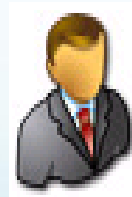


The LDAP Directory Life After Sun

A story of migration



Alban MEUNIER

IdM Senior consultant

ameunier@smartwavesa.com

www.smartwavesa.com

Agenda

- Introduction
- Common layer
- Migrate a standalone instance
- Migrate a replicated infra
- Migrate a complex LDAP infra
- Conclusion

Agenda

- **Introduction**
- Common layer
- Migrate a standalone instance
- Migrate a replicated infra
- Migrate a complex LDAP infra
- Conclusion

Introduction

- Ageing versions of former directories market leaders
 - Sun Directory 5.2
 - Novell eDirectory 8.7
- Compatibility matrix of applications has changed
 - Solaris and Suse ↘↘
 - Sun and Novell directories ↘↘↘
 - MS Active Directory ↗↗↗
 - LDAP V3, OpenLDAP ↗↗
 - IBM, TDS, OpenDJ, Apache DS, Redhat DS ↗
- Open source went out universities
 - Political trend on public sector
 - Ready for critical applications
 - Several enterprise grade level projects

Agenda

- Introduction
- **Common layer**
- Migrate a standalone instance
- Migrate a replicated infra
- Migrate a complex LDAP infra
- Conclusion

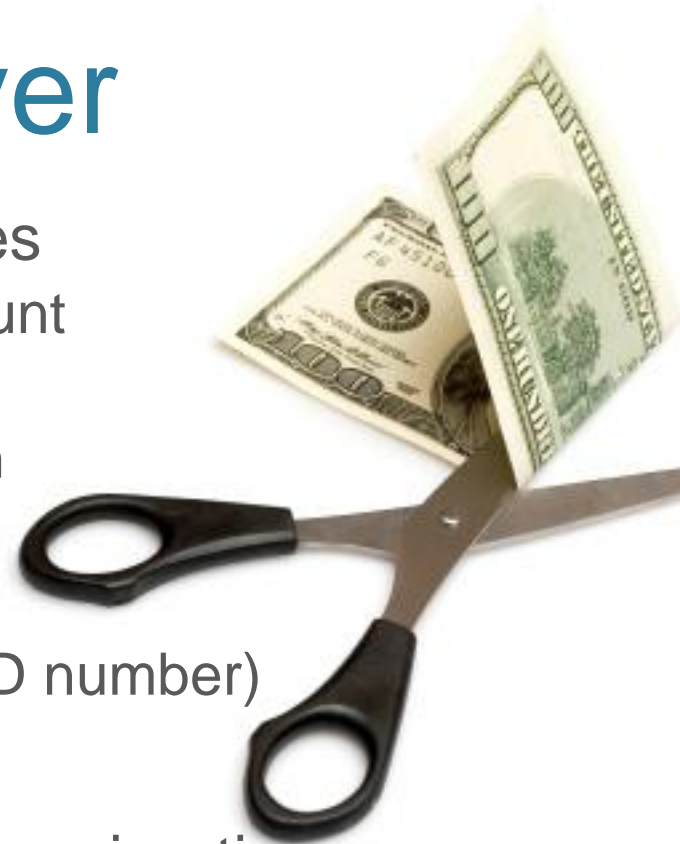
Common layer

- The directory you operate is unique
 - Fast
 - Stable
 - Effortless to operate
 - Fits all the current needs
 - Low/no more support cost
 - Well designed with no need to improve
- Unique? Probably not....



Common layer

- Limited implementation of best practices
 - Intensive usage of default admin account
 - Poor password policy
 - Use of unsecure LDAP communication
 - Logs not consolidated
 - No regular DRP tests
 - Lazy schema extension (no unique OID number)
 - Minimum/no periodic reports
- External constraints force you to plan a migration
 - Better Microsoft integration (AD, SharePoint)
 - New OS, virtualisation,
 - New editor strategic partnerships
 - Delegated operation (contractor, self service, apps owner)



Common layer

Anticipate and choose your migration path



Start with a good preparation

- Data cleaning
 - Attributes with no value
 - Unify data format
 - Unused entries
- Schema check
 - Identify unused extensions
 - Have your IANA PEN ready
<http://pen.iana.org/pen/PenApplication.page>
 - Indexes
- Third party: inventory and DNS alias
 - Scripts, application config
 - DNS, load balancers, LDAP proxies, virtual directory



Start with a good preparation

- Well known complex features
 - Define minimum performance metrics
 - Multiple intricate nested groups
 - ACL's
 - avoid redundancy and conflicting rules
 - limit personal ACLs and privilege group/sub tree
 - Check the best way to track fine grain changes
 - Change log, audit log, persistent search
 - External tool for delta evaluation
 - Identity management, provisioning
 - Supported control
 - Server-Side Sort Control, Virtual List View Control, ...
 - Persistent Search Control, Proxy Authorisation Control, Get Effective Rights Control,

```
ldapsearch -s base -b "" (objectclass=*) supportedControl
```

The password case

- The password policies
 - Identify each one and get
 - complexity
 - entries concerned
 - inheritance
 - Get the special attributes like
 - Pointers to the password policy
 - Failed login count
 - Locked status
- Internal key for password encryption
- Gettable or not
- Compatible hash or not



The operational attributes are often lost or changed

- TimeStamp
 - Creation
 - Modification
 - Last login
- DN
 - Created by, Updated by
 - Parent entry, referral
- Other
 - Nb of subordinates
 - Internal entry ID
 - Tombstone and replication data
- Virtual attributes

Different LDAPv3 implementation

- Schema
 - inetorgPerson vs user
 - groupOfName vs groupOfUniqueName
 - naming attributes (users with uid vs cn)
- DIT
 - An entry could be a container or a leaf
- ACL
 - No standard for the syntax
 - Several types (global, default, custom, dynamic)
- Plug-ins, overlay, extensions, DSML
- Virtual attributes

Install a DEV environment

- Check supported control
 - If all you need is present 😊
 - If not, you will have to ☹
 - find a workaround in the client applications
 - develop a custom extension of the directory if possible
 - change the version/vendor of the new directory
- Check existing vendor schema
 - Check syntax of attributes editor schema (DN, timestamp)
 - Check required and optional attributes
 - Adapt if necessary (script changes for future update)
- Extend the schema using OID
- Set indexes and virtual attributes (if supported)



Tune the DEV environment

- Activate LDAPS/TSL and HTTPS
- Adjust anonymous access
- Rewrite the ACLs, referrals
- Rewrite the password policies
- Plug-ins, overlay, extension, DSML
- Implement regular monitoring (snmp, logs, scripts, ...)
- Think periodic reports (dedicated tools, custom script or standard tools with <http://myvd.sourceforge.net/bridge.html>)
- Update best practices and docs



Install a PROD environment

- Install as DEV but
 - Rename and/or use non default admins
 - Use complex and dedicated passwords
 - Use crypted disk volumes
 - Use dedicated system user and avoid root
 - Use scripted installation +++
 - Bind to network interface
- Set the certificates
 - CA certificate
 - Instance certificate
 - Replication certificates
 - Activate LDAPS, TLS, HTTPS
 - Clients certificate store



Backup and restore

- Backup
 - Old directory
 - New directory with no data
- TEST full restore
 - Old directory (on a new machine)
 - New directory
 - Environment
 - Engine
 - Instance
 - Configuration
- TEST at least one rollback
- Define procedure and time for rollback



Go Live

- Communicate about changes and potential service disruption
- Load data in the new directory (detailed in next slides)
- Check list
- Eventually apply delta from old directory
- Open firewalls, switch DNS alias
- Restart some client applications
- Get confident with the new directory
- Decommission the old directory



Agenda

- Introduction
- Common layer
- **Migrate a standalone instance**
- Migrate a replicated infra
- Migrate a complex LDAP infra
- Conclusion

Standalone Compatible directory

- Example of compatible directories
 - Same editor N → N+x release (including Sun → Oracle)
 - Same origine like
 - Sun → Redhat DS, CentOS DS, 389
 - OpenDS → OpenDJ, Oracle Unified Directory
- Set the replication
 - Configure ONE WAY flow
 - Old to new
 - 2 ways are rarely supported
 - Initialise the new directory with data from the old one

Standalone

Not compatible directory

- On the old directory
 - activate the changelog/audit/persistent search tool
 - prepare delta export and import automation (coexistence)

- Export data in LDIF

- Full DB if possible to avoid virtual attributes and referrals
- Data without following referrals

- **Adapt** the export file to be compliant with new directory

- +++++ script +++++
 - Normalise DN (‘, ’ → ‘, ’ case)
 - Add: objectClasse, default values
 - Remove: system attributes, incompatible attributes/objectclass
 - Change: attribute name, trim spaces, date format, DIT, referrals

```

sub clean_line() {
    ($ligne) = @_ ;
    chomp $ligne;
    if ( $origin_LANG eq "nl" ) { # nl
        if ( $ligne =~ m/<\div>/ ) { return ""; }
        if ( $ligne =~ m/^\{Wikimedia.\}/ ) { return ""; }
        $ligne =~ s/"/"/g;
    } elsif ( $origin_LANG eq "en" ) { # en
        $ligne =~ s/^\{.\*\}\./ / ;
    } elsif ( $origin_LANG eq "it" ) { # it
        if ( $ligne =~ m/^\[Image:.*$/ ||
            $ligne =~ m/^\[Image:.*$/ ) { return ""; }
    }
    $ligne =~ s/\\|\/|\/g ; #rien interne renommé
    $ligne =~ s/#[^\]]*/ /g ; #ancres
    if ( $ligne =~ m/^\|\/| || $ligne =~ m/\/|\/ / ) { return ""; } #tableau
    if ( $ligne =~ m/^\|\/ / ) { return ""; } #<tr> / <td>

    if ( $ligne =~ m/<?[A-Za-z0-9]*>/ ) {
        die ("Erreur : balise html à la ligne $. : \n$ligne\n");
    }
    if ( $ligne =~ m/==.*==/ ) {
        return $ligne;
    } elsif ( $ligne =~ m/\[.(*)\]/ ) {

```

Standalone

Not compatible directory

- Import LDIF in new directory
 - When possible, use bulk import tools
- On the new directory
 - activate the changelog/audit/persistent search tool
 - prepare delta export and import automation (rollback)
 - +++++ script +++++
 - Normalize DN (‘, ’ → ‘,’ case)
 - Add: objectClass, default values
 - Remove: system attributes, incompatible attributes/objectclass
 - Change: attribute name, trim spaces, date format, DIT, referrals
 -

Agenda

- Introduction
- Common layer
- Migrate a standalone instance
- **Migrate a replicated infra**
- Migrate a complex LDAP infra
- Conclusion

Replicated infra Compatible directory

- Set the replication
 - Configure ONE WAY flow
 - If nb of existing replica is already at it's max supported, unconfigure one replica
 - Old to new
 - 2 ways are rarely supported ☹
 - Initialise the new directory with data from one old one
 - Adapt the procedure with referral, multiple dbs, ...



Replicated infra

Not compatible directory

- On every old directory instances with write capabilities
 - activate the changelog/audit/persistent search tool
 - prepare delta export and import automation (coexistence from consolidated export timestamp sorted)
- Export in LDIF (full DB if possible)
- Adapt the export file to be compliant with new directory
- Import LDIF in one of the new directories set in MMR
 - When possible, use bulk import tools
- On every new directory with write capabilities,
 - activate the changelog/audit/persistent search tool
 - prepare delta export and import automation (rollback)

Agenda

- Introduction
- Common layer
- Migrate a standalone instance
- Migrate a replicated infra
- **Migrate a complex LDAP infra**
- Conclusion

What is a complex infra



- Multiple backends
- Referrals
- Replication topology with hubs
- LDAP access through virtual directory complex rules
- Instance with intensive write operations

Complex infra Compatible directory

- Prepare the new topology and try to simplify taking advantage of new machine capabilities
- Set the replication
 - Configure ONE WAY flow
 - If nb of existing replica is already at it's max, unconfigure one replica
 - Old to new
 - 2 ways are rarely supported
 - Initialise the new directory with data from the old one using REPLICATION over LDAP and not using binary feeding

Complex infra

Not compatible directory

- Try decrease the number of old directory instances with write capabilities
 - Adapt DNS alias
 - Use LDAP proxy
 - to separate write and read requests
 - to migrate step by step
- Use hub replica to decrease the network traffic
- Design the new replica topology to minimise the number of servers on recent hardware
 - Bandwidth 100/1000 Mbps → 1000/10000 Mbps
 - RAM 4/8 Go → 32/64 Go
 - CPU 4/8 x 1 core → 4/16 x 8 cores
 - Local store → SAN with separate log volume



Agenda

- Introduction
- Common layer
- Migrate a standalone instance
- Migrate a replicated infra
- Migrate a complex LDAP infra
- **Conclusion**

Conclusion

- LDAP v3 is a standard that can't guaranty the migration success due to many different vendor implementations
- Don't underestimate the technical efforts for scripting
- A good migration requires a good preparation
- A good opportunity to
 - improve your control on directories
 - open to new services (VoIP, identity federation, ...)
- Most of directory migrations are success stories even if directories are considered as a commodity

The LDAP Directory Life After Sun

That's all Folks!