4

Identity and Access Management in NFV

IAM (Identity and Access Management) is widely accepted as the first defense line of today's ICT infrastructures and services. Some of the major functions include authentication, authorization, accounting and access control. In this chapter, we first present the basic functions and basic implementations of IAM and then discuss their NFV-based implementations. We finally provide a comprative analysis between these two variants.

IAM is a very broad topic that covers both technical and non-technical areas, involving business processes, technologies, and policies for managing digital identities, monitoring network access and controlling access to company assets. In other words, IAM is about how to enable the right individuals to access the right resources at the right time for the right reasons. Technically speaking, IAM is used to initiate, capture, record and manage user identities and their related access permissions to information assets in an automated way. As a result, access privileges are granted to the users according to the interpretation of policy rules, which are then enforced by a sequence of authentication, authorization and auditing functions.

The implication of IAM has two independent elements: *identity management* and *access management*. Identity management describes the process of authentication, authorization and user privileges across

system boundaries, whereas access management is more focused on the access control to verify whether users are granted privileges to access particular services or resources. The decision result is evaluated based on policy rules, user's roles and other elements that are predefined by the administrators.

4.1. Major functions

More specifically, the IAM framework is composed of the following functions [IDE 12], which are briefly explained as follows:

– Authentication: a process of determining whether the user credentials are authentic. Authentication is activated only when the users intend to access information in the system. Then, the users are required to prove their right and identity, basically through the username and password or biometric identities. The system verifies user identity by matching the provided credentials with a specific abstract user object that is stored in the system. Once the two objects match, the access is authenticated. To date, there are several types of authentication that have been widely used in the current ICT context such as tokens, public keys, certificates and biometric authentication [CLO 12, HAV 07].

– Authorization: a process of allowing users to perform an action with the resource that they are granted, i.e. preventing users from accessing the resources that they are not allowed to access [HAV 07]. For example, if a user tries to write a file with only read permission, then the authorization fails and the requested operation for writing could be rejected. Furthermore, the users can be authenticated using a certain identity but they can be authorized to access a specific resource under a different identity. Many authorization functions are built upon the four following methods:

- Discretionary Access Control (DAC): which allows users or administrators to define an access control list (ACL) in regards to specific resources, such as which users can access the resource and what privileges they are granted. An example of DAC was proposed by [WAN 11] to mitigate user privacy and data leakage problems in collaboration clouds.

- Mandatory access control (MAC): which is defined by the administrator to manage the access control based on policy and cannot be modified or changed by users. The policy specifies the access rules for the requested services/resources. Some examples of MAC are used for end-to-end access control in Web applications [HIC 10] and in commodity OS to support intrusion detection [SHA 11].

- Role-base-access control (RBAC): which is based on defining a list of business roles and permissions and privileges are then granted to each role. Some concrete examples related to RBAC were discussed in [KER 03, KER 05].

- Attribute-based access control (ABAC): which uses attributes as the policy building blocks to define access control rules and describe access requests. Authorization-based ABAC typically relies on the evaluation of attributes (users), targeted resources, desired action (read, write), and access control/policy rules to verify whether access right are granted. The ABAC is widely used in various domains such as authorization services [LEE 08], Web services [CAP 14] and data protection services [IRW 05, IRW 09].

– Auditing: it is a process of recording security events related to the accounting and traceability process. It can provide historic information about when and how a user accessed the assets, and whether there were any attempts to violate authentication policies. The historic information of user status is stored in the log files for further analysis.

– User management: the area of user management in the IAM context is not only related to user management but also covers password management, role management and user provisioning. User management is one of the authentication features that provides administrators with the ability to identify and control the state of users when logged into the system. It encompasses a set of administrative functions such as identity creation, propagation, and maintenance of user identity and privileges. This enables the administrator to have better granularity to control the user authentication and manage the lifespan of user accounts through user lifecycle management, thus ranging from the initial stage of authentication to the final stage when the user logs out of the system. The flexibility provided by user management allows

administrators to implement IAM efficiently with a closer match to the security policy.

In addition, user management incorporates the *central user repository*, providing storage for user data and data delivery to other services when it is required. The aggregation of data is kept and maintained in the repository. The user repository can be located either in a distribution network composed of multiple databases/files or a local area directly accessible by the user without having to travel across the network. An example of a central user repository is the lightweight directory access protocol (LDAP) [YEO 95] that is an industry standard application protocol for accessing and maintaining distributed directory information over the IP network. The concept of LDAP is based on a hierarchical information structure (a simple tree hierarchy) in order to deal with several kinds of information stored in the directories. Starting from the root directory (the source of the tree), it branches out to, for example but not limited to, countries, organizations, organizational units (e.g. divisions and department) and individuals (e.g. users, files and shared resources).

4.2. Case studies

We exemplify the applications of IAM in several typical scenarios, illustrating their implementation, deployment and management, so that NFV-based implementations can be compared.

IT scenario. In [RAN 07], the authors reported the problems that were experienced by the South African Social Security Agency (SASSA), which is responsible for distributing grants to underprivileged citizens. It has been estimated that approximately 187.5 million dollars are lost annually due to fraud.

According to the social grant distribution in the South Africa, the organization consists of four main components: (1) South African Department of Home Affairs (SADHA), which is responsible for issuing to South African citizens; (2) South African Social Security Agency (SASSA), which is formed by the Ministry of Social Development to distribute grants; (3) distribution companies, which are

responsible for the actual payment of the social grants to the eligible recipients; and (4) social grant recipient. In particular, there are two types of processes related to the social grant recipient: the registration process and authentication process. In the registration process, all the recipients must be registered with a payment system. Four good fingerprints from the recipient and the recipient's information (e.g. recipient's photograph, biometric data, type of grant they are eligible to receive and history of payment) are stored in the databases. This personnel information is also replicated and encoded onto the smart card before issuing the smart card to the recipients. Once enrollment into the company's database is complete, the recipient can be paid the grant. The authentication process is activated when receiving grants, the smart card is swiped and the beneficiary places their fingers onto a biometric reader. The fingerprints are verified with the fingerprint's information stored in the database and those encoded in the smart card. If the authentication is successful, the recipients can receive the financial grant.

4.2.1. *Telco scenarios: mobile devices and networks*

In [ARD 06], location-based access control policies were proposed for telco scenarios by considering both users' location and their credentials. Compared with the conventional access control systems, more parties are involved: requesters, the access control engine (ACE) and the location service, as shown in Figure 4.1:

– Requesters: whose access request to a service must be authorized by a location-based access control (LBAC) system.

– Access control engine (ACE): if the evaluation result of access requests is matched to LBAC policies, then the ACE enforces access control to the available services.

– Location service: which provides the location information to ACE, by measuring position as well as the environmental condition of requester.

Technically, ACE receives access requests, evaluates policies and returns answers. It communicates with the location service to acquire

the location information of the requester. To describe how the access control has been operated, the authors define an access control rule with 4-tuples of request form (*user_id, SIM, action, object_id*), where: *User_id* is an optional identifier of the requester who makes the request; *SIM* is the optional SIM card number; *action* is an action being requested; *object_id* is the identifier of the object on which the user wishes to perform the action. Thus, access is granted if the subject expression evaluates to 'true' for every applicable rule.

Figure 4.1. *Location-based conditions in access control policies [ARD 06]*

However, user privacy in location-based services remains an important issue [CHO 09]. With an untrustworthy location service provider, the revealed private location information of the requester could be abused by adversaries. Therefore, location privacy-based anonymity solutions for the purpose of blinding user's requests/queries were proposed by [TEE 10], allowing requesters to send requests or queries to the LBS servers without revealing their personal information.

The proposed framework is classified into two major parts: authentication and querying processes; both the processes are done via *anonymity (trusted third party)* as described in [MAL 08].

– Authentication process: during the authentication process, a one-way hash function technique has been applied to provide better privacy authentication. In addition, location blurring (or *K-anonymity*) is used

to hide the actual location when the requester needs to interact with an untrusted service provider.

– Querying process: in the querying processes, time fuzzy logic is used to examine the degree of confidence about whether the requester is requesting the service under the right privileges.

4.2.2. *Public clouds*

Due to the fact that cloud service providers may have different users, access control and user identity privacy protection is extremely complicated in the multi-tenant environment. Therefore, in [XIO 13], the authors proposed an approach called *privacy preserving access management (PRAM)* to address identity privacy and access control concerns in cloud services by: (1) using both blind signature and hash chains which are used to protect identity privacy and secure authentication; (2) integrating on-demand access control with a service level agreement (SLA) to provide flexible fine-grained access management. As shown in Figure 4.2, the PRAM consists of five components: users, cloud service provider, registration servers, authentication and policy decision point (PDP).

– Users: the first time, a user U must register at a registration server RS, which issues authorized credential SID to U. This SID will be used for further authentication with the PDP when a user attempts to access a cloud service.

– Cloud service provider (CSP): which is in charge of providing the cloud service data to authorized users.

– Registration server (RS): which is responsible for the registration of all the users and all the kinds of cloud services.

– Authentication: which refers to the process of determining whether U is who they claim to be. In order to evaluate the access control decision, PRAM adopts the attribute-based access control mechanism [JIN 12] and the access control policies stored in the policy repository of PDP. If the authentication is successful, PDP allows U to access the requested service and relays this decision message back to PEP.

– Policy decision point (PDP): which is connected with the access control policy repository and policy enforcement point (PEP). In particular, the PEP is responsible for receiving message requests from U, forwarding this message to PDP for taking decision, and finally returning the decision result back to U. When PDP receives the requested message from U, it then first authenticates U to access the requested service in the cloud. The evaluation is done based on the description of the user's attributes and SLA. The PDP finally issues the decision result to PEP, which then uses it to inform U (access or reject).

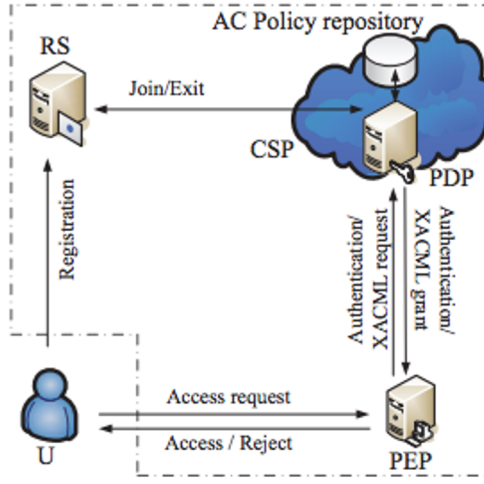

Figure 4.2. *The PRAM architecture [XIO 13]*

Although a lot of IAM schemes for public clouds [IRW 09, XIO 13, GHA 13] have been designed, there are no standards available. Basically, the designs need to meet the following requirements, as suggested by the authors of [YAN 14]:

– Strong and flexible authentication: one-time password (OTP) and multi-factor authentication should be available as alternative options.

– Data loss prevention: it should be able to monitor, protect and verify the security of data during processing as well as stored in the cloud.

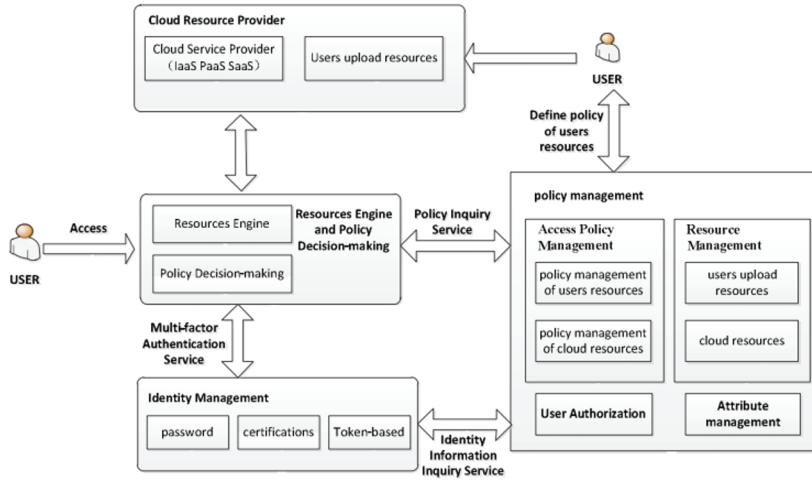

Figure 4.3. *Identity and access management architecture [YAN 14]*

To meet the requirements, the authors proposed an IAM architecture, as shown in Figure 4.3, which consists of four components: cloud resource provider, identity management (IdM), policy management (PM) and resource engine and policy decision-making (REPD). Their specific roles are explained as follows:

– Cloud resource provider is responsible for providing access to resources based on a user's asserted identity and privilege.

– Identity management (IdM) is used to manage users and their identities, issue credentials, and authenticate and assert the user's identity.

– Policy management (PM) enforces access rules that associates users with resources. In particular, it ensures that provisioning requests conform to the policies that are defined through four functions: attribute management, user authorization, resource management and access policy management.

– Resource engine and policy decision-making (REPD), which has two functions: (i) determining whether to allow users to access the requested resources and (ii) finding resources that meet user request. After REPD receives user requests, the REPD submits the

authentication request to the IdM. If authenticated, the REPD then submits a query to PM. Once authorized, the user can gain access to the requested resources, otherwise they are denied.

4.2.3. *Collaborative network scenarios*

Several organizations have recognized the benefits of being involved in inter-organization, multi-disciplinary and collaborative projects that may require diverse resources to be shared among participants. However, the conventional IAM solution is not sufficient to support robust and flexible access control in such collaborative network scenarios. In [RUB 15], the authors proposed a federated and distributed access management to support automated resource sharing in the collaborative network environment, allowing each entity to possibly implement their own security domain and their own dedicated federated access management infrastructure. In particular, resource providers are required to guarantee that the information and resources have been released only to trusted collaborators within the community. To do that, various types of security policies are specified to ensure the degrees of assurance. Figure 4.4(a) presents its framework and Figure 4.4(b) illustrates the process of federated access management by defining three main components: actor, targets and context.

– Actor: refers to end users (e.g. human agents) or subjects (e.g. computer processes) acting on behalf of users. When users request access to the resources/services, it involves access control entities. The access control entities then contact local attributes (if the request exists in the same security domain) or federate attributes (if across collaborative network environments) to check whether users have the right privilege to access the requested resources. However, local attributes are related to federated attributes through *attribute derivation rules (AD-rules)*, which define how local attributes are ultimately related to federated ones. The AD-rules can be organized into a graph-like structure, known as an *attribute derivation graph (AD-graph)*, which presents how attributes are related to permission.

– Targets: targets are the protected resources within a security domain.

– Context: context is the running environment, e.g. operating system and supporting platform, where a given request is issued.

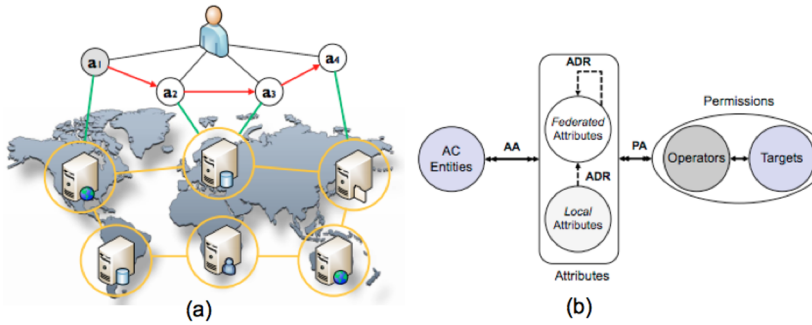

Figure 4.4. *(a) A federated access management framework and (b) a model for federated access management [RUB 15]*

4.3. NFV-based implementations

To date, we have seen some efforts on virtualizing IAM mechanisms through the use of virtualization technology to support a large number of VNFs running in a NFV environment. For example, in [JAC 14], access control virtualized network function (AC-VNF) is proposed to control a large number of VNF appliances and authenticate the end users. The objective of AC-VNF is the provisioning of security services and providing more concrete access control to the services and policy enforcement. In particular, the authentication and authorization mechanisms are required to verify the VNF appliances whether they are eligible to access the requested resources or not. Furthermore, the service providers who own network infrastructures and share resources can define arbitrary security policies based on their needs, so each VNF appliance can be controlled independently according to those pre-defined policy rules.

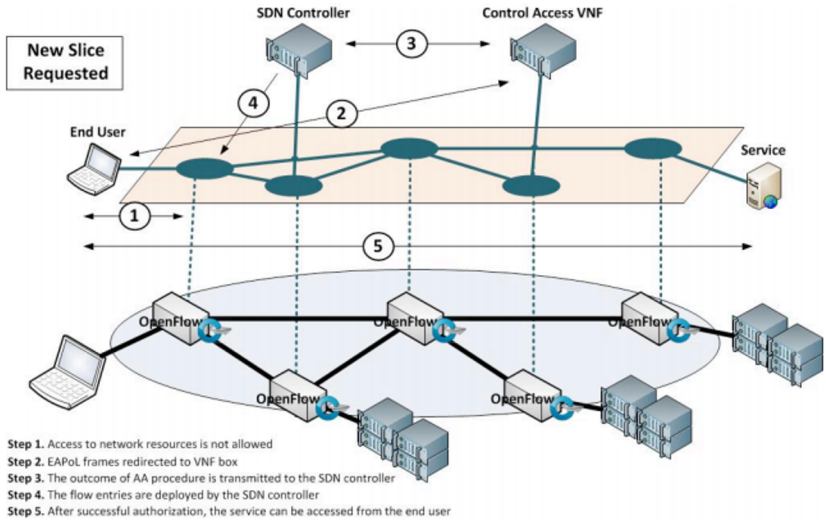

Figure 4.5. *An access control VNF procedure [JAC 14]*

The AC-VNF is implemented based on the IEEE 802.1X standard (IEEE standard for port-based network access control (PNAC)) and a modified version of the standard to implement the access control per service (instead of per port). Figure 4.5 shows the AC-VNF architecture interacting between four different domains: (1) *End user*, who requests for the services and the request can be initiated both in a VM from another virtual server or a physical PC. The machine running the user's request is directly connected to one of the physical ports of any OpenFlow switches; (2) *SDN controller*, which is a software application maintaining the state of user traffic. It enforces the flow entries by enabling or disabling them to access the requested resources/services. In addition, it performs two tasks: redirect user's request to access control VNF box for further authentication and redirect end user to requested services after the user has been successfully authenticated; (3) *Access control VNF*, which is built on a VM with a Linux distribution and uses the modified versions of HostAP [HOS 12] and WPA supplicant [WPA 12]. It acts as software authenticator and implements the IEEE 802.1X-based authentication and authorization (AA) mechanism. Once the AA procedure succeeds, the access control VNF box then transmits the evaluation result to the

SDN/NFV controller; (4) *Service*, which is a specific type of resource provided by CSPs. The owner of resources can control them independently, using different security procedures.

To summarize regarding the automated deployment of an access control VNF, in which it is possible to activate or deactivate the access control VNF without disrupting other services: network operators can easily reconfigure the access control VNF by using the provided tools. This approach is particularly well adapted to a stateless SDN/NFV infrastructure, as the access control VNF is able to store the data needed for the AA procedure. However, when considering the design, it needs to be carefully addressed in terms of the configuration, not only for the VMs (supporting deployed access control VNF), but also the underlying infrastructure for maintaining the proper isolation.

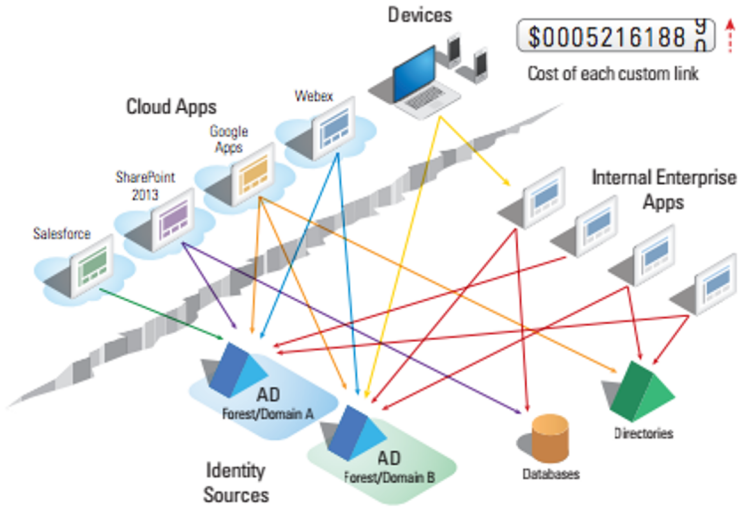

Figure 4.6. *N-squared problem with traditional identity infrastructure [LOG 14]. For a color version of this figure, see www.iste.co.uk/zhang/networks.zip*

Radiant Logic Inc. [LOG 14] pointed out that identity infrastructures face the traditional challenges of multiple links to multiple sources and targets, which create an unmanageable "n-squared" problem,

as shown in Figure 4.6. The problem implies that there are many custom links, each one is extremely expensive to manage and deploy, inflexible to maintain, and complex in terms of hard-coded point-to-point connection. To solve the problem, the authors proposed a virtualized IAM solution based on the federated identity system, which provides flexible management, increases security, flexibility and end user experience with seamless single sign-on. In particular, a large number of heterogeneous and distributed identity modules (e.g. active directory, LDAP directories, databases and APIs) can be presented as a global view of identity and attributed over the federation access layer. The federated identity can be operated on the fly for further user evaluation. It adopts federation standards such as SAML 2.0 (security assertion markup language) [SAM 08], OpenID Connect [CON 14] and OAuth 2.0 [HAR 12], which are designed to better manage security and address complexity. Figure 4.7 shows the concept of a federated identity system based on virtualization and the architecture involves two parties:

– Service provider (SP), which provides the functionality of the application and controls the access to the resources. It delegates authentication and attribution management to a trusted external identity source.

– Identity provider (IdP), which manages user's identities, their profiles, and provides users with access to the new applications. In particular, the IdP adopts a common authentication method (based on standard-based tokens) for verifying user identity and an identity hub is leveraged to support the IdP for easier and more efficient routing. The identity hub has a central repository (e.g. virtual meta-directories) for integrating and synchronizing all the identity directories from heterogeneous systems via an abstraction layer.

Another example design is reported in [WRA 10], which proposes content-aware identity and access management solution to protect user information in a virtualization environment by controlling identities, access and information usage. The major advantages include: (1) privileged user management; (2) fine-grained access controls on virtual hosts and guests; (3) enhanced user activity and compliance

reporting; (4) sensitive data discovery and information protection on virtualized systems; (5) extension of identity and access management capabilities to virtual system and applications. The architecture is shown in Figure 4.8, which contains three major parts:

– Role and access policy management: it is prespecified and defines different roles for accessing the VMs.

– System and application access: these applications will be running on the VMs and monitored by security apps.

– Virtual system and applications: which consist of three main modules: (1) *privileged user management*, which controls privilege users (e.g. role-based) along with fine-grained access control; (2) *compliance reporting*, which collects users' activities from all the event logs; (3) *information protection*, which facilitates the management of sensitive data.

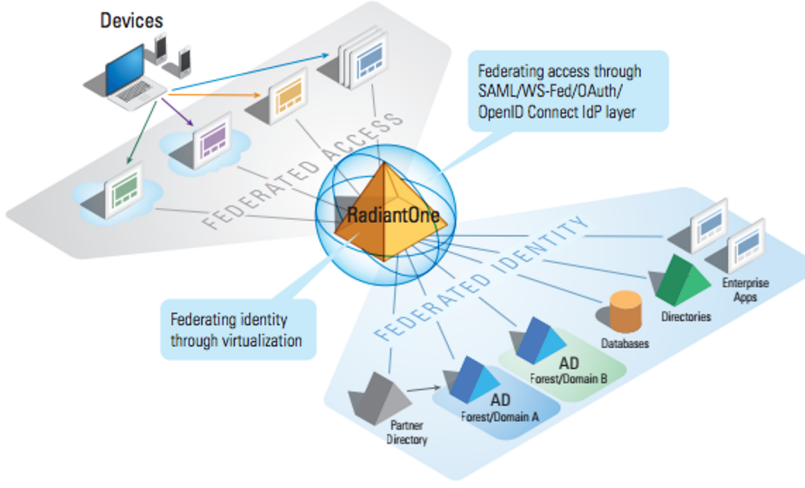

Figure 4.7. *Federated identity system through virtualization [LOG 14]. For a color version of this figure, see www.iste.co.uk/zhang/networks.zip*

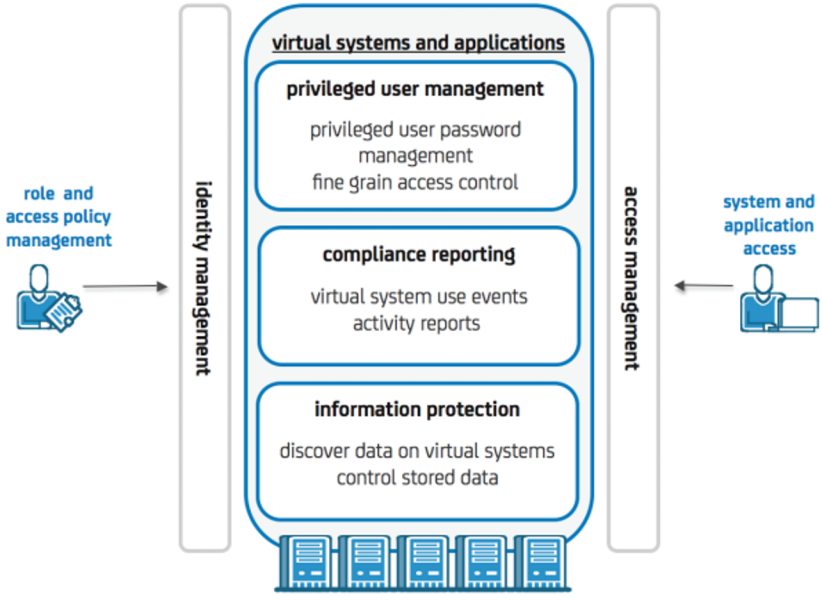

Figure 4.8. *A content-aware identity and access management through virtual environment [WRA 10]*

4.4. Comparative analysis

To compare the legacy IAM mechanisms with their implementations in NFV, we take the designs proposed by [JAC 14, LOG 14, WRA 10] as reference and analyze the following metrics, while the results are summarized in Table 4.1:

– Specific functions: the access control VNF box is implemented based on virtualization technology to provide security services and ensure that access privileges are granted to the right resources according to the policy rules. These policy decisions are centralized in a single point to assure the coherence in the definition. In [JAC 14], the AC-VNF is implemented based on FlowNAC [MAT 14], inspired by the IEEE 802.1X standard of the port-based network access control (PNAC), which is a basic NAC solution for enforcing the access control at the port level (e.g. the physical port of the network node). It is designed

to maintain the whole authentication and authorization (AA) process in the data plane to further avoid overload on the control plane. When end users are requested to access the services/resources, this associated traffic coming from the end users will be redirected to the access control VNF box, which then identifies the end users (e.g. authentication) and applies a policy (e.g. authorization) as a result of a decision made at the policy decision point (PDP). The outcome of AA procedure is then transmitted to the SDN/NFV controller. Once the end users are authorized, the flow-based port entries are generated, thus the end users are allowed to access the requested services.

More specifically, the access control service relies on a virtualized infrastructure, implemented based on layer 2 (MAC address). Thus, one physical port is shared by multiple users through the use of virtual port (each virtual port is identified by the MAC address of the user) so that the users are isolated virtually. Although it is acceptable in terms of flexibility, cost and physical layer independence, there are limitations on capacity, security and overhead. It is worth noting that VLANs can possibly create 4094 different VLANs for the same network, and each VLAN is assigned a unique ID between 0 and 4093 [CIS 12]. Thus, if a network spans more than one geographical location, the traffic needs to go through the third parties. This exposes the traffic to potential sniffing and man-in-the-middle attacks, which are hard to deal with, unless higher layers offer additional security mechanisms. Additionally, if the VLANs rely on port-based or MAC-based configuration, it requires lots of effort and time to manage the network.

An alternative solution to improve the VLAN limitation is VXLAN [MAH 14], which can be implemented to support flexible and large-scale virtualized multi-tenant environment over a shared common physical infrastructure like a cloud.

– Cost: it is well recognized that NFV can reduce the total cost of hardware acquisition and capital investment through the use of commodity standard hardware platforms. For example, HP reports that the cost for a small-scale NFV deployment can be reduced from $34,015 to $27,828 (about 18%), whereas for a large-scale deployment cost can be reduced from $18,935 to $14,435 (about 24%) [HP 14a]. It is worth

mentioning, however, that although the cost for hardware investment, installation, configuration and power consumption is decreased, the software cost could increase.

The example design reported in [LOG 14] clearly illustrates the advantages of NFV-based IMA implementation for solving "n-squared" problems, as shown in Figure 4.6. In particular, using IAM based on virtualization can significantly reduce the costs from hardware investment and management complexity, while access policy rules are allowed to be redefined and reconfigured remotely in real time.

– Complexity of lifecycle management: instead of deploying network equipment at customer sites and using them to provide a set of pre-defined services, NFV makes it possible to deploy hardware at provider sites and provision services dynamically using centralized management tools. Therefore, both cost and complexity resulting from service deployment and configuration can be reduced. Thus, security functions like virtual IAM can be quickly deployed and easily managed through a centralized management platform. As shown in the given examples [JAC 14, LOG 14, WRA 10], with IAM based on virtualization, administrators can flexibly deploy access control boxes, define security policies, assign specific actions according to user's attributes and role-based access control on the fly. NFV-based IAM especially allows security administrators to keep better control over the upcoming users' requests and offer efficient fine-grained policy enforcement.

– Effectiveness: if there is a large number of independent VNF appliances deployed in the NFV infrastructure, the effectiveness of management and communication overhead will become an important concern. A question arises regarding the location where an access control VNF box should be installed and the number that has to be deployed in order to achieve the best performance. Let us revisit Figure 4.5: if the end users are located closely to the requesting service's server, while the access control VNF box is deployed on a VM which is distant, then the SDN/NFV controller redirects the network traffic to the access control VNF box, regardless of the physical location of the end users. As such, a very large volume of traffic over virtual network switches and routers would be created, consuming lots of bandwidth.

This clearly indicates that it is necessary to take into account the trade-off between security objectives and performance efficiency during the deployment of access control VNF box.

– Availability (single point of failure of a centralized controller): in the example designed in [JAC 14], a controller is deployed to redirect all the traffic to an access control VNF box. While the centralized management can get a global view of the network, the platform or controller itself could be overloaded, failed or possibly attacked. As a result, the overall network latency will be heavily degraded. To share the processing load, a distributed control scheme was proposed in virtualized networks [ZUC 15]. In particular, multiple controller instances are created and organized in *clusters*, while the benefits of centralized network control remains. As such, the network control workload can be distributed across the cluster by deploying each instance of SND/NFV controller on dedicated VMs. Thus, a better trade-off between scalability and centralization can be achieved.

– Scalability: theoretically, NFV can make network services agile, cost-effective and scalable, allowing network operators to dial up/down network capacity as demand changes and providing elastic scale in/out of network resources assigned to VNFs depending on the dynamic traffic load and resource management. For example, in [JAC 14], a NFV-based scalable service deployment is proposed, where an access control VNF box can be activated or deactivated in real time without disrupting other services. Furthermore, the authors of [WRA 10] proposed a scalable access management to help network operators to manage the risk of data loss, allowing all the user requests and resources to be controlled efficiently, and policy rules to be dynamically inserted into existing VNF IAM box. Since the virtualized IAM box usually contains a large number of databases and directories, which are used to store user identities, attributes and policy roles, a scalable virtual directory is presented in [LOG 14], which aims to reduce the complexity of meta-directories when they are deployed or upgraded. The identities across heterogeneous sources can be easily integrated, providing a logical view, no matter where or how they are stored.

Example design of IAM	**Easy deployment (no specific function required)**	**Cost saving**	**Complexity reduction of lifecycle management**	**Effectiveness**	**Availability (e.g. single point of failure)**	**Scalability**
IAM for social grants in South Africa [RAN 07]	/	X	X	/	X	X
Location-based access control policies [ARD 06]	X	X	X	X	X	X
Access control-based anonymous location [TEE 10]	/	X	X	/	X	X
PRAM: privacy preserving access Management for cloud [XIO 13]	/	X	X	/	X	/
IAM solution for cloud [XIO 13]	/	/	X	/	X	/
Federated access management for collaborative network [RUB 15]	X	X	X	X	X	/
Agent-based framework for access control and trust management [SHA 05]	X	/	X	/	X	/

Access control virtualized network function (AC-VNF) [JAC 14]	X	/	/	X	X	/
Virtualized IAM solution based on federated identity service [LOG 14]	X	/	/	/	X	/
Content-aware identity and access management [WRA 10]	-	/	/	/	X	/

Notations: "/" and "X" denote that NFV characteristics are satisfied and not satisfied, respectively; "-" means data sources are insufficient.

Table 4.1. *Gap analysis between conventional and virtualized IAM*