### How to manage non personal (system) accounts

Customer often ask me for best practices regarding management of non-personal or highly privileged accounts in the process of implementing an Identity and Access Management (IAM) solution. This is an interesting question, because in an IAM project, we try to manage all kinds of accounts, but this type of account is different from accounts that are owned by end users. This type of accounts can’t directly be related to a uniquely identifiable person, nor are they the result of the ‘joiner – mover - leaver’ HR processes in an organization. So, how do you manage the existence of such an account?

**Types of non personal accounts**

There are Non Personal Accounts (NPA’s) and Non-personal System Accounts: (NPSA’s). We can identify:

- **Admin or root account**

The admin or root account of Windows and Linux or Unix servers is highly privileged system account on the respective platforms.

o It is authorized at the highest level

o It has access to every file and process running on a platform.

o The ‘root’ or ‘Admin’ has the permissions to change the behavior of their component;

o Commands can be run from it as well as react to responses of the system.

o Operational use of the account needs to be monitored continuously.

- **Superuser account** It’s a business information system or application account, that looks a lot like ‘root’. It is there when the system is installed, it’s a system account. The Superuser has permission to modify, making it a risk critical account in an information system. Like Sap\* in a Sap environment.

- **Service account**

Accounts for middleware processes like DBMS’s, ESB’s or other ICT components that run on top of the Windows or Linux operating systems. A special form of a non-personal account is an application account in a DBMS to give database access to an application.

- **Batch user account**

An account used by a batch job process, it is most commonly used for scheduled batch jobs, like nightly file transfers.

**NPA characteristics**

NPSA’s have a few characteristics in common. They are non-personal and they are not directly connected to a person. Login with the account doesn’t leave an audit trace showing which person is actually using it. And, of course, NPSA’s are very powerful and so use of them should be tightly controlled.

Service and batch accounts also have a specific similarity: one typically doesn’t login with such an account, these accounts are not used interactively. In most cases such an account is only used as a placeholder with some permissions to perform specific tasks, like running a webserver with the limited capability to process https-requests and write log files.

Modern IAM solutions can be implemented to facilitate provisioning of personal accounts for specific user functionality. Since non-personal accounts are not an attribute of an identity, there is not a sole user that can be connected to an NPA, so an IAM solution is not suitable to manage them.

**If not an IAM problem, then what?** These accounts belong to the component that they manage. The Windows operating system comes with the Administrator account, Linux comes with root. You cannot install Linux without a root account. You may not by default be able to login with it (as on Ubuntu), but the account is there. So by installing an OS, you automatically get the ‘God’‑account. There is no choice, it is the result of the change process that leads to the implementation of the OS. Such an NPSA should only be used in a controlled manner from specific processes, like an incident management process (admin may be needed to assist in a catastrophe), or the change management process (the admin permissions may be required to perform an infrastructural change).

The same is true for service accounts: when installing a middleware component, like a database management system, the account is created to enable the service, hence the name service account. Again, you have no choice. You might install the service using a ‘root’ – type account, but that will result in a security violation, Thou shalt not run any service as root!

And again, for batch account the same is true again: a batch process is created as the result of a change request. The batch tasks are created to support an information system, or a business process. The batch job is created to make it possible to schedule the automatic execution of the tasks. A batch account is created to make it possible to use resources on the system.

This leads to the following conclusion:

**Non-personal accounts have to be managed in the change management process.**

**This has the following implications:** - The account has to be registered in the configuration management database, it is an attribute of the component that it belongs to. Admin belongs to the Active Directory. Root belongs to a linux server. The account named ‘Oracle’ probably belongs to an Oracle dbms instance: the dbms is a managed component and the account name is Oracle. - The account has an owner, who is accountable for the use of the account. Admin and root belong to the manager of the ICT department. The SAP account is owned by the system owner of the SAP system - The interactive accounts should only be used for infrastructural changes or calamities. - The non-personal system accounts should never be used interactively for operational tasks. - The passwords of these accounts must remain secret. They should be secured by means of an envelope procedure, a password vault, or by using a Privileged Account Management system (PAM, like CyberArk, Hitachi PAM or CA PAM to name just a few). Any use has to be related to a service management ticket (an incident or a change).

So, there you have it. Non-personal accounts must not be managed in an IAM solution, they have to be managed by the Change Management processes in an organization. Either they are owned by ICT or by the system owner who owns the information system that the account is used for. You should not manage privileged accounts in an IAM solution. And if you have to execute tasks with one of these accounts: use a Privileged Account Management system to secure it.