A Novel Zero-Trust Network Access Control Scheme based on the Security Profile of Devices and Users

P. García-Teodoro, J. Camacho, G. Maciá-Fernández, J.A. Gómez-Hernández,V.J. López-Marín

Network Engineering & Security Group (https://nesg.ugr.es)
CITIC - University of Granada
Email: {*pgteodor,josecamacho,gmacia,jagomez*}*@ugr.es, victorjlopez@correo.ugr.es*

Abstract

Security constitutes a principal concern for communication networks and services at present. This way, threats should be under control to minimize risks over time in real environments. With this aim, we introduce here a new approach for access control aimed to strengthen security in corporate networks and service providers related environments. Our proposal, named SADAC (*Security Attribute-based Dynamic Access Control*) presents three main novel features: *(i)* security related attributes regarding both configuration and operation are considered for network access control of final devices/users; *(ii)* a dynamic supervision procedure is implemented to evaluate the security profile associated to devices/users over time and, if so, to apply corresponding access restrictions; and *(iii)* a supervision procedure that also permits to diagnose the causes of inadequate security behaviours, so that the final devices/users can adapt their configuration and/or operation. We describe the overall access control methodology as well as the aspects for its implementation. In particular, we present and evaluate the specific deployment of SADAC for a corporate WiFi environment supported on a Raspberry Pi-based AP to provide Internet access to mobile devices. Through this experimentation we can conclude the convenience of adopting the approach for improving security by minimizing risks in network and communication environments.

Keywords: Access control, Anomaly detection, Security profile, Zero Trust Access

1. ICT Security: A Right and A Commitment

The social and economic dependence on the information and communication technologies (ICTs) is continuously increasing. Society has not only taken advantage from the various ICT benefits but it has also inherited their (not few) limitations and drawbacks. One of such problems is that of security, since systems and services are becoming more complex and widely used and, as a natural consequence, the associated exposure to vulnerabilities and threats is increasing [1, 2].

Security risks are mainly derived from vulnerabilities in software and systems [3], but also human action is a principal way of infection in the digital world [4, 5]. Social engineering is increasingly becoming a relevant methodology to succeed in attacks, either targeted or discretionally [6]. This way, malicious or even unconscious users' behaviour can put into risk other users and systems, *e.g.*, by installing inadvisable software, visiting non-recommendable websites, resubmitting malicious ads and links, etc.

In summary, either due to users' unawareness or conscious malice, it is not recommended that ICT security solely relies on final users and/or devices. This is especially critical in certain environments like those with mobility capabilities, where users' personal devices are entitled to connect and share corporate infrastructures and resources, which usually gives way to the adoption of policies like BYOD (*Bring Your Own Device*) [7, 8]. Likewise, IoT related environments are especially prone to become victims of security threats [9, 10] due to: *(i)* the autonomous operation of the final devices, and *(ii)* the high exposition to (even simple) attacks because of their low computation capacities. In all these cases, individual malicious behaviours can put into serious risk an overall organization or network. Given the expected ubiquity of IoT in the future Internet, this poses a major challenge.

In this context, the usual security mechanisms deployed by corporate networks and ISP (Internet Service Provider) administrators are mainly focused on providing authentication, confidentiality and integrity through solutions like AAA, WPAx, etc., which is not enough to fight against malicious behaviours as evidenced by the impact of continuous security incidents around the world and the consequent social alarm generated on this topic. Hence, we postulate the necessity of strengthening security by deploying more ambitious pro-active solutions, just because networks and communications security should be both a right and a commitment, for users and providers.

Our specific proposal consists on a zero trust access control procedure in which, in addition to traditional access control schemes (password-based, use of digital certificates, etc.), the security profiles of users and/or devices are dynamically estimated and then considered as an additional factor to decide if allowing, limiting or even denying access to resources and network services over time. The approach is named SADAC (standing for *Security Attribute-based Dynamic Access Control*), with the following remarkable contributions:

- Security related attributes or parameters are considered to take decisions about the access of users and devices to common infrastructures and resources.
- Such decisions are taken dynamically over time during the communication and not only (as usual) at the initial association of the device/user to the network.
- In order to take access control decisions, SADAC makes use of MSNM (*Multivariate Statistical Network Monitoring*), a machine learning (ML) monitoring and anomaly detection solution introduced by the authors with a main proved benefit: MSNM provides diagnosis capabilities on the user/device side, which allows to identify the causes of a detected problem, thus helping to solve potentially harmful situations.

According to the above, the rest of the paper is organized as follows. Section 2 discusses fundamentals on access control schemes, with special emphasis on ABAC since it constitutes a basic flexible and promising (despite its existence for a number of years) access model. Based on that, Section 3 introduces SADAC as a novel security-based implementation for access control, which relies on two main elements:

- A dynamic policy enforcement point capable of dynamically granting or not the access of a device to the network environment, as described in Section 3.2.
- An MSNM anomaly-based detection module, which estimates the security profile exhibited by users and devices in order to dynamically decide about the previously referred access (Section 3.3).

After the description of the overall system, Section 4 presents the experimentation carried out to evaluate the access control proposal in a real

WiFi network corporate environment for mobile devices. Based on that, a further discussion about some practical deployment aspects for SADAC is provided in Section 5. Finally, Section 6 presents the main conclusions and contributions of the work.

2. Related Work on Security Provision Through Access Control

Access control is a principal security mechanism which assures that only authorized subjects can access to certain resources for a given action when specific environment conditions are accomplished [11]. In this definition, four elements are essential: *subjects* (*e.g.*, users, devices), *resources* or *objects* (*e.g.*, web pages, bank accounts, database records), *actions* (*e.g.*, read, write, execute) and environment *conditions* (*e.g.*, date, location).

In 1994, Sandhu *et al.* [12] presented the fundamentals and principles of access control implementations, describing different known models that conform the basis for most of current access control implementations [11]:

- *Mandatory Access Control* (MAC). Both subjects and resources are assigned a security level. The security level of a resource reflects the sensitivity of its information, while that of the subject (also named *clearance level*) indicates the confidence level on it for not disclosing sensitive information.

- *Discretionary Access Control* (DAC). Rules termed *Access Control Lists*, ACLs, are assigned to provide access to every resource for specific subjects. For every subject and object, these rules specify which actions are allowed.

- *Role-Based Access Control* (RBAC). Roles are defined to group subjects or resources. Then, ACLs are assigned to govern the access between the different groups. This is an intermediate model between MAC and DAC, as it provides greater flexibility than MAC while it is more manageable than DAC. This advantage has boosted its acceptance in corporate security management scenarios.

An example of the MAC scheme is the typical classification of data in the hierarchical labels: Top Secret, Secret, Confidential and Unclassified. DAC and RBAC schemes have also been implemented in most common operating systems (Linux/Unix, OS X and Windows). Although previous access control schemes are currently well known and widely utilized, many authors agree that they suffer from certain limitations [13], mainly due to the fact that they only consider subjects' identity to govern the access control while many other relevant aspects remain unconsidered, namely: *(i)* subjects' attributes other than identity (*e.g.*, geographical location, device used for access, dynamic reputation of user, etc.), *(ii)* environment conditions (*e.g.*, time/date of access, network congestion, business state, etc.), or *(iii)* resource attributes (*e.g.*, service state, business conditions, etc.).

For this reason, jointly with the emergence of the *Service Oriented Architecture* (SOA), a new model was proposed in 2003 through the OASIS standard body called *Extensible Access Control Markup Language* (XACML). This approach was termed *Attribute Based Access Control* (ABAC), and it consists precisely in the definition of ACLs that consider different attributes of either subjects, resources, actions or environment conditions.

After this proposal, and due to a lack of consensus about ABAC features, an extensive guide to ABAC was contributed in 2014 by the National Institute of Standards and Technology (NIST), which was updated in 2019 through [14]. First, this document provides a definition and a description of its functional components. Second, it discusses the implementation process of ABAC within a corporation. The focus of the guide is on illustrating the main challenges that appear in such an implementation.

According to [14], the main functional elements of the ABAC architecture are (see Figure 1) :

- *Policy Information Point* (PIP), which is in charge of the retrieval of attributes from subjects, resources and environment.
- *Policy Administration Point* (PAP), which generates, manages and stores the policies into a policy database.
- *Policy Decision Point* (PDP), which takes the decisions on access control by evaluating the applicable policies.
- *Policy Enforcement Point* (PEP), which enforces the actions decided by PDP as a response to an access request from a given subject to a protected object. Some possible actions to be taken are allowing, denying, restricting or logging access.
- *Context Handler*, which is an optional component aimed to execute the workflow logic that defines the order in which policies and attributes

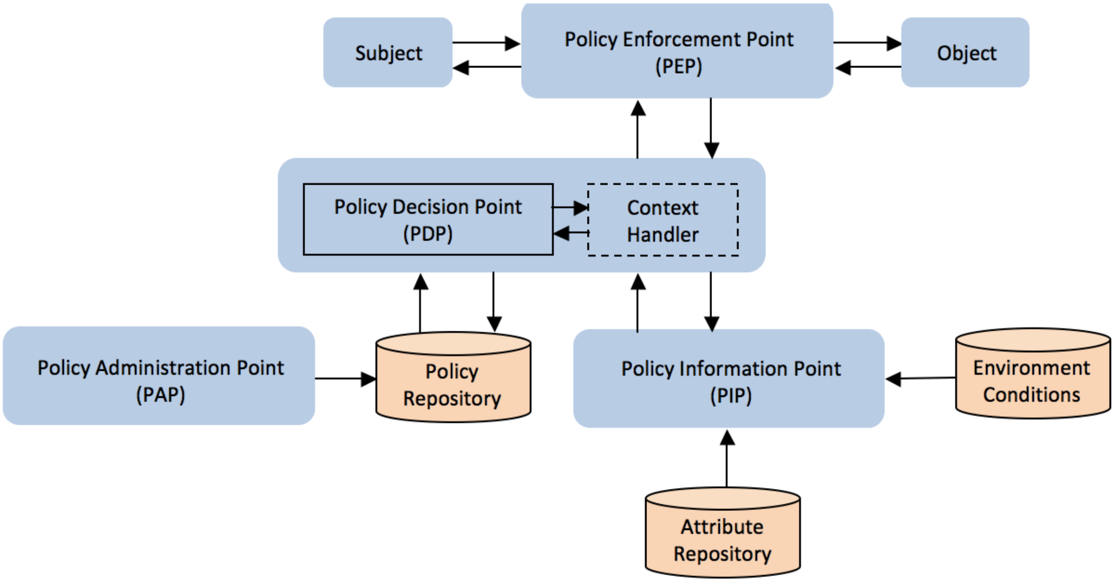

Subject
Policy Enforcement Point (PEP)
Object
Policy Decision Point (PDP)
Context Handler
Policy Administration Point (PAP)
Policy Repository
Policy Information Point (PIP)
Environment Conditions
Attribute Repository

Figure 1: Functional elements of the ABAC architecture.

> are retrieved and enforced. For example, attributes may be retrieved in advance of an access request, or cached to avoid the inherent delay in the retrieval process at the time of the access request.

Despite the long time since ABAC exists, this access control scheme is still considered a “next generation” authorization model because it provides robust, context-aware access control to resources. However, only few implementations for access control schemes in general and ABAC in particular are contributed until now. Specific use cases include cloud storage and the Internet of Things [15, 16, 17, 18, 19, 20], where the use of blockchain technology is recurrently considered to prevent data from manipulation or unauthorized access [21, 22, 23].

It is worth to note the use of ABAC in the Zero Trust Architecture (ZTA) [24, 25, 26]. In this model, it is assumed that all the devices in a given environment are unreliable so that it is required they get continuously authenticated based on different types of attributes. In [27], authors present some challenges regarding ZTA (*e.g.*, policies, controls, services) and discuss related solutions. Other recent and more specific works in this area are as follows. The work in [28] is focused on Zero Trust Networking (ZTN) with the aim of specifying the design of required policy languages including a generic firewall policy language to express firewall rules. Authors design a mechanism to map these rules to specific firewall syntax and to install the rules on the firewall.

More recently, authors in [29] make use of mutable attributes about subjects, resources and environment as well as trust levels that are continuously monitored to obtain an authorization for consumer IoT. Whenever change is detected, the corresponding authorization rules, including both access control rules and trust level expressions, are re-evaluated. Yao *et al.* propose in [30] a dynamic and fine-grained access control and authorization system to trust users according to their behaviour. Authors in [31] are more focused on establishing the real identity of the user than on access control itself. Also in 2021, authors in [32] propose a novel access control policy based on a zero-trust network by explicitly restricting the incoming network traffic to substantiate MAC spoofing attacks in the software-defined network (SDN) paradigm of cloud computing. The multiplicative increase and additive decrease algorithm helps to detect the advanced MAC spoofing attack before penetrating the SDN-based cloud resources.

In the above overall context, we introduce here SADAC, a novel zero trust network access control scheme where subjects' security related attributes are considered to dynamically authorize them to operate into a corporate communication infrastructure. Experimentally particularized for mobile devices and users, as we will show later in the document, the security attributes are estimated through a ML tool named MSNM, where multidimensional features regarding communications (*e.g.*, ports, IP addresses, data volume, duration), applications installed and permissions involved, resource consumption (*e.g.*, RAM, CPU, battery), and device protection mechanisms (*e.g.*, screen locking method) are analyzed. Moreover, it is worth to mention that SADAC presents diagnosis capabilities to allow identifying specific causes for access restrictions.

3. SADAC: Security Attribute-based Dynamic Access Control

Similarly to concepts like *software-as-a-service* or *malware-as-a-service*, that of *security-as-a-service* (SECaaS) [33, 34] is gaining traction in recent years. SECaaS consists of a business model in which a service provider integrates security services into a corporate infrastructure (generally on a subscription basis) more effectively than most individuals or corporations can provide on their own. These security services can include authentication, antivirus, antimalware/spyware, intrusion detection, penetration testing, and security event management, among others. A brief but interesting report about the topic can be found in [35], as well as in [36] from a market perspective.

In this general context, we introduce here a novel approach for dynamic access control based on security attributes of the subject, which is named SADAC (*Security Attribute-based Dynamic Access Control*) and mainly intended to be applied into corporate networks and ISP related environments according to the general operational workflow shown in Figure 2. The main aspects that characterize SADAC are:

1. Subject (user and device) security level must be known by the network in order to provide access to services and resources (*i.e.*, objects). For this purpose, a *security profile* for every subject is dynamically estimated from certain security related attributes.
2. Such a security profile is monitored over time, both at the beginning of the communication and also during it following a zero trust basis. This way, access is granted (if so) at the beginning and renewed periodically based on the security level obtained for a subject and the accessed object. In case security threats exist, the access can be limited or even, if necessary, denied anytime. In addition, SADAC can provide recommendations to the subject in order to subsequently fix the problems diagnosed.

The practical implementation of SADAC to allow the previous operation implies some specific considerations for the general ABAC architecture in Figure 1:

- *Policy diversity.* Different types or levels of access (*e.g.*, complete, null, partial to some services, limited in speed) could be defined through policy rules into the policy repository.

- *Security related attribute repository.* The subject security profile is dynamically updated over time and introduced into the corresponding attribute repository.

- *Dynamic Policy Enforcement Point* (dPEP). The access levels or privileges defined can be assigned and dynamically modified according to

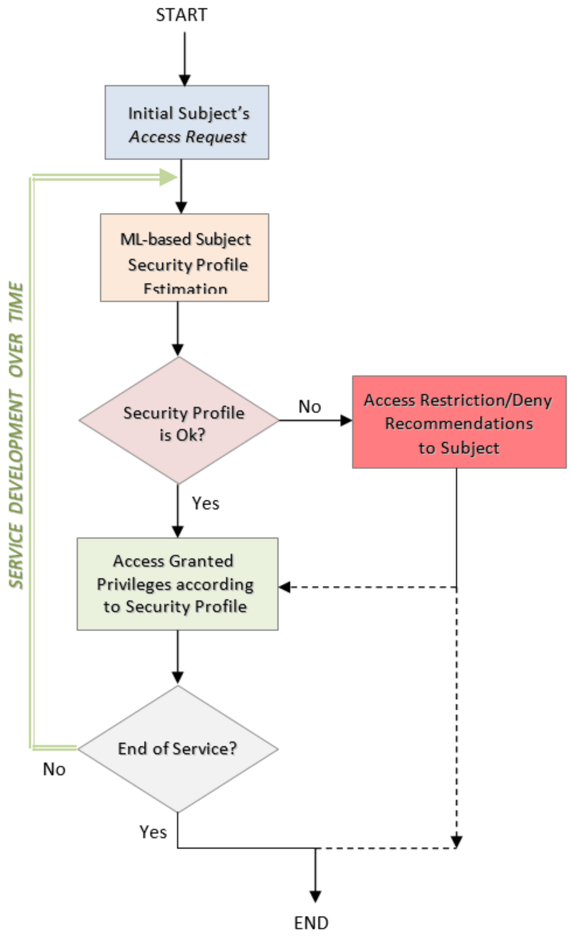


Figure 2: General operational workflow for SADAC.

the subject security profile. For that, the PEP module should be configured to operate dynamically over time and not only when a subject generates an explicit access request to the network. For example, periodically for a given communication, each time a transmission is performed by the subject, asynchronously under demand from an external supervision module, etc.

- *Dynamic Policy Decision Point* (dPDP). The dynamic enforcement of policies (dynamic PEP) implies taking decisions accordingly (dynamic PDP). For this reason, PDP should be equipped with algorithms ca-

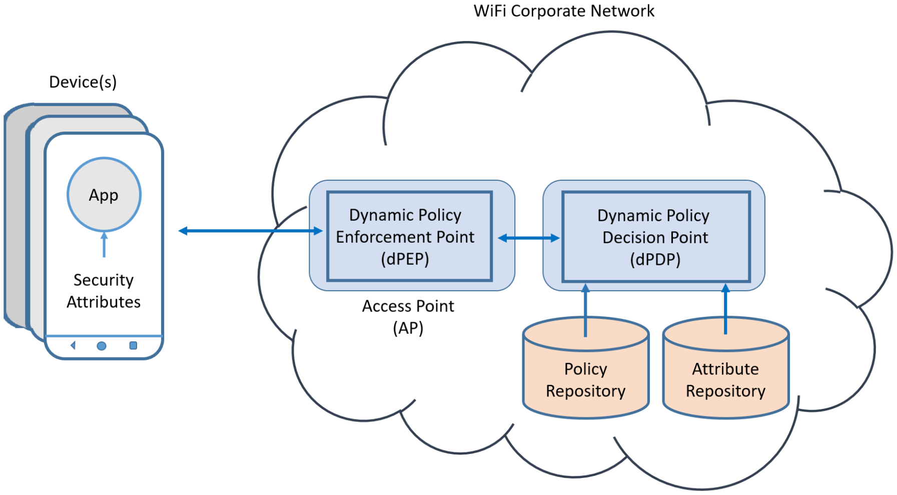


Figure 3: SADAC architecture for WiFi environments.

pable of taking decisions over time based on the evolution of security profiles. Despite the existence of a huge amount of algorithms for this purpose in the literature, we choose MSNM for our implementation [37], as we shall describe below.

We have implemented SADAC here to control the access of mobile devices to a WiFi corporate environment. For that, Figure 3 shows the specific architecture considered, where:

- Each mobile device interacts with the environment through an Access Point (AP). Such interaction is double-faced. On the one hand, a mobile-network dialogue to manage the access itself is carried out. On the other hand, the device provides to the network some specific security attributes or features. This process is eased by the use of a SADAC specific app that might be downloaded from the ISP network to be installed on the device.
- The AP implements the dPEP module, so that:
  1. It receives the security features associated to a given device/user, which can be performed just once at the beginning of the association to the AP or periodically over time.

2. The AP forwards the security features to the dPDP, which will estimate the corresponding security profiles for the device/user.
3. In case the dPDP concludes a given device/user does not accomplish a security policy, it instructs the AP/dPEP to limit or even deny the device continue accessing the network.

- As previously stated, the dPDP module is in charge of estimating the security profile associated to each mobile device from the corresponding security attributes. In addition, it will take decisions about granting or limiting the access to the network. To achieve it, the security attribute repository for the devices on the network as well as the security policy repository established for the environment are considered.

In the rest of the Section, each of the abovementioned entities and elements are better described.

3.1. SADAC Mobile Device App

Devices and users demanding access to the corporate network are required to provide security related attributes as defined by the ISP or the network administrator. For that, a specific app is intended to ease the process. It will act as a part of the *policy information point* in Figure 1 (see Section 2) to dynamically retrieve the required security related attributes.

Based on the existence of the SADAC related mobile app, the interaction between the mobile device and the AP is as follows:

1. As usual in WiFi environments, the device must show a valid credential to the AP (*e.g.*, a shared key in WiFi-Personal systems, or a certificate in WiFi-Entreprise variants) to get the initial access.
2. If the authentication fails, the access is rejected. Otherwise, a temporal network access is provided to the device so that the AP will subsequently establish a connection with the app on the device, as follows:

    - The AP first demands the connection on the specific app port (see next Section for details) for the target device.
    - After confirming a valid version of the app, it will obtain the security parameters associated to the operated mobile/user as defined by the operator (and established in the app program), and will send the attributes to the AP.

- The AP consults the dPDP, which will decide if the security profile of the mobile device/user accomplishes the security policy of the network. According to that decision, the AP: *(i)* will grant the access to the device or, if not, *(ii)* it is rejected and the device notified about the problem. In the last case, the user will be able to solve the problem (*e.g.*, by updating OS or installed applications and permissions) and free to try a new access.

It is important to remark that the mentioned connections (app-AP/dPEP and AP/dPEP-dPDP) correspond to secure TLS/SSL connections, where digital certificates are considered. This allows providing authentication of the parts, as well as confidentiality and integrity for the shared information.

We have developed *AMon* [38] as the specific SADAC related app to be installed on the devices to access the corporate WiFi network. As described in the mentioned paper, AMon does not require root privileges to autonomously collect a number of features of the device it is installed on. The attributes collected by Amon are:

(a) *Configuration related attributes.* They refer to the operative state of the final device in terms of its configuration. Some of the features to be considered here are: running software, OS version and installed patches, changes in the `/etc/hosts` file, saved non-protected WiFi networks, resource consumption, navigation history to blacklisted sites, etc.
Additional specific features for mobile devices are application permissions, jailbreak existence, premium calls or SMSs, security mechanisms to log on the device (PIN number, fingerprint, graphical pattern), among others. Note that the collection of this kind of information is feasible in most devices through the use of libraries like the Android API for developers [39].

(b) *Communications related attributes.* This kind of attributes describe the subject (device/user) behaviour exhibited over time in terms of the network activities carried out: type of communications performed, accessed services, navigation profile, traffic rate, IP addresses visited, packet volume, etc.
Although the simpler solution for a network operator to monitor this information is doing it on the network side, in this case many of the characteristics of network traffic could be hindered by encrypted communications or the use of proxies. For this reason, the monitoring of

traffic in the own monitored device is also recommended (*e.g.*, by means of tools like NetGuard [40]). In addition, as we shall see below, this allows to reduce network resources consumption.

Whatever the specific set of features considered, the sequence of them will serve as the input to the dPDP module at a given instant.

3.2. Dynamic Policy Enforcement Point

As previously stated, the dPEP functionality is integrated into the AP, its main function being to act as an intermediary between the mobile device and the corporate network to manage access to the infrastructure and services by optionally taking into account the security profile of the subject.

Most of the specific operation of the AP/dPEP has been already described. Resuming it: *(i)* first, the AP expects to receive connection/association requests from mobile devices. In such a case, a challenge is demanded as usual (password, credential,...) to the terminal; *(ii)* in case of success, the AP will grant temporal access to the device and then will contact with the SADAC related app installed on the device to demand security attributes at the current time; *(iii)* after receiving the security features associated to the device/user, the AP will forward them to the dPDP, which will estimate the corresponding security profile for the device/use; and *(iv)* in case the dPDP concludes a given device/user does not accomplish the security policy, it intructs the AP/dPEP to restrict the device accessing the network; otherwise, it is allowed to operate on the environment according to its authorization level.

Some relevant specific details regarding our current AP/dPEP implementation are as follows. It is is here developed over a Raspberry Pi Model-B 8GB (RPi OS) by using the tool *hostapd* [41]. This tool has a daemon with a double function: *a)* to implement a WiFi AP, as well as *b)* an authentication server.

This tool has two modules to deploy the AP: *wpa_supplicant* and *hostapd*. The latter includes the packages *dnsmasq*, *netfilter-persistent* and *iptables-persistent*, which are related to network administration and management. Moreover, the file *hostapd.conf* contains some parameters to configure the AP operation: SSID, encryption type, etc.

Additionally, our specific implementation includes three novel parameters:

- *security_level_check*, to activate the access control based on the security profiles of the devices or, if not, to work like a standard AP. That is, the SADAC functionality will be deployed or not through this configuration parameter.

- *slc_port*, to set the communication port used by the SADAC related app installed on the mobile devices.

- *slc_interval*, to fix a time interval (in seconds) used by the AP to periodically collect the security attributes from devices. This parameter is so important because it defines the dynamic nature of the PEP module. If this parameter is set to 0, the security attributes are demanded just once at the association instant of the mobile with the AP.

3.3. Dynamic Policy Decision Point

As previously discussed, the dPDP module is in charge of estimating the security profile associated to each mobile device and, from it, to take the decision about granting or limiting the access to the network. This way, the dPDP module can be easily deployed as a corporate/intranet service with direct access to the AP/dPEP to:

1. Receive security features from mobile devices (via the AP), thus creating a security attribute repository.
2. Estimate the associated security profile for them.
3. Obtain a normality model for the overall environment and, thus, derive a behavioural policy.
4. Instruct the AP to accept or limit the access according to the individual profiles and the security policies considered.

The most novel and relevant issue at this point concerns the subjects' security profile itself, both regarding its estimation over time and the decision taking procedure involved in the dynamic access control. SADAC deals with these two aspects through MSNM as described in what follows (for more technical details, see [37]).

3.3.1. Security profile estimation

As previously indicated, security profiles are built by using a ML-methodology known as MSNM. As in any typical ML-based estimator, two main elements compose MSNM:

1. *Feature generator*, which is in charge of extracting the characteristics that represent the state of the subject. The set of characteristics at a given instant t are defined as an observation, o_t.
2. *Estimation module*, which determines whether a given observation accomplishes or not the expected behaviour defined for the target system by doing a comparison with a 'normality' model.

- *Feature generation*

The collected attributes or features for a given subject over time are parameterized in MSNM following a *feature-as-a-counter* (FAAC) basis, which means that the value of a feature is a counter of the number of times that a given event occurs during a monitoring period T (*e.g.*, number of times the destination port 443 is accessed). Thus, for a monitoring period of duration T, we will obtain a set of observations of features $O = o_1, o_2, \ldots o_T$, where $o_t \in [1, T] = \{faac_1^t, facc_2^t, \ldots, facc_N^t\}$ and $faac_i^t$ is the numeric value of the feature i at instant t. It is important to say that the FAAC approach implies that a large number of features might be defined for an observation (*e.g.*, to monitor the access to every TCP and UDP ports a total of 65535*2 features would have to be defined). To deal with such a big dimensionality, MSNM uses PCA (*Principal Component Analysis*) techniques.

The sequence of time observations O will serve as the input to the estimation module.

- *Estimation module*

Based on the parameterization previously discussed, MSNM defines the following general process regarding behaviour modeling and estimation:

1. *Modeling phase.* Based on a set of observations O^τ specifically collected from calibration data, a PCA model is first trained. For this purpose, every observation is decomposed using PCA into a *model part* and a *residual part* as follows:

$$\mathbf{x}_n = \sum_{a=1}^{A} \mathbf{t}_{a_n}.\mathbf{p}_a + \mathbf{e}_n \tag{1}$$

where, in the model part, p_a is the loading column vector corresponding to the a-th principal component (PC) and t_{an} corresponds to the so called *score* of the n-th observation in that PC; e_n is the column vector that represents the residual part of the observation; and A is the total number of PCs in the model.

The obtained model is the matrix P, which is built with the different p_a as column vectors and that contains the A principal component vectors.

2. *Monitoring/Detection phase.* From such a base model, P, subsequent observations o_t from every subject are decomposed using Eq. (1). Then, the security level for every subject is estimated by calculating two statistics: the *D-statistic*, which is computed from the scores, and the *Q-statistic*, which compresses the residuals:

$$D_n = \sum_{a=1}^{A} \left(\frac{t_{an} - \mu_{t_a}}{\sigma_{t_a}} \right)^2 \tag{2}$$

$$Q_n = \mathbf{e'}_n . \mathbf{e}_n \tag{3}$$

where μ_{t_a} and σ_{t_a} stand for the mean and the standard deviation of the scores of the a-th PC in the calibration data. Usually, x_n are centered so the value of μ_{t_a} is 0.

3. *Diagnosis.* In addition to the previous main phases, upper control limits (UCLs) can be derived for *D-st* and *Q-st*. If outliers are found from the UCLs, that is, if some of the statistics for the devices exceed the limits, they are diagnosed using a MSNM diagnosis tools, like oMEDA [42]. If the diagnosis reveals a security problem, the anomalous samples are discarded and steps 1-3 are repeated.

Based on the above, each input observation o_t will be assigned with the pair <*D-st,Q-st*>^{t} (with no private or sensitive information) to summarize the subject security profile corresponding to the instant t.

3.3.2. Access control decision

For every summarized security profile <*D-st,Q-st*>^{t}, an access decision is taken by the dPDP module. For that, both *D-st* and *Q-st* are compared to the estimated UCLs. If any of these limits is surpassed, the occurrence of an anomaly is identified, and the access might be properly modified according to the established access policies.

As previously indicated, in addition to the ability of determining the occurrence of deviations from the expected security profile, MSNM includes a set of diagnosis tools (*e.g.*, oMEDA) which help elucidate the specific features of the subject (regarding configuration or communications) that are really anomalous when evaluated against the model P [37]. This is a major advantage of MSNM over other black-box machine learning procedures, and it allows the subject to understand and solve the potential threats that caused an anomaly from the service provider's point of view, so as to obtain better access privileges.

4. Experimental Results

After describing the functionallity and the specific implementation of SADAC for WiFi environments, this section presents a proof-of-concept aimed to demonstrate the usefulness and validity of the proposal in a real corporate network. Although the functionality of SADAC, as already described through the paper, has been completely implemented by authors, two main restrictions are considered here:

- Only two access policies exist: accept or deny, so that no partial access to infrastructure is considered for devices/users.
- The experimentation below is performed in an offline basis instead in an online/realtime mode.

Previous aspects, however, do not constitute a critical operational restriction from the functional point of view and the capabilities of the access control approach introduced in this work.

4.1. Experimental Data

The specific experimentation scenario deployed involves the monitoring and study of a total of 83 final mobile devices in a university campus that have been monitored during 205 days to collect a number of features regarding their associated configuration and communication profiles, as described in Section 3.1. Table 1 summarizes the set of 46 attributes gathered.

It is also worth to mention that the mobile devices involved in the experimentation belong to volunteer users (students and teachers) who have signed an agreement to allow the monitoring process to obtain the mentioned individual data for scientific purposes. The global data are publicly available at `https://github.com/nesg-ugr/mdsm-dataset`.

Group	**Subgroup**	**Attribute**	**Description**
Configuration (21)	Application (5)	appIs macId name / version permissions	Application identifer MAC address Android package name / version Permission list of the package
	Status (3)	ramUsage batteryLevel cpuUsage	Usage of RAM Level of battery % CPU in use
	Security (4)	unkownSources developerOptions secure rooted	Software from unkown sources? Developer option is active? Device with locking mechanisms? Device rooted?
	Device (9)	mac device (4) sdk cpuCores ramTotal batteryTotal	MAC address of the device Device description (model, etc.) SDK used # of cores Total of RAM installed Total of battery capacity
Communications (25)	Traffic (14)	macID packageName time duration protocol saddr / daddr sport / dport sentBytes receiveBystes sentPackets receivePackets tcpFlags	MAC address identification Package responsible for the flow Time of communication Flow duration Communication protocol Source / Destination address Source / Destination port # of bytes sent # of bytes received # of packets sent # of packets recieved Flow state: new, active, closed, ...
	Bluetooth (2)	macID bluetoothDevice	MAC address id Description of the bluetooth device
	Wifi (3)	macID SSID security	MAC address identification SSID of the WiFI's connected WiFi security mechanism
	Connectivity (6)	macId data wifi bluetooth gps airplane	MAC address Data network active? WiFi active? Bluetooth active? GPS active? Airplane mode active?

Table 1: Attributes collected by AMon.

4.2. Results

Based on the previous scenario, we analyze the performance of SADAC exclusively from the perspective of the security provided. That is, the operational performance regarding computational cost, resources usage, agility in communications between entities and modules, etc. are ignored for the moment.

To carry out the intended analysis on the security related performance, two successive stages are done. In a first stage, the overall behavioural model P is estimated from an attribute training dataset. In the second stage, the features of every device are analyzed to obtain the associated security profile $<D\text{-}st, Q\text{-}st>^t$ and determining potential deviations in its behaviour with respect to P. If so, access restrictions are applied and the affected devices/users are properly notified.

4.2.1. Stage I: Training

For training purposes, we consider the attribute related data corresponding to the first 139 days monitored by our SADAC related app, which correspond to a total of 67 trustable devices.

Making use of the MSNM methodology, six anomalous mobile devices are detected: those labeled as *D428*, *D1*, *D25*, *D860*, *D394*, and *D190*[1]. Figure 4 shows that the values of *D-st* and *Q-st* statistics for such devices exceed the corresponding UCLs.

After diagnosing the anomalies with oMEDA, we observe deviations in the permissions list per device. A more detailed diagnosis of the most anomalous behaviour, that for *D428*, reveals significant deviations in 36 of the total of 158 app permissions monitored in comparison with normal devices. Figure 5 shows the number of apps using each of the four most used permissions: REQUEST_INSTALL_PACKAGE, SET_PREFERED_APPLICATION, BIND_QUICK_SETTING, and SET_TIME_ZONE. In particular, we can see a huge difference in the usage of REQUEST_INSTALL_PACKAGE in *D428* with respect to the rest of devices. This permission is considered dangerous because it is related with malware spreading [43].

From the above, we consider the six anomalous devices as outliers and remove them from the training dataset. With that in mind, the final model

[1] It is important to mention that the device identifier values are intended to anonymize the devices and, thus, independent of the number of them.

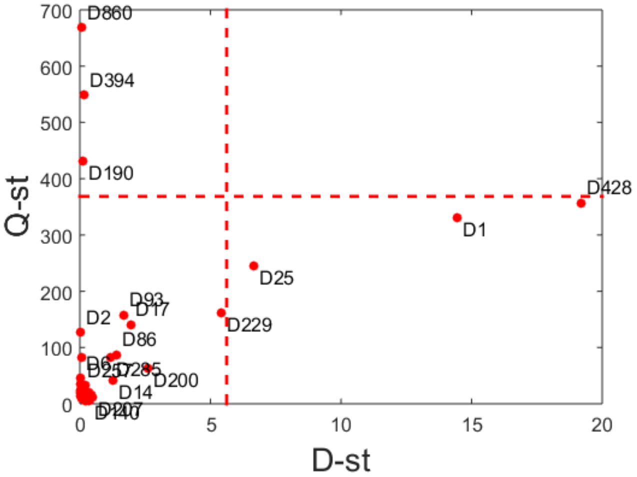


Figure 4: *D-st* and *Q-st* values for devices used in training, where dashed lines represent the threshold values.

P associated to the expected behaviour of our overall mobile environment is estimated to be subsequently used by the dPDP module.

4.2.2. Stage II: Access control decision

Once the 'normality' model P is obtained, devices either requiring initial access to the infrastructure or already attached to it are evaluated against P over time by the dPDP module. For that, the gathered activity of all the 83 available mobile devices during the whole sampling period is analysed with experimentation purposes.

Figure 6 shows D and Q statistics for some devices, where device *D116* exhibits a notoriously anomalous behaviour (very far from the control limits) regarding configuration profiles. From the diagnosis module, we conclude that the cause of such an anomaly is the permission WRITE_SYNC_SETTINGS (Figure 7). The threat associated to this permission is due to the possibility of data synchronization with external devices and entities.

Regarding communication profiles, Figure 8 shows the values of *D-st* and *Q-st* for daily traffic samples, where days without traffic are removed from the picture. The existence of days with anomalous behaviour are clearly observed. After diagnosing the anomaly with oMEDA, we conclude that one of them is motivated by very low traffic, which is not usually a real problem. However, another anomaly is due to *BitTorrent* traffic generated by the

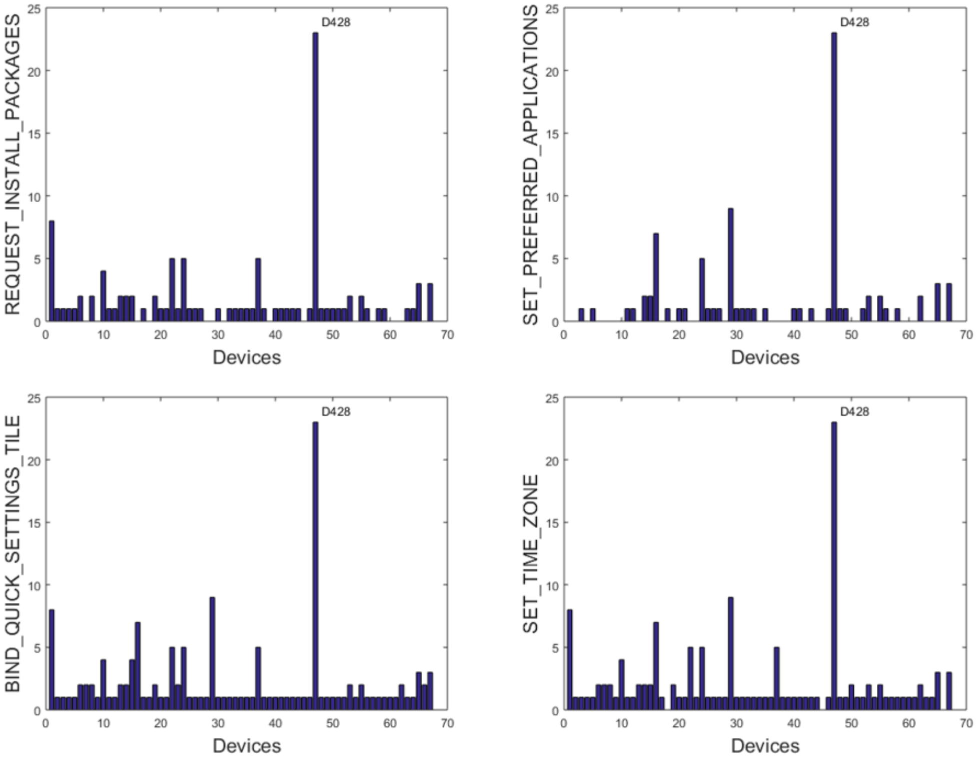


Figure 5: Number of apps using four of the most significant permissions of the anomalous device *D428* in training stage.

device *D473*, which can constitute a security risk depending on the security policy of the access provider. Likewise, the anomaly appeared around 130-th day, is due to traffic from devices *D260* and *D1*, which involve an abnormal amount of *NetBios* related traffic.

From the above, the causes detected as anomalies could be used again to restrict the access to the devices to our infrastructure at a given instant. However, the final users could be additionally notified to solve the problem and, if so, to try to access the infrastructure again.

As an example about the access control procedure itself performed at the AP/dPEP entity from the anomaly detection carried out by the dPDP, Figure 9 shows an example of a fragment of the AP logfile during experimentation. The first red box represents the communication between a device and

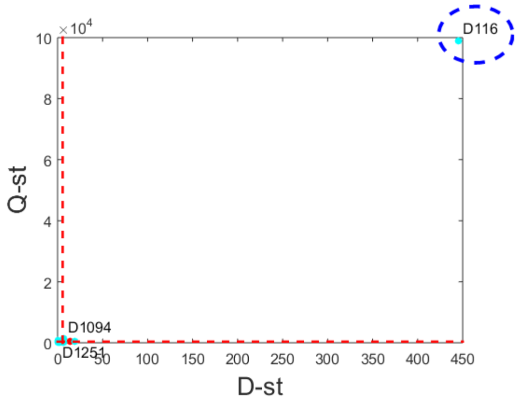


Figure 6: *D-st* and *Q-st* values for devices in Stage I.

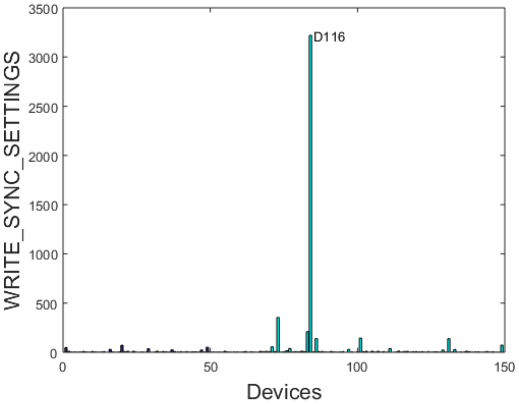


Figure 7: Number of apps using the permission causing the anomaly on device *D116*.

the AP to obtain the security parameters after the initial challenge being successfully solved. In this case, the device is considered 'secure' and the connection is granted. In the second red box, the log shows a case in which the AP receives the order to reject the device because it does not present and adequate security level. In this particular case, because it is concluded to be *rooted* and, thus, it is not a trusted device and is marked as 'insecure'. Although this aspect is not detailed here in-depth, in our specific implementation the rooted device is included into a blacklist in order to mark it as a critical device.

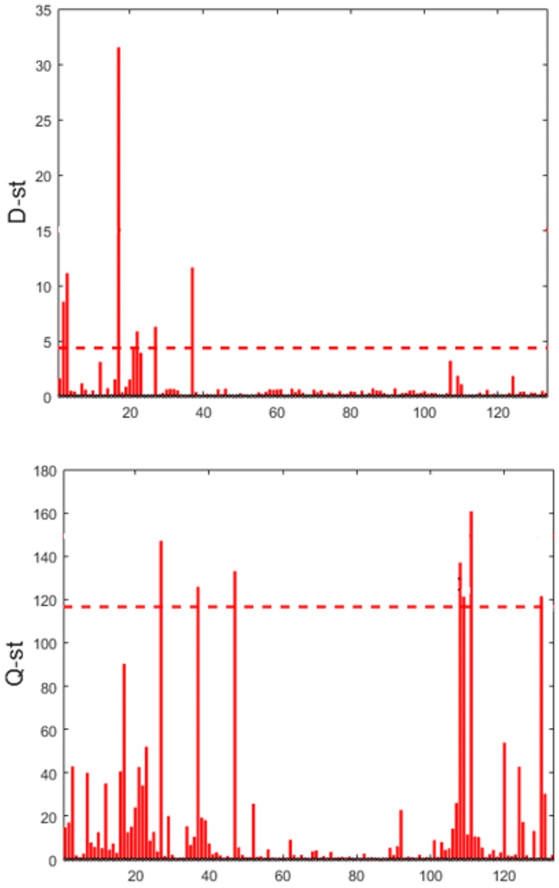
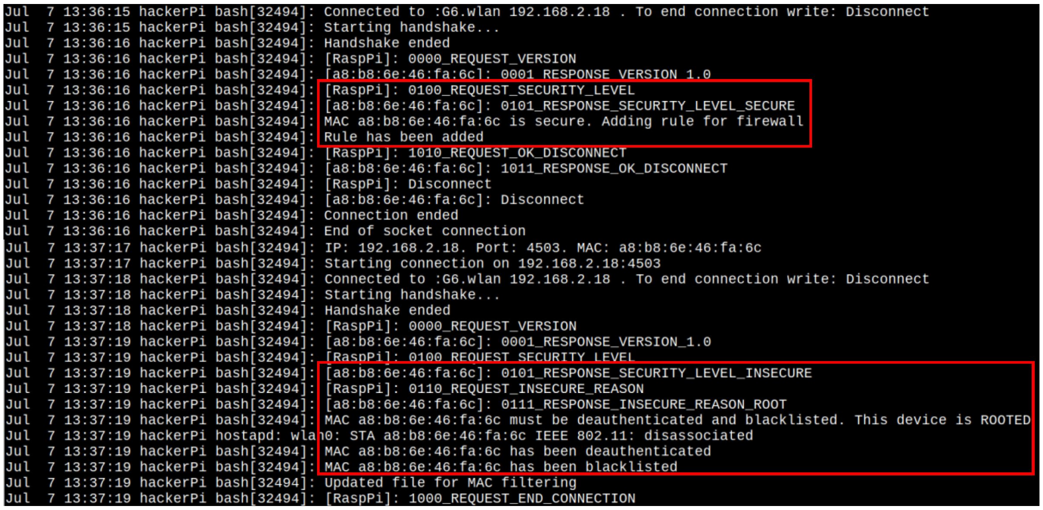


Figure 8: *Q-st* and *D-st* values for the detected anomalies in network traffic in Stage II.



```
Jul  7 13:36:15 hackerPi bash[32494]: Connected to :G6.wlan 192.168.2.18 . To end connection write: Disconnect
Jul  7 13:36:15 hackerPi bash[32494]: Starting handshake...
Jul  7 13:36:16 hackerPi bash[32494]: Handshake ended
Jul  7 13:36:16 hackerPi bash[32494]: [RaspPi]: 0000_REQUEST_VERSION
Jul  7 13:36:16 hackerPi bash[32494]: [a8:b8:6e:46:fa:6c]: 0001_RESPONSE_VERSION_1.0
Jul  7 13:36:16 hackerPi bash[32494]: [RaspPi]: 0100_REQUEST_SECURITY_LEVEL
Jul  7 13:36:16 hackerPi bash[32494]: [a8:b8:6e:46:fa:6c]: 0101_RESPONSE_SECURITY_LEVEL_SECURE
Jul  7 13:36:16 hackerPi bash[32494]: MAC a8:b8:6e:46:fa:6c is secure. Adding rule for firewall
Jul  7 13:36:16 hackerPi bash[32494]: Rule has been added
Jul  7 13:36:16 hackerPi bash[32494]: [RaspPi]: 1010_REQUEST_OK_DISCONNECT
Jul  7 13:36:16 hackerPi bash[32494]: [a8:b8:6e:46:fa:6c]: 1011_RESPONSE_OK_DISCONNECT
Jul  7 13:36:16 hackerPi bash[32494]: [RaspPi]: Disconnect
Jul  7 13:36:16 hackerPi bash[32494]: [a8:b8:6e:46:fa:6c]: Disconnect
Jul  7 13:36:16 hackerPi bash[32494]: Connection ended
Jul  7 13:36:16 hackerPi bash[32494]: End of socket connection
Jul  7 13:37:17 hackerPi bash[32494]: IP: 192.168.2.18. Port: 4503. MAC: a8:b8:6e:46:fa:6c
Jul  7 13:37:17 hackerPi bash[32494]: Starting connection on 192.168.2.18:4503
Jul  7 13:37:18 hackerPi bash[32494]: Connected to :G6.wlan 192.168.2.18 . To end connection write: Disconnect
Jul  7 13:37:18 hackerPi bash[32494]: Starting handshake...
Jul  7 13:37:18 hackerPi bash[32494]: Handshake ended
Jul  7 13:37:18 hackerPi bash[32494]: [RaspPi]: 0000_REQUEST_VERSION
Jul  7 13:37:19 hackerPi bash[32494]: [a8:b8:6e:46:fa:6c]: 0001_RESPONSE_VERSION_1.0
Jul  7 13:37:19 hackerPi bash[32494]: [RaspPi]: 0100_REQUEST_SECURITY_LEVEL
Jul  7 13:37:19 hackerPi bash[32494]: [a8:b8:6e:46:fa:6c]: 0101_RESPONSE_SECURITY_LEVEL_INSECURE
Jul  7 13:37:19 hackerPi bash[32494]: [RaspPi]: 0110_REQUEST_INSECURE_REASON
Jul  7 13:37:19 hackerPi bash[32494]: [a8:b8:6e:46:fa:6c]: 0111_RESPONSE_INSECURE_REASON_ROOT
Jul  7 13:37:19 hackerPi bash[32494]: MAC a8:b8:6e:46:fa:6c must be deauthenticated and blacklisted. This device is ROOTED
Jul  7 13:37:19 hackerPi hostapd: wlan0: STA a8:b8:6e:46:fa:6c IEEE 802.11: disassociated
Jul  7 13:37:19 hackerPi bash[32494]: MAC a8:b8:6e:46:fa:6c has been deauthenticated
Jul  7 13:37:19 hackerPi bash[32494]: MAC a8:b8:6e:46:fa:6c has been blacklisted
Jul  7 13:37:19 hackerPi bash[32494]: Updated file for MAC filtering
Jul  7 13:37:19 hackerPi bash[32494]: [RaspPi]: 1000_REQUEST_END_CONNECTION
```

Figure 9: Example of a fragment of the AP logfile.

5. Further Discussion

Once the feasibility of SADAC as a valid security-based access control scheme has been shown, it is important to remark some practical key issues about it. Deploying SADAC on real communication networks, either corporate or ISPs, has to deal with some principal aspects: legality, scalability and reliability, among others.

Regarding legality, we must remark again the necessity of preserving subject privacy. The relevance of this topic is dealt with in works like [44], where techniques like blockchain are proposed to prevent data from manipulation or unauthorized access. In SADAC, beyond the provision of confidentiality, authentication and integrity through the use of TLS/SSL communications, privacy can be also achieved as follows:

- SADAC can be easily re-designed so that the attributes used to estimate the normality model associated to a subject remain local, which guarantees privacy for the associated information. This is relevant in particular for the diagnosing process, which only requires the model P for computation over a given observation.

- Regarding the monitoring process, only the UCLs and the summarized security profiles in terms of the pairs $<D\text{-}st, Q\text{-}st>^t$ are strictly needed.

From (a) above, the subjects are able to compute the $<$*D-st,Q-st*$>^t$ statistics and share them with the network (AP/dPEP).

- Since neither D nor Q contain any sensitive information about the primary monitored features or attributes, the network (AP/dPEP and dPDP) will not be aware of private information about subjects.

Beyond privacy compliance, because of the 'intrusive' nature of any analysis and the strict access control scheme proposed here, a previous contract between the subject and the network provider could be required. In this case, the use of SADAC has to be part of the corresponding SLA between the parties. Such a contract must include the installation and operation of the SADAC related app on the user devices.

With respect to scalability, we must consider that monitoring and analyzing individual communications from the network side could imply high computational cost and high wideband consumption for information sharing. Regarding computational cost, the use of the app on the subject's device to estimate $<$*D-st,Q-st*$>^t$ as described above will reduce the computational cost at the network side. Although this can also be viewed as an inconvenient from the subject side, its acceptance and collaboration can be achieved through a double benefit: *(a)* the global reduction of security risks for the community; and *(b)* some potential advantages to the subject from the network in terms of high performance in access, or significant restrictions in other case.

Regarding wideband consumption, it is clear that SADAC operation will imply low network load, as only the parameters *D-st* and *Q-st* need to be transmitted to inform about the security profile for a given subject.

In addition to the above, some kind of distributed solution should be considered to deploy SADAC on complex environments. For example, by establishing a hierarchical structure from the very entrance points in the network (*e.g.*, APs for WiFi or terminal routers in wired environments). As shown in [45], MSNM can also be successfully used in this direction to scale the problem of behaviour estimation of users and systems.

Finally, reliability is a key concern too. Provided the delicate nature of granting users with access to systems and resources, the potential cancellation of this right must be supported by irrefutable evidences about the harmful behaviour of the user affected. From this perspective, some cautions might be adopted by network administrators to implement SADAC:

- Signature- or ACL-based estimation approaches can be used as a complement to conclude real malicious behaviours (*e.g.*, the installation of some well-known unreliable OS version or application), as anomaly like estimators are suitable to provide unacceptable false positives [46, 47].

- Once inadequate security profiles are detected, the corresponding subject can be properly advised and kindly invited to solve the associated problems before adopting further restrictions on the access. As explained, the diagnosis capabilities of MSNM are relevant for this purpose.

- Infrastructures with different confidence and service levels can be used by providers to attend the different types of trustworthy subjects.

6. Conclusion and Future Work

ICT security is a principal concern nowadays. This situation will become even more severe with the next adoption of technologies like IoT and BYOD. Provided the increasing relevance and impact of security threats, we introduce here SADAC as a novel attribute based access control implementation where the security profile of the subjects in a communication environment (devices/users) is dynamically evaluated in order to grant, limit or deny access to services and resources of the network over time.

The system has been proved here with real data from real users and devices on WiFi corporate environments. The results obtained are promising and show the viability of the proposal to control access in dynamic environments.

SADAC is security-based, dynamic adaptive and diagnosis capable. Moreover, it can be complemented with the use of other attributes (either security related or not) and conditions to take more complex and ambitious access control decisions. Moreover, it would be easily extended to strengthen user privacy by moving the estimation and diagnosis modules to the final devices.

Despite the sensitive nature of the problem itself and the possible solutions to be adopted, we firmly think that the security profile of users and devices should be necessarily considered as a confidence measure to provide access to ICT environments.

Acknowledgement

This work is partly supported by the Spanish Ministry of Economy and Competitiveness and ERDF (European Regional Development Fund) funds through projects TIN2017-83494-R and PID2020-114495RB-I00.

References

[1] Cisco: ”2021 Cybersecurity Threat Trends: Phishing, Crypto Top the List”. Security report, 2021. Available at `https://umbrella.cisco.com/info/2021-cyber-security-threat-trends-phishing-crypto-top-the-list`.

[2] TrendMicro: ”Toward a New Momentum. Trend Micro Security Predictions for 2022”. Security report, 2022. Available at `https://documents.trendmicro.com/assets/rpt/rpt-toward-a-new-momentum-trend-micro-security-predictions-for-2022.pdf`.

[3] J. Peng, M. Guo, J. Quan: “Software Vulnerability and Application Security Risk”. Information Resources Management Journal (IRMJ), vol. 32, n. 1, pp. 1-10, 2019 (DOI: 10.4018/IRMJ.2019010103).

[4] J.J. Jeong, J. Mihelcic, G.C. Oliver, C. Rudolph: “Towards an Improved Understanding of Human Factors in Cybersecurity”. 5th IEEE International Conference on Collaboration and Internet Computing (CIC), pp. 338-345, 2019 (DOI: 10.1109/CIC48465.2019.00047).

[5] E. Kadena, M. Gupi: “Human Factors in Cibersecurity: Risks and Impacts”. Security Science Journal, vol. 2, n. 2, pp. 51-64, 2021 (DOI: 10.37458/ssj.2.2.3).

[6] Z. Wang, H. Zhu, L. Sun: ”Social Engineering in Cybersecurity: Effect Mechanisms, Human Vulnerabilities and Attack Methods”. IEEE Access, pp. 11895-11910, 2021 (DOI: 10.1109/ACCESS.2021.3051633).

[7] U.M. Neves, F.L. Mello: ”BYOD with Security”. Journal of Information Security and Cryptography (Enigma), vol. 5, n. 1, pp. 40-47, 2019 (DOI: 10.17648/jisc.v5i1.70).

[8] M. Ratchford, O. El-Gayar, C. Noteboom, Y. Wang: ”BYOD Security Issues: A Systematic Literature Review”. Information Security Journal: A Global Perspective, pp. 1-21, 2021 (DOI: 10.1080/19393555.2021.1923873).

[9] K.M. Sadique, R. Rahmani, P. Johannesson: ”Towards Security on Internet of Things: Applications and Challenges in Technology”. Procedia Computer Science, vol. 141, pp. 199-206, 2018 (DOI: 10.1016/j.procs.2018.10.168).

[10] N. Mishra, S. Pandya: ”Internet of Things Applications, Security Challenges, Attacks, Intrusion Detection, and Future Visions: A Systematic Review”. IEEE Access, vol. 9, pp. 59353-59377, 2021 (DOI: 10.1109/ACCESS.2021.3073408).

[11] M. Benantar: “Access Control Systems: Security, Identity Management and Trust Models”. Springer, 2006 (DOI: 10.1007/0-387-27716-1).

[12] R.S. Sandhu, P. Samarati: “Access Control: Principle and Practice”. IEEE Communications Magazine, vol. 32, n. 9, pp. 40-48, 1994 (DOI: 10.1109/35.312842).

[13] V.C. Hu, D.F. Ferraiolo, R. Chandramouli, D.R. Kuhn: “Attribute-based Access Control”. Artech-House, 2018.

[14] V.C. Hu, D. Ferraiolo, R. Kuhn, A. Schnitzer, K. Sandlin, R. Miller, K. Scarfone: “Guide to Attribute Based Access Control (ABAC) Definition and Considerations”. NIST Special Publication, 800-162, 2019 (DOI: 10.6028/NIST.SP.800-162).

[15] Y. Hao, C. Huang, J. Ni, H. Rong, M. Xian, X. Shen: “Fine-Grained Data Access Control with Attribute-hiding Policy for Cloud-based IoT”. Computer Networks, vol. 153, pp. 1-10, 2019 (DOI: 10.1016/j.comnet.2019.02.008).

[16] S. Ravidas, A. Lekidis, F. Paci, N. Zannone: ”Access Control in Internet-of-Things: A Survey”. Journal of Network and Computer Applications, vol. 144, n. 15, pp. 79-101, 2019 (DOI:10.1016/j.jnca.2019.06.017).

[17] A.S.M. Kayesa, W. Rahayua, P. Wattersa, M. Alazabb, T. Dillona, E. Chang: ”Achieving Security Scalability and Flexibility using Fog-based

Context-Aware Access Control". Future Generation Computer Systems, vol. 107, pp. 307-323, 2020 (DOI: 10.1016/j.future.2020.02.001).

[18] M.U. Aftab, A. Oluwasanmi, A. Alharbi, O. Sohaib, X. Nie, Z. Qin, S.T. Ngo: "Secure and Dynamic Access Control for the Internet of Things (IoT) Based Traffic System". PeerJ Computer Science, pp. 1-26, 2021 (DOI: 10.7717/peerj-cs.471).

[19] S. Bhatt, T. K. Pham, M. Gupta, J. Benson, J. Park, R. Sandhu, "Attribute-Based Access Control for AWS Internet of Things and Secure Industries of the Future". IEEE Access, vol. 9, pp. 107200-107223, 2021 (DOI: 10.1109/ACCESS.2021.3101218).

[20] Y. Zhang, M. Yutaka, M. Sasabe, S. Kasahara: "Attribute-Based Access Control for Smart Cities: A Smart-Contract-Driven Framework". IEEE Internet of Things Journal, vol. 8, n. 8, pp. 6372-6384, 2021 (DOI: 10.1109/JIOT.2020.3033434).

[21] Q. Lyu, Y. Qi, Z. Zhang, H. Liu, Q. Wang, N. Zheng: "SBAC: A Secure Blockchain-based Access Control Framework for Information-centric Networking". Journal of Network and Computer Applications, vol. 149, pp. 1-17, 2020 (DOI: doi.org/10.1016/j.jnca.2019.102444).

[22] F. Ghaffari, E. Bertin, N. Crespi, S. Behrad, J. Hatin, "A Novel Access Control Method Via Smart Contracts for Internet-Based Service Provisioning". IEEE Access, vol. 9, pp. 81253-81273, 2021 (DOI: 10.1109/ACCESS.2021.3085831).

[23] Y. Liu, M. Qiu, J. Liu, M. Liu: "Blockchain-Based Access Control Approaches". 8th IEEE International Conference on Cyber Security and Cloud Computing (CSCloud)/7th IEEE International Conference on Edge Computing and Scalable Cloud (EdgeCom), pp. 1-6, 2021 (DOI: I 10.1109/CSCloud-EdgeCom52276.2021.0003).

[24] S. Rose, O. Borchert, S. Mitchell, S. Connelly: "Zero Trust Architecture". NIST Special Publication, 800-207, 2020 (DOI: 10.6028/NIST.SP.800-207).

[25] J. Garvis, J.W. Chapman: "Zero Trust Security, An Enterprise Guide". Apress, Berkeley, CA, 2021 (DOI:10.1007/978-1-4842-6702-8).

[26] M. Shore, S. Zeadally, A. Keshariya: "Zero Trust: The What, How, Why, and When". Computer, vol. 54, no. 11, pp. 26-35, 2021, (DOI: 10.1109/MC.2021.3090018).

[27] E. Bertino, K. Brancik: "Services for Zero Trust Architectures - A Research Roadmap". IEEE International Conference on Web Services (ICWS), pp. 14-20, 2021 (DOI: 10.1109/ICWS53863.2021.00016).

[28] R. Vanickis, P. Jacob, S. Dehghanzadeh, B. Lee: "Access Control Policy Enforcement for Zero-Trust-Networking". 29th Irish Signals and Systems Conference (ISSC), pp. 1-6, 2018 (DOI: 10.1109/ISSC.2018.8585365).

[29] T. Dimitrakos, T. Dilshener, A. Kravtsov, A. LaMarra, F. Martinelli, A. Rizos, A. Rosetti, A. Saracino: "Trust Aware Continuous Authorization for Zero Trust in Consumer Internet of Things". 19th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), pp. 1801-1812, 2020 (DOI: 10.1109/TrustCom50675.2020.00247).

[30] Q. Yao, Q. Wang, X. Zhang, J. Fei: "Dynamic Access Control and Authorization System based on Zero-trust Architecture". International Conference on Control, Robotics and Intelligent System (CCRIS), ACM, pp. 123–127, 2020. (DOI: 10.1145/3437802.3437824).

[31] Y.G. Wu, W.H. Yan, J.Z. Wang: "Real Identity Based Access Control Technology Under Zero Trust Architecture". International Conference on Wireless Communications and Smart Grid (ICWCSG), pp. 18-22, 2021 (DOI: 10.1109/ICWCSG53609.2021.00011).

[32] S. Mandal, D. Ali Khan, S. Jain: "Cloud-Based Zero Trust Access Control Policy: An Approach to Support Work-From-Home Driven by COVID-19 Pandemic". Next Generation Computing, vol. 39, pp. 559-662, 2021 (DOI: 10.1007/s00354-021-00130-6).

[33] M. Carvalho: "SECaaS-Security as a service". Information Systems. Security Association (ISSA) Journal, pp. 20-24, 2011.

[34] D. Sharma, C.A. Dhote, M. Potey: "Security-as-a-Service from Clouds: A Comprehensive Analysis". International Journal of Computer Applications, vol. 67, no. 3, pp. 1-4, 2011.

[35] A. Peterson: “Top 5 Security-as-a-Service Providers”. Available at https://technologyadvice.com/blog/information-technology/security-as-a-service-saas-software-providers/, 2021.

[36] MarketsandMarkets: “Security as a Service Market”. Available at https://www.marketsandmarkets.com/Market-Reports/security-as-a-service-market-132531603.html. Accessed in February 2022.

[37] J. Camacho, A. Pérez-Villegas, P. García-Teodoro, G. Maciá-Fernández: ”PCA-based Multivariate Statistical Network Monitoring for Anomaly Detection”. Computers & Security, vol. 59, pp. 118-137, 2016 (DOI: 10.1016/j.cose.2016.02.008).

[38] J. A. Gómez-Hernández, P. García-Teodoro, J. A. Holgado-Terriza, G. Maciá-Fernández, J. Camacho-Páez, M. Robles-Carrillo: ”AMon: A Monitoring Multidimensional Feature Application to Secure Android Environments”. IEEE Security and Privacy Workshop (SPW), pp. 31-36, 2021 (DOI: 10.1109/SPW53761.2021.00013).

[39] Android: “API Reference”. Available at https://developer.android.com/reference.

[40] M. Bokhorst: ”NetGuard: A Simple Way to Block Access to the Internet per Application”. Available at https://github.com/M66B/NetGuard/.

[41] J. Malinen: ”Developers’ Documentation for WPA_Supplicant and Hostapd”. Available at http://w1.fi/wpa_supplicant/devel/.

[42] M. Fuentes-García, G. Maciá-Fernández, J. Camacho: ”Evaluation of diagnosis methods in PCA-based Multivariate Statistical Process Control”. Chemometrics and Intelligent Laboratory Systems, vol. 172, pp. 194-210, 2018 (DOI: 10.1016/j.chemolab.2017.12.008).

[43] Eybisi: “Mobile Malware Analysis: Tricks Used in Anubis”. Available at https://eybisi.run/Mobile-Malware-Analysis-Tricks-used-in-Anubis/.

[44] Y. Xu, Q. Zeng, G. Wang, C. Zhang, J. Ren, Y. Zhang: ”An Efficient Privacy-Enhanced Attribute-based Access Control Mechanism”. Concurrency and Computation. Practice and Experience, vol. 32, n. 5, pp. 1-10, 2019 (DOI: 10.1002/cpe.5556).

[45] G. Maciá-Fernández, J. Camacho, P. García-Teodoro, R.A. Rodríguez-Gómez: ”Hierarchical PCA-based Multivariate Statistical Network Monitoring for Anomaly Detection”. 8th IEEE International Workshop on Information Forensics and Security (WIFS), pp. 1-6, 2016 (DOI: 10.1109/WIFS.2016.7823895).

[46] Q.S. Qassim, A.M. Zin, M.J.A. Aziz: “Anomaly-based Network IDS False Alarm Filter Using Cluster-based Alarm Classification Approach”. International Journal of Security and Networks, vol. 12, n. 1, pp. 13-26, 2016 (DOI: 10.1504/IJSN.2017.081056).

[47] D.S. Vijayakuma, S. Ganapathy: “Machine Learning Approach to Combat False Alarms in Wireless Intrusion Detection System”. Computer and Information Science, vol. 11, n. 3, pp. 67-81, 2018 (DOI: 10.5539/cis.v11n3p67).