# Introduction to Identity Management

## Identity Management Workshop

Victoriano Giralt

Central Computing Facility
University of Málaga, Spain

Chişinău, Replublic of Moldova
May, 15th 2007

# Disclaimer

This presentation is freely based on material borrowed from:
Keith Hazelton

Sr. IT Architect, University of Wisconsin-Madison

as presented at Porto EuroCAMP by
Ken Klingenstein

Director, Internet2 Middleware and Security

All merits should be attributed to them, and all errors to myself.

# Overview

1 What is Identity Management (IdM)?

# Overview

1. What is Identity Management (IdM)?
2. The Identity Management Stone Age

# Overview

1. What is Identity Management (IdM)?
2. The Identity Management Stone Age
3. A better vision for IdM

# Overview

1. What is Identity Management (IdM)?
2. The Identity Management Stone Age
3. A better vision for IdM
4. Basic IdM functions

# Overview

1. What is Identity Management (IdM)?
2. The Identity Management Stone Age
3. A better vision for IdM
4. Basic IdM functions
5. Demands on IT and how IdM helps

# IdM definition

## What is all this about?

We need to know what we will be talking about

# IdM definition

## What is all this about?

We need to know what we will be talking about

- What is Identity Management?

# IdM definition

## What is all this about?

We need to know what we will be talking about

- What is Identity Management?

> *"Identity management is the set of business processes, and a supporting infrastructure, for the creation, maintenance, and use of digital identities."*

The Burton Group (a research firm specializing in IT infrastructure for the enterprise)

# IdM definition

## What is all this about?

We need to know what we will be talking about

- What is Identity Management?

> *"Identity management is the set of business processes, and a supporting infrastructure, for the creation, maintenance, and use of digital identities."*

The Burton Group (a research firm specializing in IT infrastructure for the enterprise)

- Identity Management, in this sense, is often called "Identity and Access Management" (IAM)

# IdM Frequent Terms

## What do this *buzz* words mean?

We need to understand what others are talking about

# IdM Frequent Terms

## What do this *buzz* words mean?

We need to understand what others are talking about

- Digital Id

### Digital Identity

The collection of bits of identity information about you in all the relevant IT systems at your institution.
The identity must be unique inside a given domain.

# IdM Frequent Terms

## What do this *buzz* words mean?

We need to understand what others are talking about

- AuthN
- Digital Id

**Authentication**

The process that allows to verify the identity of a principal, by any means,
be them electronic or physical.
This proof of identity is also known as credentials.

# IdM Frequent Terms

## What do this *buzz* words mean?

We need to understand what others are talking about

- AuthR
- AuthN
- Digital Id

**Authorisation**

The process that validates the user's rights on a given resource, and, usually, enforces them. Also seen in the wild as AuthS (British spelling) or AuthZ (American spelling).

# IdM Frequent Terms

## What do this *buzz* words mean?

We need to understand what others are talking about

- AAI
- AuthR
- AuthN
- Digital Id

**Authentication and Authorisation Infrastructure**

A coordinated set of systems that allows institutions
to collaborate in exchanging identity data
to control the access to services
by their respective members.

# IdM Frequent Terms

## What do this *buzz* words mean?

We need to understand what others are talking about

- IdP
- AAI
- AuthR
- AuthN
- Digital Id

### Identity Provider

A.K.A. identity source.
The institution that holds
all the necesary information for
identifying a principal,
be it a person, a system or a service.

# IdM Frequent Terms

## What do this *buzz* words mean?

We need to understand what others are talking about

- SP
- IdP
- AAI
- AuthR
- AuthN
- Digital Id

**Service Provider**

A.K.A. identity consumer.
Someone that needs to know the identity of
a principal and, probably,
some associated information,
in order to grant access to a resource.

# IdM Frequent Terms

## What do this *buzz* words mean?

We need to understand what others are talking about

- SoR
- SP
- IdP
- AAI
- AuthR
- AuthN
- Digital Id

### System of Record

Those systems that collect data about indviduals
i.e., through which individuals enter
the organization.
For example: student registration or
Human Resources.

# IdM by example

## student Lisa

Let's see Lisa interacting with some University systems

# IdM by example

## student Lisa

Let's see Lisa interacting with some University systems

- “Hi! I'm Lisa.”

# IdM by example

## student Lisa

Let's see Lisa interacting with some University systems

- "Hi! I'm Lisa." (*Identity*)

# IdM by example

## student Lisa

Let's see Lisa interacting with some University systems

- "Hi! I'm Lisa." (*Identity*)
- ". . . and here're my NetID / password to prove it."

# IdM by example

## student Lisa

Let’s see Lisa interacting with some University systems

- “Hi! I’m Lisa.” (*Identity*)
- “. . . and here’re my NetID / password to prove it.” (*AuthN*)

# IdM by example

## student Lisa

Let's see Lisa interacting with some University systems

- "Hi! I'm Lisa." (*Identity*)
- ". . . and here're my NetID / password to prove it." (*AuthN*)
- "I want to do upload my assignments."

## IdM by example

### student Lisa

Let's see Lisa interacting with some University systems

- "Hi! I'm Lisa." (*Identity*)
- ". . . and here're my NetID / password to prove it." (*AuthN*)
- "I want to do upload my assignments."
  ☺ (AuthR: Allowing Lisa to use the services
  to which she's entitled)

# IdM by example

## student Lisa

Let's see Lisa interacting with some University systems

- "Hi! I'm Lisa." (*Identity*)
- "...and here're my NetID / password to prove it." (*AuthN*)
- "I want to do upload my assignments."
  ☺ (AuthR: Allowing Lisa to use the services
  to which she's entitled)
- "And I want to change my grade
  in last semester's Physics course."

# IdM by example
## student Lisa

Let's see Lisa interacting with some University systems

- "Hi! I'm Lisa." (*Identity*)
- ". . . and here're my NetID / password to prove it." (*AuthN*)
- "I want to do upload my assignments."
  ☺ (AuthR: Allowing Lisa to use the services
  to which she's entitled)
- "And I want to change my grade
  in last semester's Physics course."
  ☹ (AuthR: Preventing her from doing things
  she's not supposed to do)

# IdM by example

## New hire, Assistant Professor Alice

Some needs for Alice before she is in the payroll.

# IdM by example

## New hire, Assistant Professor Alice

Some needs for Alice before she is in the payroll.

- The Department Head wants her to have an e-mail account to give her a running start.

# IdM by example

## New hire, Assistant Professor Alice

Some needs for Alice before she is in the payroll.

- The Department Head wants her to
  have an e-mail account to give her a running start.
- How does she get into our system and get set up with the accounts and services appropriate to faculty?

# Some common questions

## to several IdM scenarios

In many IdM scenarios, this set of questions should be answered.

# Some common questions

## to several IdM scenarios

In many IdM scenarios, this set of questions should be answered.

- Are the people using these services who they claim to be?
- Are they a member of our campus community?
- Have they been given permission?
- Is their privacy being protected?

# Some common questions

## to several IdM scenarios

In many IdM scenarios, this set of questions should be answered.

- Are the people using these services who they claim to be?
- Are they a member of our campus community?
- Have they been given permission?
- Is their privacy being protected?

We can feel the smell of policy and process issues lurking nearby.

# The basic IdM functions

## those that any system needs

There are three functions a system should provide

# The basic IdM functions

## those that any system needs

There are three functions a system should provide

- AuthN: Verify the identity of principals seeking access to a service or resource

# The basic IdM functions

## those that any system needs

There are three functions a system should provide

- AuthN: Verify the identity of principals seeking access to a service or resource
- AuthR: Validate that the principal has the rights to accomplish the intended operation

# The basic IdM functions

## those that any system needs

There are three functions a system should provide

- AuthN: Verify the identity of principals seeking access to a service or resource
- AuthR: Validate that the principal has the rights to accomplish the intended operation
- Log: Track access to services / resources

# In the stone age

## tribes were issolated

In an organization that has not dawned to IdM

# In the stone age

## tribes were issolated

In an organization that has not dawned to IdM

- Every application for itself performs the IdM functions

# In the stone age

## tribes were issolated

In an organization that has not dawned to IdM

- Every application for itself performs the IdM functions
- User list, credentials, if you're on the list, you're in AuthN *IS* AuthR

# In the stone age

## tribes were issolated

In an organization that has not dawned to IdM

- Every application for itself performs the IdM functions
- User list, credentials, if you're on the list, you're in AuthN *IS* AuthR
- Some identifiers are assigned nationally with uncertain value locally

# A better vision of IdM

## a cure to the yellow stickers syndrome

IAM as a middleware layer at the service of any number of applications,
which needs an expanded function set

# A better vision of IdM

## a cure to the yellow stickers syndrome

IAM as a middleware layer at the service of
any number of applications,
which needs an expanded function set

- Reflect: Track changes to institutional data from changes in SoR and other IdM components

# A better vision of IdM

## a cure to the yellow stickers syndrome

IAM as a middleware layer at the service of
any number of applications,
which needs an expanded function set

- Reflect: Track changes to institutional data from changes in SoR and other IdM components
- Join: Establish & maintain person identity across SoR

# A better vision of IdM

## a cure to the yellow stickers syndrome

IAM as a middleware layer at the service of
any number of applications,
which needs an expanded function set

- Reflect: Track changes to institutional data from changes in SoR and other IdM components
- Join: Establish & maintain person identity across SoR
- Credential: issue digital credentials to people in the community

# Identity fragmentation

## the cancer of IdM

There are two important elements for the diagnose of the disease

# Identity fragmentation

## the cancer of IdM

There are two important elements for the diagnose of the disease

- For any given person in the community,
  do we know which entry in each system's data store
  carry bits of their identity?

# Identity fragmentation

## the cancer of IdM

There are two important elements for the diagnose of the disease

- For any given person in the community,
  do we know which entry in each system's data store carry bits of their identity?
- How many systems can create a "person record"?
  more than one $=>$ identity fragmentation

# The Join

## we have a cure for cancer (in IdM)

The number one cure for identity fragmentation is

# The Join

## we have a cure for cancer (in IdM)

The number one cure for identity fragmentation is: *The Join*

# The Join

## we have a cure for cancer (in IdM)

The number one cure for identity fragmentation is: *The Join*

For it, we have to use bussiness logic to

## The Join

we have a cure for cancer (in IdM)

The number one cure for identity fragmentation is: *The Join*
For it, we have to use bussiness logic to

- Establish which records correspond to the same person
- Maintain that identity join in the face of changes to data in collected systems

# Identity Information Access

## Implementig The Join

In order to implement The Join,
we need to access indentity information

# Identity Information Access

## Implementig The Join

In order to implement The Join,
we need to access indentity information

- Some direct from the Enterprise Directory via reflection from SoR

# Identity Information Access

## Implementig The Join

In order to implement The Join,
we need to access indentity information

- Some direct from the Enterprise Directory via reflection from SoR
- Some other bits, reached through identifier crosswalks

| Registry ID | Sys A ID | Sys B ID | Sys C ID | Sys D ID |
|---|---|---|---|---|
| 3a104e59 | fsmith32 | 86443 | freds | 864164 |
| 8c2f916d | abecker1 | 45209 | amyb | 752731 |

# If you can't integrate, *federate*

## another way to cure identity fragmentation

The second best cure for identity fragmentation is

# If you can't integrate, *federate*

## another way to cure identity fragmentation

The second best cure for identity fragmentation is: *Federation*

# If you can't integrate, *federate*

## another way to cure identity fragmentation

The second best cure for identity fragmentation is: *Federation*
Federated IdM

# If you can't integrate, *federate*

## another way to cure identity fragmentation

The second best cure for identity fragmentation is: *Federation*
Federated IdM

- Rely on the Identity Management infrastructure of one or more institutions or units

# If you can't integrate, *federate*

## another way to cure identity fragmentation

The second best cure for identity fragmentation is: *Federation*
Federated IdM

- Rely on the Identity Management infrastructure of one or more institutions or units
- To authenticate and pass authorization-related information to service providers or resource hosts

# If you can't integrate, *federate*

## another way to cure identity fragmentation

The second best cure for identity fragmentation is: *Federation*
Federated IdM

- Rely on the Identity Management infrastructure of one or more institutions or units
- To authenticate and pass authorization-related information to service providers or resource hosts
- Via institution-to-provider agreements

# If you can't integrate, *federate*

## another way to cure identity fragmentation

The second best cure for identity fragmentation is: *Federation*
Federated IdM

- Rely on the Identity Management infrastructure of one or more institutions or units
- To authenticate and pass authorization-related information to service providers or resource hosts
- Via institution-to-provider agreements
- Facilitated by common membership in a federation

# Expand the basic functions set

## new views require new ways of doing things

This new approach to doing IdM require some new functions

# Expand the basic functions set

## new views require new ways of doing things

This new approach to doing IdM require some new functions

- Mng. Affil.: Manage affiliation and group information

# Expand the basic functions set

## new views require new ways of doing things

This new approach to doing IdM require some new functions

- Mng. Affil.: Manage affiliation and group information
- Mng. Priv.: Manage privileges and permissions
  at system and resource level

# Expand the basic functions set

## new views require new ways of doing things

This new approach to doing IdM require some new functions

- Mng. Affil.: Manage affiliation and group information
- Mng. Priv.: Manage privileges and permissions at system and resource level
- Provision: Push IAM info out to systems and services as required

# Expand the basic functions set

## new views require new ways of doing things

This new approach to doing IdM require some new functions

- Mng. Affil.: Manage affiliation and group information
- Mng. Priv.: Manage privileges and permissions at system and resource level
- Provision: Push IAM info out to systems and services as required
- Relay: Make access control / authorization information available to services and resources at run time

# Expand the basic functions set

## new views require new ways of doing things

This new approach to doing IdM require some new functions

- Mng. Affil.: Manage affiliation and group information
- Mng. Priv.: Manage privileges and permissions at system and resource level
- Provision: Push IAM info out to systems and services as required
- Relay: Make access control / authorization information available to services and resources at run time
- AuthR: Make the allow deny decision independent of AuthN

# Managing privileges and roles

## Who does what

Role-Based Access Control (*RBAC*) model

# Managing privileges and roles

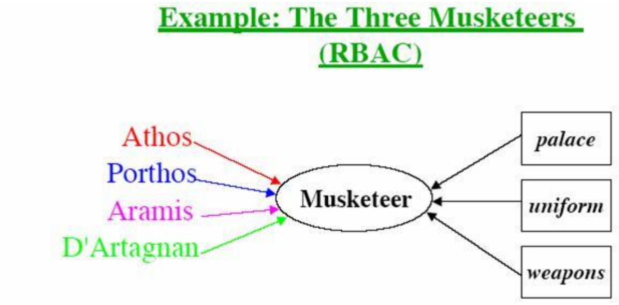
## Who does what

Role-Based Access Control (*RBAC*) model

- Users are placed into groups
- Privileges are assigned to groups
- Groups can be arranged into hierarchies to effectively bestow privileges

# Managing privileges and roles

## A nice example



Example: The Three Musketeers (RBAC)

# Provisioning

## Getting identity information where it needs to be

This is a process designed for getting identity into applications with *an attitude* by

# Provisioning

## Getting identity information where it needs to be

This is a process designed for getting identity into applications with *an attitude* by

- Exporting reformatted information to them in a form they understand

# Provisioning

## Getting identity information where it needs to be

This is a process designed for getting identity into applications with *an attitude* by

- Exporting reformatted information to them in a form they understand
- Using either app-provided APIs

# Provisioning

## Getting identity information where it needs to be

This is a process designed for getting identity into applications with *an attitude* by

- Exporting reformatted information to them in a form they understand
- Using either app-provided APIs
- Or tricks to write to their internal store

# Provisioning

## Getting identity information where it needs to be

This is a process designed for getting identity into applications with *an attitude* by

- Exporting reformatted information to them in a form they understand
- Using either app-provided APIs
- Or tricks to write to their internal store

# Provisioning

## Getting identity information where it needs to be

This is a process designed for getting identity into applications with *an attitude* by

- Exporting reformatted information to them in a form they understand
- Using either app-provided APIs
- Or tricks to write to their internal store

Change happens, so this is an ongoing process

# Application/IdM integration

## bringing applications to the future

There are two modes for integrating IdM and applications

# Application/IdM integration

## bringing applications to the future

There are two modes for integrating IdM and applications

- For domesticated applications:
  Provide them with the full set of IdM functions

# Application/IdM integration

## bringing applications to the future

There are two modes for integrating IdM and applications

- For domesticated applications:
  Provide them with the full set of IdM functions
- For applications with attitude included:
  Meet them more than halfway by provisioning

# We have a single SoR

## should we use it as the Entreprise Directory?

Before deciding on the use of a single SoR as the Directory, some questions should be answered

# We have a single SoR

## should we use it as the Entreprise Directory?

Before deciding on the use of a single SoR as the Directory, some questions should be answered

- Who “owns“ the system?

# We have a single SoR

## should we use it as the Entreprise Directory?

Before deciding on the use of a single SoR as the Directory, some questions should be answered

- Who “owns“ the system?
- Do the owners perceive they run a shared infrastruture?

# We have a single SoR

## should we use it as the Entreprise Directory?

Before deciding on the use of a single SoR as the Directory, some questions should be answered

- Who “owns“ the system?
- Do the owners perceive they run a shared infrastruture?
- Will any “external” populations ever become “internal“?

# We have a single SoR

## should we use it as the Entreprise Directory?

Before deciding on the use of a single SoR as the Directory, some questions should be answered

- Who “owns“ the system?
- Do the owners perceive they run a shared infrastruture?
- Will any “external” populations ever become “internal“?
- How does the system score when confronted to the basic IdM functions?

# Some policy issues

## the ”recredential“ function: NetID

On the life cycle of digital identities

# Some policy issues

## the ”recredential“ function: NetID

On the life cycle of digital identities

- When to assign / activate?

# Some policy issues

## the "recredential" function: NetID

On the life cycle of digital identities

- When to assign / activate?As early as possible

# Some policy issues

## the ”recredential“ function: NetID

On the life cycle of digital identities

- When to assign / activate?As early as possible
- Who gets them?

# Some policy issues

## the ”recredential“ function: NetID

On the life cycle of digital identities

- When to assign / activate?As early as possible
- Who gets them?
- “Guest” NetIDs (temporary, identity-less)

# Some policy issues

## the ”recredential“ function: NetID

On the life cycle of digital identities

- When to assign / activate?As early as possible
- Who gets them?
- “Guest” NetIDs (temporary, identity-less)
- When to reassign?

# Some policy issues

## the ”recredential“ function: NetID

On the life cycle of digital identities

- When to assign / activate?As early as possible
- Who gets them?
- “Guest” NetIDs (temporary, identity-less)
- When to reassign?Never

# Some policy issues

## the ”recredential“ function: NetID

On the life cycle of digital identities

- When to assign / activate?As early as possible
- Who gets them?
- “Guest” NetIDs (temporary, identity-less)
- When to reassign?Never, except . . .

# Some policy issues

## the ”recredential“ function: NetID

On the life cycle of digital identities

- When to assign / activate?As early as possible
- Who gets them?
- “Guest” NetIDs (temporary, identity-less)
- When to reassign?Never, except . . .
- Who can handle them?

# Requirements

## old and new, and then some

What IT is being asked to do

# Requirements

## old and new, and then some

What IT is being asked to do

- Automatic creation and deletion of computer accounts

# Requirements

## old and new, and then some

What IT is being asked to do

- Automatic creation and deletion of computer accounts
- Personnel records access for legal compliance

# Requirements

## old and new, and then some

What IT is being asked to do

- Automatic creation and deletion of computer accounts
- Personnel records access for legal compliance
- One stop for university services

# Requirements

## old and new, and then some

What IT is being asked to do

- Automatic creation and deletion of computer accounts
- Personnel records access for legal compliance
- One stop for university services
- Comply with a growing list of policy mandates

# Requirements

## old and new, and then some

What IT is being asked to do

- Automatic creation and deletion of computer accounts
- Personnel records access for legal compliance
- One stop for university services
- Comply with a growing list of policy mandates
- Increase the level of security protections
in the face of a steady stream of new threats

# Requirements

## old and new, and then some

What IT is being asked to do

- Automatic creation and deletion of computer accounts
- Personnel records access for legal compliance
- One stop for university services
- Comply with a growing list of policy mandates
- Increase the level of security protections
  in the face of a steady stream of new threats
- Serve new populations (alumni, applicants, Bologna, . . . )

# Requirements

## old and new, and then some

What IT is being asked to do

- Automatic creation and deletion of computer accounts
- Personnel records access for legal compliance
- One stop for university services
- Comply with a growing list of policy mandates
- Increase the level of security protections
  in the face of a steady stream of new threats
- Serve new populations (alumni, applicants, Bologna, . . . )
- More requests for new services and
  new combinations of services

# Requirements

old and new, and then some

What IT is being asked to do

- Automatic creation and deletion of computer accounts
- Personnel records access for legal compliance
- One stop for university services
- Comply with a growing list of policy mandates
- Increase the level of security protections in the face of a steady stream of new threats
- Serve new populations (alumni, applicants, Bologna, . . . )
- More requests for new services and new combinations of services
- Increased interest in eBusiness

## Requirements

old and new, and then some

Looks overwhelming

# Requirements

## old and new, and then some

Looks overwhelming

It *IS*

## Requirements
old and new, and then some

Looks overwhelming

It *IS*

And there is an Identity Management aspect to each and every one of these items

# IdM as a helping aid
## IdM rescues haired IT profesionals

How full IdM layer helps

# IdM as a helping aid

## IdM rescues haired IT profesionals

How full IdM layer helps

- Improves scalability: IdM process automation

# IdM as a helping aid

## IdM rescues haired IT profesionals

How full IdM layer helps

- Improves scalability: IdM process automation
- Reduces complexity of IT ecosystem

# IdM as a helping aid

## IdM rescues haired IT profesionals

How full IdM layer helps

- Improves scalability: IdM process automation
- Reduces complexity of IT ecosystem
  complexity seen as friction $=>$ wasted resources

# IdM as a helping aid

## IdM rescues haired IT profesionals

How full IdM layer helps

- Improves scalability: IdM process automation
- Reduces complexity of IT ecosystem
  complexity seen as friction => wasted resources
- Improved user experience

# IdM as a helping aid

## IdM rescues haired IT profesionals

How full IdM layer helps

- Improves scalability: IdM process automation
- Reduces complexity of IT ecosystem
  complexity seen as friction => wasted resources
- Improved user experience
- Functional specialization

# IdM as a helping aid

## IdM rescues haired IT profesionals

How full IdM layer helps

- Improves scalability: IdM process automation
- Reduces complexity of IT ecosystem
  complexity seen as friction $=>$ wasted resources
- Improved user experience
- Functional specialization
  Application developers can concentrate on application-specific functionality

# Evolution of IdM

## from construction to integration

The way of doing things is changing

# Evolution of IdM

## from construction to integration

The way of doing things is changing

- Construction

# Evolution of IdM

## from construction to integration

The way of doing things is changing

- Construction
  Raw materials into systems

# Evolution of IdM

## from construction to integration

The way of doing things is changing

- Construction
  Raw materials into systems
- Integration

# Evolution of IdM

## from construction to integration

The way of doing things is changing

- Construction
  Raw materials into systems
- Integration
  - Subsystems into whole systems

# Evolution of IdM

## from construction to integration

The way of doing things is changing

- Construction
  Raw materials into systems
- Integration
  - Subsystems into whole systems
  - Multiple systems into ecosystems

# Evolution of IdM

## from construction to integration

The way of doing things is changing

- Construction
  Raw materials into systems
- Integration
  - Subsystems into whole systems
  - Multiple systems into ecosystems
- We are all moving from construction to integration

# IdM Functions

## the extended set

| | |
|---|---|
| *Reflect* | data of interest |
| *Join* | identity across SoR |
| *Credential* | NetID, other |
| *Manage Affil/Groups* | AuthR info |
| *Manage Privileges* | more AuthR info |
| *Provision* | Get AuthNR info into app space |
| *Relay* | AuthR info to app on request |
| *Authenticate* | identity claim |
| *Authorise* | access decision (allow / deny) |
| *Log* | for audit, accounting, diagnose, . . . |

# Same functions

## different packaging

And finally . . .

# Same functions

## different packaging

And finally . . .

- Your IdM infrastructure (existing or planned) may be different from mine

# Same functions
## different packaging

And finally . . .

- Your IdM infrastructure (existing or planned) may be different from mine
- But somewhere, somehow
  the set of IdM functions is getting done

# Same functions
## different packaging

And finally . . .

- Your IdM infrastructure (existing or planned) may be different from mine
- But somewhere, somehow
  the set of IdM functions is getting done

# Same functions

## different packaging

And finally . . .

- Your IdM infrastructure (existing or planned) may be different from mine
- But somewhere, somehow the set of IdM functions is getting done
- We can compare our solutions by looking at the various packagings of the IdM functions