Gap analysis of standard activity in Annex will be updated continuously to reflect the latest contents.*

1. **Introduction**

CASB is a system that provides separate security features for SaaS applications. It serves as a platform to meet security demands from each customer effectively without public cloud service providers’ burden to implement more complicated security features to meet the exactly same security demands.

The main components for the 4-tier CASB are a secure agent, a CASB proxy, a CASB inline gateway, and a CASB secure API. They are positioned between devices of cloud service users and cloud service servers. If they independently operate security control without any prearranged interaction, possible duplicates of security control undermine the overall quality of cloud service. Furthermore, it would raise many problems such as inconsistency or desynchronizing of security policy applied to a company, too.

For efficient CASB service, we need methods to provide stable service and to prevent overload of a specific CASB by considering the processing performance of CASB for each tier and distributing cloud service requests by CASB.

Furthermore, since each CASB component has different purposes and intents, a client of a cloud service user must pass and be checked with all CASBs provided when it accesses to a cloud server. Especially, we must consider the way not to bypass CASB because smartphone outside in BYOD(Bring Your Own Device) environment can access to cloud server directly.

This document describes a protocol to minimize duplicated actions of security control, to improve availability of security control, to synchronize security policy of each CASB, and to prevent cloud service user accessing to the cloud server by bypassing CASB in 4-tier CASB environment consisting of secure agent, CASB proxy, CASB inline gateway, and CASB secure API. And including this, this document shows the result of simulation and performance evaluation for 4-tier CASB. This document will include test in various service.

1. **Scope**

This document is to provide a framework of 4-tier CASB with following;

* Access Control Protocol for Cloud Service Security in 4-tier CASB
* Security control process for efficient cloud service security in 4-tier CASB environments
* Secure communication protocols between CASBs in 4-tier CASB settings
* Methods to manage security control for CASB and non-CASB secure devices in BYOD(Bring Your Own Device) environments
* Simulation and performance evaluation of the framework

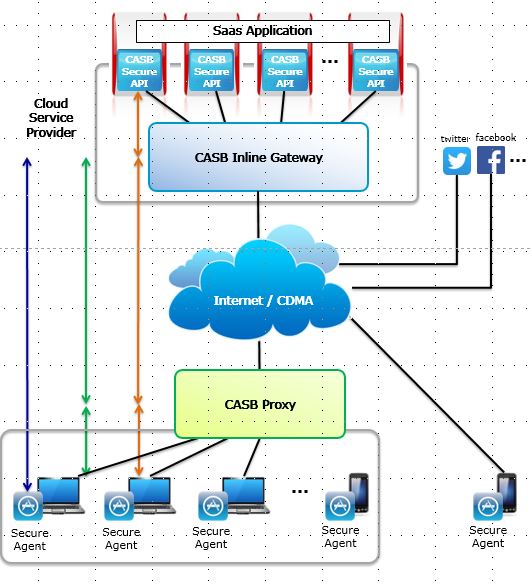
1. **Terms and definitions**

CASB: on-premises, or cloud-based security policy enforcement points, placed between cloud service consumers and cloud service providers to combine and interject enterprise security policies as the cloud-based resources are accessed.

SaaS(Software as a Service): software that is owned, delivered and managed remotely by one or more providers.

Public Cloud Computing: a style of computing where scalable and elastic IT-enabled capabilities are provided as a service to external customers using Internet technologies

1. **Structure of 4-tier Cloud Access Security Broker**



To guarantee two key requirements of CASBs, which are security of SaaS applications, and security of SaaS users, the structure needs CASB inline gateways, CASB proxies, and CASB secure APIs. Secure agents are needed to support such a structure from the client side. Thus the structure of CASBs should be formed with 4-tier of CASB secure API, CASB inline gateway, CASB proxy, and secure agent as in the figure above.

* CASB Secure API

Generally, the cloud service provider supplies service users with a huge number of SaaS applications. Because service traffic varies depending on usage frequency and usage pattern of users and consumption for each login, security controls by the CASB gateway alone may cause availability issues of SaaS service. So the provider of SaaS applications is required to supply SaaS service of his applications to users without those availability issues anytime and anywhere.

CASB Secure API satisfies such a requirement and exists inside SaaS applications as a library. Applying CASB secure API is completed when the development company of SaaS applications finishes implementing integration with the library authorized by the cloud service provider.

* CASB Inline Gateway

Generally, the public cloud service provider supplies services using SaaS applications from various vendors. The credibility of the public cloud service depends on whether users are accurately charged based on how much they have used the service. Furthermore, exact billing depends on whether to accurately measure how much each user has used certain SaaS applications, and the biggest obstacle to exact billing is usage of SaaS applications by unauthorized users or identity thieves.

CASB inline gateways control security at the gateway of the SaaS system as an appliance. While SaaS service is usually provided in encrypted data as in SSL, CASB inline gateways operate inside the system, so they do not concern with encrypted data.

* CASB Proxy

As the number of SaaS applications increases recently, the leakage of inside information through usage of not only authorized SaaS applications by company or agency but also unauthorized ones has become serious. It prompts effective control over all SaaS applications used by members of a company or an agency, and the CASB proxy fills the need. The CASB proxy is placed inside a company or an agency as a common proxy and performs security controls on all devices.

CASB registers SaaS applications authorized by company or agency. It performs security controls based on defined rules or defers them to the CASB inline gateway. Depending on security policies, it performs various security controls on unauthorized SaaS applications.

CASB proxy operates to a client like a server and to a server like a client. It can perform security controls in encrypted data of SaaS service.

* Secure Agent

Secure agents are basic client programs to manage all functions of CASB effectively. Typically, it supports settings of the CASB proxy, processes load balance of CASBs based on service type, runs encryption functionality like SSL provided by SaaS applications, handles its own encryption and decryption, and so on. For mobile devices, it provides VPN to prevent security bypass and to force the devices to access CASB.

1. **Access Control Protocol**

Since cloud service is provided via HTTP, CASB may choose to request further actions of security control by adding information of its security control in the HTTP header. Once CASB inline gateway and CASB proxy execute their security control, it accordingly adds the following metadata as the request message in the table below to outbound information.

|  |  |  |
| --- | --- | --- |
| Field name | Description | Example |
| CASB-agentID | User ID authenticated through the secure agent | CASB-agentID: [ask@abc.com](mailto:ask@abc.com) |
| CASB-agentIP | IP address from the secure agent when the device of the agent connects to CASB | CASB-agentIP: 10.8.0.3 |
| CASB-SCAN | Whether CASB has executed an action of security control (yes/no) | CASB-SCAN:yes |

When such a protocol is applied, CASB inline gateway and CASB proxy parse the HTTP header before executing their security control. If the value of CASB-SCAN is yes, then CASB determines whether further actions of security control are necessary.

And when CASB secure API executes its security control, it accordingly adds the following metadata as the reply message in the table below to outbound information.

|  |  |  |
| --- | --- | --- |
| Field name | Description | Example |
| CASB-API-domain | Server domain applying API | CASB-API-domain: www.abc.co.kr |
| CASB-agentID | User ID authenticated through the secure agent | CASB-agentID: [ask@abc.com](mailto:ask@abc.com) |
| CASB-agentIP | IP address from the secure agent when the device of the agent connects to CASB | CASB-agentIP: 10.8.0.3 |
| CASB-API-SCAN | Whether CASB secure API has executed an action of security control (yes/no) | CASB-API-SCAN: yes |

When such a protocol is applied, CASB inline gateway and CASB proxy parse the HTTP header before executing their security control. If the value of CASB-API-SCAN is yes, then CASB inline gateway and CASB proxy will skip the security control for the application.

Furthermore, we need method for load balance because the overload of specific CASB could occur if the protocol only to prevent the overlapping access control is used. CASB can share system availability information between 4-tier CASBs by adding CASB service availability information to the header of this protocol.

The CASB system availability meta information added to these HTTP headers enables the CASB to perform or bypass security control for a cloud service request based on the availability of each CASB system.

In particular, using the CASB availability information added to the HTTP header, the secure agent can designate the CASB to perform security control for a cloud service request in the 4-tier environment.

|  |  |  |  |
| --- | --- | --- | --- |
| Field name | Description | Value | Example |
| CASB-Type | CASB information : Inline Gateway CASB, Proxy CASB or API CASB | Inline Gateway CASB, Proxy CASB, API CASB | CASB-type: Proxy CASB |
| CASB-Capacity | CASB service performance level | High, Mid, Low | CASB-Capacity: High |
| CASB-Act | CASB security control action status | Yes, No | CASB-Act: Yes |
| CASB-svTarget | Policy URL information of target CASB | Inline Gateway CASB, Proxy CASB, API CASB | CASB-policyCallbackURL: https://www.abc.com/policy |

In a 4-tier CASB environment, the four-layer CASB system consists of secure agent, proxy CASB, inline gateway CASB, and CASB secure API, and HTTP requests of the user's cloud service are transmitted to each CASB system through secure agent installed in user PC.

Through the information included in the received HTTP header, the CASB of each layer can check the availability information of all CASB systems running in the 4-tier CASB and information on whether CASB security control is executed for the request.

HTTP requests of the user's cloud service are transmitted to the secure agent through all CASB systems in the 4-tier environment as HTTP responses. The secure agent analyzes the received HTTP header information to check the availability of all CASB systems in the 4-tier environment.

Based on the availability performance information of the CASB system, the secure agent selects the CASB system to execute the security control for the cloud service request of the user, adds information to the HTTP header, and transmits it. The CASB determines whether the CASB information for the security service target is included in the header of the corresponding HTTP and then determines whether to execute the CASB security service.

1. **Security Control Process**

Because security controls of CASB must work with security policy of each organization, security control process of CASB begins with setting up security policies accurately. However, it is difficult for a security officer to manage the security policies of CASBs consistently when organization uses multiple CASBs.

When user uses cloud service, multiple heterogeneous CASBs can located in service flow. And then, if the security policies of each CASB are set differently, it can cause serious problems. So, when one CASB security policy is updated, other CASB security policy must be automatically updated.

Since cloud service uses HTTP, CASB can synchronize policies by adding information of security policies to the HTTP header. For the synchronization, the version of security policy, the date when security policy is set, and the CallbackURL address must be shared within CASB. In order to synchronize the latest CASB policies, the following metadata is included in the HTTP header.

|  |  |  |
| --- | --- | --- |
| Field name | Description | Example |
| CASB-policyVER | The Policy version | CASB-policyVER: 1.0.1.2 |
| CASB-policyDATE | The date when the policy is set | CASB-policyDATE: 2017-07-24 13:22:30 |
| CASB-policyCallbackURL | The Callback URL address to be used when an update of policy information is required | CASB-policyCallbackURL: https://www.abc.com/policy |
| CASB-type | CASB inline gateway or CASB Proxy | CASB-type: CASB Proxy |

When policies change in a CASB proxy or a CASB inline gateway during network communication with SaaS applications, the corresponding component adds new metadata of such changed policies in an HTTP header. A CASB inline gateway or CASB secure API parses the HTTP header for CASB-policyVER or CASB-policyDATE before executing security control. If the currently applied version turns out to be an older version, the component that detects it requests policy information at the value of CASB-policyCallbackURL and updates the information.

1. **Method to manage security control in BYOD**

When a CASB proxy is used, client devices in a corporation or an organization forcibly go through CASBs when accessing to a cloud server, but in a BYOD environment, a cloud server user can access using a smart phone or the like, so that access by a non-CASB secure device(secure agent was installed, but not used) occurs.

In this case, although access control can be done through secure API, a security vulnerability may appear when a company using cloud service uses CASB proxy for protecting information inside company from exposure to cloud service such as Facebook.

Therefore, access to the cloud server should be applied to ensure that the client is checked or passed through the CASB. A method for preventing bypassing by non-CASB is to mount VPN(Virtual Private Network) and MDM(Mobile Device Management) function on secure agent.

In other words, the VPN and MDM modules are embedded in the secure agent, and it is registered in the management system at the initial installation. The management system checks whether the secure agent is running. In this case, it is possible to force the device to access to the cloud server through the CASB proxy under VPN while the secure agent is running.

In addition, it is possible to prevent bypass access with the secure agent removed. In case of security sensitive devices, security incidents can be prevented by initializing the smart phone through MDM when the secure agent is deleted or turned off.

1. **Simulation and Performance evaluation**

Since CASB is placed between SaaS application user and SaaS application server, latency would occur. So it is very important to minimize it for cloud service quality. We had proposed framework of 4-tier CASB as the method to reduce latency. And now we show the result of simulation test and its performance.

To verify effect of protocol we proposed for 4-tier CASB framework, we tested the performance of CASB for 5 cloud services we developed

We compared the cases with and without CASB, and in the case of “with CASB”, we compared the case where each CASB operates completely independently and the case where we use protocol we proposed.

First, the test environment is summarized as follows.

* Test overview and environment
  + Test Period : 2018. 7.1 ~ 2018. 9.30.(3 months)
  + Number of testers: 20 Users
  + User Test Environment

|  |  |  |
| --- | --- | --- |
|  | PC | Mobile |
| Processor | Intel Core i5, i7 | Exynos 9810, 8895 |
| System type | 32-bit, 64-bit | 64-bit |
| Memory | 4GB, 8GB, 32GB | 4GB, 6GB |
| Operating System | Windows 7, 8, 8.1ProK, 10 | Android 8.0 |

* + Cloud Service Operating Environment

|  |  |
| --- | --- |
| Processor | Intel Xeon E5-2609, E5-2630 |
| Memory | 16GB, 32GB |
| Operating System | CentOS 6.6, Ubuntu 14.04 |
| Cloud Operating S/W | OpenStack(ver. Icehouse, ver. mitaka) |

* Cloud Services for Test
  + File sharing Cloud Application : SeaFile, Pydio
  + RSS feed Reader Cloud Application: Sismics Reader
  + SNS Cloud Application : GNU Social, ELGG
  + HRM Cloud Application : Orange HRM
  + Web Blog Cloud Application : Solo
* Test method
  + Access to test target cloud service and attempt various actions
  + Modify the security policy for cloud service in case of each access onto the service
  + Access cloud service with 2-tier, 3-tier and 4-tier CASB and without CASB
  + Distinguish the status of the function that prevents duplex check of security control and execute test
  + Distinguish the status on each system structure and the function that prevents duplex check of security control and measure latency about the cloud service use
* Test data
  + Number of requests : 1,851,258 cases
  + Cloud System without CASB : 329,275 cases
  + Cloud System with 2-tier CASB : 471,233 cases
  + Cloud System with 3-tier CASB: 553,622 cases
  + Cloud System with 4-tier CASB: 497,128 cases
  + Volume of Network Traffic : about 12.7 TB(including FileStream data)
  + Cloud System without CASB : about 2TB
  + Cloud System with 2-tier CASB: about 3TB
  + Cloud System with 3-tier CASB: about 4.1TB
  + Cloud System with 4-tier CASB: about 3.6TB
* Test result

Before CASB was applied, the cloud service use latency appeared as an average of 37ms in 3 months. When CASB was applied but there was no access control protocol, which is the security control duplex preventive protocol, around 3~12 times greater latency resulted every time CASB system was added. However, in the case with access control protocol, around 3 times greater latency results even in the 4-tier CASB system so it appeared to be effective in the multi tier system.

|  |  |  |
| --- | --- | --- |
|  | No use of protocol we propose | Use of protocol we propose |
| System without CASB | 37ms | |
| System with 2-tier CASB | 362ms | 323ms |
| System with 3-tier CASB | 836ms | 340ms |
| System with 4-tier CASB | 1203ms | 369ms |

1. **References**

[1] http://www.gartner.com

[2] https://www.joins.co.kr/w/site/Network\_Programming/AdvancedComm/HTTP

[3] https://en.wikipedia.org/wiki/List\_of\_HTTP\_header\_fields

**Annex 1**

**GAP ANALYSYS OF STANDARD ACTIVITY**

**ON CLOUD ACCESS SERVICE BROKER**

1. **CASB Vendor**

CASB vendors develop solutions in one of the next five elements.

* Shadow IT

Shadow IT in cloud service poses hidden threats and possibly incurs costs from them. Usually companies use hundreds of cloud services. Most of them are affected in shadow IT and exposed to a security blind spot. Therefore employees of such companies does not meet their companies’ security requirements, and they likely use vulnerable and costly cloud service. CASB should be able to find cloud services in shadow IT and detect access of those services. It should also be able to analyze usages of cloud services and estimate risk indexes of cloud applications. Existing CASB solutions checks for unknown cloud applications and for accessibility of such applications, but some CASB vendors have analysis of usage and estimation of risk index included in their solutions as an option.

* Compliance Monitoring

Cloud service customers should comply with rules to protect data and personal information when such data is transferred to cloud service. CASB should retrieve and sort company data and should know the policy template that contains regular expressions. It should also be able to support policy execution and exception policy that include data block, encryption, and deletion. Most of CASB vendors have those aforementioned features except the policy template. Thus additional work for standardization is necessary for the policy template.

* Threat Protection

Organizations must prevent their members who use cloud storage service and its client from exposed to malicious codes or security threats. They must detect and prevent their members who upload files infected with malicious codes or unauthorized users who attempt to access cloud service or data. Those members should be able to get protection from many different security threats from cloud service or malicious codes. CASB can support detection and removal of malicious codes, analysis of user actions, and analysis of network. It can also maintain detection of security threats and other kinds of threats. Not all CASB solutions include all those functionalities. Therefore we need to work on standardization of threat protection.

* Encryption

Confidentiality must be preserved when data is stored in the storage of cloud service. CASB must protect such data from unauthorized access, sniffing during data transmission, and hidden threats like a backdoor. CASB should be able to choose to support not just file level encryption and field level encryption but also stronger encryption module. It should also maintain such encryption keys. Solutions from most CASB vendors support file level encryption and maintenance of encryption keys. Encryption algorithm and protocols for key exchange needs to be included in CASB standardization.

* IAM(Identity and Access Management)

The basic principle of IAM, “the right people gain access to the right materials at the right time”, is also applied to cloud service. CASB should guarantee right access to data stored at various cloud services. It can support context-based access control, DRM technology, single sign-on for cloud service, and IAM technology integrated with 3rd party. Most CASB vendors currently only include context-based access control for their solutions. Few other CASB solutions with IAM support the rest. Standardization should be underway so that CASB systems of different configurations can share context-based access control policies.

1. **Organization for Standardization**

Currently there is no standards organization for CASB. But we have progress of standards for some CASB technologies, and we would like to describe such efforts further.

* NIST(National Institute of Standards and Technology)

NIST (National Institute of Standards and Technology) revealed a technical roadmap for cloud computing in November 2011. It includes cloud computing standard plans for interoperability, data mobility, and security measures between cloud services. It published “Guidelines on Security and Privacy in Public Cloud Computing” in December 2011, and the document includes the following CASB concepts for standardization.

Governance: It outlines policies, procedures, implementations, and tests cloud service and monitoring for organizations. It needs to establish audit tools and structures to ensure tasks for the organizations are clearly defined during the system lifecycle.

Compliance: It forces organizations to comply with security and protection of personal information. Users need to understand organization rules including data layout of cloud computing, protection of personal information, security maintenance, record maintenance, electronic search.

Trust: Cloud service providers should guarantee enough visibility for users to gain access to cloud or to control security and personal information. They need to establish threat management programs to encounter various attacks without affecting the system lifecycle.

Identity and Access Management: Organizations need to find right methods to protect and support authentication, authorization, and access control.

Data protection: Cloud service providers should have data management solutions that support access control and protection on data related to organizations.

Data protection: Cloud service providers should have data management solutions that support access control and protection on data related to organizations.

NIST has published a few documents for cloud computing standardization: “NIST Cloud Computing Standards Roadmap” in 2013, “Cloud Computing Forensic, Hypervisor Deployment” in 2014, “Definition of Microservices, Application Containers and System Virtual Machines” in 2016, and “Mobile Device Security:Cloud and Hybrid Builds” in 2019. It has not published any standardization documents since.

* ITU-T(International Telecommunication Union Telecommunication

Standardization Sector)

“Security framework for cloud computing” published at ITU-T on October 2015 describes a security framework for cloud computing and analyzes security threats and challenges in the cloud computing environment. The document also provides a framework methodology that mitigates such threats and supports security features that address security issues.

In March 2016, 'Guidelines of operational security for cloud computing' indicated that security threats from the perspective of cloud service providers could occur as described below:

Threats by cloud service providers: attackers can lower availability on cloud computing service through unauthorized access to the cloud service management interface. Personnel inside cloud service providers may accidently lower security measures for cloud service, which can allow attackers to infiltrate specific cloud service systems.

Therefore general operational security guidelines for cloud computing provide a set of security measures and detailed security activities for operations and maintenance to help resolve security issues that arise during cloud computing operations.

In September 2016, 'Guidelines for service customer data security' advised that security threats from the viewpoint of cloud computing service customers could occur as described below.

Threats by cloud service customers: cloud service customers may undermine data validity, data management policy, and data confidentiality by missing or leaking their own data. Attackers may hijack privileges of authorized users and fabricate data for those users. Users’ negligence like sharing the administrator account or managing account information poorly can lead to serious problems.

To prevent this, we provide general security guidelines on customer data to analyze a security life cycle of customer data and suggest security requirements for each step of the data life cycle.

In March 2018, ‘Data security requirements for the monitoring service of cloud computing’ analyses data security requirements for the monitoring service of cloud computing could occur as described below.

Monitoring service of cloud computing includes monitoring data scope requirements, monitoring data lifecycle, security requirements of monitoring data acquisition and security requirements of monitoring data storage. Monitoring data scope requirements include the necessary monitoring scope that cloud service providers (CSPs) should provide to maintain cloud security and the biggest monitoring scope of CSPs. Monitoring data lifecycle includes data creation, data store, data use, data migrate, data present, data destroy and data backup. Monitoring acquisition determines security requirements of the acquisition techniques of monitoring service. Monitoring data storage determines security requirements for CSPs to store the monitoring data

As of May 2019, we are currently standardizing four work items, X.SRIaaS, X.SRNaaS, X.SRCaaS, X.GSBDaaS and X.sgcc. The standardization scope of each is as follows.

ITU-T X.SRIaaS “Security Requirements of public Infrastructure as a service (IaaS) in cloud computing” was drafted because special protection is necessary against security threats to IaaS environments that share computing, storage, and network service. The goal of the document is that strategies, designs, and operating environments about security proposed by it would help IaaS cloud service providers improve overall security level.

ITU-T X.SRNaaS “Security Requirements of Network as a Service (NaaS) in Cloud Computing” analyzes security threats about network virtualization and application security vulnerability about NaaS that provides cloud service customers with transport connectivity. It would help NaaS providers resolve security issues.

ITU-T X.SRCaaS “Security Requirements for Communication as a service application environments” intends to fulfill security requirements about identity theft, orchestration security, multi devices security, countering spam, privacy protection, and infrastructure attack in CaaS.

ITU-T X.GSBDaaS intends to fulfill security requirements about distribute computing system, non-relational data storage, audit, and disaster recovery of big data in BDaaS.

ITU-T X.sgcc “Security guidelines for container in cloud computing environment” analyses security threats and challenges on container in cloud computing environment, and provides the security guidelines and reference security framework for container in cloud.

ITU-T is not currently standardizing on CASB, and only standardization of security requirements for IaaS, NaaS, CaaS, GSBDaaS, and application of cloud service is in progress.

* ISO/IEC(international Organization for Standardization/International Electrotechnical Commission)

Also ISO/IEC does not have any ongoing research or standardization on CASB. As for a standard related to cloud technology, it provided an overview of cloud computing with definitions of cloud computing in “ISO/IEC 17788:2014” on October 2014.

In “ISO/IEC 17789:2014” on October 2014, it described cloud computing reference architecture which includes the cloud computing roles, cloud computing activities, and the cloud computing functional components and their relationships.

It provided guidance regarding information security controls and implementation that apply to the provision and usage of cloud service in “ISO/IEC 27017:2015” on December 2015. The guidance presents standards for cloud service provides and cloud service customers.

In “ISO/IEC 27036-4:2016” on October 2016, it defined guidelines that enable cloud service provider to effectively manage information security risks by identifying such risks associated with cloud service usage. Those guidelines also enable organizations using cloud service to address risks associated with the import and use of cloud services that can impact information security.

In “ISO/IEC 19944:2017” on October 2017, it described systems associated with devices that used cloud services by expanding existing cloud computing terms and reference architectures from ISO/IEC 17788 and ISO/IEC 17789. It specifies various kinds of data within cloud computing and the data flow among cloud services, cloud service customers, and cloud service users.

The currently ongoing work program includes “ISO/IEC DIS 22624 – Taxonomy based data handling for cloud services”, “ISO/IEC PDTR 23188 – Edge computing landscape”, “ISO/IEC NP TR 23187 – Interacting with cloud service partners(CSNs)”, “ISO/IEC NP TR 23951 – Best practices for cloud SLA metrics”, “ISO/IEC PDTR 23613 – Cloud service metering and billing elements” “ISO/IEC AWI 23751 – Data sharing agreement((DSA) framework”, “ISO/IEC CD 22123 – Concepts and terminology” and “ISO/IEC PDTS 23167 – Common Technologies and Techniques”.

1. **Reference**

[1] Bitglass-Cloud Security Solution Brief

[2] CipherCloud – The CipherCloud Trust Platform

[3] Netskope – Netskope Active Platform

[4] Skyhigh – Cloud Access Security Broker

[5] CloudLock – www.cloudlock.com/platform

[6] BlueCoat – BlueCoat Cloud Data Protection Gateway

[7] NIST SP800-144 “Guidelines on Security and Privacy in Public Cloud Computing”

[8] NIST - http://csrc.nist.gov/publications/PubsDrafts.html

[9] ITU-T “Security in Telecommunications and Information Technology”

[10] ITU-T “LS/o/r on updated information in the cloud computing standards roadmap”

[11] ITU-T “Summaries for Recommendations under development in Q8/17”

[12] ITU-T “Report of Question 8/17”

[13] ITU-T “https://www.itu.int/ITU-T/recommendations/search.aspx”

[14] ITU-T “https://www.itu.int/ITU-T/workprog/wp\_item.aspx

[15] ISO/IEC “https://www.iso.org/advanced-search/x/”