

# OpenLDAP Replication Workshop

Nov 4, 2019

LDAPCon, Sofia

# Objective

This tutorial takes the attendee through the process of setting up a multi-tier OpenLDAP replication network. The use case covers installation and configuration of a multi-master cluster and selectively replicating entries based on filters. The documentation provides the instructions for installing onto Centos 7 virtual machines including sample configurations and test cases used to verify completion.

# Introductions

- Maryanne Normann
  - [mnormann@symas.com](mailto:mnormann@symas.com)
- Shawn McKinney
  - [smckinney@symas.com](mailto:smckinney@symas.com)

# Agenda

1. Introduce LDAP and Common Use Cases
2. Describe Sample System Requirements & Architecture
3. Describe Test Configuration
4. Describe Sample Project
5. Break 1
6. Student Environment Preparation
7. Install/Test Masters
8. Break 2
9. Install/Test Replicas
10. Wrap-up

# Lightweight Directory Access Protocol

---

From Wikipedia, the free encyclopedia

The **Lightweight Directory Access Protocol** (**LDAP**; [/ˈɛldæp/](#)) is an open, vendor-neutral, industry standard [application protocol](#) for accessing and maintaining distributed directory information services over an [Internet Protocol](#) (IP) network.<sup>[1]</sup> [Directory services](#) play an important role in developing [intranet](#) and Internet applications by allowing the sharing of information about users, systems, networks, services, and applications throughout the network.<sup>[2]</sup> As examples, directory services may provide any organized set of records, often with a hierarchical structure, such as a corporate [email](#) directory. Similarly, a [telephone directory](#) is a list of subscribers with an address and a phone number.

LDAP is specified in a series of [Internet Engineering Task Force](#) (IETF) Standard Track publications called [Request for Comments](#) (RFCs), using the description language [ASN.1](#). The latest specification is Version 3, published as [RFC 4511](#) [↗](#).



# RFC2307bis

Network Working Group

Internet-Draft

Obsoletes: [2307](#) (if approved)

Intended status: Informational

Internet-Draft

LDAP NameService Schema

L. Howard

PADL Software

H. Chu, Ed.

Symas Corp.

August 2009

## **An Approach for Using LDAP as a Network Information Service draft-howard-rfc2307bis-02.txt**

This document describes a mechanism for mapping entities related to TCP/IP and the UNIX system [[UNIX](#)] into [[X.500](#)] entries so that they may be resolved with the Lightweight Directory Access Protocol [[RFC4511](#)]. A set of attribute types and object classes are proposed, along with specific guidelines for interpreting them. The intention is to assist the deployment of LDAP as an organizational nameservice. No proposed solutions are intended as standards for the Internet. Rather, it is hoped that a general consensus will emerge as to the appropriate solution to such problems, leading eventually to the adoption of standards. The proposed mechanism has already been implemented with some success.



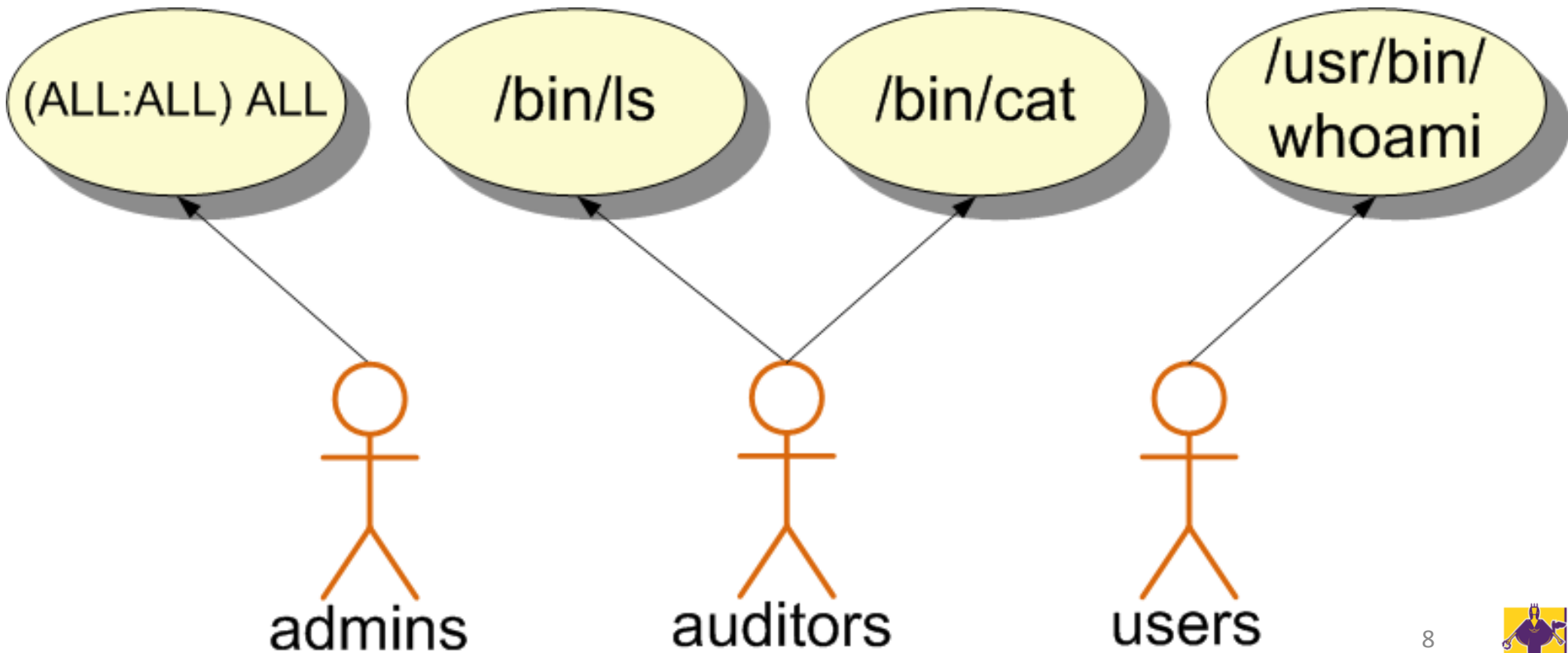
# Use RFC2307bis LDAP Schema

```
( 1.3.6.1.1.1.2.0 NAME 'posixAccount' SUP top AUXILIARY  
  DESC 'Abstraction of an account with POSIX attributes'  
  MUST ( cn $ uid $ uidNumber $ gidNumber $ homeDirectory )  
  MAY ( authPassword $ userPassword $ loginShell $ gecos $  
        description ) )
```

```
( 1.3.6.1.1.1.2.2 NAME 'posixGroup' SUP top AUXILIARY  
  DESC 'Abstraction of a group of accounts'  
  MUST gidNumber  
  MAY ( authPassword $ userPassword $ memberUid $  
        description ) )
```



# sudo





# sudo LDAP Schema

```
objectclass ( 1.3.6.1.4.1.15953.9.2.1
  NAME 'sudoRole' SUP top STRUCTURAL
  DESC 'Sudoer Entries'
  MUST ( cn )
  MAY ( sudoUser $ sudoHost $ sudoCommand
    $ sudoRunAs $ sudoRunAsUser
    $ sudoRunAsGroup $ sudoOption
    $ sudoNotBefore $ sudoNotAfter
    $ sudoOrder $ description )
)
```



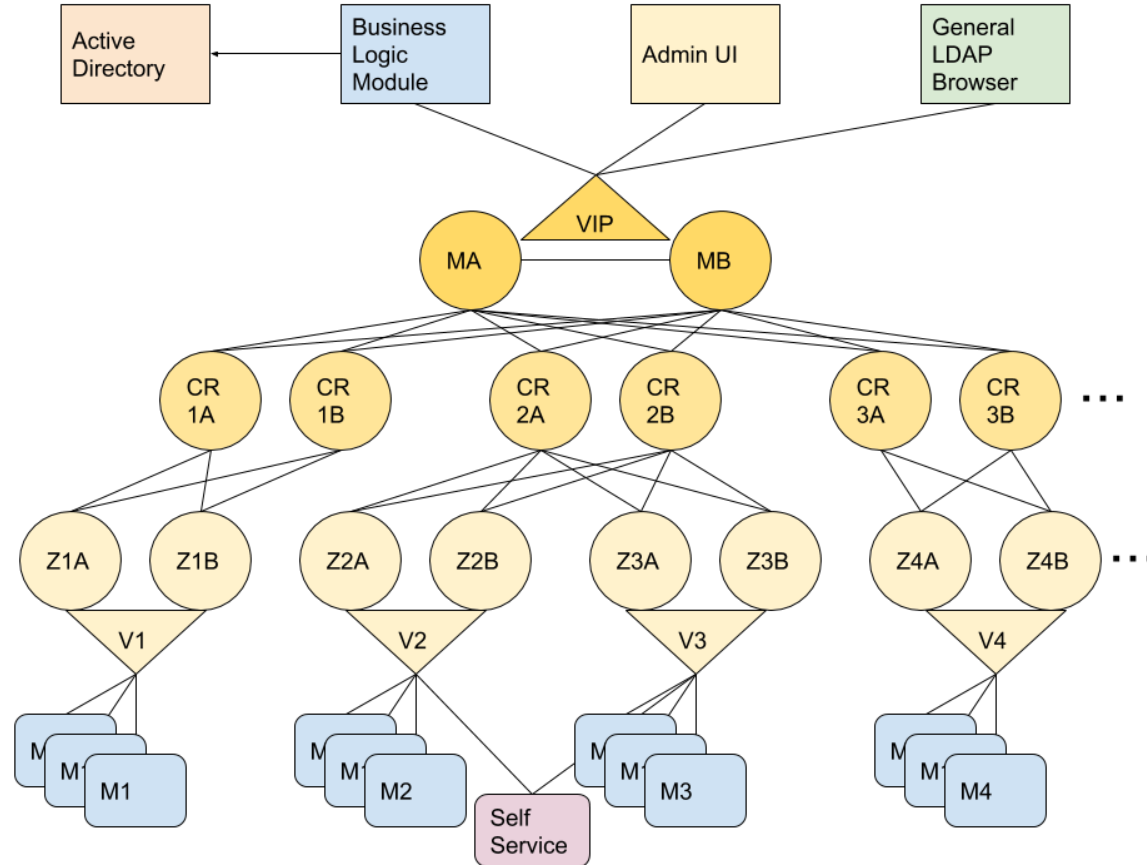
# User to Role to Machine

<---- Zone 1 ----> <----- Zone 2 -----> <---- Zone 3 ---->

User- Role- Machine	m1001	m1002	m1003	m2010	m2020	m2030	m3100	m3200	m3300
Huey	Admin	Admin	Admin						
Dewey				Auditor	Auditor	Auditor			
Louie							User	User	User



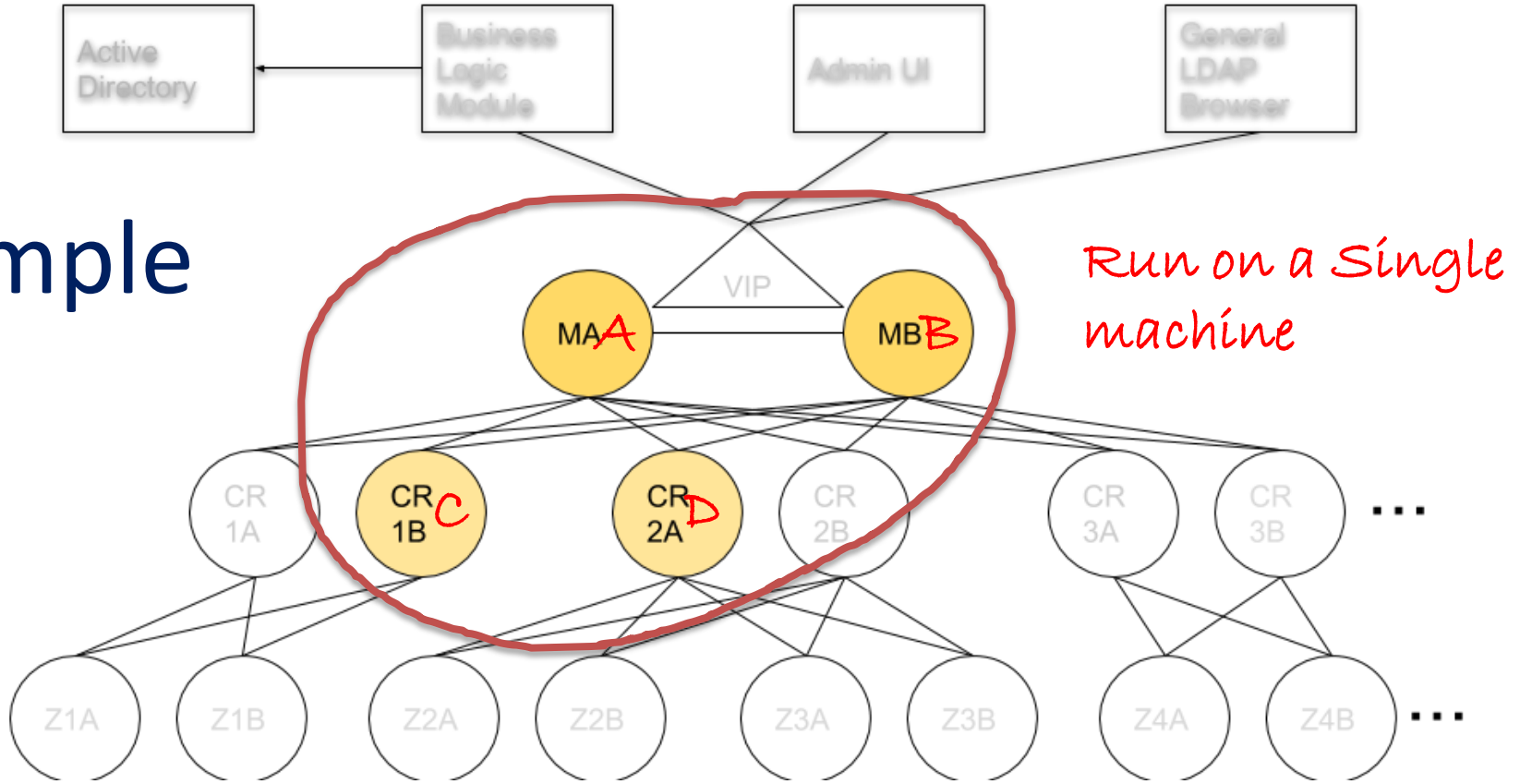
# Target Architecture



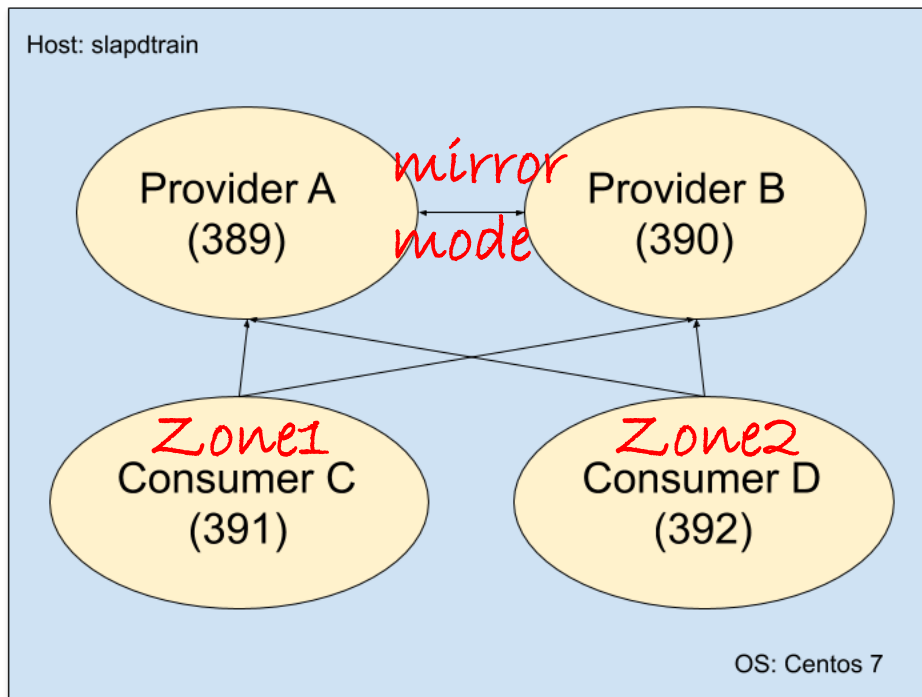
# Business Logic Module

- Lightweight IdM
- Single purpose mapping from \*any\* data stream to LDAPv3
- Data mappings
- Simplified customizations, like userid mappings, emails, etc.

# Sample



# Sample Architecture



# Details about the sample

- Two masters, MirrorMode, Delta Synchrepl
- Two read-only replicas, Delta Synchrepl with masters, selectively replicate based on filter.
- Access log and Synchprov overlays on master
- All running on a single machine.



# Mirror Mode

MirrorMode can be termed as Active-Active Hot-Standby, therefore an external server or device is needed to manage which provider is currently active.





# Sample Configuration

- Four instances of slapd running on a single machine.
- Usage of slapd.conf
  - slapd-a.conf (master)
  - slapd-b.conf (master)
  - slapd-c.conf (replica)
  - slapd.d.conf (replica)



# Master Config

- slapd-a.conf
- slapd-b.conf

# Security Options

# File contains CA certificates.

```
TLSCACertificateFile "/etc/openldap/certs/ca-cert.pem"
```

# TLSCertificateFile <filename>

# File contains the slapd server certificate.

```
TLSCertificateFile "/etc/openldap/certs/server-cert.pem"
```

# TLSCertificateKeyFile <filename>

# File contains the slapd server private key.

# Private key is not be protected with a password.

```
TLSCertificateKeyFile "/etc/openldap/certs/server-key.pem"
```

# More Security Options

```
TLSCipherSuite HIGH:-SSLv3
```

```
TLSVerifyClient try
```

```
# Only accept TLS 1.2
```

```
TLSProtocolMin 3.3
```

```
# Only accept TLS connections
```

```
#security tls=128
```

# Master Identity Mapping

# Id mapping:

authz-regexp

```
"email=student@ldap.org,cn=([^\n  
*),ou=Training,o=Symas,l=Sofia,s  
t=Sofia City,c=BG"
```

```
"cn=replicator,ou=admin,dc=examp  
le,dc=com"
```

# Master Modules

<code>modulepath</code>	<code>"/usr/lib64/openldap"</code>
<code>moduleload</code>	<code>accesslog.la</code>
<code>moduleload</code>	<code>back_monitor.la</code>
<code>moduleload</code>	<code>pw-sha2</code>
<code>moduleload</code>	<code>refint</code>
<code>moduleload</code>	<code>sssvlv</code>
<code>moduleload</code>	<code>syncprov.la</code>
<code>moduleload</code>	<code>ppolicy.la</code>



```
# ENABLE MIRROR MODE
mirrormode      TRUE
```

# Master Overlays

```
#-----
```

```
# OVERLAY [SYNCPROV]
```

```
#-----
```

```
overlay          syncprov
syncprov-checkpoint 100 10
syncprov-sessionlog 10000
syncprov-reloadhint TRUE
```

```
#-----
```

```
# OVERLAY [ACCESSLOG]
```

```
#-----
```

```
overlay          accesslog
logdb            cn=accesslog
logops           writes
logsuccess       TRUE
logpurge 24:00 01+00:00
```



# More Overlays

```
#-----  
# OVERLAY [REFINT]  
#-----  
overlay refint  
refint_attributes member owner
```

```
#-----  
# SSSVLV overlay  
#-----  
overlay sssvlv
```

```
#-----  
# ppolicy overlay  
#-----  
overlay ppolicy  
ppolicy_hash_cleartext
```





# Main DB

#####

# Main Database

# (lmdb-backed)

#####

database        mdb

suffix         "dc=example,dc=com"

rootdn         "cn=Manager,dc=example,dc=com"

# Cleartext passwords, especially for the  
rootdn, should be avoided. See  
slappasswd(8) and slapd.conf(5) for details  
describing encrypted passwords.

rootpw         secret

directory       "/var/lib/ldap/sample-a"



```
#####  
# Accesslog database  
#####
```

# Accesslog DB

```
database          mdb  
directory         /var/lib/ldap/accesslog-a  
maxsize          5120000  
suffix            "cn=accesslog"  
rootdn            "cn=accesslog"  
index             default eq  
index  
    objectClass,entryCSN,entryUUID,reqEnd,reqRe  
    sult,reqStart
```



# Replica Config

- slapd-c.conf
- slapd-d.conf

# Security Options

# File contains CA certificates.

```
TLSCACertificateFile "/etc/openldap/certs/ca-cert.pem"
```

# TLSCertificateFile <filename>

# File contains the slapd server certificate.

```
TLSCertificateFile "/etc/openldap/certs/server-cert.pem"
```

# TLSCertificateKeyFile <filename>

# File contains the slapd server private key.

# Private key is not be protected with a password.

```
TLSCertificateKeyFile "/etc/openldap/certs/server-key.pem"
```

# More Security Options

`TLSCipherSuite HIGH:-SSLv3`

`TLSVerifyClient try`

`# Only accept TLS 1.2`

`TLSProtocolMin 3.3`

`# Only accept TLS connections`

`#security tls=128`

# Replica Modules

```
modulepath
```

```
"/usr/lib64/openldap"
```

```
moduleload
```

```
back_monitor.la
```

```
moduleload
```

```
pw-sha2
```

```
moduleload
```

```
ppolicy.la
```

syncrepl

# Replication

```
rid=21
provider=ldap://slapdtrain:389
bindmethod=sasl
saslmech=external
starttls=yes
tls_cacert=/etc/openldap/certs/ca-cert.pem
tls_cert=/etc/openldap/certs/server-cert.pem
tls_key=/etc/openldap/certs/server-key.pem
tls_reqcert=demand
type=refreshAndPersist
searchbase="dc=example,dc=com"
filter="( | (sampleZone=Z1) ( ! (objectClass=samplePerson) ) ) "
scope=sub
schemachecking=on
retry="60 5 300 +"
network-timeout=30
keepalive=180:30:60
sizeLimit=unlimited
timelimit=unlimited
```



# What's missing from this example?

- Chain Overlay
- Referrals
- cn=config (dynamic config)

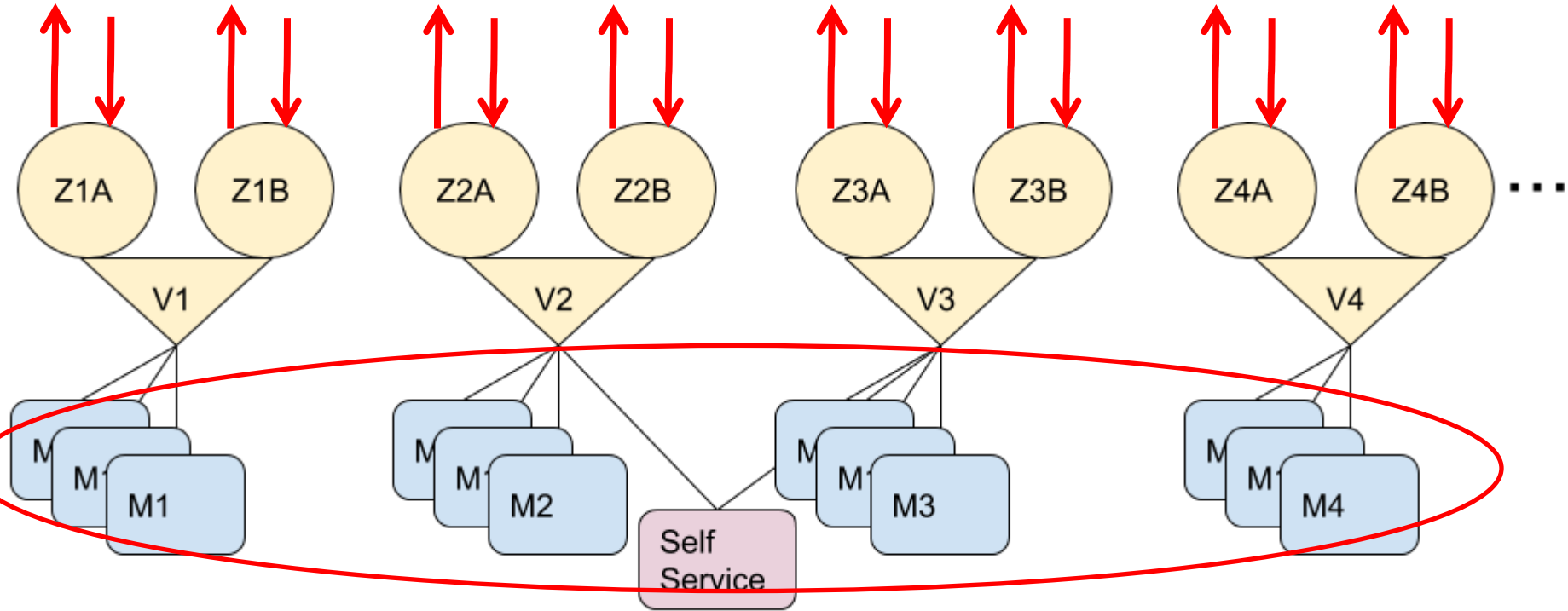
To enforce  
password policies  
on the read-only  
replicas

recommended  
approach





# Password Policy Replication



# More on Password Policies

- Password events occurring in the read-only replicas are referred upwards via referrals.
- Chaining Overlay handles the referrals on behalf of the client.
- Updates referred to the masters are then replicated back downward.

# Testing



# Zone Testing

User	Zone 1	Zone 2
Huey	True	True
Louie	False	True
Dewey	True	False

# Group Testing

User	sampleUsers *	sampleAdmins	sampleAuditors
Huey	True	True	False
Louie	True	False	True
Dewey	True	False	False

\* default posixGroup



# Sudo Role Testing

Groups	sudoUser	sudoAdmin	sudoAuditor
sampleAdmins	False	True	False
sampleAuditors	False	False	True
sampleUsers	True	False	False



Huey	Zone 1	Zone 2
Admin	True	True
Auditor	False	False
User	True	True

Louie	Zone 1	Zone 2
Admin	False	False
Auditor	False	True
User	False	True

Dewey	Zone 1	Zone 2
Admin	False	False
Auditor	False	False
User	True	False



# Sample Project Contents

- <https://github.com/shawnmckinney/openldap-mmr-tutorial>
  - README's for VirtualBox Setup
  - README's for Master and Replicas
  - Sample Config for OpenLDAP
  - Test Data and Certs
  - Script to check if replication in synch



# Sample Project

1. [README-virtualbox-centos7-install.md](#)
  - Only if not using supplied .ova images
2. [README-IMPORT-VM.md](#)
  - Enable supplied .ova into VMware or VirtualBox
3. [README-MASTERS.md](#)
  - Install OpenLDAP and configure masters A & B
4. [README-REPLICAS.md](#)
  - Configure replicas C & D



# Contact Info

Web:

<http://symas.com>

Email:

[mnormann@symas.com](mailto:mnormann@symas.com)

Email:

[smckinney@symas.com](mailto:smckinney@symas.com)

