Efficient Datacenters Management for Network and Security Operations

Lucian Paiusescu
University
POLITEHNICA of
Bucharest
Bucharest, Romania
lucian.paiusescu@upb.ro

Mihai Barbulescu
University
POLITEHNICA of
Bucharest
Bucharest, Romania
mihai@roedu.net

Valeriu Vraciu
Alexandru Ioan Cuza
University of Iasi
Bucharest, Romania
valeriu@roedu.net

Mihai Carabas
University
POLITEHNICA of
Bucharest
Bucharest, Romania
mihai.carabas@cs.pub.ro

Abstract— **Identity management refers to the process of employing emerging technologies to manage information about the identity of users and control access to organization resources. The goal of identity management is to improve productivity and security while lowering costs associated with managing users and their identities, attributes, and credentials. The purpose of this document is to offer a broad overview of the IAM project that is being developed and tested within RoEduNet's infrastructure and giving the opportunity to bring a contribution that would benefit the organization. In this paper we are presenting the architecture with the components used and how they integrate as well as the deployment of the application in two locations across the country to provide a fault tolerance high availability scenario.**

Keywords— **identity, access, management, tacacs, sql, replication, network, RoEduNet, LDAP**

I. Introduction

Identity and access management (IAM) is a framework that facilitates the management of electronic or digital identities. The framework includes the organizational policies for managing digital identity as well as the technologies needed to support identity management. [1]

With an IAM solution, people responsible with the network management, can control user access to critical information within their organizations. Identity and access management products offer role-based access control, which lets system administrators regulate access to systems or networks based on the roles of individual users within the organization. The roles can be dynamically defined within the application and support granular customization of access rights.

An Identity and Access Management solution is a must in terms of infrastructure security as it enables the right people to access the appropriate resources when they are allowed to do so. The goal is to address a need to ensure resources are protected against unauthorized access and at the same time compliance requirements are well met within the infrastructure. For an organization an IAM solution can be a very valuable asset as it can reduce administration efforts and reduce costs if developed internally. [2]

There are various IAM commercial solutions on the market that provides all the necessary AAA level and much more, though, not many are willing to give in to the costs, especially public institutions funded by the state budget such as educational and research institutions. The alternative is to build an IAM solution internally using open source tools and link them together through a central component to achieve a solid working product that can be used in production.

When a user attempts to access a system or data, he or she first makes a claim of identity, typically by entering a username into the system. The system must then verify this claim of identity through an authentication process. Authentication may use basic knowledge-based techniques, such as passwords, or rely upon other methods, such as a centralized directory service based authentication. Once a user successfully completes the authentication process, the IAM system must then verify the user's authorization to perform the requested activity. The fact that a user proves his or her identity is not sufficient to gain access — the system must also ensure that users perform actions only within their scope of authority. Having an IAM platform can solve this administration burden in the situation where there are many network devices and you need the ability to provision access on all of them for a certain user, instead of going through each network device and configure access you can configure from a single centralized point.

Within an organization the directory service is one of the most important part when it comes to integrating various applications to work together. The IAM platform needs a way of storing or retrieving information about users or groups from a centralized directory such as LDAP or Active Directory, thus, everything can be managed from a single location by using conventional means, a web browser to interact with the system. [3]

This paper describes the protocols that the IAM platform covers to ease the process of administration for Network or System Engineers within a network infrastructure by solving the AAA burden by linking the appropriate open source components to achieve the desired outcome.

The reason we started this project is because there are no open source or free IAM solutions that covers the network part by providing an easy way to manage network equipments in terms of security context (AAA) making use of tacacs+ and radius protocols integrated in a centralized identity and access management system.

II. Design Overview

The IAM platform is composed of a set of components which are designed to provide a specific communication protocol for network devices to achieve as best as possible the granularity and strict security contexts, for example for CISCO equipments we will be using the tac_plus component which talks natively using the TACACS+ protocol which allows us to enforce full AAA capabilities and for the non-CISCO equipments we will use the FreeRADIUS component which will cover a wide variety of vendors including endpoint port-security through 802.1x. As for the services and applications running on servers they will communicate through the central IAM component that will act as a LDAP proxy, relaying queries to a directory service such as a LDAP, thus, not exposing the real directory server within the infrastructure. It also provides a caching mechanism to the directory service in case it is temporarily unavailable. The authorization part consists of filtering the LDAP query responses. Applications will check if the user has a special attribute set in order to provide access or not. The central component will provide a web interface to easily manage identity and access management. All devices will be configured to query the IAM platform in order to establish whether a user has access to a specific service or otherwise, as well as every action a user will make will be logged and stored centrally on the IAM component for future review at any given time.

Figure 1 depicts the IAM platform architecture that centrally integrates the components that provides, on the left, network management and security contexts and on the right, providing means for various applications and services to allow authorized access to specific users.

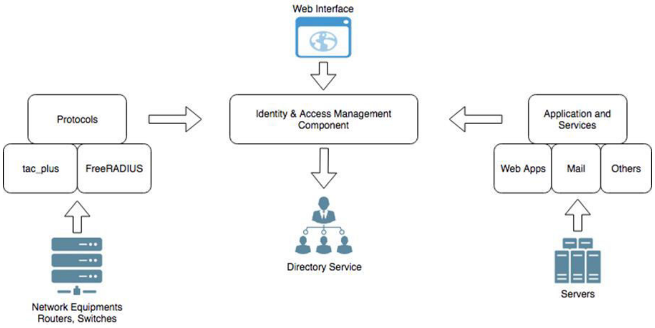


Fig. 1: Architecture

III. Implementation

The IAM platform depends on the central component which is the main application that controls the configuration and state of the other components that covers the network protocols. It provides a web interface written in PHP and as a backend for storing data it uses a MySQL database. The web interface approach offers an easy and convenient way to interact with the system. Through the web interface it is possible to configure identities (users), groups, access control lists and device access control. Users must already be present in the directory service, the interface will allow you to select the users that need to have specific roles and access in terms of who can login in order to manage a specific network device and what this user can do.

Let’s say for example there is a user – junior network administrator that we need to provide access for, on the network equipment. In order for the user to have access an identity must be created before he can login. An identity must be created for each user that we need to grant special access. Authorization can be set directly for that identity or it can be assigned to a role which makes it easier to scale when there are many identities to manage. For each identity we can assign a role or also known as a group (example operators, admins) that will inherit the authorization permissions from. The role or group must be created where we can configure all the permissions we need to allow for users that will be assigned in. In an identity or group entry we can specify an access control list (ACL) where we can restrict access to an user or group of users to not be able to login on specific network equipments, say for example, some of them do not need to have access to core routers just access switches instead. For a network equipment to be able to communicate with the IAM platform a device access control entry must be created in order to allow that specific device to query the system, it’s a security measure to authorize only trusted devices that are configured with the correct secret key. After every operation that involves adding, modifying or deleting entries the configuration must be applied by click on the Apply button. The apply function generates proper config files for both tac_plus+ and FreeRADIUS applications, and lastly it restarts the services (daemons) to initialize with the new configuration in place, as shown in Figure 2.

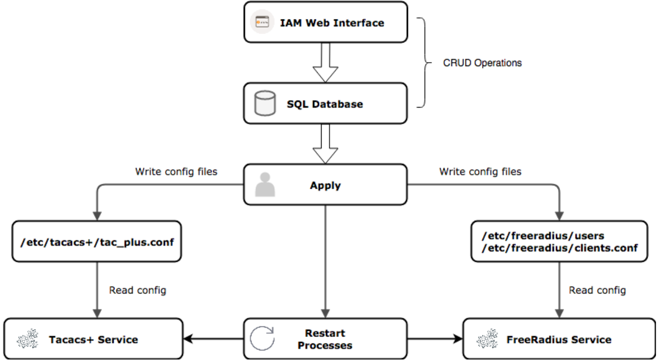


Fig. 2: Configuration cycle

All the entries are stored in the MySQL database from where the generator function reads and writes config files. Config files must not be modified manually as they will be overwritten each time new configuration is applied through the web interface.

It is also possible to have two IAM systems synchronized in a high availability scenario as it provides replication capabilities based on MySQL replication mechanism used in a master – master method. In a master – master method you are able to operate on both systems but not at the same time, the login session does not permit to be connected on both systems at once with the same user to avoid any possible conflicts. As databases keep in sync automatically, the IAM system has a script running in a cron job every minute that checks if there are changes in the database and if there are new modifications then regenerate config files to keep them in sync with the other system where the CRUD operations have been made on.

We have named the IAM platform as ACS Management which is short for Access Control System Management.

Below we are presenting the IAM platform’s file structure and a brief description of each file’ purpose:

ACS-Management:

- index.php - main file that initializes the interface, prints html
- config.php - config file holding global settings
- files/ - media folder where css, js and images reside
  |- *.png
  |- *.css
  |- *.js
  |- (meda files)
- include/ - library folder
  |- acl.inc.php - functions that provide access control list capability
  |- ajax.inc.php - used in XHR operations and provides the Apply function
  |- devices.inc.php - device control access – trust based on secret key
  |- functions.inc.php - various small functions used widely
  |- global.inc.php - provides general application settings content

|- groups.inc.php - provides group functionality content

|- help.inc.php - help documentation

|- istoric.inc.php - view who and what modifications were made using interface

|- login.inc.php - login/logout session functionality in the interface

|- main.inc.php - intermediary content used in displaying various sections

|- mysql.inc.php - mysql connector

|- replication.php* - replication script that must be run via CLI in a cron job

|- users.inc.php - provides users functionality content

IV. TESTING ENVIRONMENT

Our research is being conducted within the Romanian Educational Network (RoEduNet) institution. The Romanian educational and research network or RoEduNet represents a communication infrastructure of national interest which was defined and developed within the national education and research system. The network of RoEduNet provides data transport services between connected institutions as well as access to the European Research Educational networks and to the Internet. RoEduNet is defined on the european plan and worldwide as an NREN – National Research and Education Network – which is an institution that represents our country as each country has its own NREN. At national level, RoEduNet, connects all public institutions that are subordinated to the Romanian National Ministry of Education, Research and Innovation which involves a high responsibility and mission critical monitoring and administration tasks. The Romanian Educational Network is recognized as a critical infrastructure that operates 24x7 having in administration 8 Network Operation Centers (NOC), 2 in Bucharest, and one each in main cities: Galati, Iasi, Tg-Mures, Cluj, Timisoara, Craiova and many point of presence (PoP) placed throughout the country, as shown in Figure 3, thus, managing over 200 network equipments which gives my research opportunity to bring a centralized system in motion to ease network identity management tasks within RoEduNet's Infrastructure.

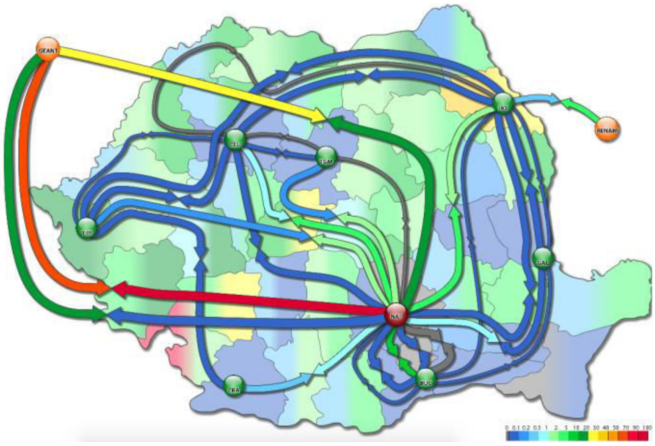


Fig. 3: RoEduNet National Infrastructure

The IAM platform, ACS Management, is installed and fully functional in two of main RoEduNet's datacenters, Bucharest and Iasi, each running on a Debian Linux in a Virtual Machine instance. The instances have been setup in two different locations in order to provide High Availability. The information data is replicated at database level through the use of the MySQL replication feature in a master-master cluster. [4]

In order to keep configuration files synchronized there is a script replication.php which tracks any changes made in the database, if new changes have been made on either instance, the script will regenerate config files and restart processes (daemons) thus having both instances with the same configuration at all times.

V. RESULT OVERVIEW

For the ACS Management application to be able to work accordingly and demonstrate its usefulness a few important settings need to be done first. User information has been configured to be retrieved from the existing LDAP Server by the ACS Management application which now allows creation of appropriate identities for the network engineers responsible of maintaining the network equipments.

All the network equipments have been configured to communicate with the ACS Management application for resolving any AAA request with the appropriate response, in a summarized view the result will be as shown in Figure 4. When a network admin is logging in on a network equipment to carry out administration tasks, proper credentials need to be entered at the login prompt. The user credentials are those from the LDAP Directory Service, and for the authentication to succeed, the user needs to have an identity configured with proper role assigned and authorization. Every time a user logins on any network equipment, the device will ask the ACS Management Server if the user exists, if correct credentials are provided and if authorized to use that service.

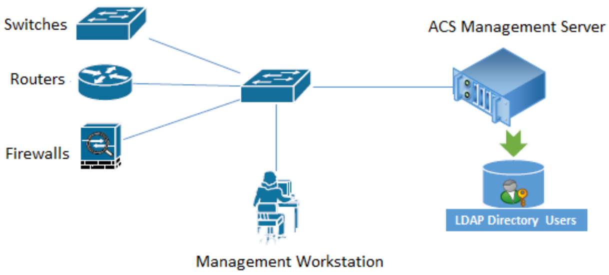
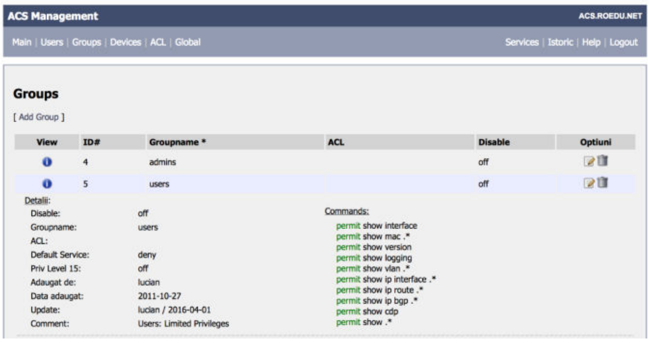


Fig. 4: Components communication overview

In our testing environment we have created identities for the appropriate users that needs to have access to login on the network equipments. Instead of making the same configuration for each identity, we have created 2 groups (roles) where authorization is configured: admins group where users have full control and for the users group a set of commands have been set in the authorization section to limit what the user can do on the equipment as shown in Figure 5, bellow.


ACS Management
ACS.ROEDU.NET
Main | Users | Groups | Devices | ACL | Global
Services | Istoric | Help | Logout

Groups

[Add Group]

View	ID#	Groupname *	ACL	Disable	Optiuni
●	4	admins		off	
●	5	users		off	

Detalii:

Disable: off
Groupname: users
ACL:
Default Service: deny
Priv Level 15: off
Adaugat de: lucian
Data adaugat: 2011-10-27
Update: lucian / 2016-04-01
Comment: Users: Limited Privileges

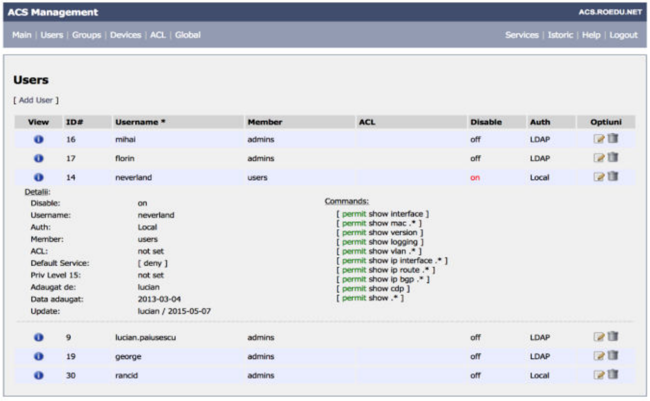
Commands:

```
permit show interface
permit show mac .*
permit show version
permit show logging
permit show vlan .*
permit show ip interface .*
permit show ip route .*
permit show ip bgp .*
permit show cdp
permit show .*
```

Fig. 5: Creating groups with permission sets

Users that are part of groups inherit the permissions set and ACL selection. As can be seen in Figure 6, we populated the users section with identities, and now we added a user neverland that is assigned to group users and we defined that the authentication method to be local, not from the LDAP directory, the other users are configured to be in admins group and authenticate using LDAP. It is possible to select multiple authentication realms, for the moment – local and LDAP can be used.

The advantage of local authentication is that if we want to create a new identity for a user that is not present in the directory service we can create one as local defining the username and password.


ACS Management
ACS.ROEDU.NET
Main | Users | Groups | Devices | ACL | Global
Services | Istoric | Help | Logout

Users

[Add User]

View	ID#	Username *	Member	ACL	Disable	Auth	Optiuni
●	16	mihai	admins		off	LDAP	
●	17	florin	admins		off	LDAP	
●	14	neverland	users		on	Local	
●	9	lucian.paiusescu	admins		off	LDAP	
●	19	george	admins		off	LDAP	
●	30	rancid	admins		off	Local	

Detalii:

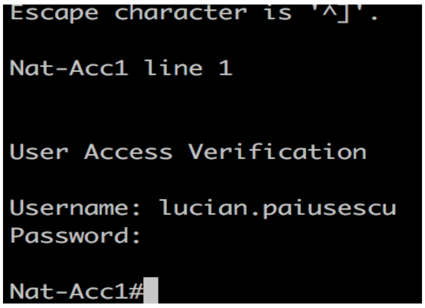
Disable: on
Username: neverland
Auth: Local
Member: users
ACL: not set
Default Service: [deny]
Priv Level 15: not set
Adaugat de: lucian
Data adaugat: 2013-03-04
Update: lucian / 2015-05-07

Commands:

```
[ permit show interface ]
[ permit show mac .* ]
[ permit show version ]
[ permit show logging ]
[ permit show vlan .* ]
[ permit show ip interface .* ]
[ permit show ip route .* ]
[ permit show ip bgp .* ]
[ permit show cdp ]
[ permit show .* ]
```

Fig. 6: Identities and roles

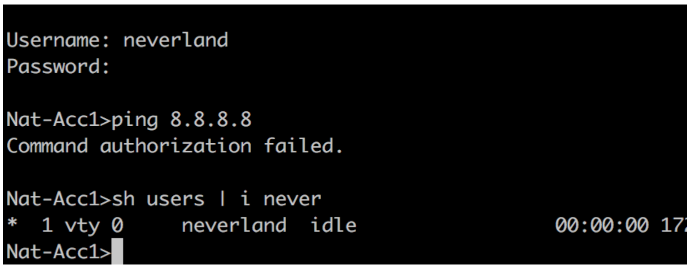
Now if you try to login to a network equipment you will be providing my credentials from the LDAP Directory and you will be granted access and full authorization as stated in the identity configured in the interface as shown in Figure 7, below.



```
Escape character is '^]'.

Nat-Acc1 line 1


User Access Verification

Username: lucian.paiusescu
Password:

Nat-Acc1#
```

Figure 7. Successful authentication based on identity set

If we connect with the user “neverland”, which is a restricted identity, we will not be able to execute any command, just the ones that have been set in the configuration section, as can be seen in the Figure 8.



```
Username: neverland
Password:

Nat-Acc1>ping 8.8.8.8
Command authorization failed.

Nat-Acc1>sh users | i never
*  1 vty 0     neverland  idle                00:00:00 17
Nat-Acc1>
```

Figure 8. Permission sets testing

Also, in order for the network devices to be able to communicate with ACS Management a Device Access Control entry must be created for each equipment with a secret key, which is the recommended method to have strict control over which devices are trusted. It is possible to define a global shared secret key that is used by all the equipments instead of creating an entry for each equipment, this method is more loose in terms of device trust as devices shares the same secret key and easier to setup.

Additionally, if we need to restrict access to a user to specific equipment’s we can define an ACL entry and specify which IP address is the user allowed to login on.

In RoEduNet’s network infrastructure the majority of the network equipments are CISCO, which is an advantage because we can make use of the TACACS+ protocol making authorization possible to be enforced on identities. For the other equipments that are not CISCO, radius is used to provide authentication and accounting, the authorization part needs further code writing in the ACS Management application because each vendor has its own way of implementing authorization attribute sets. Although there are situations where non-cisco vendors use tacacs+ protocol as well, and if supported it can be used instead of radius to provide the authorization capability without special code added. Also as the ACS Management application has the FreeRADIUS component providing radius protocol communication, it is easy to setup 802.1X on switch devices and enforce port authentication restrictions and allow only the users that have identities configured in the interface.

VI. Conclusions and Future Work

In this paper we have presented a valid open source possibility and at the same time an alternative solution to a commercial Identity and Access Management with the goal to centralize authentication, authorization and accounting with the capability to integrate with directory services or other authentication mechanisms as well as providing fault tolerance in a high availability scenario. In the document we have outlined the importance of why an Identity and Access Management solution can benefit an organization and how it can ease administration tasks especially in an environment with many network devices and at the same time providing security within the infrastructure. In this project the IAM platform is named ACS Management or Access Control System Management to which we referred to throughout the paper. We have explained the architecture which is composed of a set of components that are linked by one central component that provides a web interface to offer a convenient way to manage identities (users), groups, acl, device access control, and we have tested the application to demonstrate its usefulness and functionality within RoEduNet infrastructure.

VII. Bibliography

[1] Techtarget: Identity and access management (IAM), https://searchsecurity.techtarget.com/definition/identity-access-management-IAM-system

[2] The role of Identity Management in Information Security: Part 1 – The planning view, https://www.zdnet.com/article/the-role-of-identity-management-in-information-security-part-1-the-planning-view/

[3] Bruce Greenblatt, Building LDAP-Enabled Applications with Microsoft's Active Di- rectory and Novell's NDS, 1st Edition, Published Dec 26, 2001 by Prentice Hall.

[4] MySQL 5.5 Reference Manual, Chapter 17 Replication, https://dev.mysql.com/doc/refman/5.5/en/replication.html