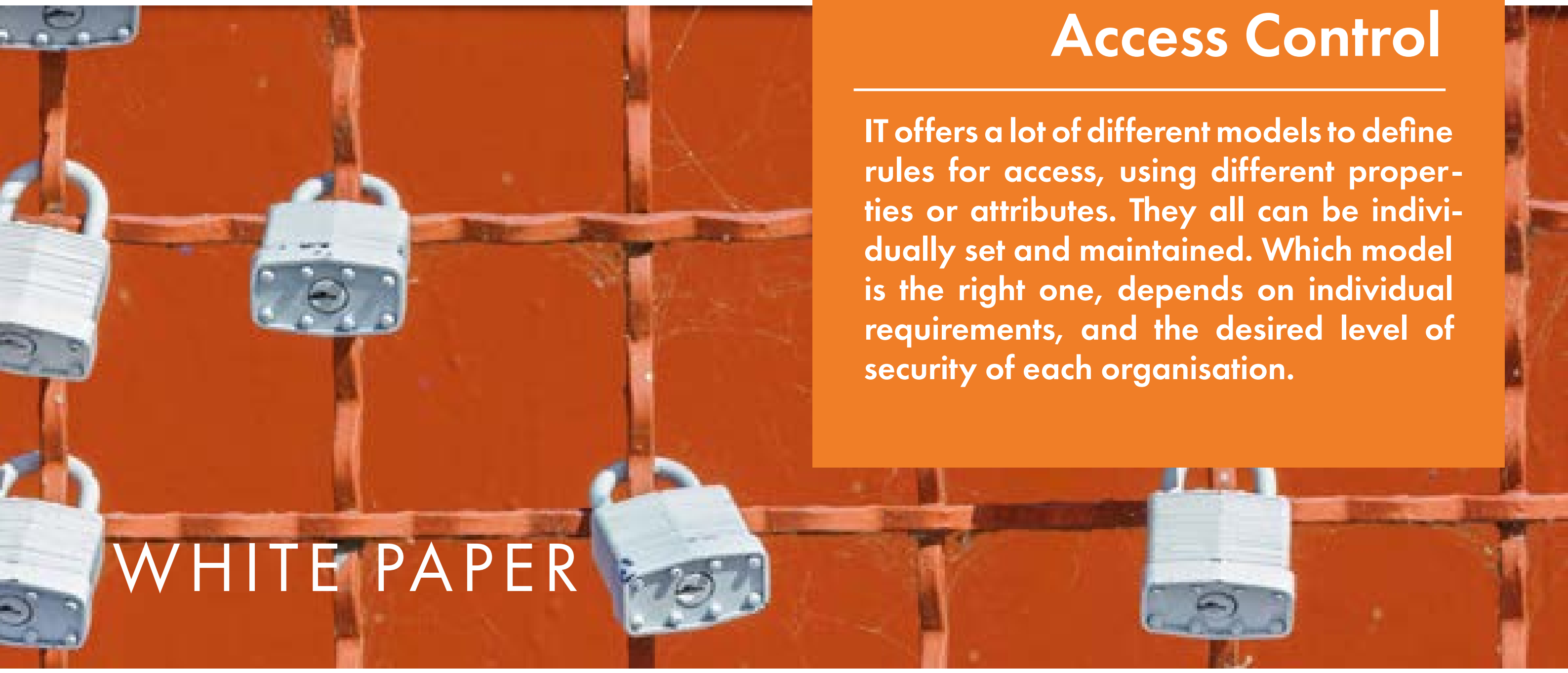
Access Control

IT offers a lot of different models to define rules for access, using different properties or attributes. They all can be individually set and maintained. Which model is the right one, depends on individual requirements, and the desired level of security of each organisation.



WHITE PAPER

With access control it is possible to limit access to certain resources based on predefined properties. In our daily lives, access control to buildings and events often works quite similarly. In the case of an event for example, the crucial property would be the possession of the appropriate ticket, or nowadays the proof of vaccination against Corona.

MANDATORY ACCESS CONTROL (MAC)

In access control according to MAC users are assigned labels which determine their respective level of clearance. Users' permissions are compiled individually, and maintained centrally; permissions cannot be changed by the users themselves. Overall, MAC is rather rigid and not easily scalable, which usually makes it a poor solution for large organisations requiring a lot of flexibility.

Nonetheless MAC is the most secure access control model and works really well for small organisations or singular organisational units, with high security requirements.

DISCRETIONARY ACCESS CONTROL (DAC)

DAC focuses on the resources to be protected, or resource groups, when assigning permissions. The resources are allocated in access control lists (ACL). These lists determine which users may access which resource. They also determine what kind of permissions users may receive, i.e. read only or write permission, which allows them to alter the resource in question. All lists can be maintained and adjusted centrally or by the respective resource owner.
DAC can be combined with MAC by adding the lists and the respective labels instead of individual profiles.

In DAC it is even possible to grant users temporary permissions using specified resource ALCs, which is handy when a user only needs access for a certain project. Unfortunately, this is also one of the biggest disadvantages of this access control model. As it is possible to be a member of multiple ACLs, e.g. the standard and the project related ACL, can result in different permission structures overwriting or contradicting each other. As a consequence users could end up being granted permission when they are not actually eligible, or lose permissions they actually are entitled to. Usually ACLs are quite complex, which also makes it difficult to maintain an overview in order to prevent such conflicts. Especially organisations with clearly separated units can benefit from this model in order to be able to quickly onboard employees in the individual units.

ROLE-BASED ACCESS CONTROL (RBAC)

In the RBAC model permissions are assigned based on roles, while the principle of the so-called least privilege is also applied. This means that all users originally only receive the minimum of permissions until they are assigned more roles within the organisation, which in turn will also give them more permissions. This of course also means that users may have multiple roles with different permissions. The role structure may be customised for the individual organisation, and can be maintained either centrally or de-centrally depending on the size of the organisation. Hierarchical structures allow for different types of administrators, i.e. ones who are only responsible for a certain part of the system, and superadministrators on the highest level who are responsible for the entire system.

If a person leaves the organisation, their roles associated with their identity can easily be withdraw, which automatically revokes all permissions, too. The same principle also works for a temporary leave. Once the person returns, the identity can easily be attributed the same roles as before the leave of absence. Thus the RBAC model is highly flexible and fairly easy to maintain. Hence it is one of the most commonly used access control strategies. However, introducing RBAC can require more time and effort. All role and permission concepts must be well thought out to use the model as effectively as possible.

ATTRIBUTE BASED ACCESS CONTROL (ABAC)

Similar to RBAC, yet more precise in configuration possibilities, there is also ABAC. Here the access permissions are not defined via roles but via individual attributes of the users, such attributes can be the security clearance, their place of work, their association with a certain organisational unit, or even roles as attributes. In comparison to RBAC this makes ABAC even more flexible.

In federated environments ABAC enables external users to gain access without having to share all information. In order to use resources within the federation, it is only necessary to share the respectively relevant attribute. This way most sensitive information (i.e. personal information) do not need to be shared. Due to the high levels of flexibility and security, as well as due to precise configuration possibilities, ABAC is well suited for large organisations with complex permission structures.

For ABAC it is crucial to create the best possible baseline conditions, for example with proper management of relevant attributes, precise definitions of business logic, and/or policies. The big advantage of these delicate structures also bears the risk of making the overall system too confusing, which can easily result in system errors. Before implementing big changes, it sometimes is the better idea to start over which then of course requires time and effort again.

RULE BASED ACCES CONTROLL (RUBAC)

Generally, access control can work without affecting user roles. RuBAC, for instance, only works with a strictly defined set of rules, which are strictly obeyed. Here, a permission is only granted when a rule explicitly states so, instead of granting access when something is not excluded (least privilege). A rule can be temporarily effective, e.g. in order to prevent access to a resource after business hours. This is especially helpful for administrative purposes, when very sensitive information is involved. Firewalls often work with RuBAC. On an organisational level RuBAC is often not sufficient, as rules need to be constantly monitored and examined, which costs time and limits flexibility. As rules are followed without exception or nuance it is also difficult to consider business logic in RuBAC.

PRIVILEGED ACCESS MANAGEMENT (PAM)

PAM is not a general access control strategy but only looks at privileged access. This affects accounts that generally require more permissions than others, such as root accounts on computers, or other similar administrative accounts as well as user accounts with which databases can be accessed. PAM is based on

the principle of least privilege. Here, privileges can be assigned to individual users, or to user groups such as administrators. One aim of PAM is to ensure that the passwords of privileged accounts are especially long and strong, and are changed on a regular basis. Another aim is to quickly detect suspicious behaviour or other peculiarities of privileged users, if they are monitored at all. This way, it is possible to prevent big security breaches and compromised systems.
However, these kind of systems are usually very complex, and can only be implemented effectively if there are enough resources available in the local IT department. If resources are not an issue, and administrators are willing to accept such a system, a high level of security can be achieved. Many best practices for access control reference PAM as the model of choice.

CONCLUSION: ACCESS CONTROL AND IDENTITY MANAGEMENT

Commonly, it is not enough to simply decide on which resources users may access. It must be ensured that the identity of a user is compatible with their permissions. This is where identity management comes into play. Nowadays, these two terms, access control and identity management, are sometimes used synonymously. More commonly, you will find them summarised as identity & access management (IAM). This goes to show how closely intertwined identity and access management really are. Only if user identities are maintained in secure processes and always kept up to date, the mechanisms of access control can effectively work. More often than not, it is very beneficial to centralise identity management for this purpose, this way there is no risk of duplicate entries in different databases of the individual services, and the administrative effort is minimised. Moreover, for a highly flexible as well as sustainable system, the IAM software suite didmos also relies on the benefits of RBAC.

Among others, the positive consequences of a well-adjusted IAM system include the reduction of the administrative effort for the system, as well as the associated lowered workload and costs in maintenance. A well thought out IAM system tailored to your organisation – using the best suited access control model – naturally also ensures system security. Additionally, internal processes can be standardised and as a consequence simplified.