Tivoli Directory Server – Schema, Access Control Lists, Password policies and Secure Socket layer

By: Shruti Maheshwari & Nilesh Patel

Introduction

Abstract

This STE will cover the security features such as Password Policy , SSL and Access Control Lists. Also, we will cover Web Admin Tool configuration.

Objectives

- Understand the security features.
- Understand how to configure security features in TDS.
- Web Admin Tool installation and configuration.

Agenda

- Before we begin
  - Important Links
  - Previous STE's
  - Planned STE's
- TDS Schema
  - What is TDS schema?
  - Object Classes and Attributes.
- Access Control Lists
  - Access Control Information
  - Non-filtered ACLs
  - Filtered ACLs

Agenda

- Password Policy
  - Types
  - Configuration- via command line and WAT
  - Common errors
- Web Admin Tool
  - Starting WebSphere Application server
  - Configuration
- Secure Socket Layer (SSL)
  - Security goals of SSL
  - Configuring server authentication
  - Configuring client authentication

Important Links

- ITDS v6.3 Package information: https://www304.ibm.com/support/docview.wss?rs=767&uid=swg24027373
- 6.3 System Requirements: http://publib.boulder.ibm.com/infocenter/tivihelp/v2r1/topic/com.ibm.IBMDS.doc/sysreq.htm
- 6.3 Product Documentation: http://publib.boulder.ibm.com/infocenter/tivihelp/v2r1/index.jsp?toc=/com.ibm.IBMDS.doc/toc.xml

Important Links

- Google Newsgroup: http://groups.google.com/group/ibm.software.ldap/topics?lnk=gschg&hl=en
- Support Site: http://www-306.ibm.com/software/sysmgmt/products/support/IBMDirectoryServer.html
- Tivoli Product Lifecycle Site: http://www-306.ibm.com/software/sysmgmt/products/support/lifecycle/
- Tivoli Software Global User Group Community: http://www.tivoli-ug.org/

STE Links

Previous STE’s

- Introduction to IBM Tivoli Directory Server:
  https://www-304.ibm.com/support/docview.wss?uid=swg27021610

STE Links

Upcoming STE's

- TDS-Back up and recovery: http://www-01.ibm.com/software/sysmgmt/products/support/TE/techex_V980536A95841W35.html
- TDS- Replication: http://www-01.ibm.com/software/sysmgmt/products/support/TE/techex_W517531B55309Q11.html
- TDS – Proxy, Performance tuning and Troubleshooting: http://www-01.ibm.com/software/sysmgmt/products/support/TE/techex_X900328J53343I07.html
- TDS Best practices ^%&*%&^%*&%&^

What is TDS Schema?

- A schema is a set of rules that governs the way that data can be stored.
- Data is stored in the directory using directory entries (LDAP information model)
- A entry consists of an :
  - objectclass
  - attributes
- Example entry in TDS :

```
cn=Mark Anthony,o=IBM,c=US
objectclass=person
objectclass=top
cn=Mark Anthony
sn=Anthony
userpassword:passw@rd123
```

TDS Schema

- In TDS 6.x , schema files are by default located at

  <InstanceLocation>/idsslapd-<instancename>/etc/

- Schema files containing objectclasses

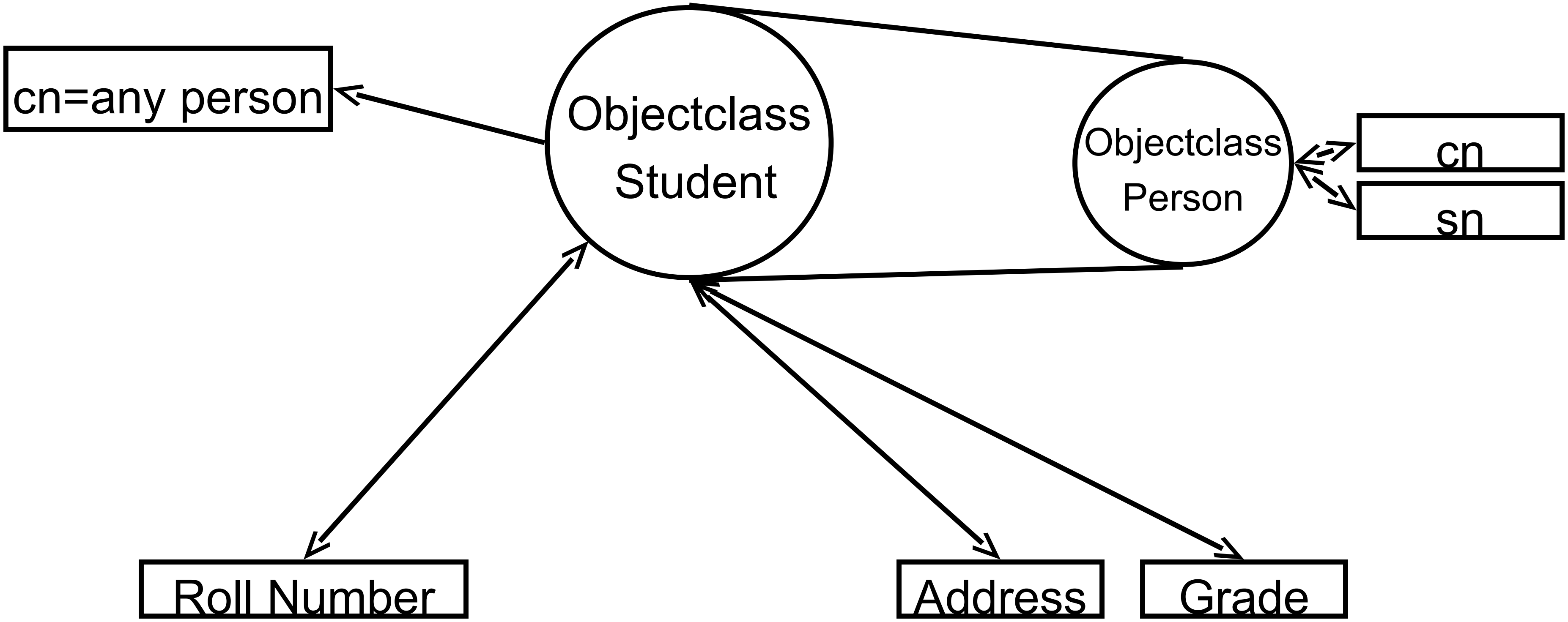
  V3.config.oc
  V3.ibm.oc
  V3.system.oc
  V3.user.oc

- Schema files containing attributes

  V3.config.at
  V3.ibm.at
  V3.system.at
  V3.user.at

- User defined schema is present in V3.modifiedschema

Picture View of Objectclass and attributes



cn=any person

Objectclass Student

Objectclass Person

cn

sn

Roll Number

Address

Grade

Object Class

- Types of Objectclass : Structural, Abstract and Auxiliary.
- Object Class Inheritance.
- Objectclass has must and optional attributes.
- Example Objectclass :

```
(
2.5.6.6
NAME 'person‘
DESC 'Defines entries that generically represent people.‘
SUP 'top‘
STRUCTURAL
MUST ( cn $ sn )
MAY ( userPassword $ telephoneNumber $ seeAlso $ description )
)
```

Attributes

- Each directory entry has a set of attributes associated with it through it’s object class.
- Actual data is contained in attribute.
- Example of attribute :

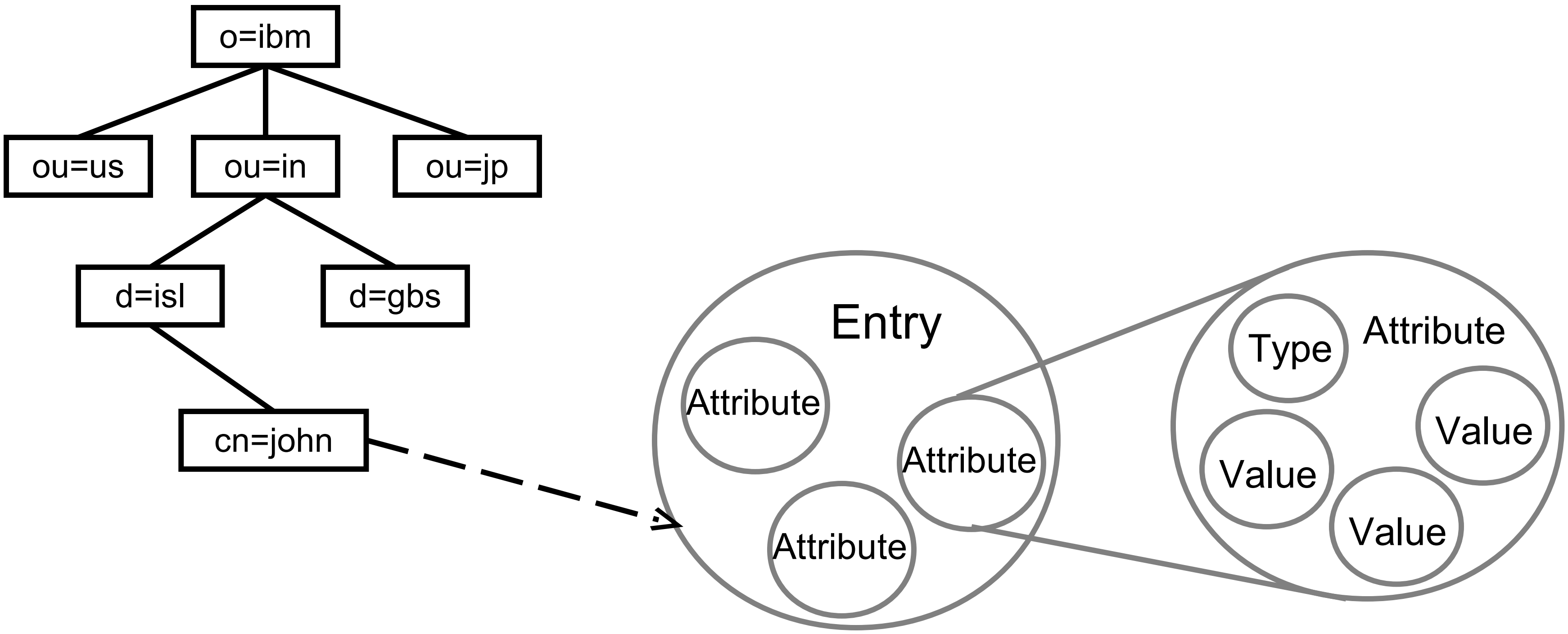
Attributetypes:

```
( 2.5.4.35
  NAME 'userPassword'
  DESC 'Holds a password value for a distinguished name.'
  EQUALITY 2.5.13.17
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.5)
```

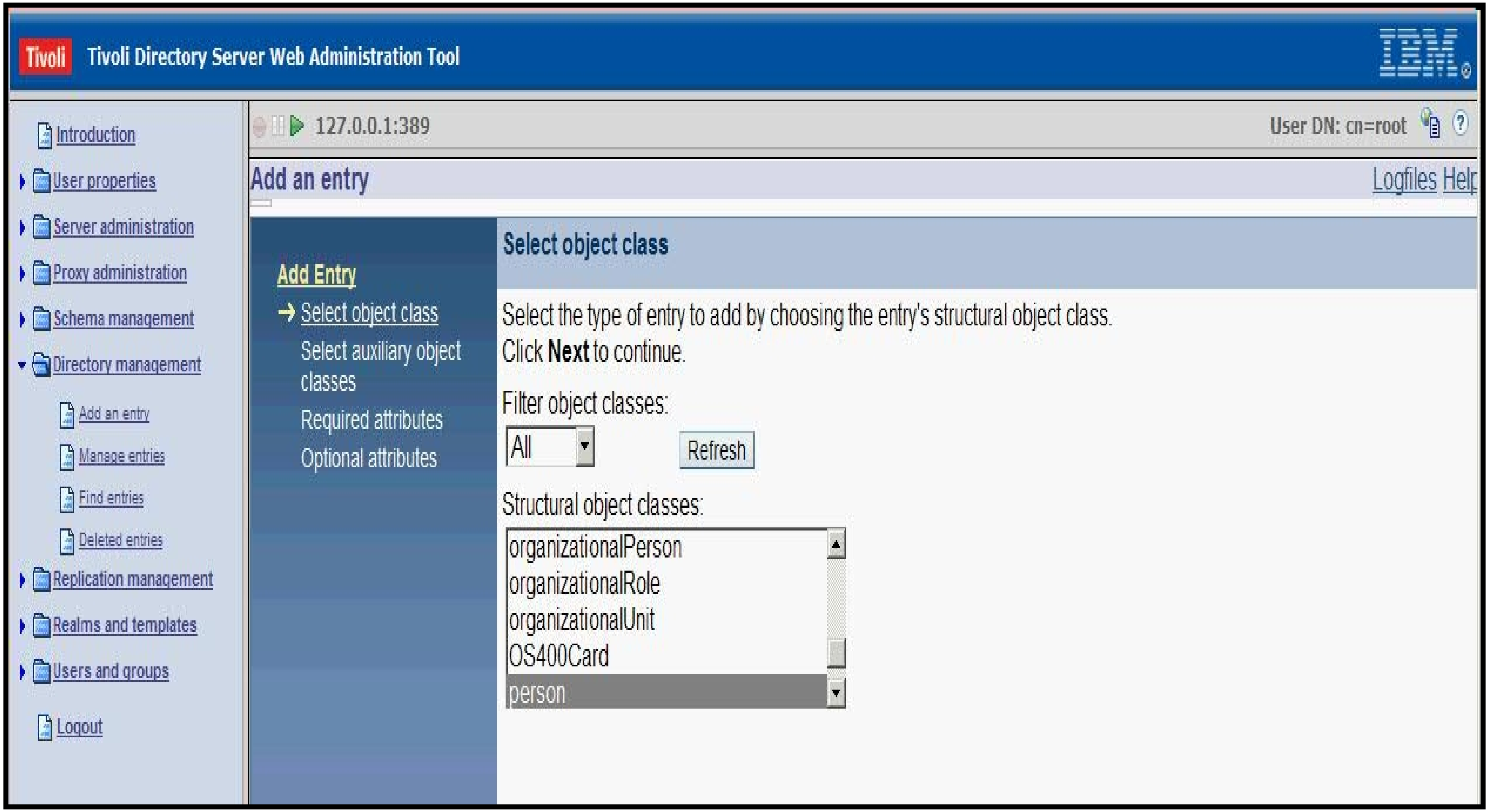
IBMAttributetypes :

```
( 2.5.4.35
 DBNAME ( 'userPassword' 'userPassword' )
 ACCESS-CLASS CRITICAL )
```

Directory Structure



Select Object Class



Select object class

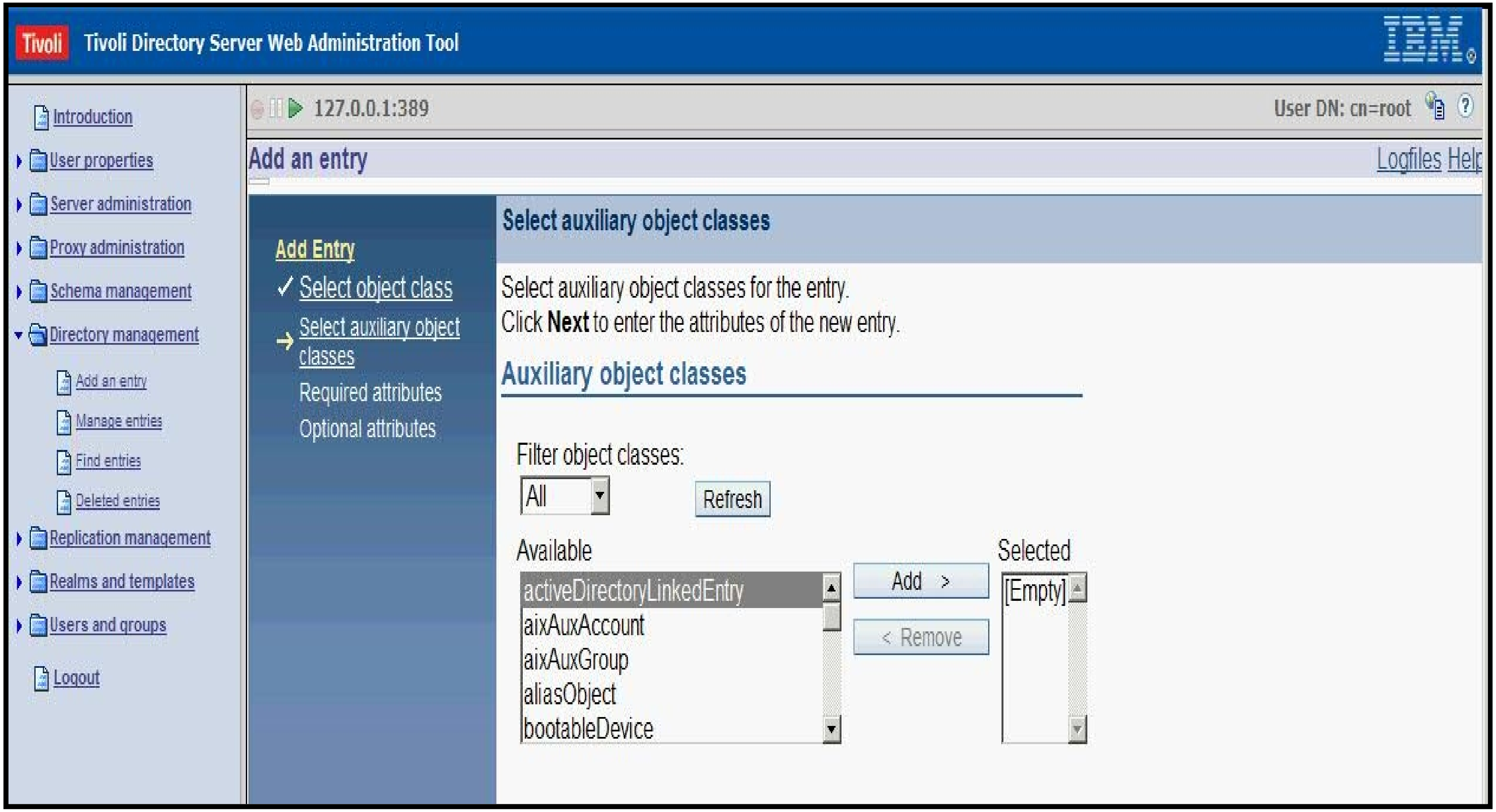
Select the type of entry to add by choosing the entry's structural object class. Click **Next** to continue.

Filter object classes: All

Structural object classes:

organizationalPerson
organizationalRole
organizationalUnit
OS400Card
person

Select Auxiliary Object Class



Add an entry

Select auxiliary object classes

Select auxiliary object classes for the entry.
Click **Next** to enter the attributes of the new entry.

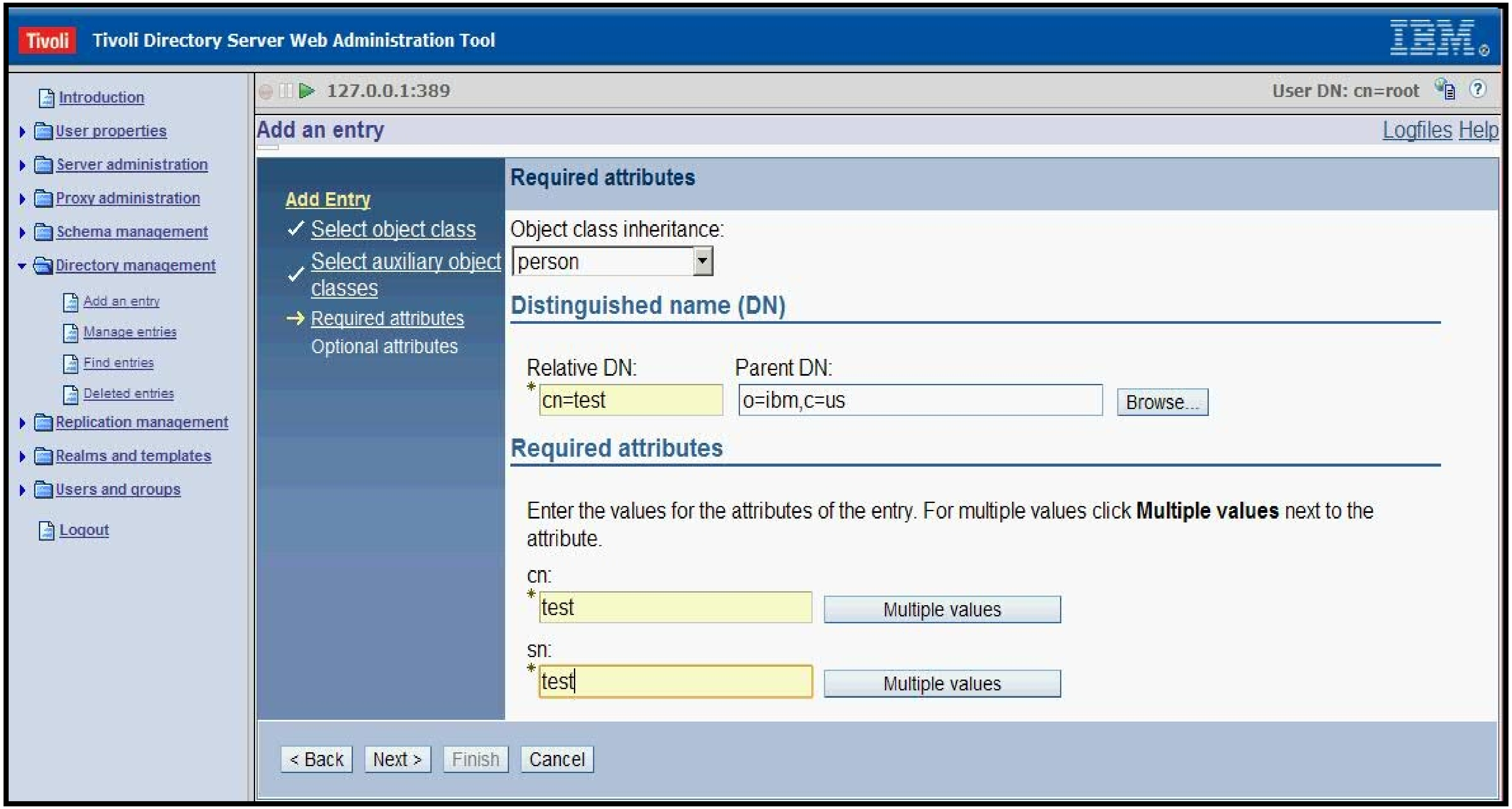
Auxiliary object classes

Filter object classes: All

Available: activeDirectoryLinkedEntry, aixAuxAccount, aixAuxGroup, aliasObject, bootableDevice

Selected: [Empty]

Enter the attributes



Add an entry

Required attributes

Object class inheritance: person

Distinguished name (DN)

Relative DN: * cn=test

Parent DN: o=ibm,c=us

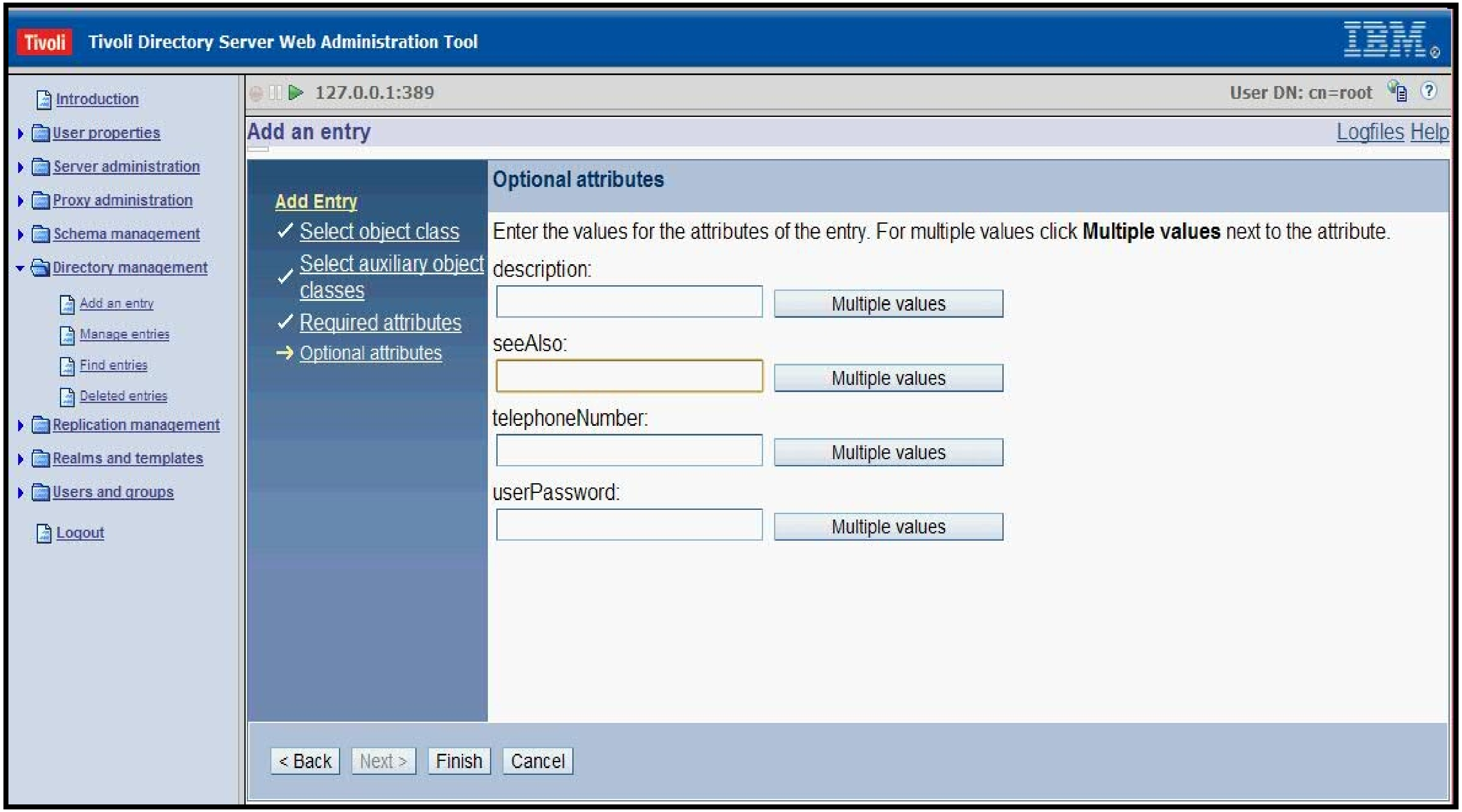
Required attributes

Enter the values for the attributes of the entry. For multiple values click **Multiple values** next to the attribute.

cn: * test

sn: * test

Optional Attributes



Optional attributes

Enter the values for the attributes of the entry. For multiple values click **Multiple values** next to the attribute.

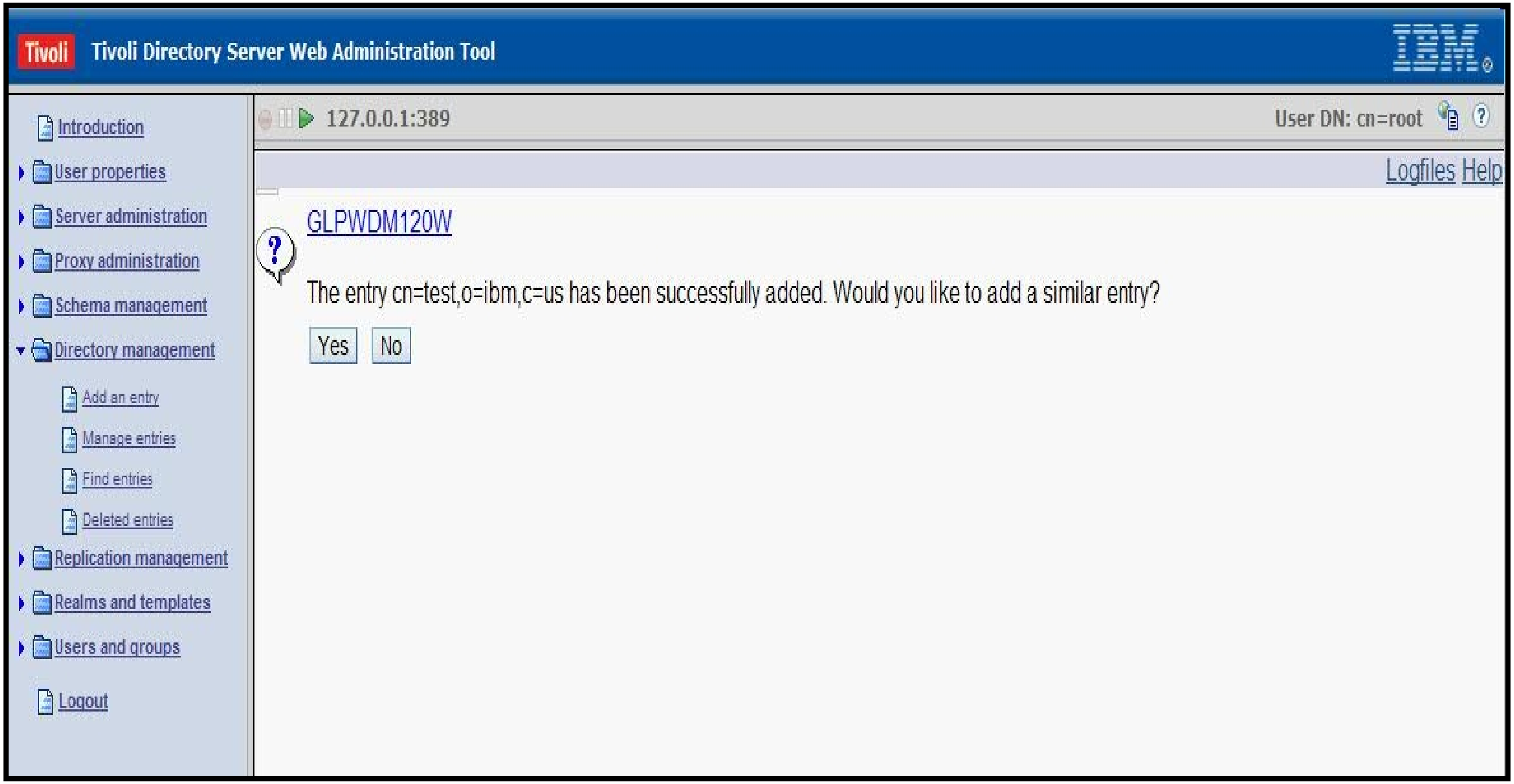
description: [ ] Multiple values

seeAlso: [ ] Multiple values

telephoneNumber: [ ] Multiple values

userPassword: [ ] Multiple values

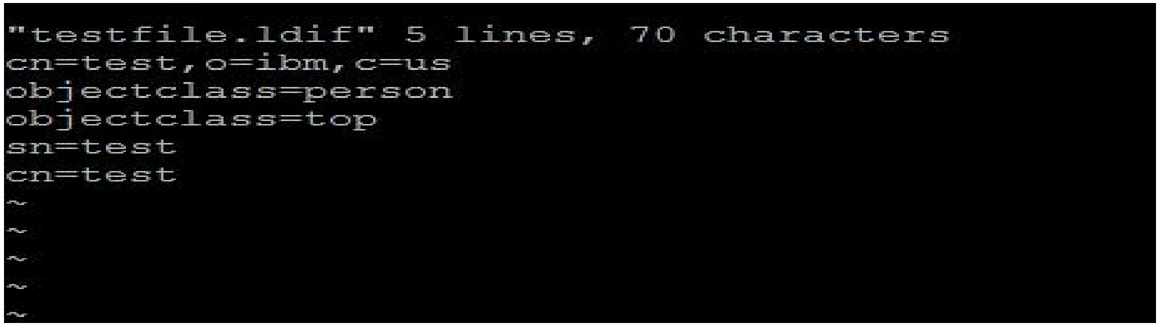
Entry Added



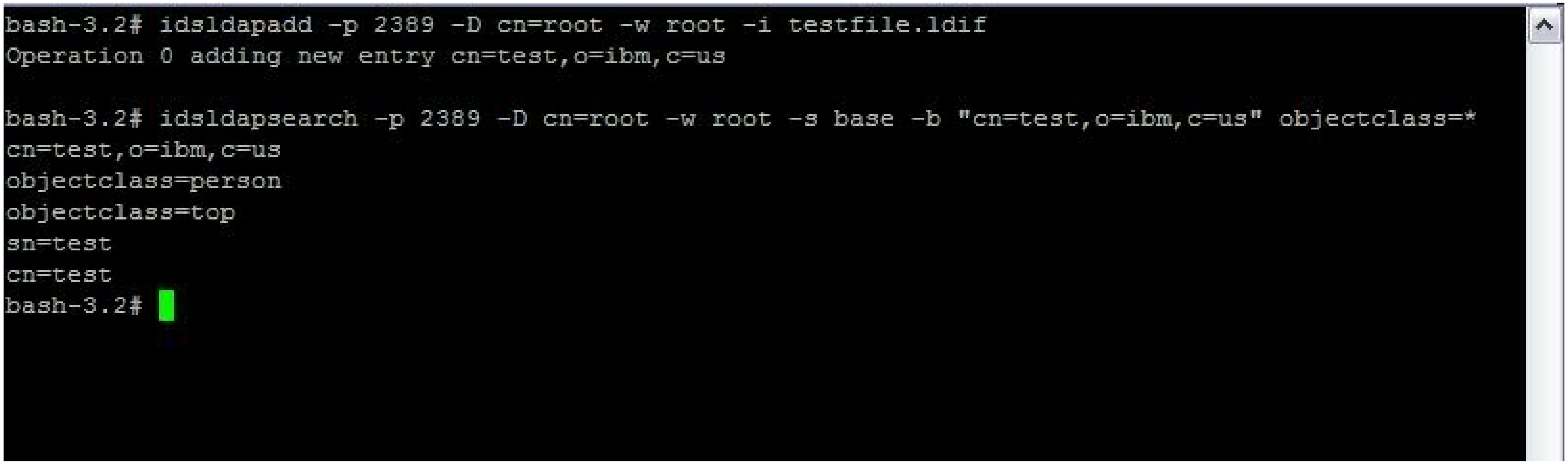
GLPWDM120W

The entry cn=test,o=ibm,c=us has been successfully added. Would you like to add a similar entry?

Adding an entry via command-line

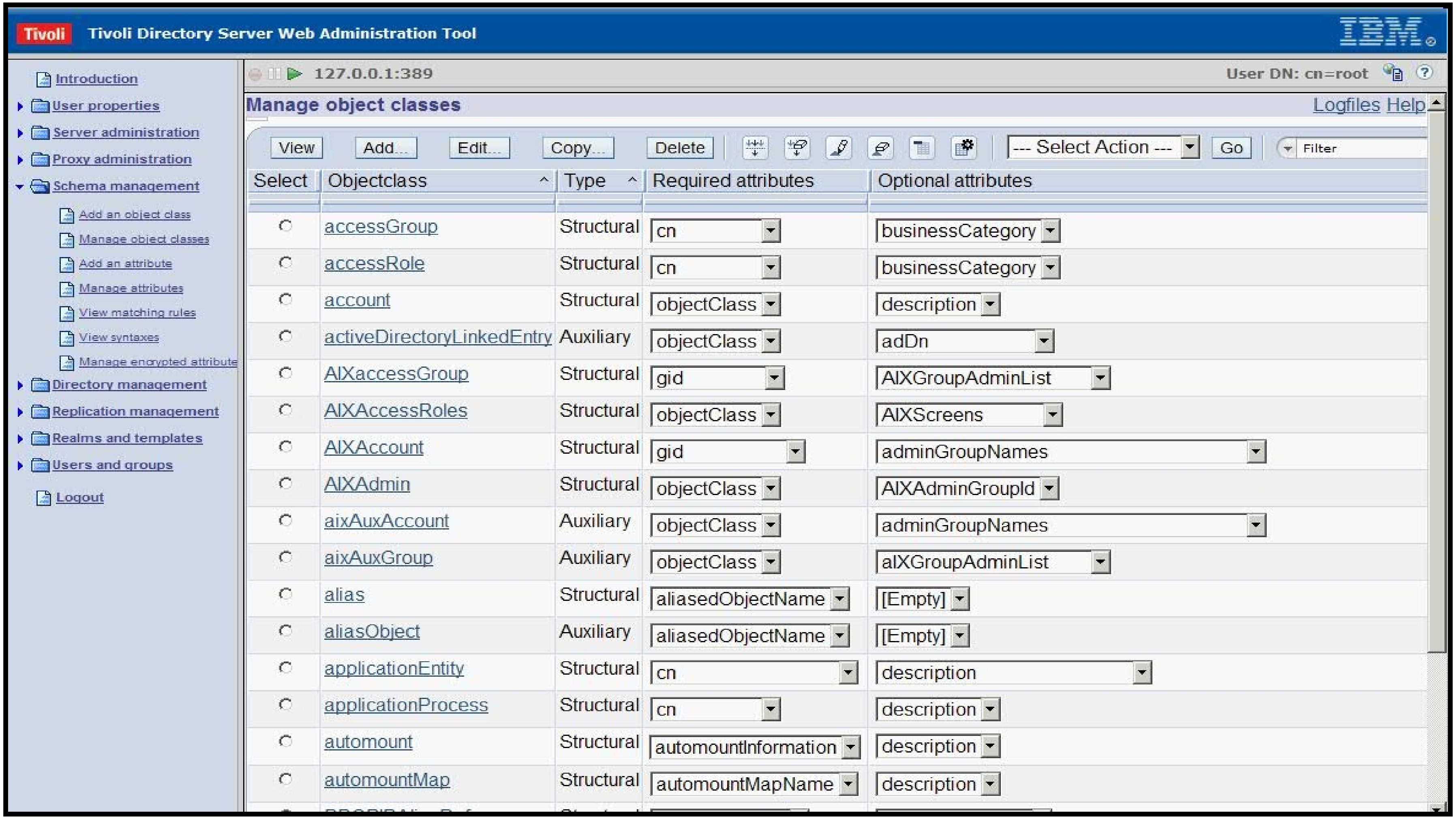


```
"testfile.ldif" 5 lines, 70 characters
cn=test,o=ibm,c=us
objectclass=person
objectclass=top
sn=test
cn=test
~
~
~
~
~
```



```
bash-3.2# idsldapadd -p 2389 -D cn=root -w root -i testfile.ldif
Operation 0 adding new entry cn=test,o=ibm,c=us

bash-3.2# idsldapsearch -p 2389 -D cn=root -w root -s base -b "cn=test,o=ibm,c=us" objectclass=*
cn=test,o=ibm,c=us
objectclass=person
objectclass=top
sn=test
cn=test
bash-3.2#
```

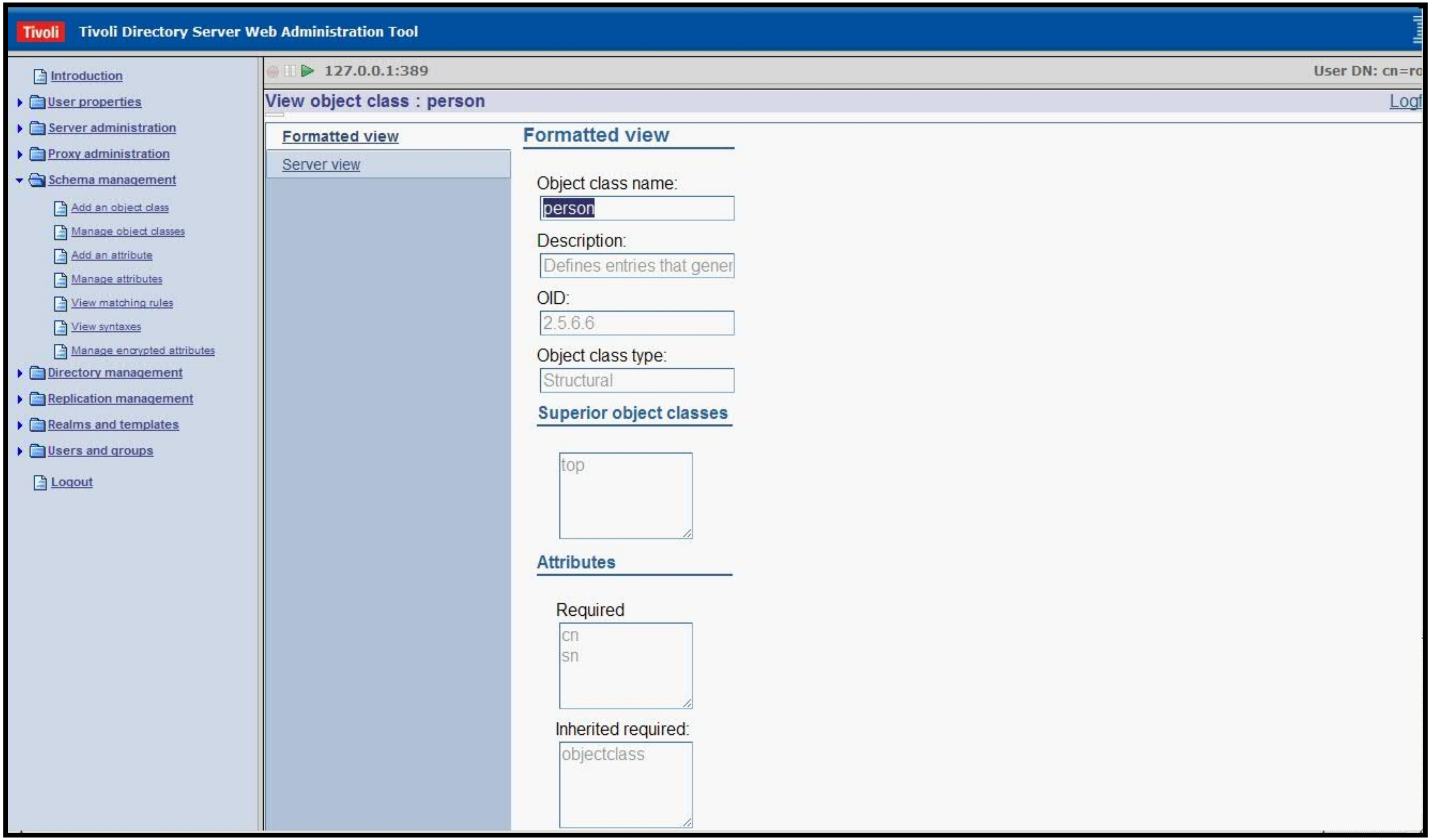
Schema Management



Manage object classes

Select	Objectclass	Type	Required attributes	Optional attributes
○	accessGroup	Structural	cn	businessCategory
○	accessRole	Structural	cn	businessCategory
○	account	Structural	objectClass	description
○	activeDirectoryLinkedEntry	Auxiliary	objectClass	adDn
○	AIXaccessGroup	Structural	gid	AIXGroupAdminList
○	AIXAccessRoles	Structural	objectClass	AIXScreens
○	AIXAccount	Structural	gid	adminGroupNames
○	AIXAdmin	Structural	objectClass	AIXAdminGroupId
○	aixAuxAccount	Auxiliary	objectClass	adminGroupNames
○	aixAuxGroup	Auxiliary	objectClass	aIXGroupAdminList
○	alias	Structural	aliasedObjectName	[Empty]
○	aliasObject	Auxiliary	aliasedObjectName	[Empty]
○	applicationEntity	Structural	cn	description
○	applicationProcess	Structural	cn	description
○	automount	Structural	automountInformation	description
○	automountMap	Structural	automountMapName	description

Schema Management (Continued)



View object class : person

Formatted view

Object class name: person

Description: Defines entries that gener

OID: 2.5.6.6

Object class type: Structural

Superior object classes

top

Attributes

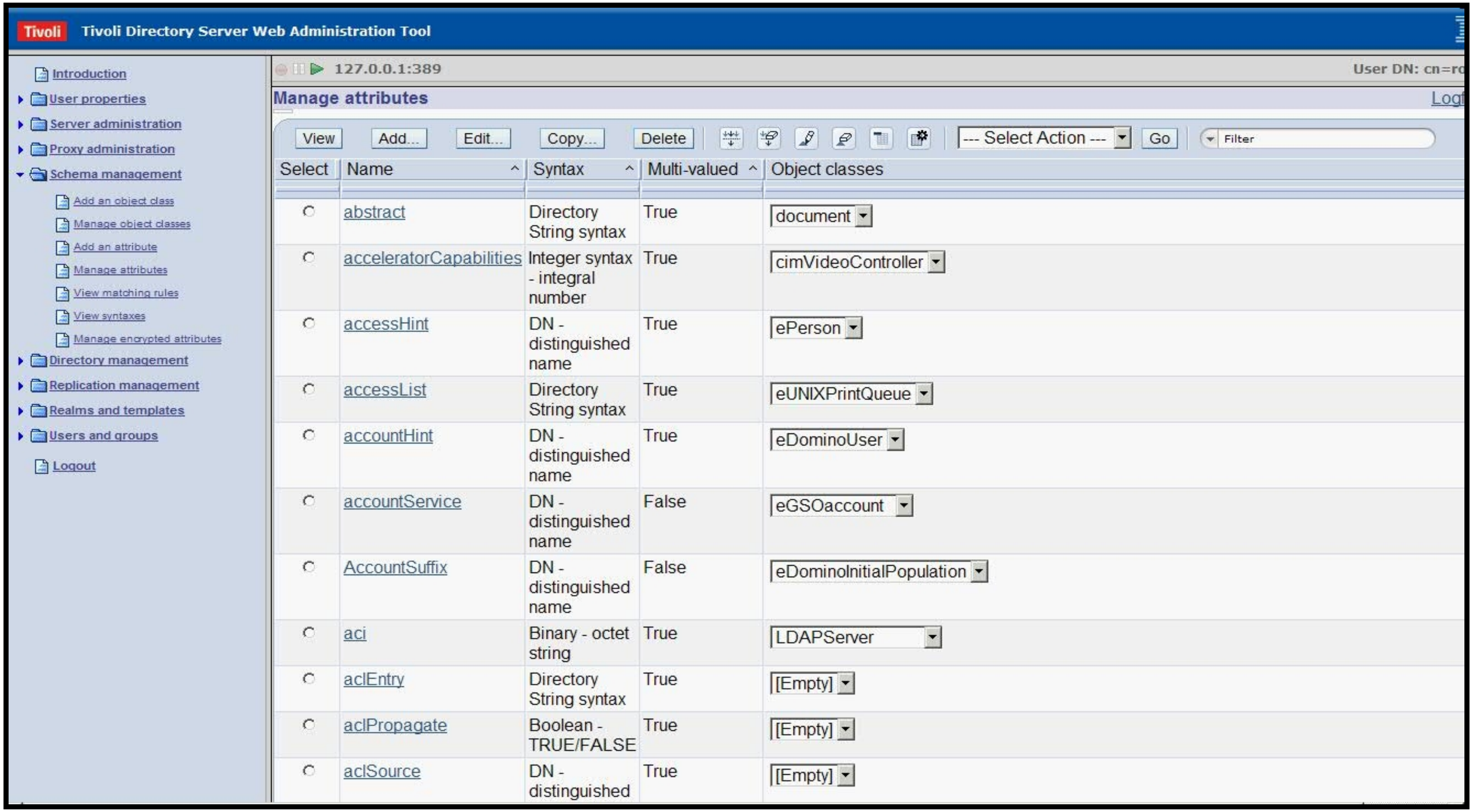
Required

cn
sn

Inherited required:

objectclass

Schema Management (Continued)



Tivoli Tivoli Directory Server Web Administration Tool

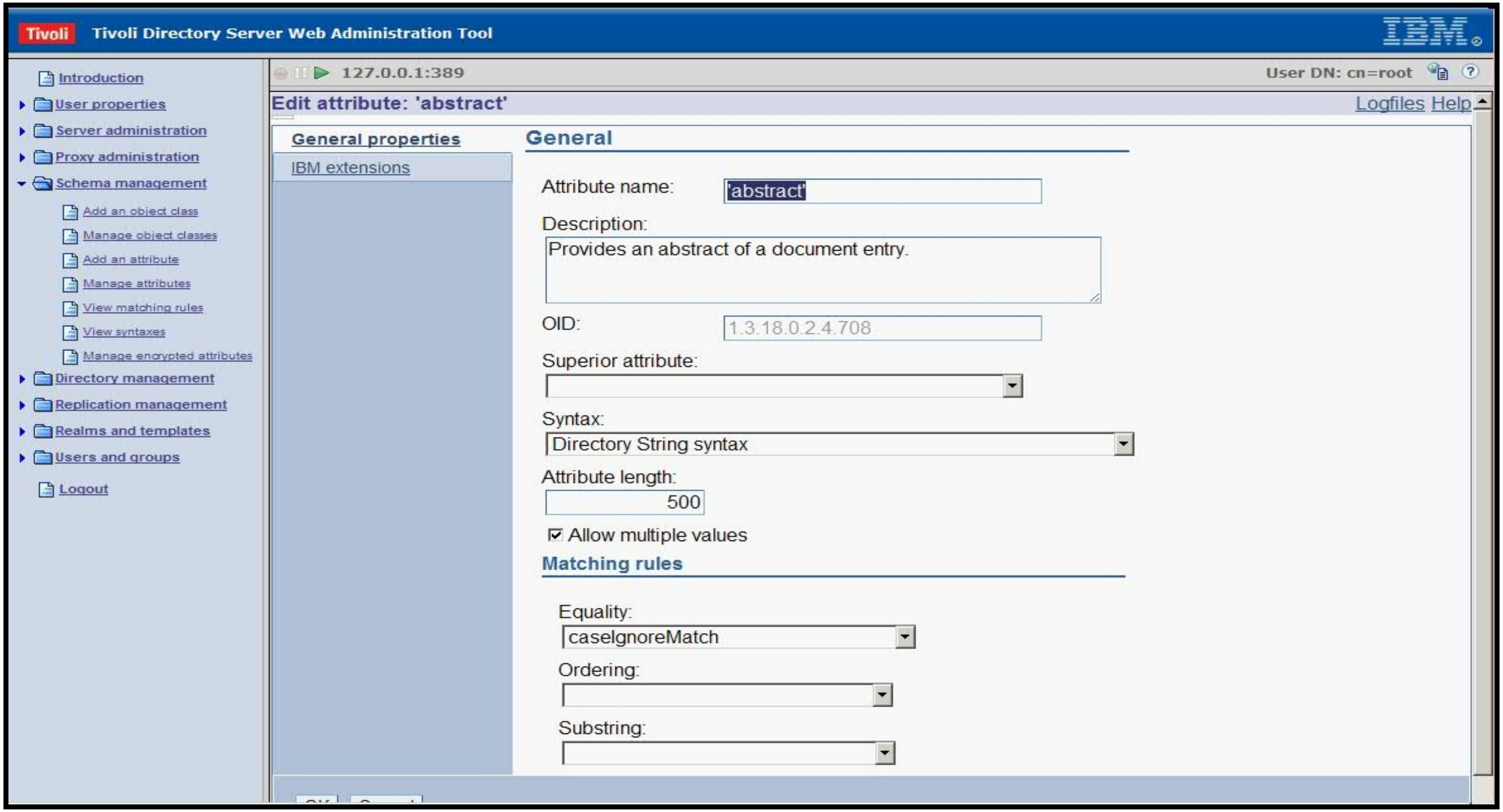
- Introduction
- User properties
- Server administration
- Proxy administration
- Schema management
  - Add an object class
  - Manage object classes
  - Add an attribute
  - Manage attributes
  - View matching rules
  - View syntaxes
  - Manage encrypted attributes
- Directory management
- Replication management
- Realms and templates
- Users and groups
- Logout

127.0.0.1:389 User DN: cn=ro

Manage attributes

Select	Name	Syntax	Multi-valued	Object classes
○	abstract	Directory String syntax	True	document
○	acceleratorCapabilities	Integer syntax - integral number	True	cimVideoController
○	accessHint	DN - distinguished name	True	ePerson
○	accessList	Directory String syntax	True	eUNIXPrintQueue
○	accountHint	DN - distinguished name	True	eDominoUser
○	accountService	DN - distinguished name	False	eGSOaccount
○	AccountSuffix	DN - distinguished name	False	eDominoInitialPopulation
○	aci	Binary - octet string	True	LDAPServer
○	aclEntry	Directory String syntax	True	[Empty]
○	aclPropagate	Boolean - TRUE/FALSE	True	[Empty]
○	aclSource	DN - distinguished	True	[Empty]

Schema Management (Continued)



Edit attribute: 'abstract'

General

Attribute name: 'abstract'

Description: Provides an abstract of a document entry.

OID: 1.3.18.0.2.4.708

Superior attribute:

Syntax: Directory String syntax

Attribute length: 500

☑ Allow multiple values

Matching rules

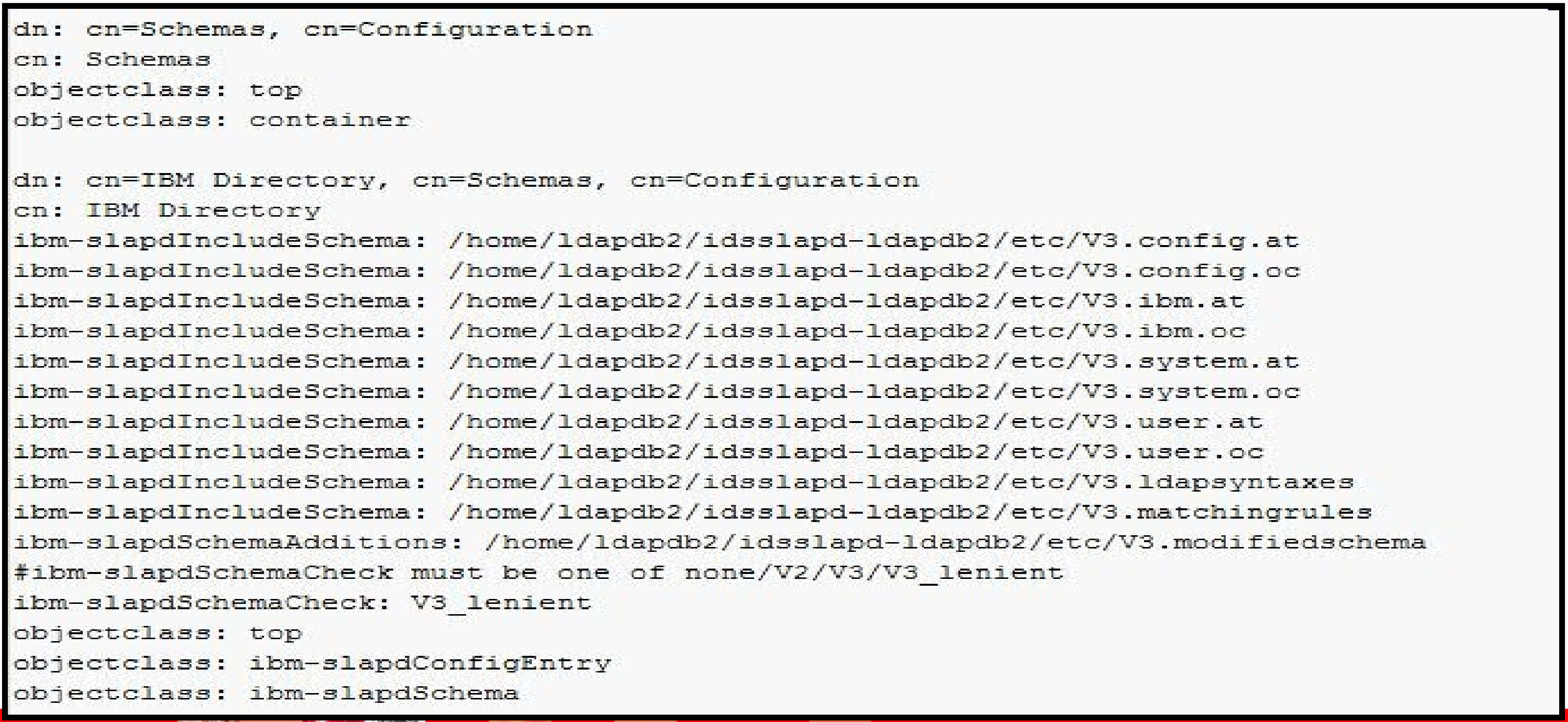
Equality: caseIgnoreMatch

Ordering:

Substring:

Schema Implementation

- Schema definitions are stored in files.
- Configuration file (**ibmslapd.conf**) lists schema files.
- LDAP clients can access the directory schema by performing a search of all objects under the **cn=schema** suffix



```
dn: cn=Schemas, cn=Configuration
cn: Schemas
objectclass: top
objectclass: container

dn: cn=IBM Directory, cn=Schemas, cn=Configuration
cn: IBM Directory
ibm-slapdIncludeSchema: /home/ldapdb2/idsslapd-ldapdb2/etc/V3.config.at
ibm-slapdIncludeSchema: /home/ldapdb2/idsslapd-ldapdb2/etc/V3.config.oc
ibm-slapdIncludeSchema: /home/ldapdb2/idsslapd-ldapdb2/etc/V3.ibm.at
ibm-slapdIncludeSchema: /home/ldapdb2/idsslapd-ldapdb2/etc/V3.ibm.oc
ibm-slapdIncludeSchema: /home/ldapdb2/idsslapd-ldapdb2/etc/V3.system.at
ibm-slapdIncludeSchema: /home/ldapdb2/idsslapd-ldapdb2/etc/V3.system.oc
ibm-slapdIncludeSchema: /home/ldapdb2/idsslapd-ldapdb2/etc/V3.user.at
ibm-slapdIncludeSchema: /home/ldapdb2/idsslapd-ldapdb2/etc/V3.user.oc
ibm-slapdIncludeSchema: /home/ldapdb2/idsslapd-ldapdb2/etc/V3.ldapsyntaxes
ibm-slapdIncludeSchema: /home/ldapdb2/idsslapd-ldapdb2/etc/V3.matchingrules
ibm-slapdSchemaAdditions: /home/ldapdb2/idsslapd-ldapdb2/etc/V3.modifiedschema
#ibm-slapdSchemaCheck must be one of none/V2/V3/V3_lenient
ibm-slapdSchemaCheck: V3_lenient
objectclass: top
objectclass: ibm-slapdConfigEntry
objectclass: ibm-slapdSchema
```

Access Control Lists

What does Authorization mean?

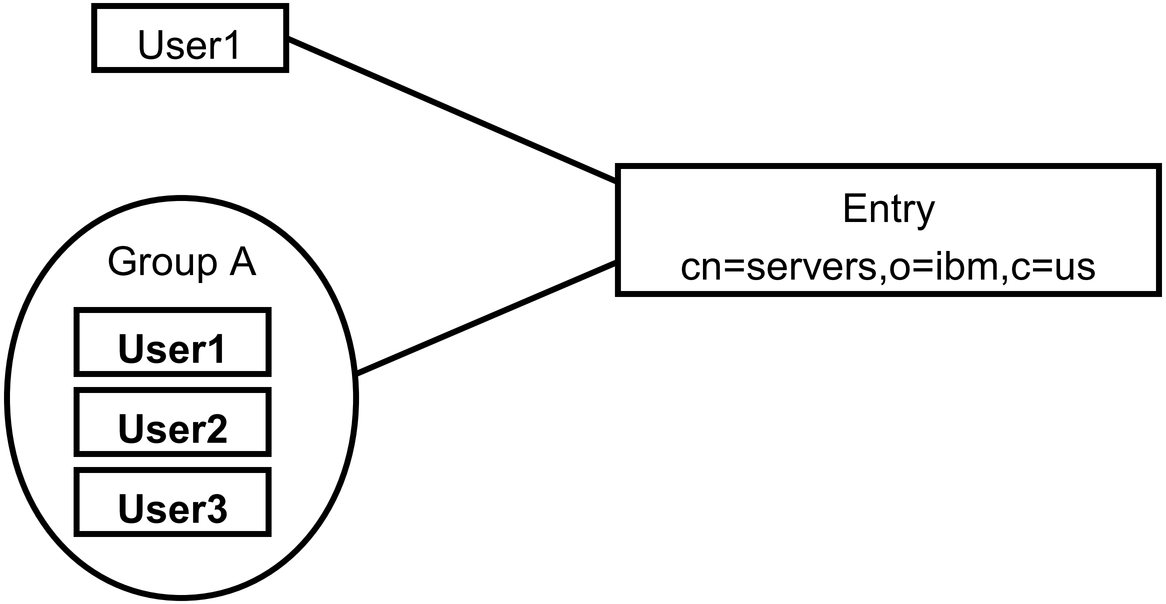
- Authorization is the concept of allowing access to resources only to those permitted to use them.
- Authorization is a process that protects computer resources by only allowing those resources to be used by resource consumers that have been granted authority to use them.
- Authorization is finding out if the person, once identified, is permitted to have the resource. This is usually determined by finding out if that person is a part of a particular group, if that person has paid admission, or has a particular level of security clearance.

In terms of Directories

- A feature to protect information stored in LDAP directory.
- Access control lists are means of controlling or restricting users from accessing different parts of the directory.
- Access control lists provide a means to protect information stored in a LDAP directory.
- Administrators use ACLs to restrict access to different portions of the directory, or specific directory entries.

How can we provide access to an entry?

- Administrators can define access for,
  1. Specific User
  2. Group



Access Control Model

The access control model defines two sets of attributes:

- The entryOwner information.
  - Controls which subjects can define the ACIs
- The Access Control Information (ACI)
  - Defines a subject's permission

EntryOwner Information

- The entryOwner information controls which subjects can define the ACIs.
- An entry Owner has full access rights to the target object.
- Attributes those define entry ownership :

  entryOwner - Explicitly defines an entry owner.

  ownerPropagate - Specifies whether the permission set is propagated to the subtree descendant entries.
- The entry owners have complete permissions to perform any operation on the object regardless of the aclEntry.
- The entry owners are the only ones who are permitted to administer the aclEntries for that object.

Access Control Information

- The ACI defines a subject’s permission to perform a given operation against certain LDAP objects.
- Ways to define ACI’s
  - Non-filtered ACLs
  - Filtered ACLs

Non-filtered ACLs

- It may be propagated to none or all of its descendant entries.
- Non-filtered ACL applies explicitly to the directory entry that contains them.
- The default behavior of the non-filtered ACL is to propagate.
- Attributes that define non-filtered ACLs are:

  aclEntry - Defines a permission set.

  aclPropagate - Specifies whether the permission set is propagated to the subtree descendant entries.

- Non-filtered ACLs inherently propagate to any comparison matched objects in the associated subtree and *aclPropagate* attribute is used to stop propagation of non-filter ACLs.

Filtered ACLs

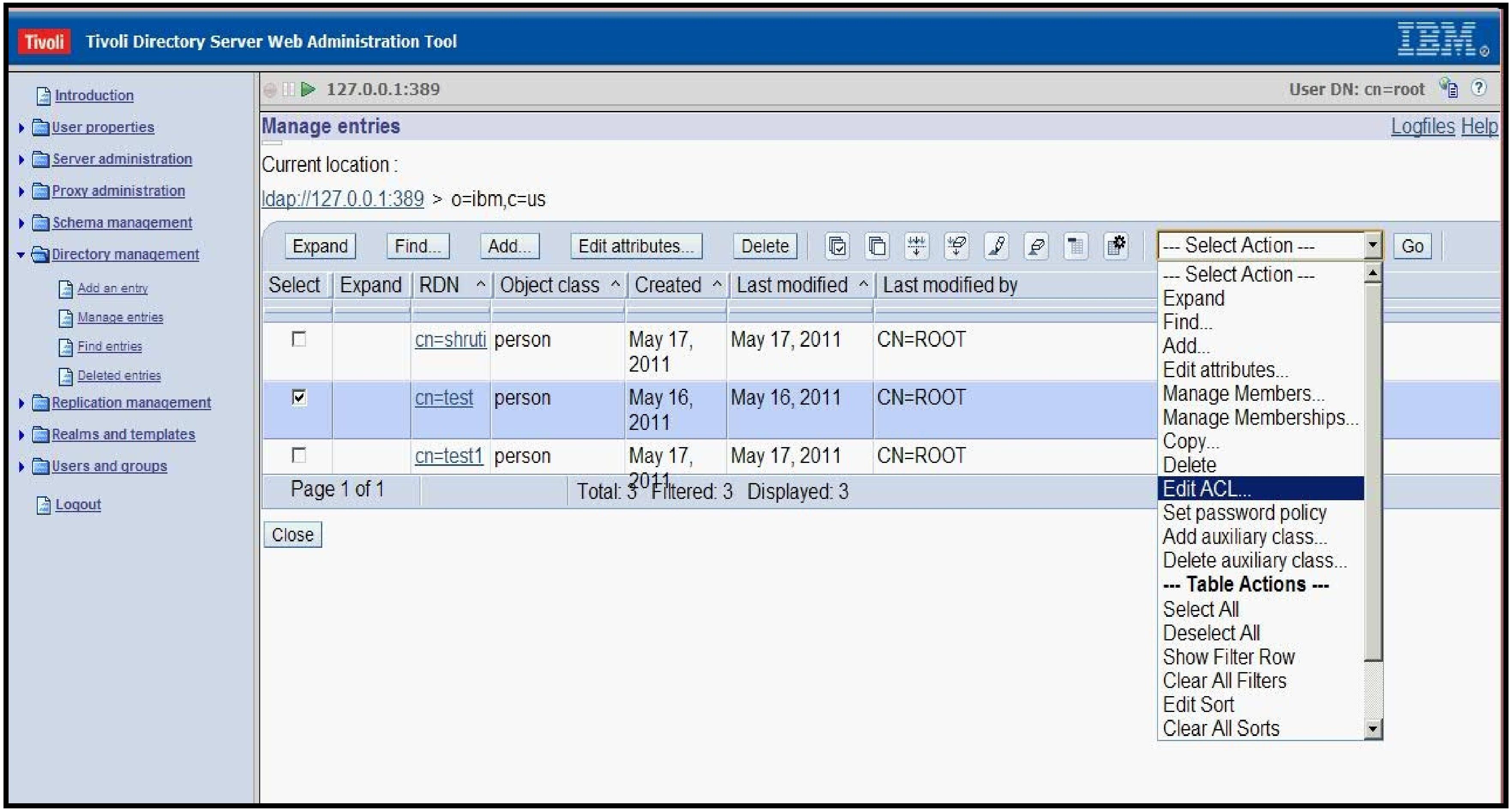
- Filter based ACLs use a specified object filter to select the directory entries to which they apply.
- The default behavior of filter-based ACLs to accumulate from the lowest containing entry, upward along the ancestor entry chain, to the highest containing entry in the DIT.
- The effective access is calculated as the union of the access rights granted, or denied, by the constituent ancestor entries.
- Attributes that define non-filtered ACLs are:

ibm-filterAclEntry –
- Same format as aclEntry, with the addition of an object filter component.

ibm-filterAclInheritThe –
- Associated ceiling attribute.
-By default it is set to true. When set to false, it terminates the accumulation.

Working with ACLs

Edit ACL entry



Current location :

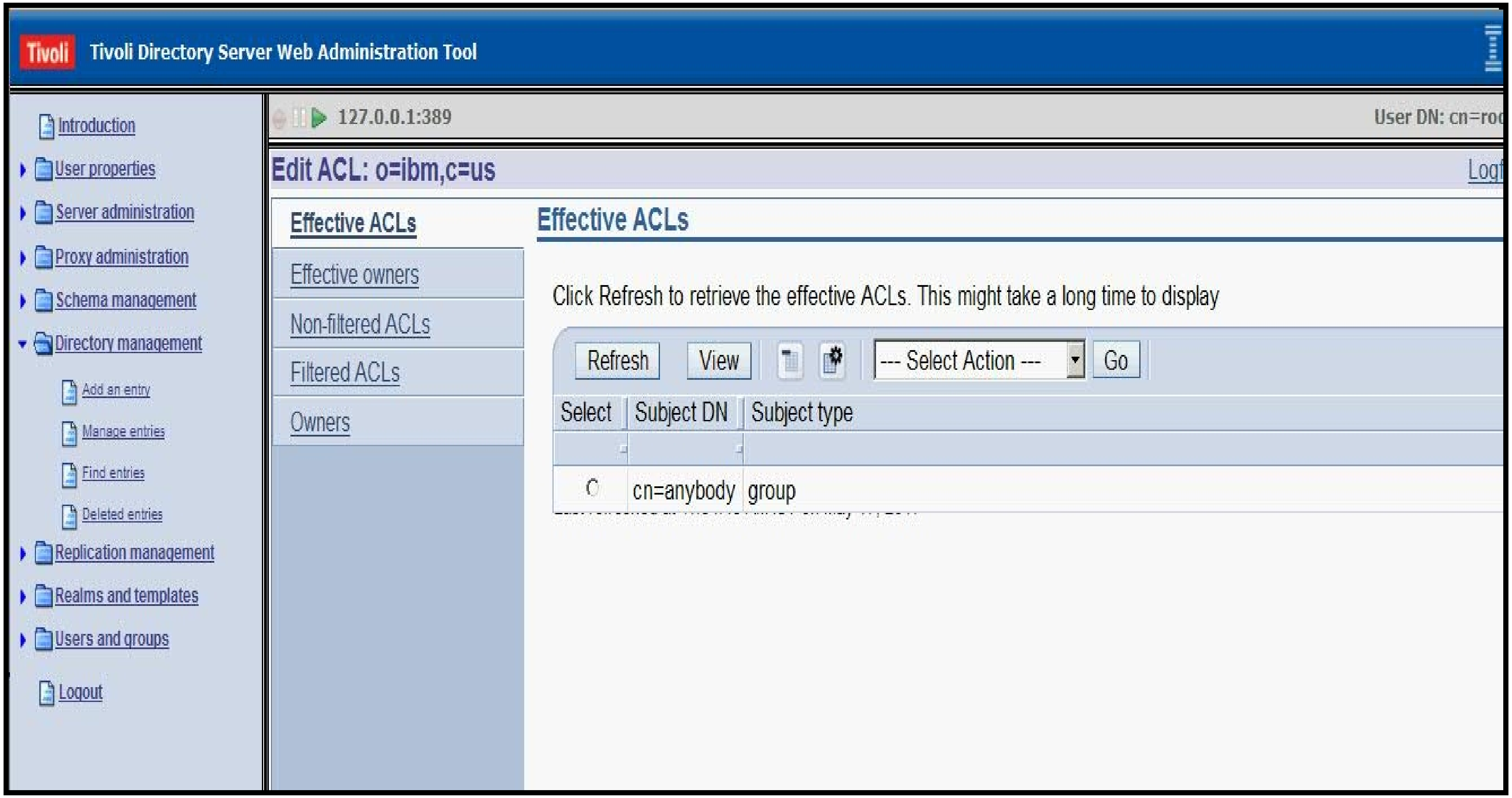
ldap://127.0.0.1:389 > o=ibm,c=us

Select	Expand	RDN	Object class	Created	Last modified	Last modified by
☐		cn=shruti	person	May 17, 2011	May 17, 2011	CN=ROOT
☑		cn=test	person	May 16, 2011	May 16, 2011	CN=ROOT
☐		cn=test1	person	May 17, 2011	May 17, 2011	CN=ROOT

Page 1 of 1 Total: 3 Filtered: 3 Displayed: 3

Working with ACLs (Continued)

Effective ACLs

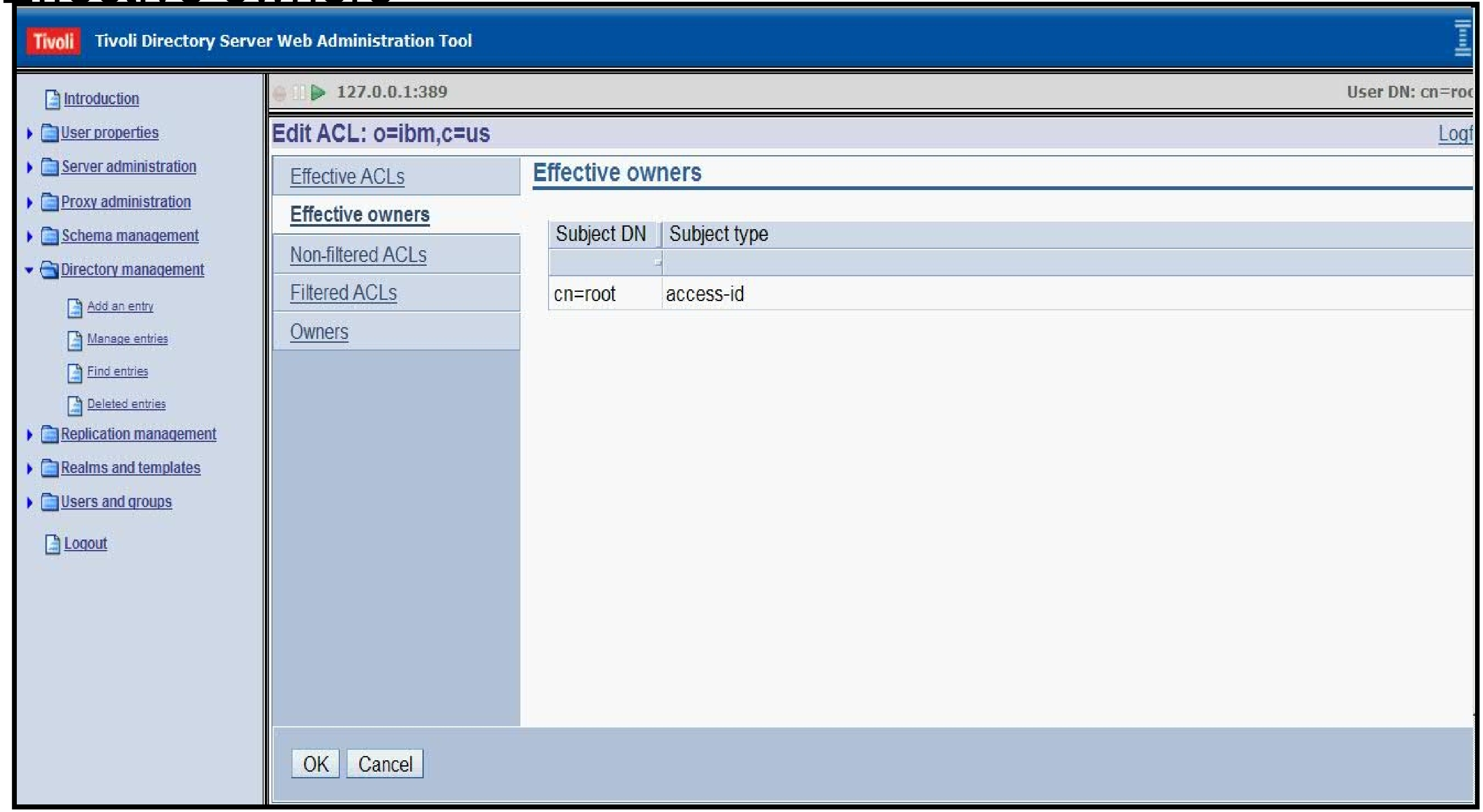


Click Refresh to retrieve the effective ACLs. This might take a long time to display

Select	Subject DN	Subject type
○	cn=anybody	group

Working with ACLs (Continued)

Effective owners



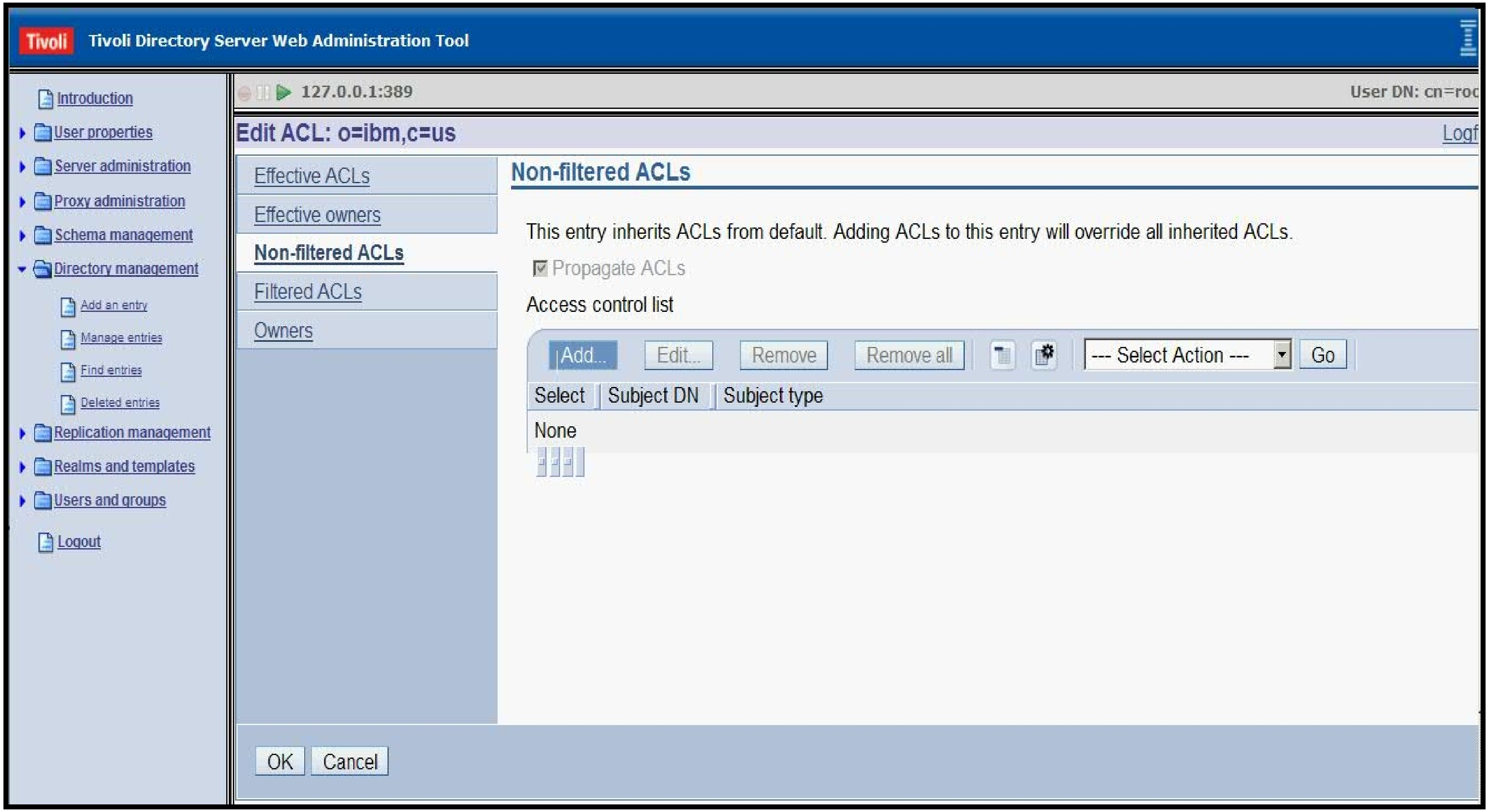
Edit ACL: o=ibm,c=us

Effective owners

Subject DN	Subject type
cn=root	access-id

Working with ACLs (Continued)

Non-filtered ACLs



Edit ACL: o=ibm,c=us

Non-filtered ACLs

This entry inherits ACLs from default. Adding ACLs to this entry will override all inherited ACLs.

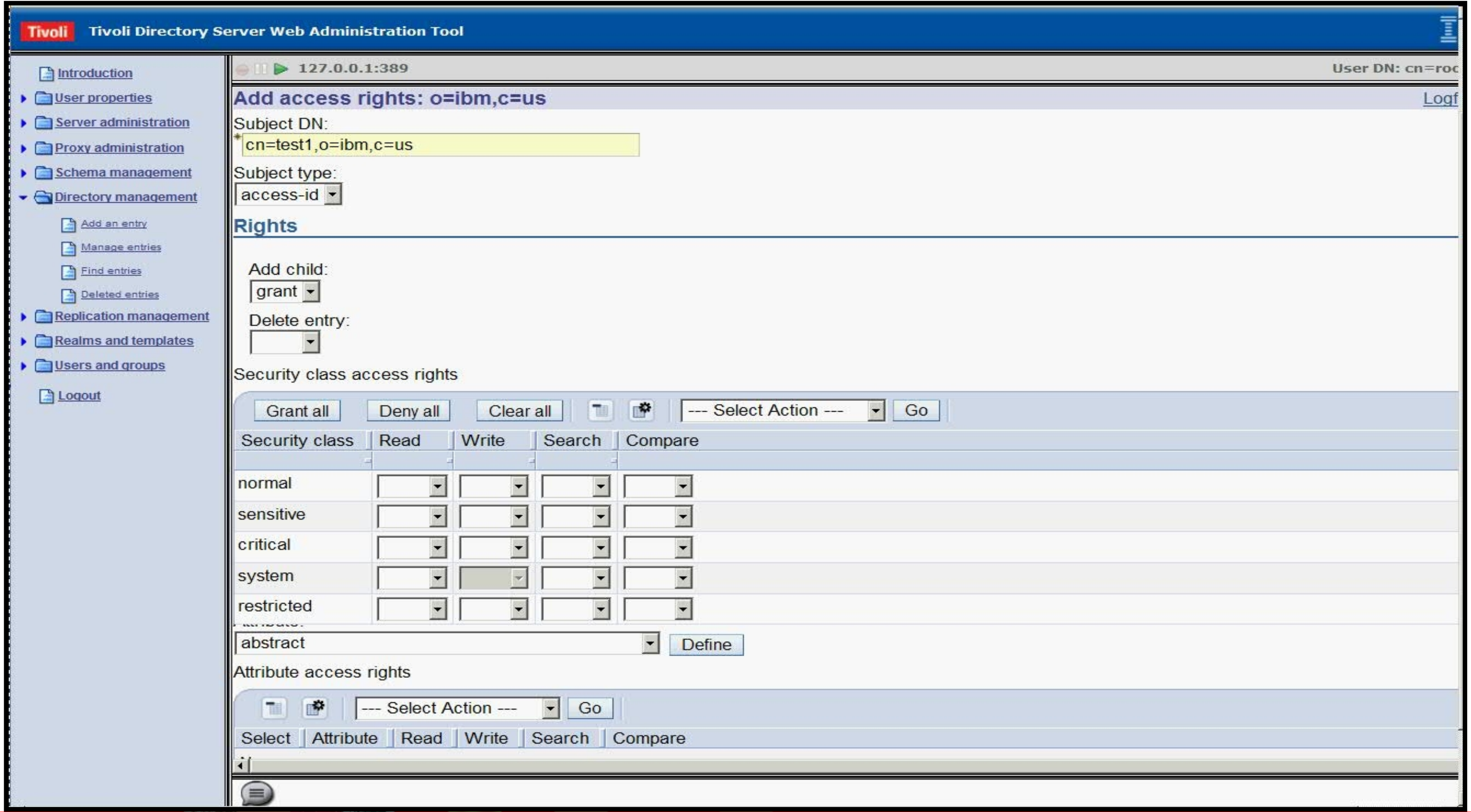
☑ Propagate ACLs

Access control list

Select	Subject DN	Subject type
None		

Working with ACLs (Continued)

Adding access rights



Add access rights: o=ibm,c=us

Subject DN: cn=test1,o=ibm,c=us

Subject type: access-id

Rights

Add child: grant

Delete entry:

Security class access rights

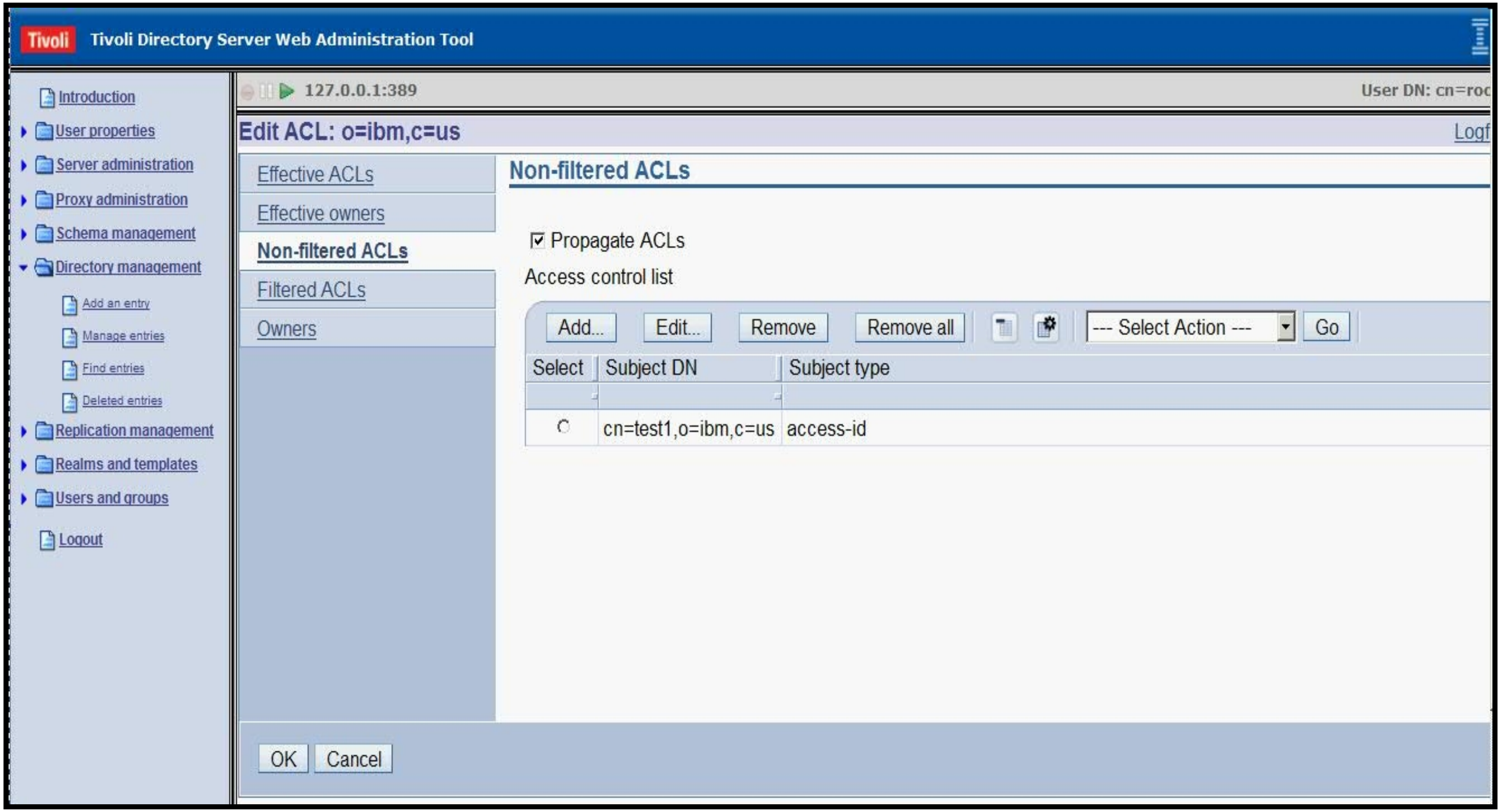
Security class	Read	Write	Search	Compare
normal				
sensitive				
critical				
system				
restricted				

Attribute: abstract

Attribute access rights

Select	Attribute	Read	Write	Search	Compare

Working with ACLs (Continued)



Edit ACL: o=ibm,c=us

Non-filtered ACLs

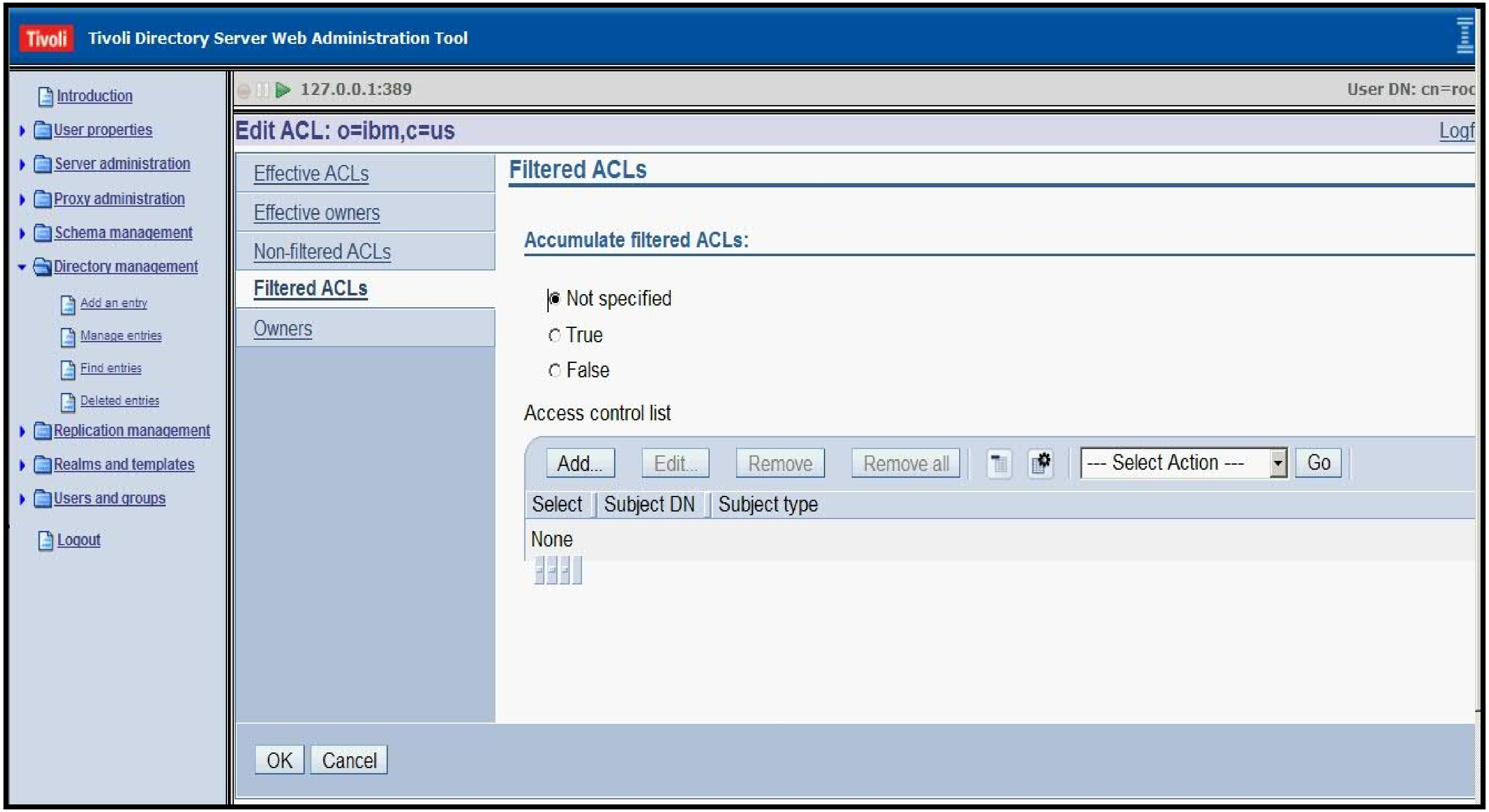
☑ Propagate ACLs

Access control list

Select	Subject DN	Subject type
○	cn=test1,o=ibm,c=us	access-id

Working with ACLs (Continued)

Filtered ACLs



Edit ACL: o=ibm,c=us

Filtered ACLs

Accumulate filtered ACLs:

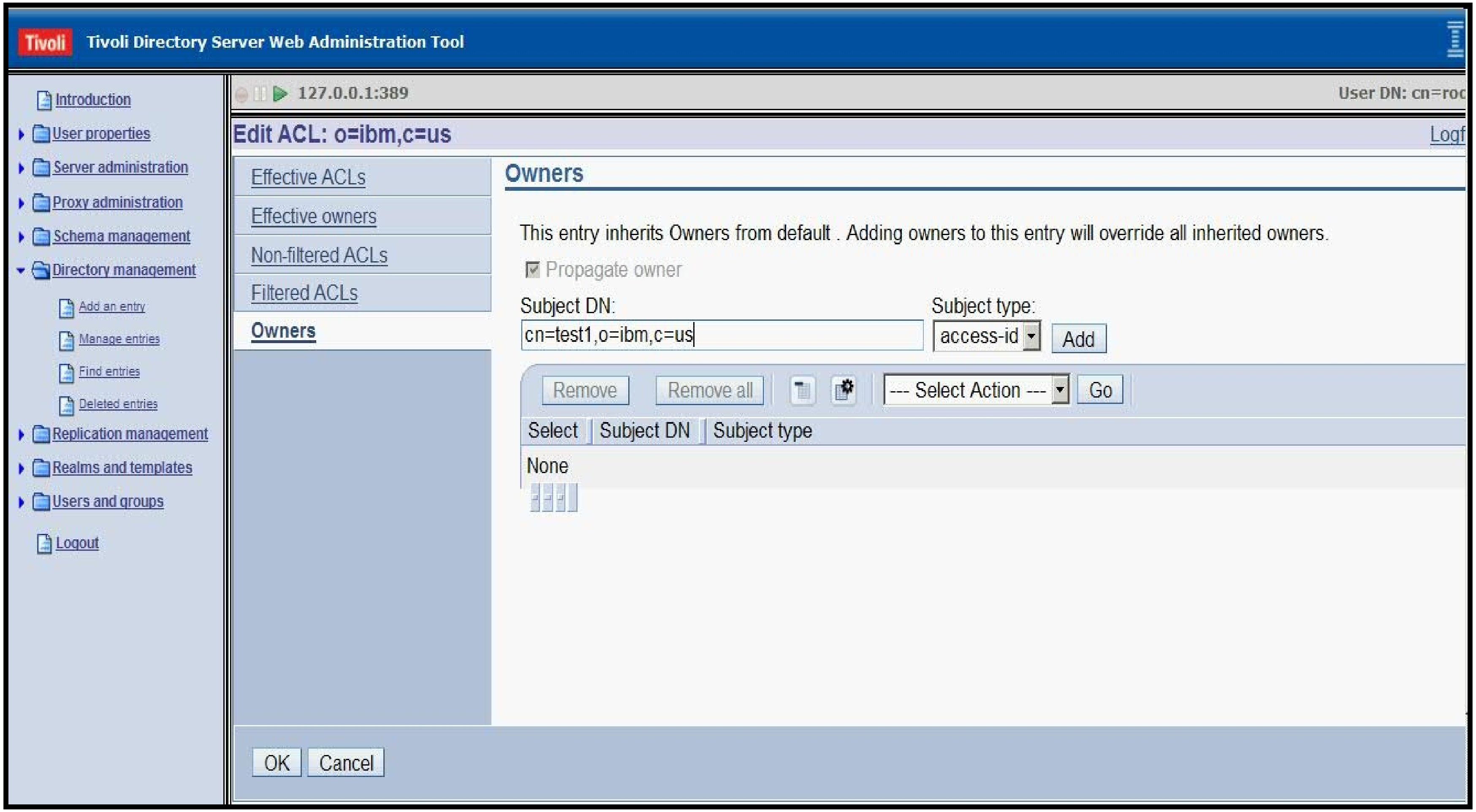
● Not specified
○ True
○ False

Access control list

Select	Subject DN	Subject type
None		

Working with ACLs (Continued)

Owners :



Edit ACL: o=ibm,c=us

Owners

This entry inherits Owners from default . Adding owners to this entry will override all inherited owners.

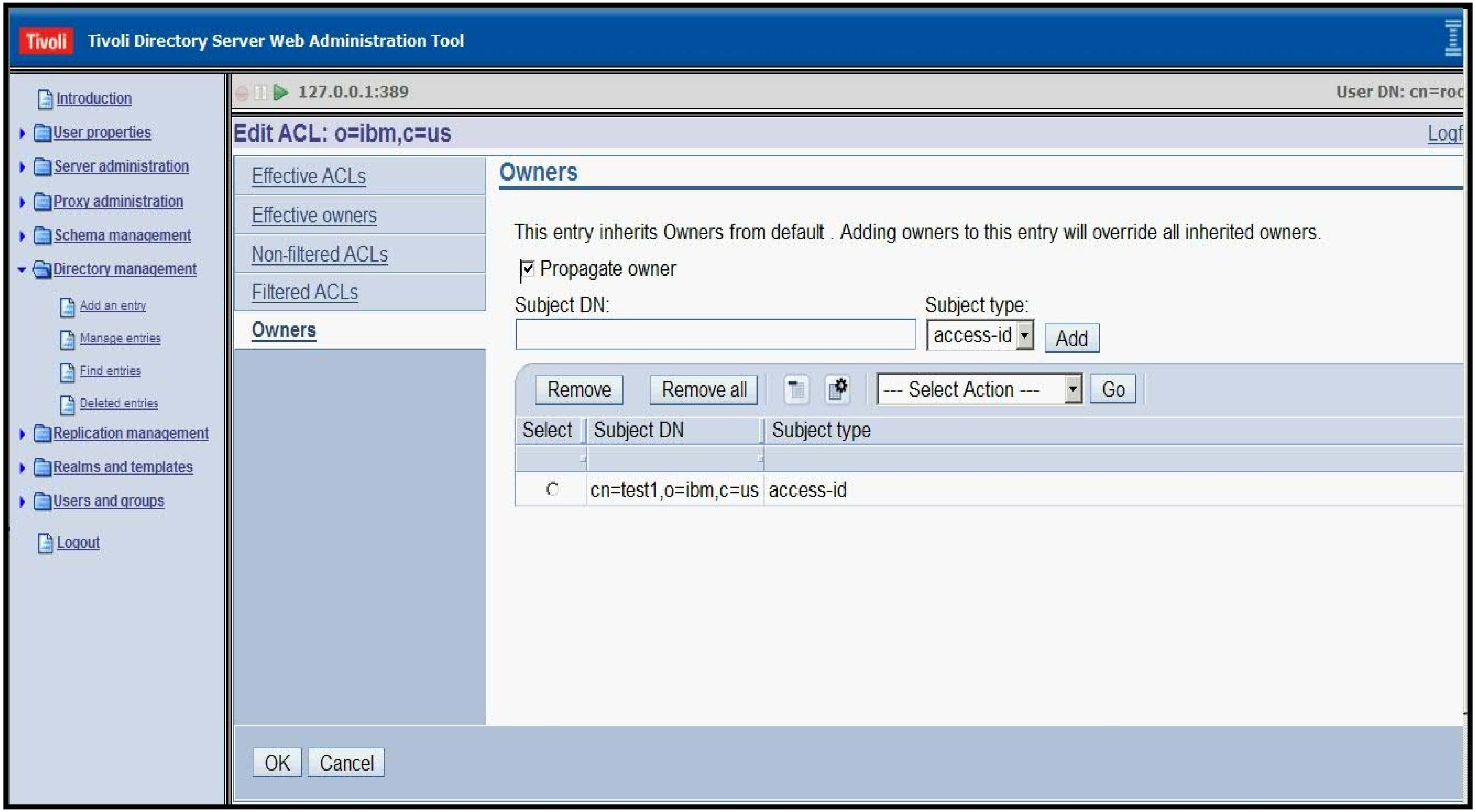
☑ Propagate owner

Subject DN: cn=test1,o=ibm,c=us

Subject type: access-id

Select	Subject DN	Subject type
None		

Working with ACLs (Continued)



Edit ACL: o=ibm,c=us

Owners

This entry inherits Owners from default . Adding owners to this entry will override all inherited owners.

☑ Propagate owner

Subject DN:

Subject type: access-id

Select	Subject DN	Subject type
○	cn=test1,o=ibm,c=us	access-id

Password Policy

- Password policy is a set of rules that controls how passwords are used and administered in IBM Tivoli Directory Server.
- These rules are made to ensure that users change their passwords periodically, and that the passwords meet the organization's syntactic password requirements.
- These rules can also restrict the reuse of old passwords and ensure that users are locked out after a defined number of failed bind attempts
- First focus : minimize threat of intruders.
- Second focus : enforce password syntax rules

Multiple password policies

- Tivoli Directory Server 6.0 users had a restriction that they could only configure a global password policy with the entry cn=pwdpolicy.
- With the release of TDS 6.1 and onwards , multiple options are available for password policies.
  - Global password policy
  - Individual password policy
  - Group password policy

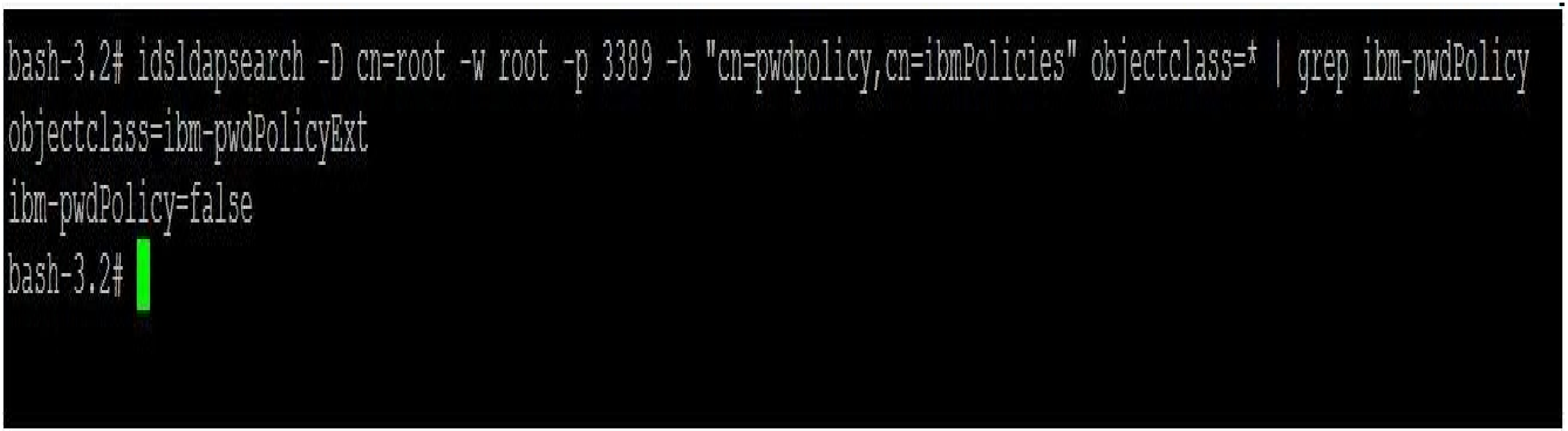
Password Policy Attributes

- pwdMinAge
- pwdMaxAge
- pwdMinLength
- pwdExpireWarning
- pwdGraceLoginLimit
- pwdLockoutDuration
- pwdMaxFailure
- pwdFailureCountInterval
- pwdMustChange
- pwdAllowUserChange
- pwdSafeModify
- passwordMinAlphaChars
- passwordMinOtherChars
- passwordMaxRepeatedChars
- passwordMinDiffChars

Password policy operational attributes

- **pwdChangedTime** - Contains the time the password was last changed.
- **pwdAccountLockedTime** - Contains the time at which the account was locked. If the account is not locked, this attribute is not present**.**
- **pwdExpirationWarned** - Contains the time at which the password expiration warning was first sent to the client.
- **pwdFailureTime** - A multi-valued attribute containing the times of previous consecutive login failures. If the last login was successful, this attribute is not present.
- **pwdGraceUseTime** - A multi-valued attribute containing the times of the previous grace logins.
- **pwdReset** - Contains the value TRUE if the password has been reset and must be changed by the user. The value is FALSE or not present otherwise.
- **ibm-pwdAccountLocked** - Indicates that the account has been administratively locked.

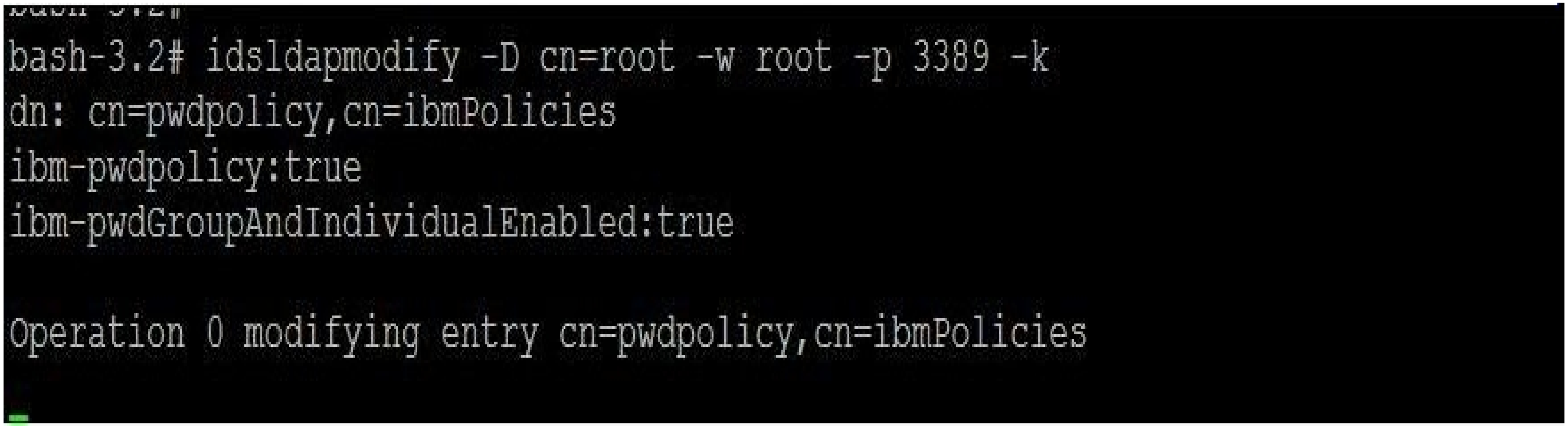
Password Policy Configuration-Command line



```
bash-3.2# idsldapsearch -D cn=root -w root -p 3389 -b "cn=pwdpolicy,cn=ibmPolicies" objectclass=* | grep ibm-pwdPolicy
objectclass=ibm-pwdPolicyExt
ibm-pwdPolicy=false
bash-3.2#
```

Password Policy Configuration-Command line(Contd.)

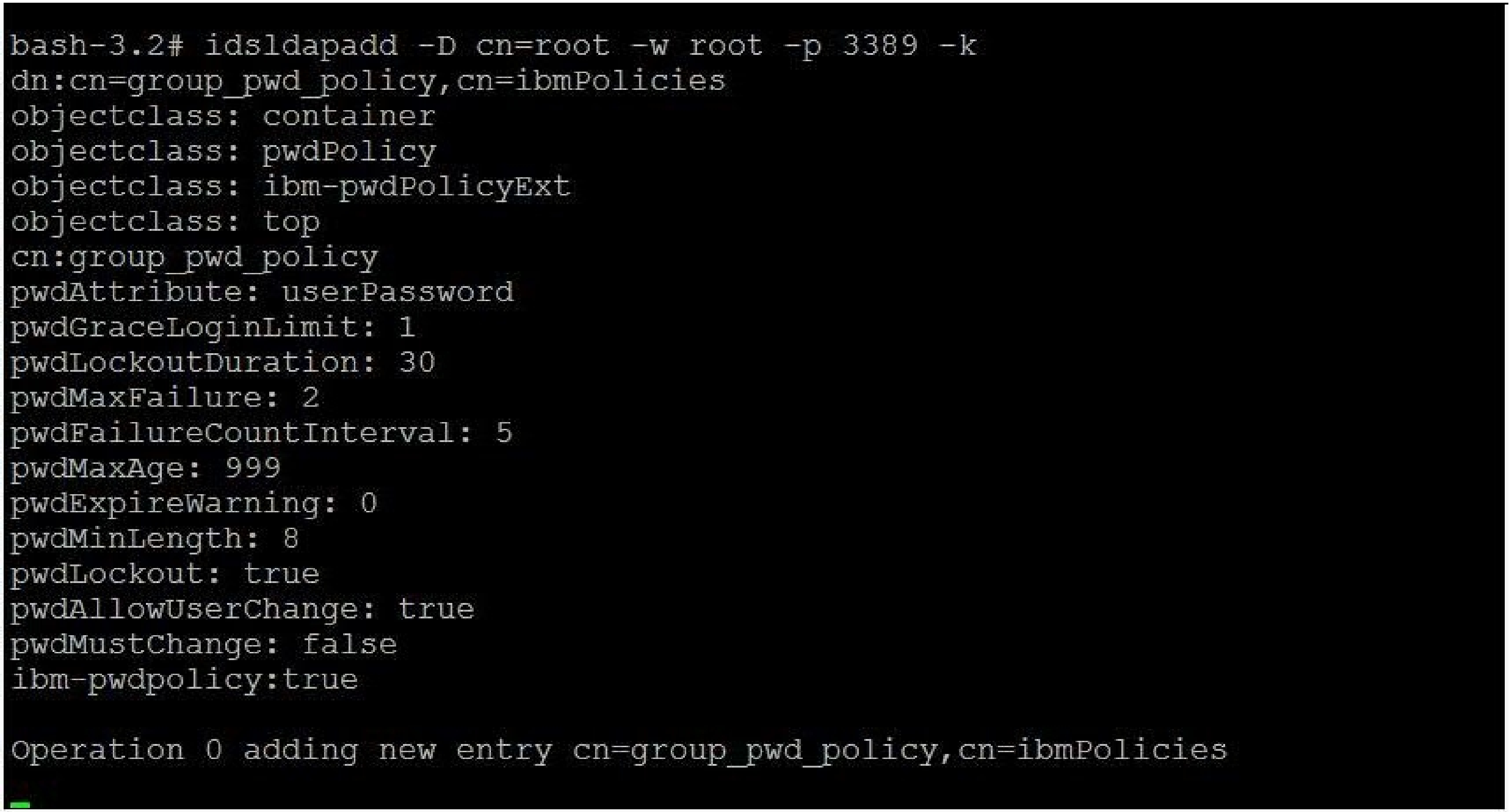
Enabling Group and individual password policy



```
bash-3.2#
bash-3.2# idsldapmodify -D cn=root -w root -p 3389 -k
dn: cn=pwdpolicy,cn=ibmPolicies
ibm-pwdpolicy:true
ibm-pwdGroupAndIndividualEnabled:true

Operation 0 modifying entry cn=pwdpolicy,cn=ibmPolicies
```

Password Policy Configuration-Command line(Contd.)

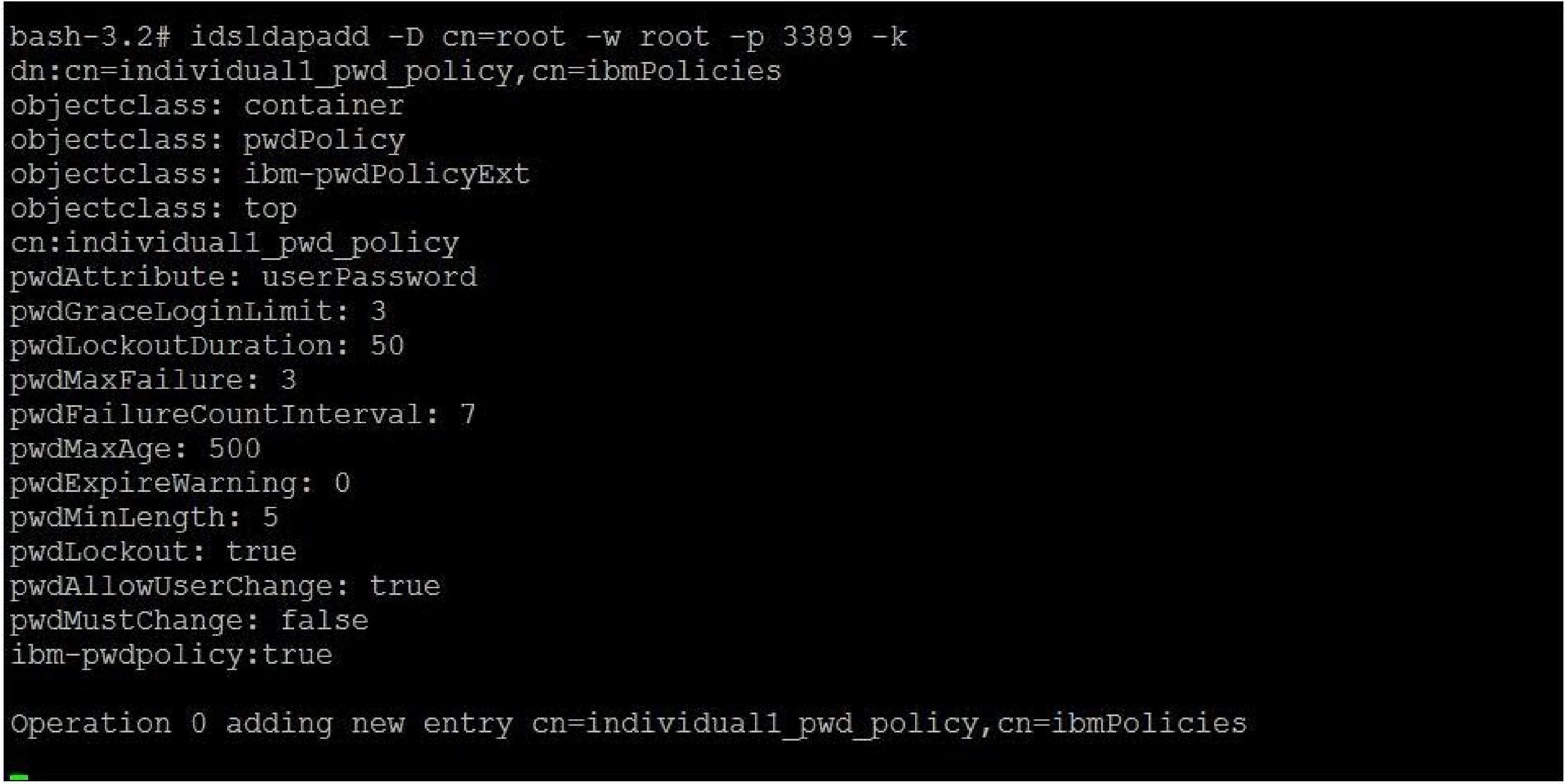
Define Group Password Policy



```
bash-3.2# idsldapadd -D cn=root -w root -p 3389 -k
dn:cn=group_pwd_policy,cn=ibmPolicies
objectclass: container
objectclass: pwdPolicy
objectclass: ibm-pwdPolicyExt
objectclass: top
cn:group_pwd_policy
pwdAttribute: userPassword
pwdGraceLoginLimit: 1
pwdLockoutDuration: 30
pwdMaxFailure: 2
pwdFailureCountInterval: 5
pwdMaxAge: 999
pwdExpireWarning: 0
pwdMinLength: 8
pwdLockout: true
pwdAllowUserChange: true
pwdMustChange: false
ibm-pwdpolicy:true

Operation 0 adding new entry cn=group_pwd_policy,cn=ibmPolicies
```

Password Policy Configuration-Command line(Contd.)

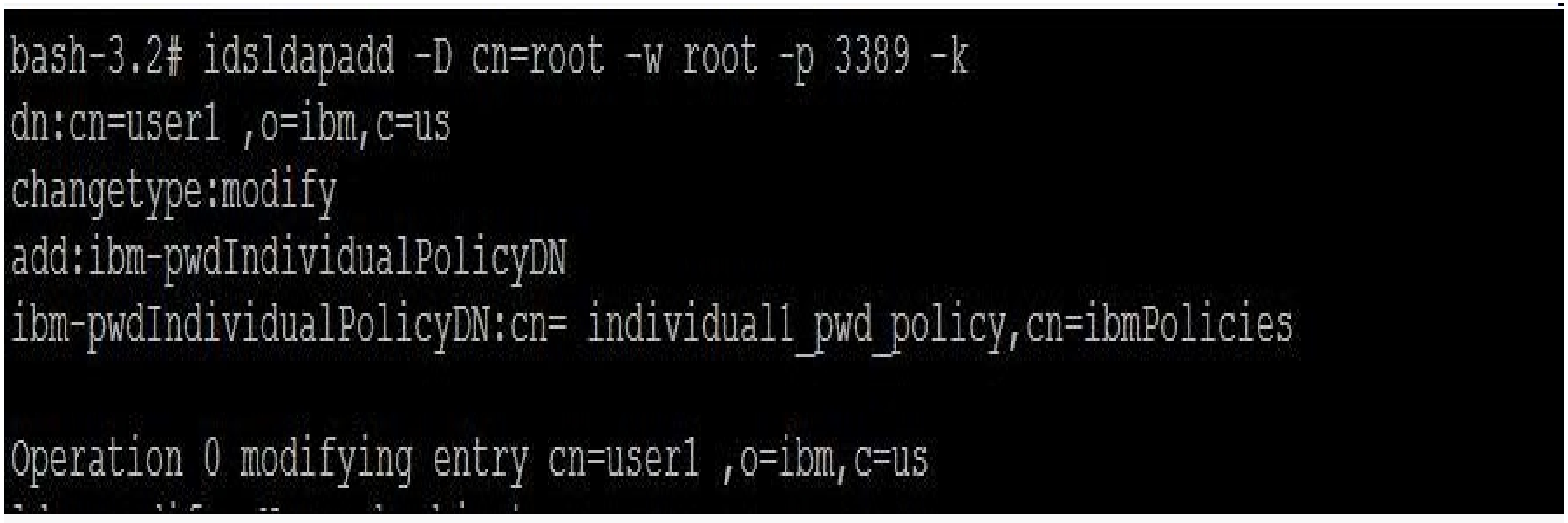
Define Individual Password Policy



```
bash-3.2# idsldapadd -D cn=root -w root -p 3389 -k
dn:cn=individual1_pwd_policy,cn=ibmPolicies
objectclass: container
objectclass: pwdPolicy
objectclass: ibm-pwdPolicyExt
objectclass: top
cn:individual1_pwd_policy
pwdAttribute: userPassword
pwdGraceLoginLimit: 3
pwdLockoutDuration: 50
pwdMaxFailure: 3
pwdFailureCountInterval: 7
pwdMaxAge: 500
pwdExpireWarning: 0
pwdMinLength: 5
pwdLockout: true
pwdAllowUserChange: true
pwdMustChange: false
ibm-pwdpolicy:true

Operation 0 adding new entry cn=individual1_pwd_policy,cn=ibmPolicies
```

Password Policy Configuration-Command line(Contd.)

Associating an individual password policy with a user.



```
bash-3.2# idsldapadd -D cn=root -w root -p 3389 -k
dn:cn=user1 ,o=ibm,c=us
changetype:modify
add:ibm-pwdIndividualPolicyDN
ibm-pwdIndividualPolicyDN:cn= individual1_pwd_policy,cn=ibmPolicies

Operation 0 modifying entry cn=user1 ,o=ibm,c=us
```

Password policy debugging practices

- **Operational Attributes** on a given user can be listed using the following ldapsearch command :

```
  idsldapsearch -D <AdminDN> -w <AdminPW> -s base -b
"<UserEntryDN>" objectclass=* +ibmpwdpolicy
```

- **Global Password Policy** settings can be listed using the following ldapsearch command :

```
  idsldapsearch -D <AdminDN> -w <AdminPW> -s base -b
"cn=pwdpolicy,cn=ibmPolicies" objectclass=*
```

Password policy debugging practices (Contd.)

- **Group / User Password Policies** can be listed using the following ldapsearch command :

```
idsldapsearch -D <AdminDN> -w <AdminPW> -s sub -b " " objectclass=ibm-pwd*
```

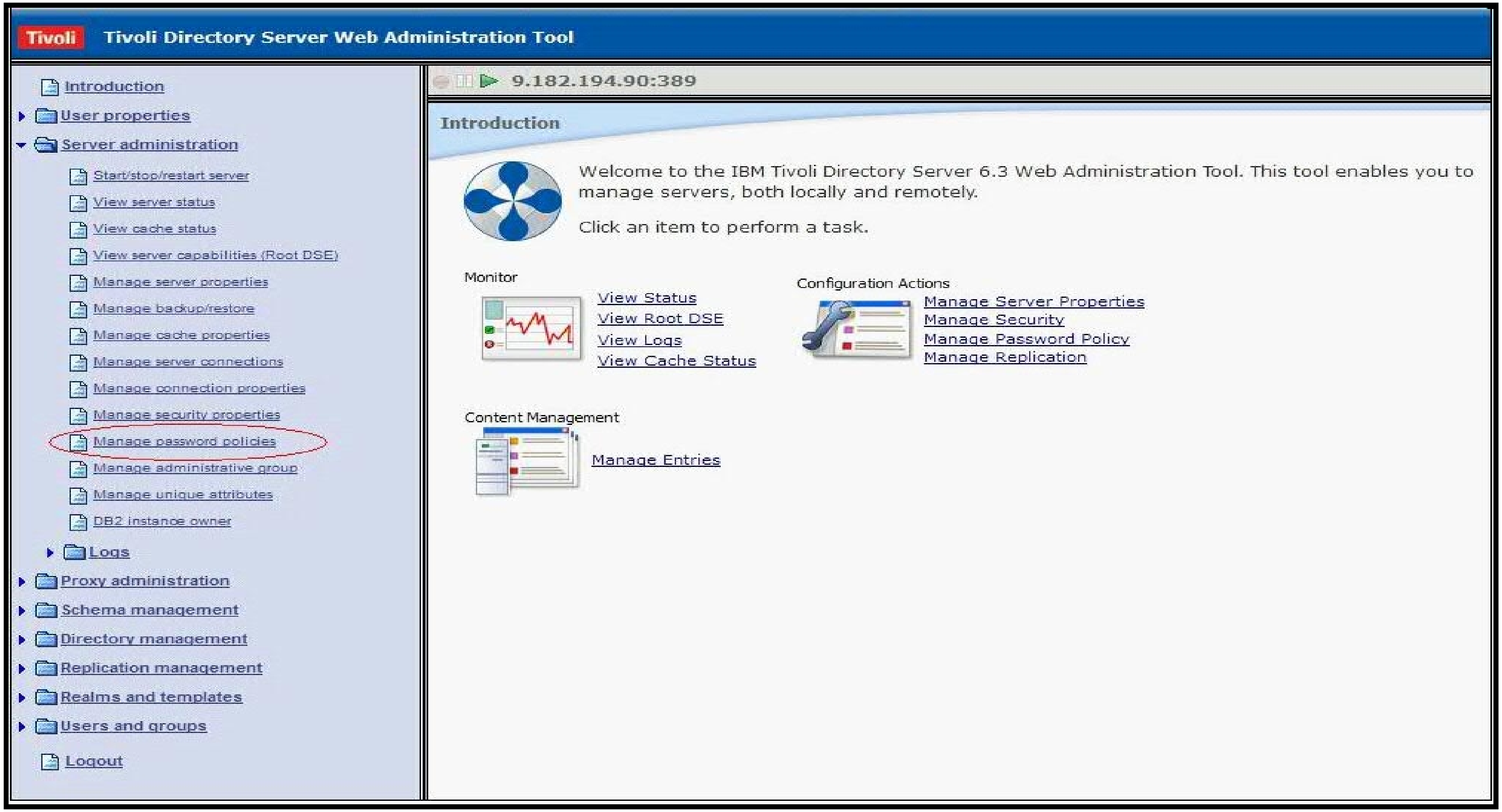
- **Effective Password Policy on a Given User** can be calculated using the following ldapexop command :

```
idsldapexop -D <AdminDN> -w <AdminPW> -op effectpwdpolicy -d "<UserEntryDN>"
```

Common Errors and Their Solutions

- Authentication error: Either the user name or password (or both) is incorrect, or the password has expired.
- Password policy rule violated: Verify the input given .
- The password policy entry DN of entry is in use and cannot be renamed or deleted.

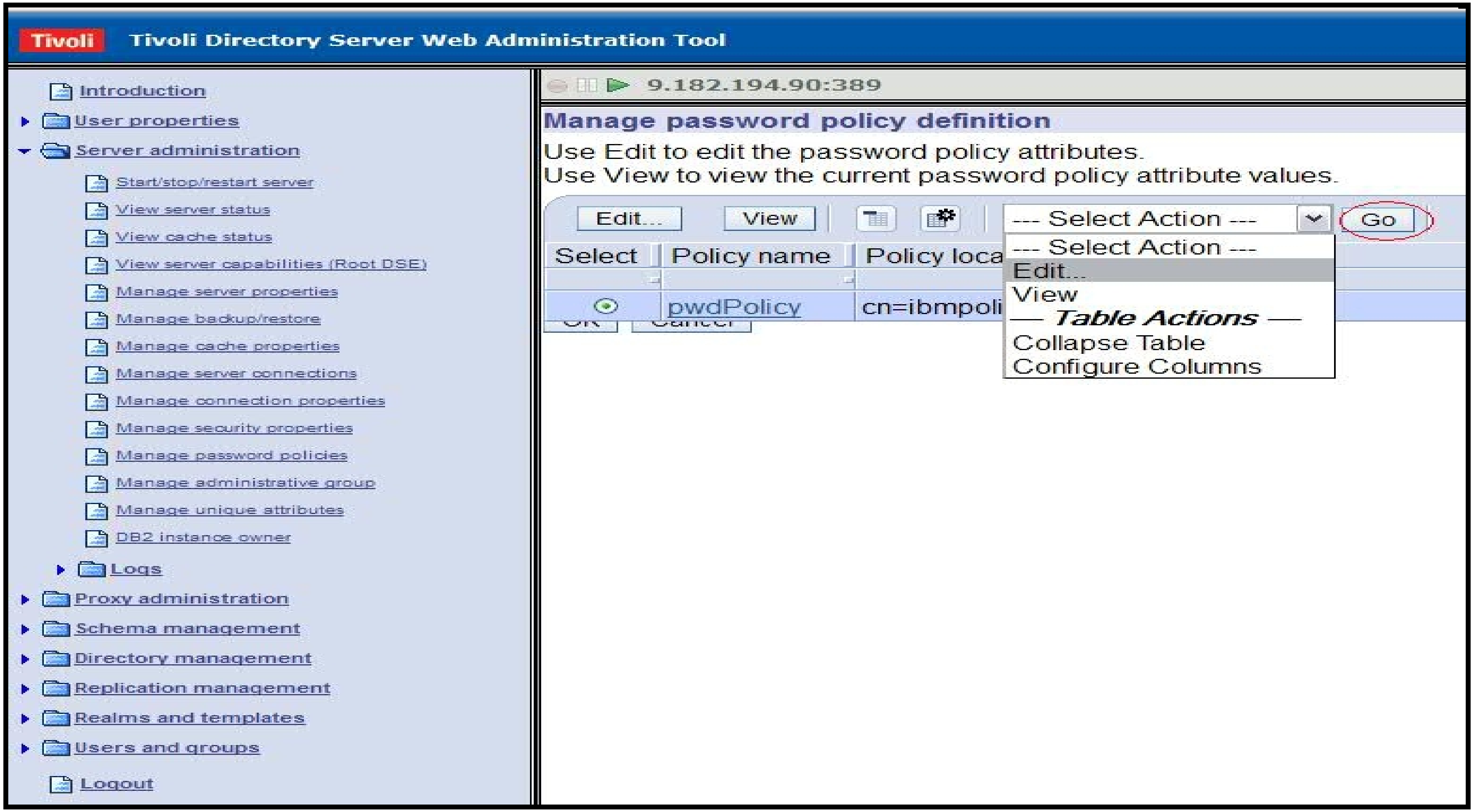
Password policy Configuration- Web AdminTool



Welcome to the IBM Tivoli Directory Server 6.3 Web Administration Tool. This tool enables you to manage servers, both locally and remotely.

Click an item to perform a task.

Password policy Configuration- Web AdminTool



Tivoli Tivoli Directory Server Web Administration Tool

- Introduction
- User properties
- Server administration
  - Start/stop/restart server
  - View server status
  - View cache status
  - View server capabilities (Root DSE)
  - Manage server properties
  - Manage backup/restore
  - Manage cache properties
  - Manage server connections
  - Manage connection properties
  - Manage security properties
  - Manage password policies
  - Manage administrative group
  - Manage unique attributes
  - DB2 instance owner
  - Logs
- Proxy administration
- Schema management
- Directory management
- Replication management
- Realms and templates
- Users and groups
- Logout

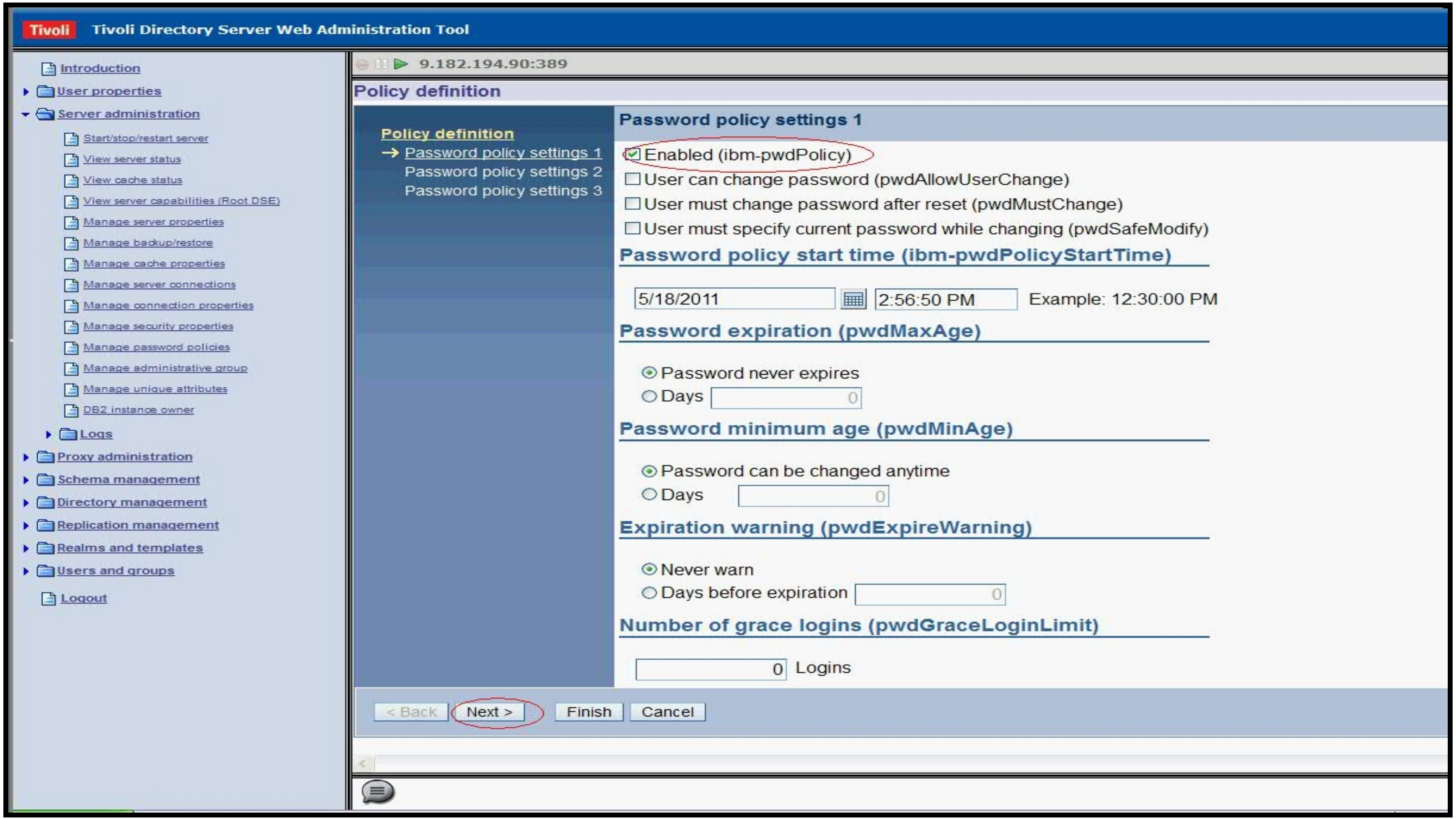
9.182.194.90:389

Manage password policy definition

Use Edit to edit the password policy attributes.
Use View to view the current password policy attribute values.

Select	Policy name	Policy loca
●	pwdPolicy	cn=ibmpoli

Password policy Configuration- Web AdminTool



Password policy settings 1

☑ Enabled (ibm-pwdPolicy)

□ User can change password (pwdAllowUserChange)

□ User must change password after reset (pwdMustChange)

□ User must specify current password while changing (pwdSafeModify)

Password policy start time (ibm-pwdPolicyStartTime)

5/18/2011 2:56:50 PM Example: 12:30:00 PM

Password expiration (pwdMaxAge)

● Password never expires

○ Days 0

Password minimum age (pwdMinAge)

● Password can be changed anytime

○ Days 0

Expiration warning (pwdExpireWarning)

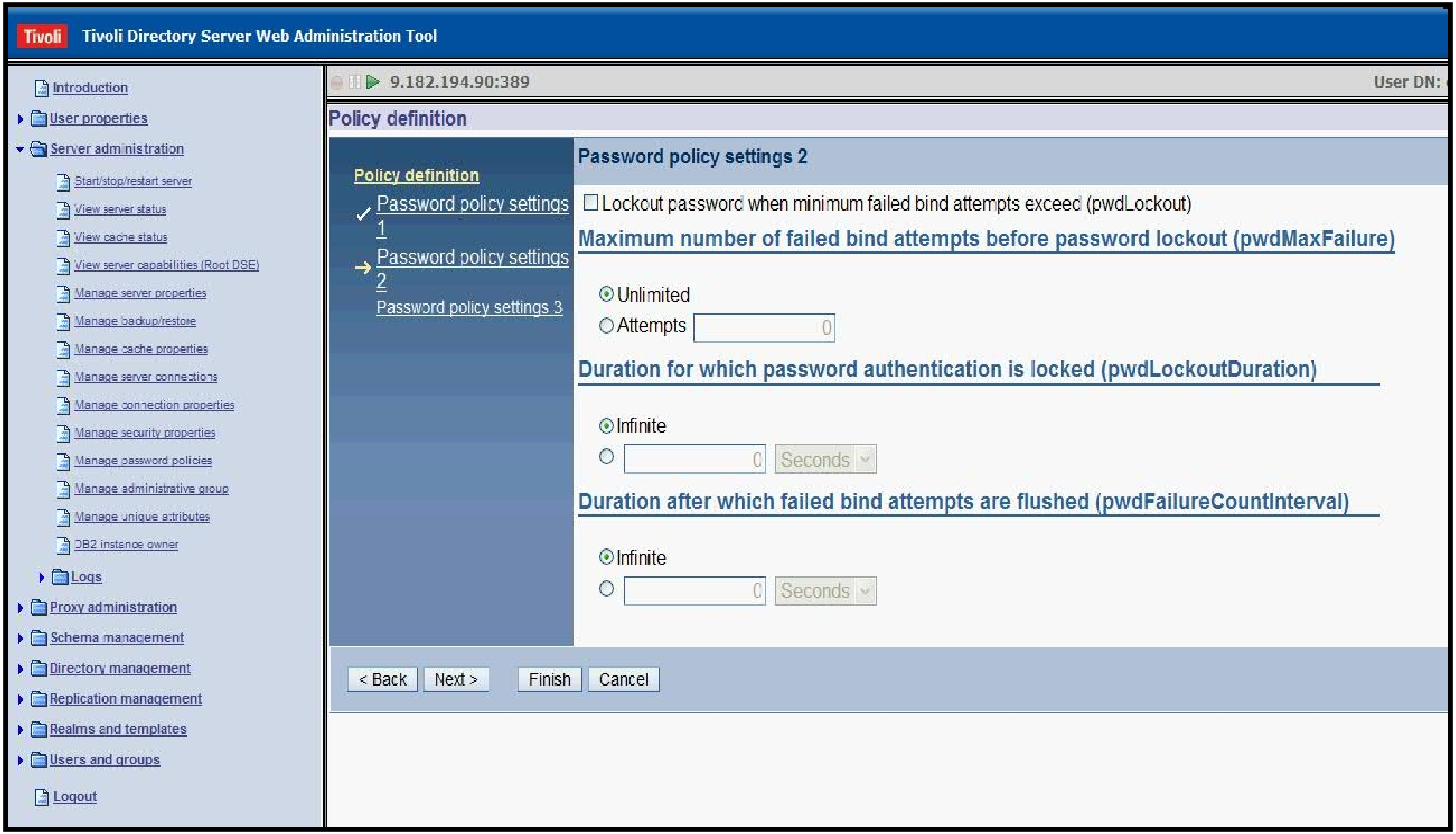
● Never warn

○ Days before expiration 0

Number of grace logins (pwdGraceLoginLimit)

0 Logins

Password policy Configuration- Web AdminTool



Password policy settings 2

□ Lockout password when minimum failed bind attempts exceed (pwdLockout)

Maximum number of failed bind attempts before password lockout (pwdMaxFailure)

● Unlimited
○ Attempts 0

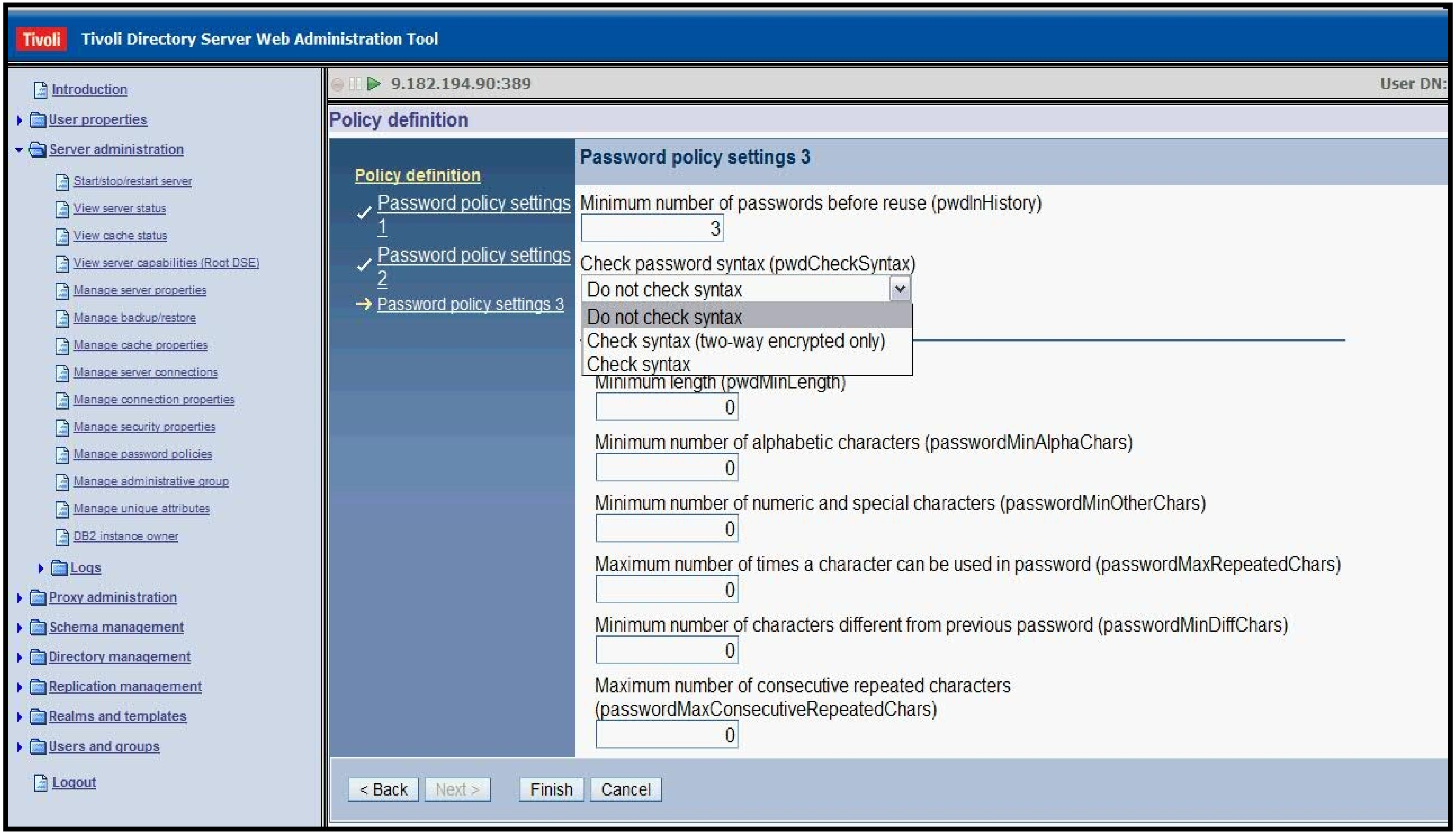
Duration for which password authentication is locked (pwdLockoutDuration)

● Infinite
○ 0 Seconds

Duration after which failed bind attempts are flushed (pwdFailureCountInterval)

● Infinite
○ 0 Seconds

Password policy Configuration- Web AdminTool



Password policy settings 3

Minimum number of passwords before reuse (pwdInHistory): 3

Check password syntax (pwdCheckSyntax): Do not check syntax

- Do not check syntax
- Check syntax (two-way encrypted only)
- Check syntax

Minimum length (pwdMinLength): 0

Minimum number of alphabetic characters (passwordMinAlphaChars): 0

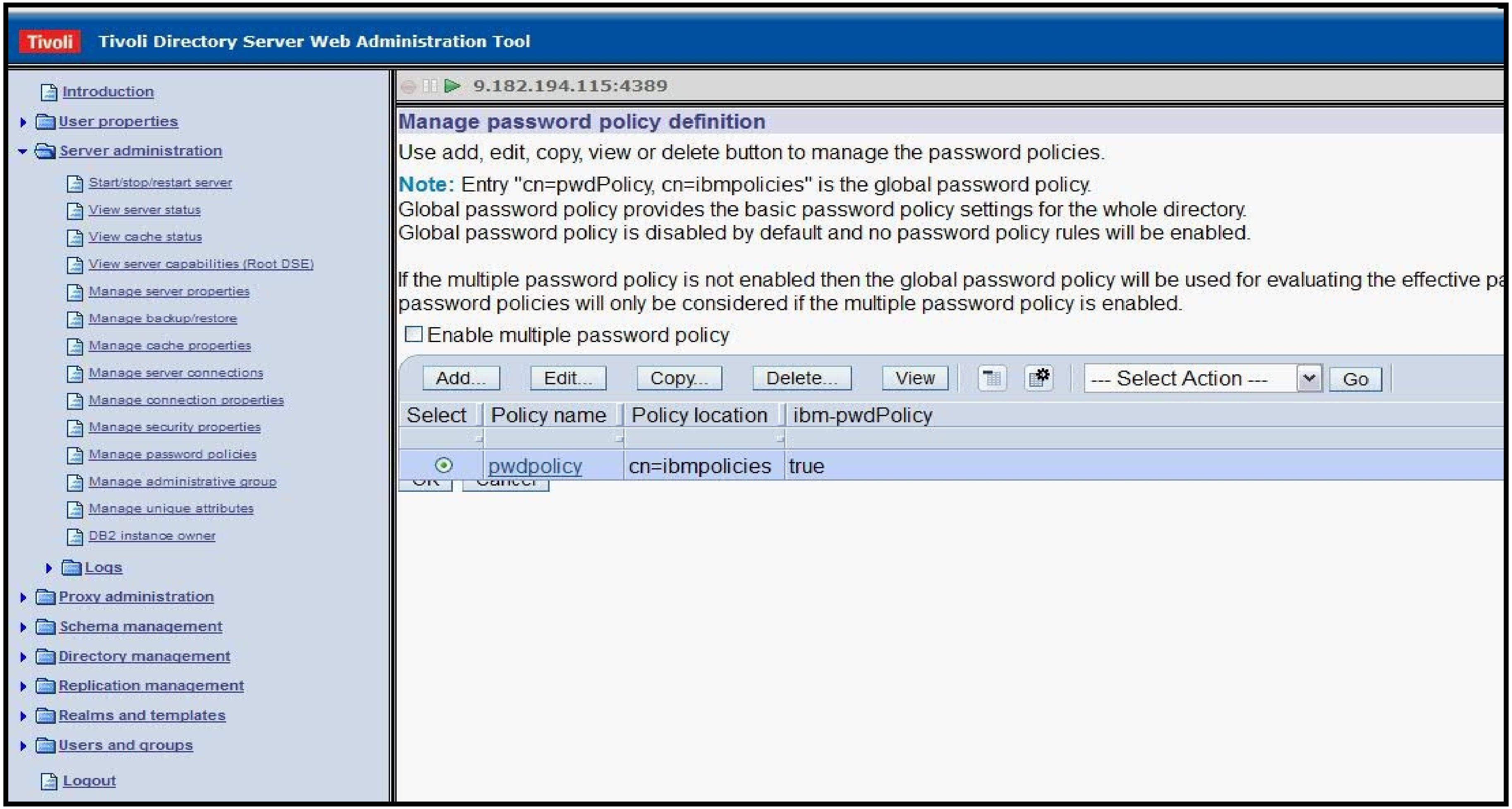
Minimum number of numeric and special characters (passwordMinOtherChars): 0

Maximum number of times a character can be used in password (passwordMaxRepeatedChars): 0

Minimum number of characters different from previous password (passwordMinDiffChars): 0

Maximum number of consecutive repeated characters (passwordMaxConsecutiveRepeatedChars): 0

Password policy Configuration- Web AdminTool



Manage password policy definition

Use add, edit, copy, view or delete button to manage the password policies.

Note: Entry "cn=pwdPolicy, cn=ibmpolicies" is the global password policy.
Global password policy provides the basic password policy settings for the whole directory.
Global password policy is disabled by default and no password policy rules will be enabled.

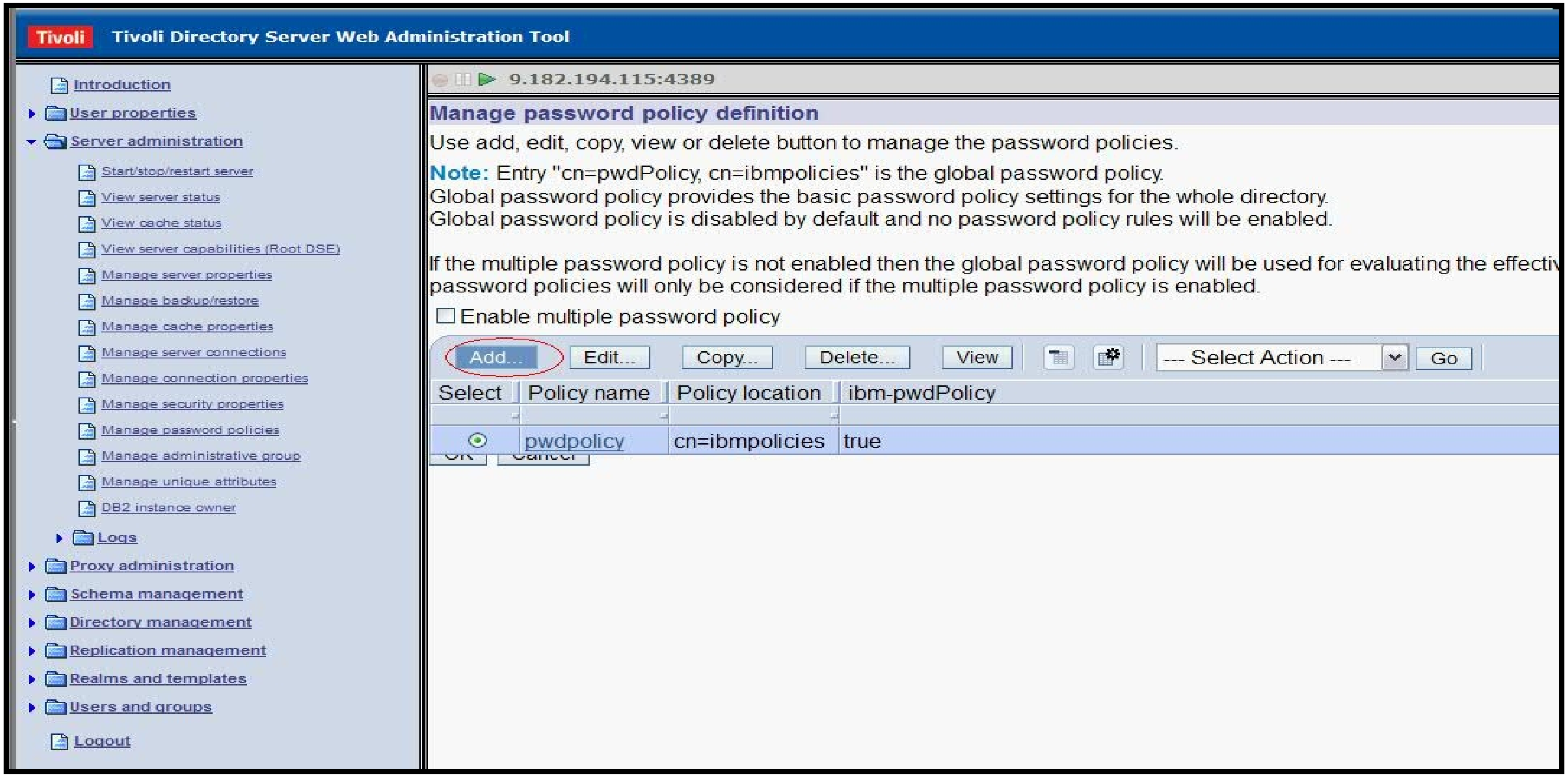
If the multiple password policy is not enabled then the global password policy will be used for evaluating the effective pa
password policies will only be considered if the multiple password policy is enabled.

☐ Enable multiple password policy

Select	Policy name	Policy location	ibm-pwdPolicy
●	pwdpolicy	cn=ibmpolicies	true

Password policy Configuration- Web AdminTool

Enabling Individual Password policy



Manage password policy definition

Use add, edit, copy, view or delete button to manage the password policies.

Note: Entry "cn=pwdPolicy, cn=ibmpolicies" is the global password policy.
Global password policy provides the basic password policy settings for the whole directory.
Global password policy is disabled by default and no password policy rules will be enabled.

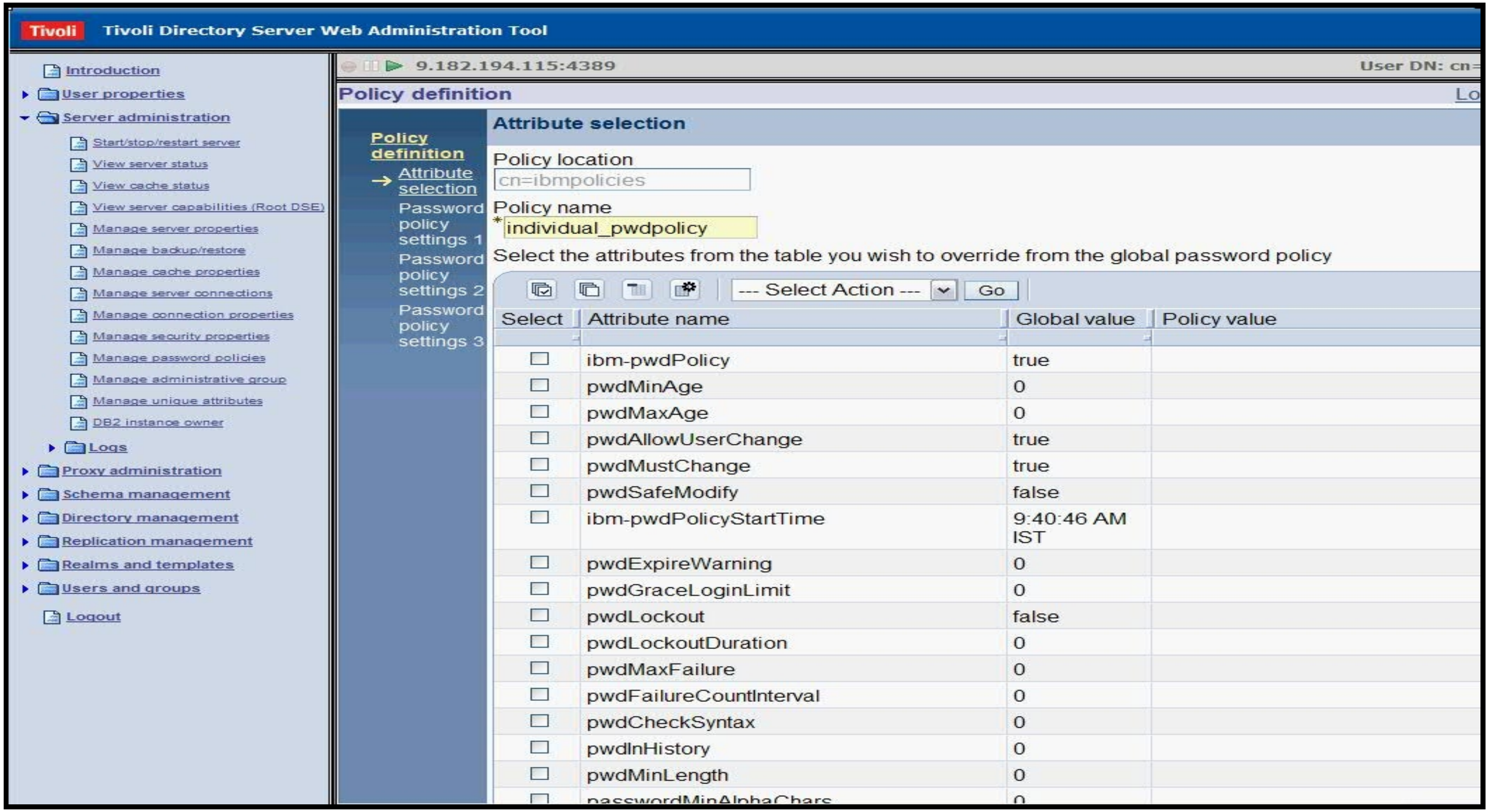
If the multiple password policy is not enabled then the global password policy will be used for evaluating the effecti
password policies will only be considered if the multiple password policy is enabled.

□ Enable multiple password policy

Select	Policy name	Policy location	ibm-pwdPolicy
●	pwdpolicy	cn=ibmpolicies	true

Password policy Configuration- Web AdminTool

Define individual password policy



Policy definition

Attribute selection

Policy location

cn=ibmpolicies

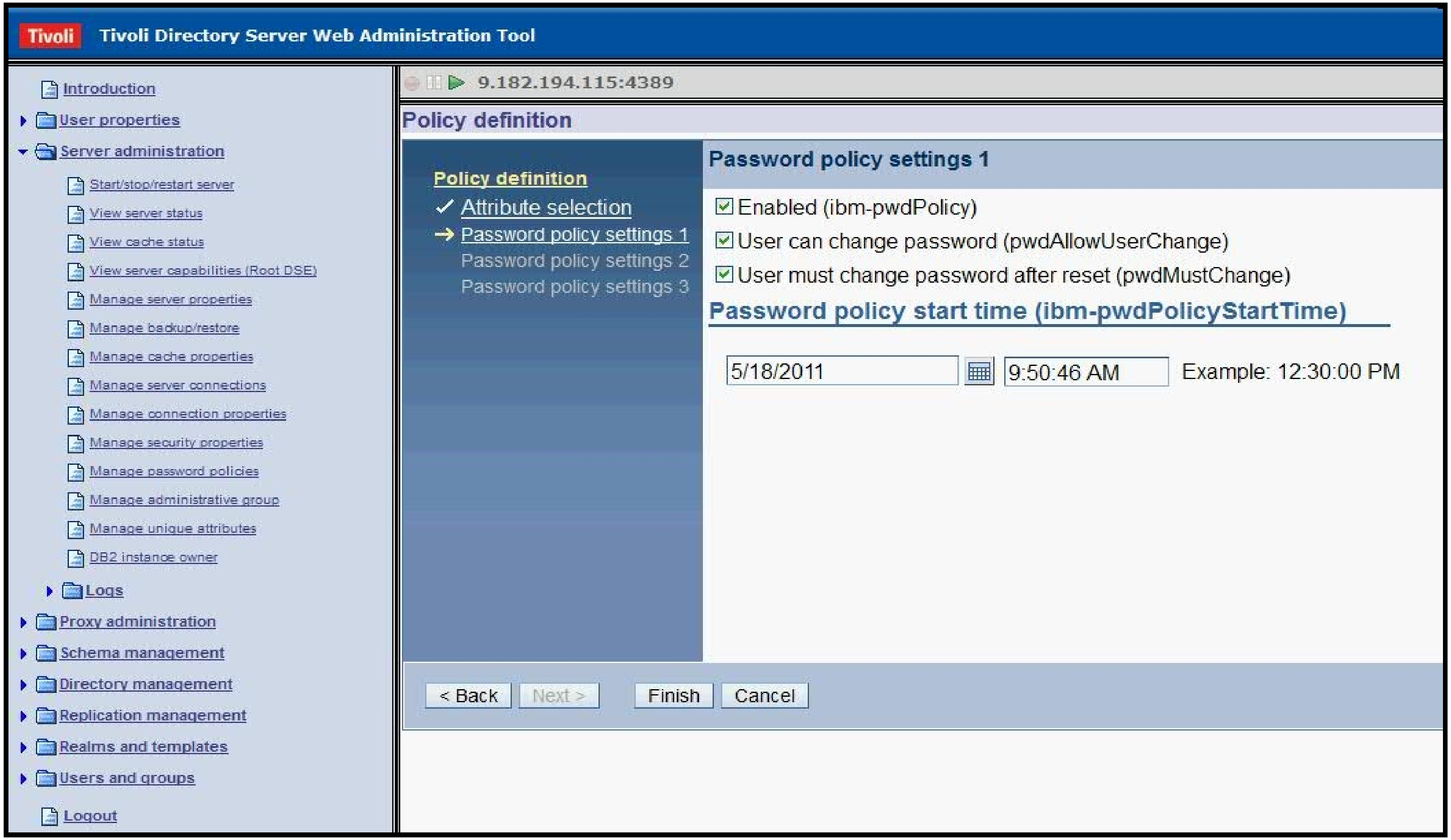
Policy name

* individual_pwdpolicy

Select the attributes from the table you wish to override from the global password policy

Select	Attribute name	Global value	Policy value
□	ibm-pwdPolicy	true	
□	pwdMinAge	0	
□	pwdMaxAge	0	
□	pwdAllowUserChange	true	
□	pwdMustChange	true	
□	pwdSafeModify	false	
□	ibm-pwdPolicyStartTime	9:40:46 AM IST	
□	pwdExpireWarning	0	
□	pwdGraceLoginLimit	0	
□	pwdLockout	false	
□	pwdLockoutDuration	0	
□	pwdMaxFailure	0	
□	pwdFailureCountInterval	0	
□	pwdCheckSyntax	0	
□	pwdInHistory	0	
□	pwdMinLength	0	
□	passwordMinAlphaChars	0	

Password policy Configuration- Web AdminTool



Password policy settings 1

☑ Enabled (ibm-pwdPolicy)

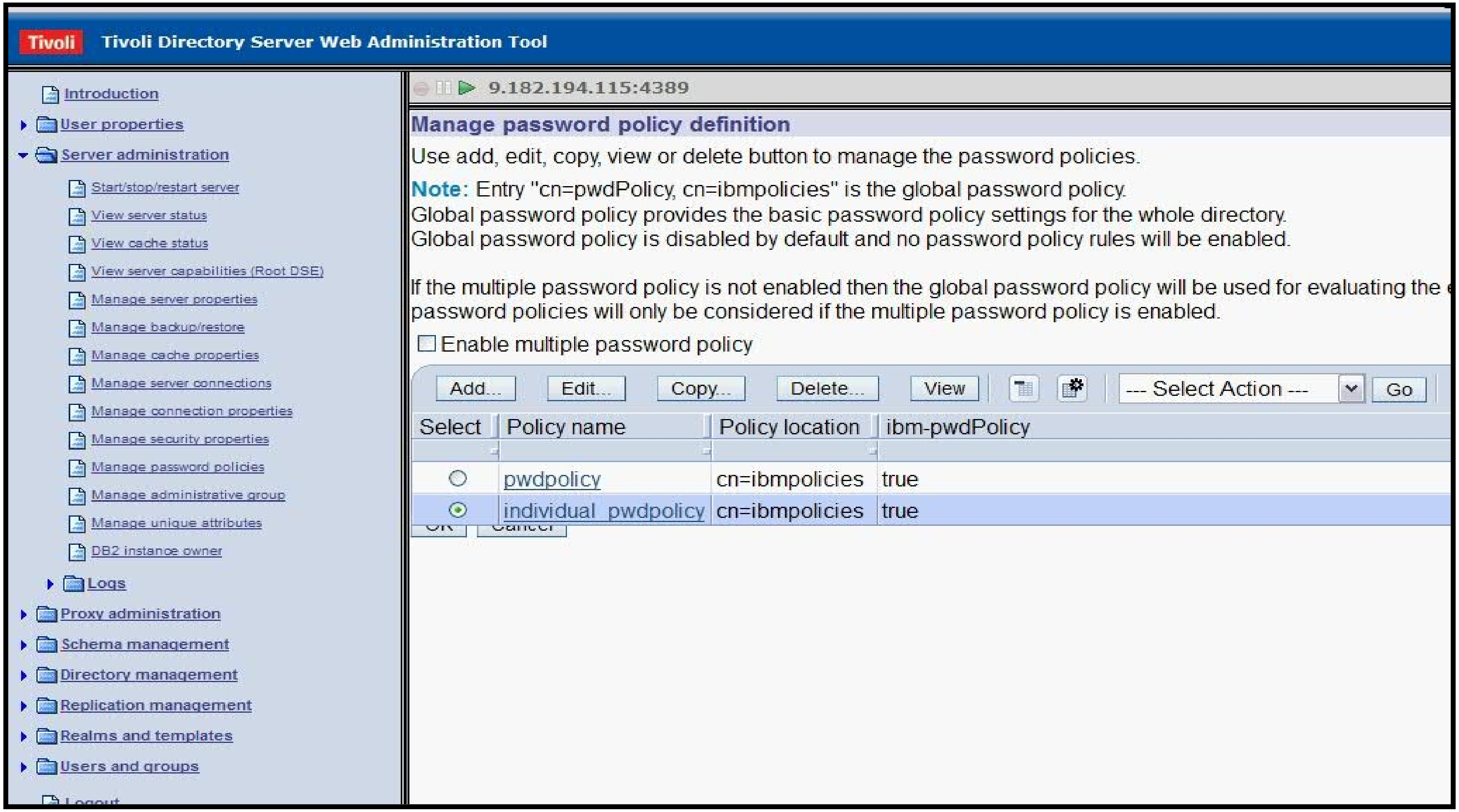
☑ User can change password (pwdAllowUserChange)

☑ User must change password after reset (pwdMustChange)

Password policy start time (ibm-pwdPolicyStartTime)

5/18/2011 9:50:46 AM Example: 12:30:00 PM

Password policy Configuration- Web AdminTool



Manage password policy definition

Use add, edit, copy, view or delete button to manage the password policies.

Note: Entry "cn=pwdPolicy, cn=ibmpolicies" is the global password policy.
Global password policy provides the basic password policy settings for the whole directory.
Global password policy is disabled by default and no password policy rules will be enabled.

If the multiple password policy is not enabled then the global password policy will be used for evaluating the e
password policies will only be considered if the multiple password policy is enabled.

☐ Enable multiple password policy

Select	Policy name	Policy location	ibm-pwdPolicy
○	pwdpolicy	cn=ibmpolicies	true
●	individual_pwdpolicy	cn=ibmpolicies	true

Web Admin Tool

- IBM Tivoli Directory Server Web Administration Tool is a graphical user interface version of ITDS.
- It is installed on an application server, such as the embedded version of IBM WebSphere® Application Server Express® (WAS) included with IBM Tivoli Directory Server, and administered through a console.
- Servers that have been added to the console can be managed through the Web Administration Tool without having to have the tool installed on each server.

Starting WebSphere Application Server

- To start the Web Administration Tool, we must start the application server in which it was installed.
- Use one of the following files to start the Web application server if you are using Embedded WebSphere Application Server.

On Windows systems,

installpath\idstools\bin\startWebadminApp.bat

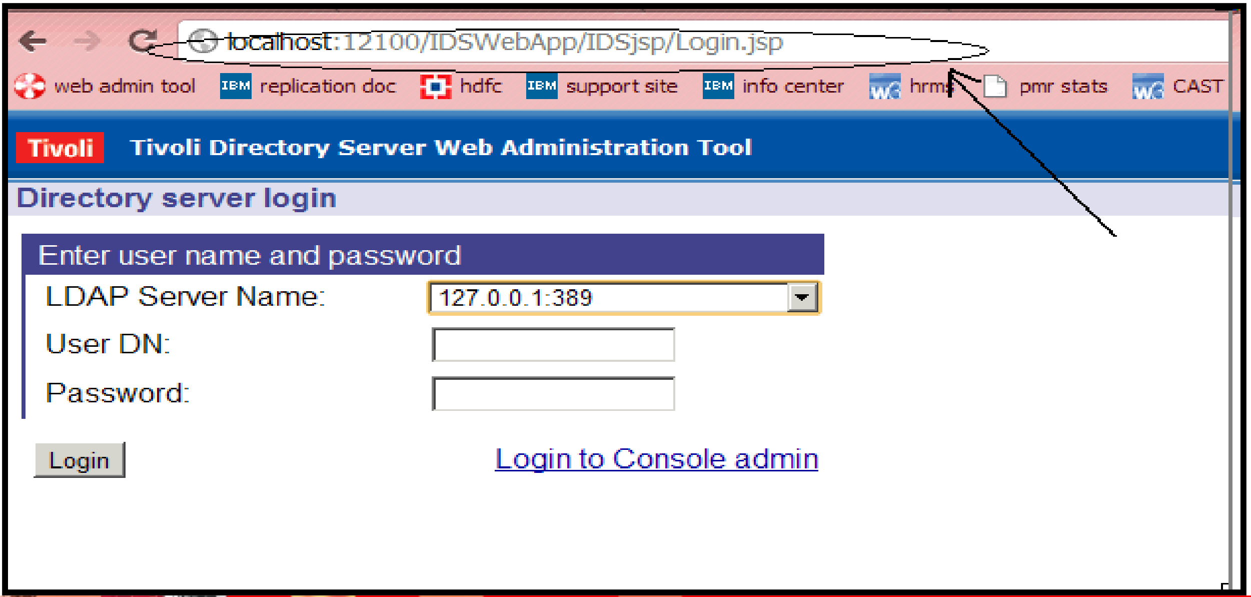
On AIX, Linux, and Solaris systems,
installpath/idstools/bin/startWebadminApp

where *installpath* is the path where you installed Tivoli Directory Server

Configuration of the Web Admin Tool

After the Web application server is started, you can start the Web

administration tool by :
From a Web browser, type the following address:
http://localhost:12100/IDSWebApp/


localhost:12100/IDSWebApp/IDSjsp/Login.jsp
web admin tool
replication doc
hdfc
support site
info center
hrms
pmr stats
CAST
Tivoli
Tivoli Directory Server Web Administration Tool

Directory server login

Enter user name and password

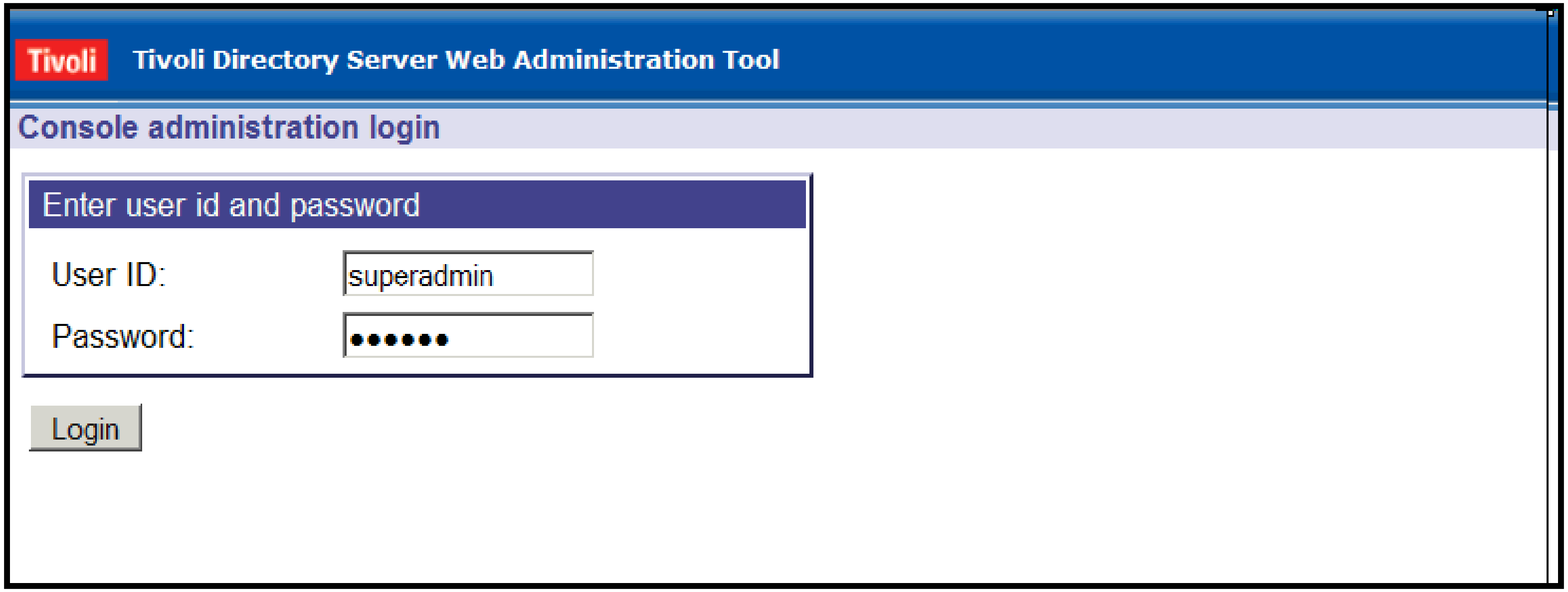
LDAP Server Name: 127.0.0.1:389

User DN:

Password:

Configuration of the Web Admin Tool (Contd.)

The initial default login to the Web Admin console is "superadmin" with the password "secret"



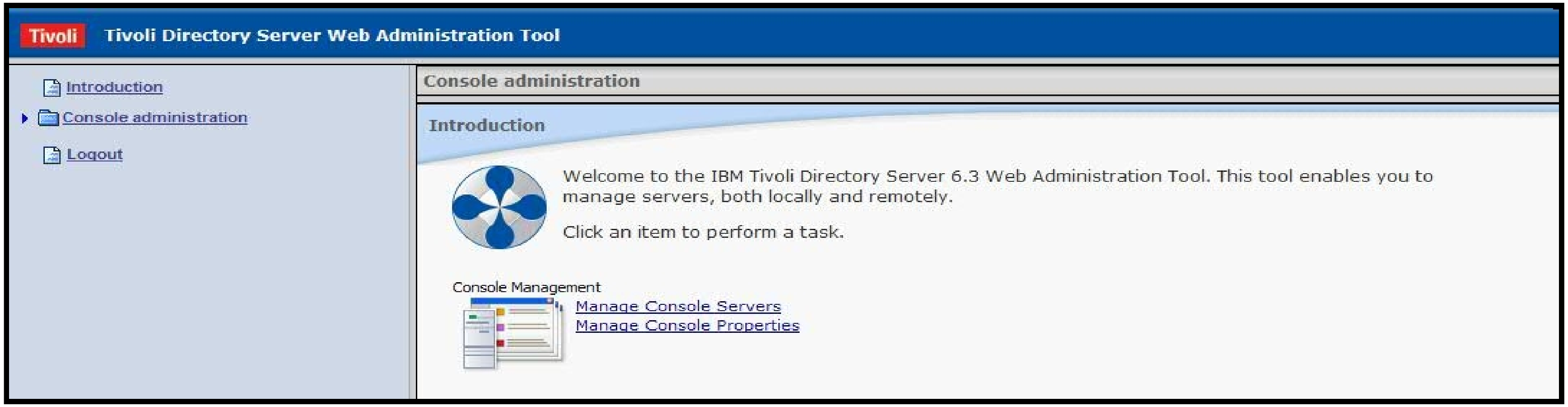
Console administration login

Enter user id and password

User ID: superadmin

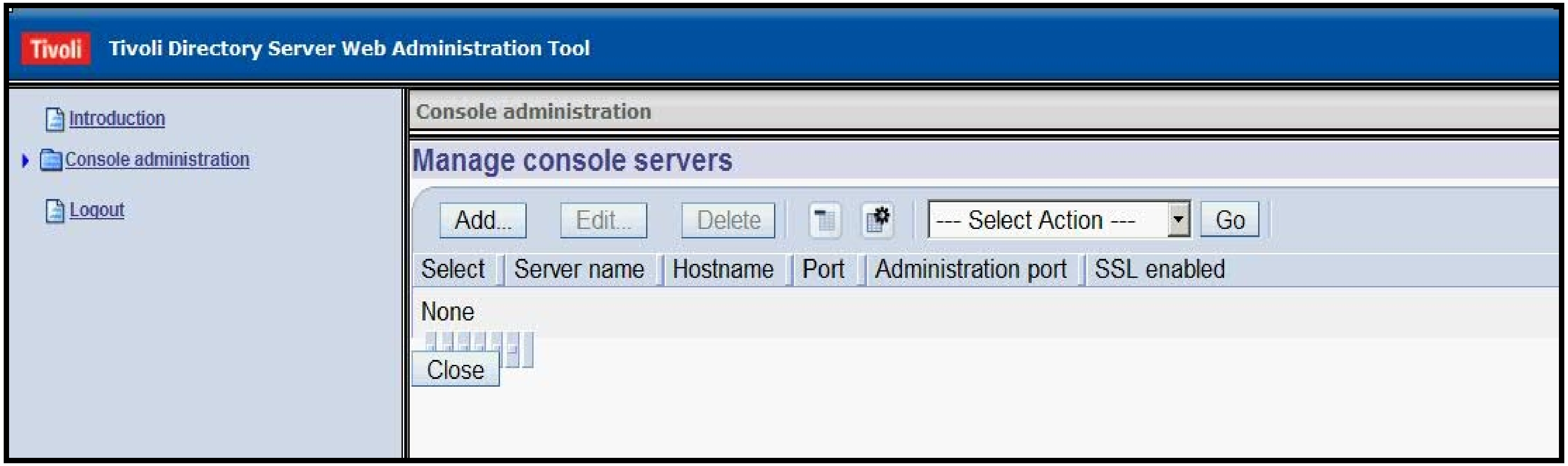
Password: ●●●●●●

Configuration of the Web Admin Tool (Contd.)



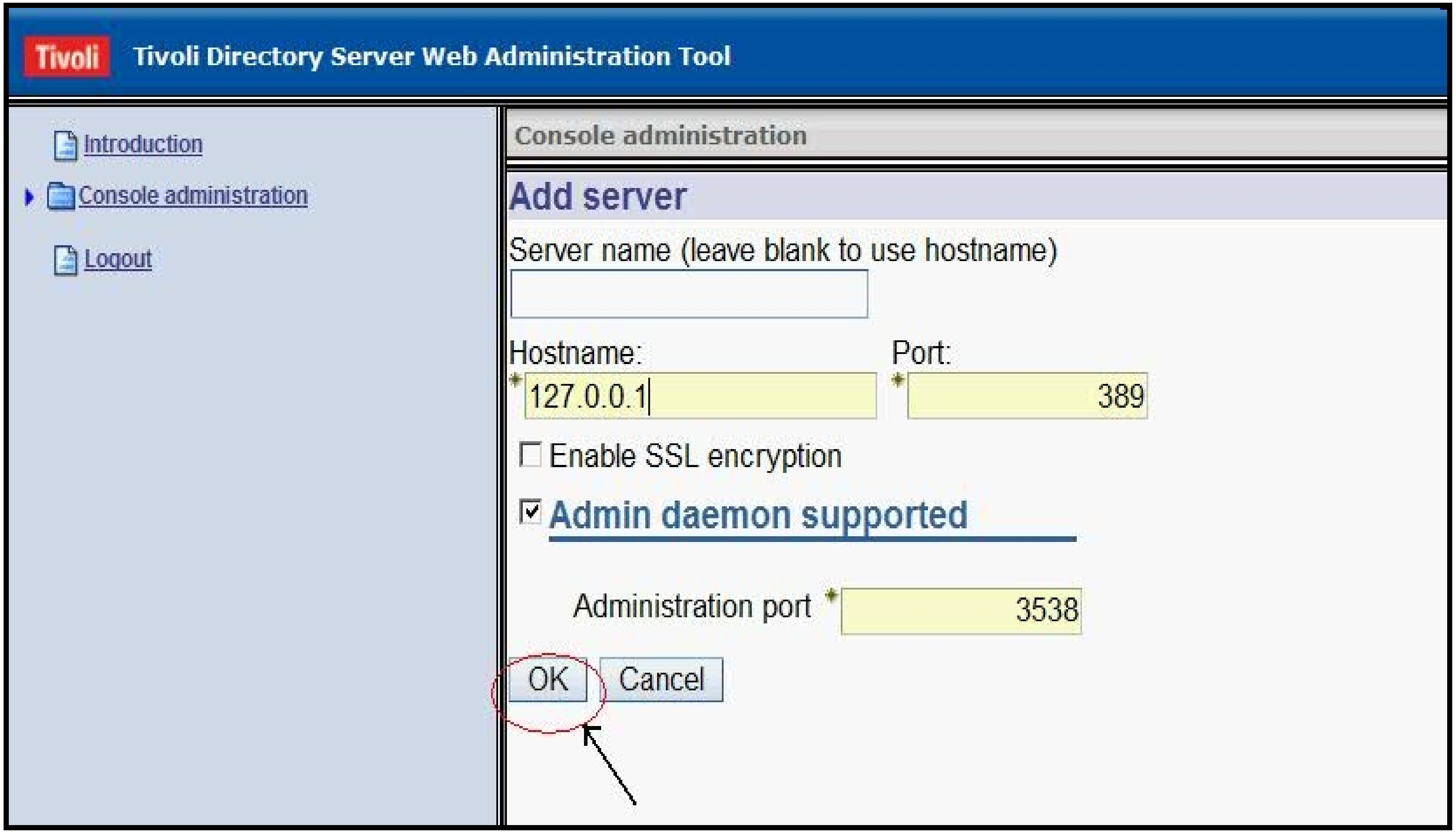
Welcome to the IBM Tivoli Directory Server 6.3 Web Administration Tool. This tool enables you to manage servers, both locally and remotely.

Click an item to perform a task.



Select	Server name	Hostname	Port	Administration port	SSL enabled
None					

Configuration of the Web Admin Tool (Contd.)



Add server

Server name (leave blank to use hostname)

Hostname: * 127.0.0.1

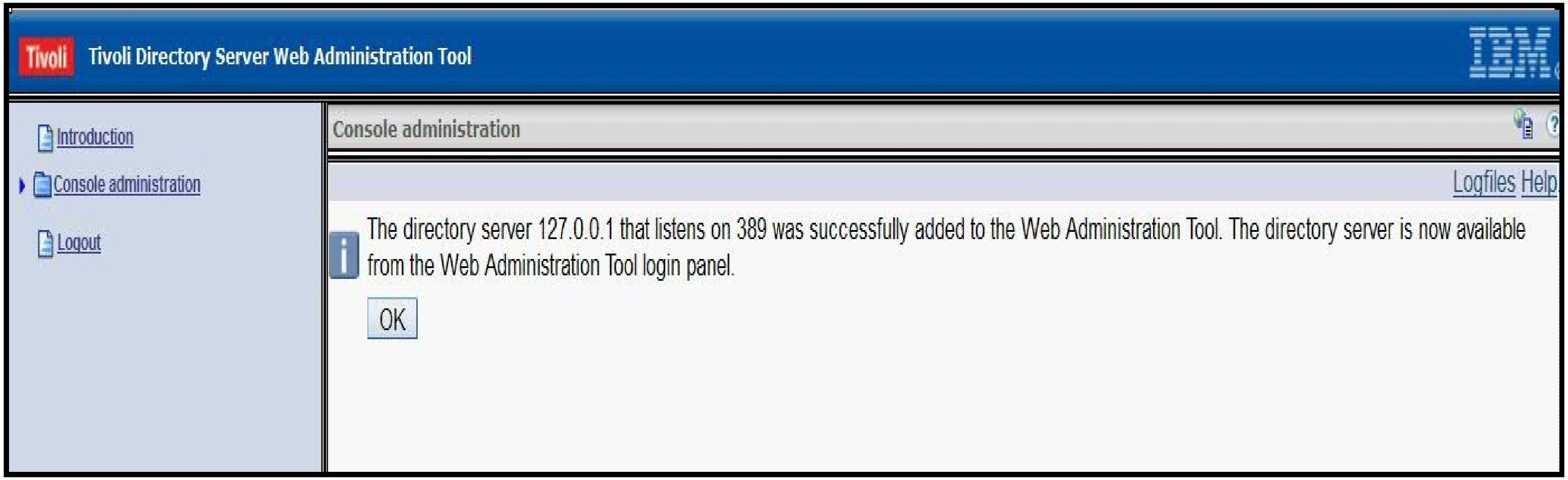
Port: * 389

☐ Enable SSL encryption

☑ Admin daemon supported

Administration port * 3538

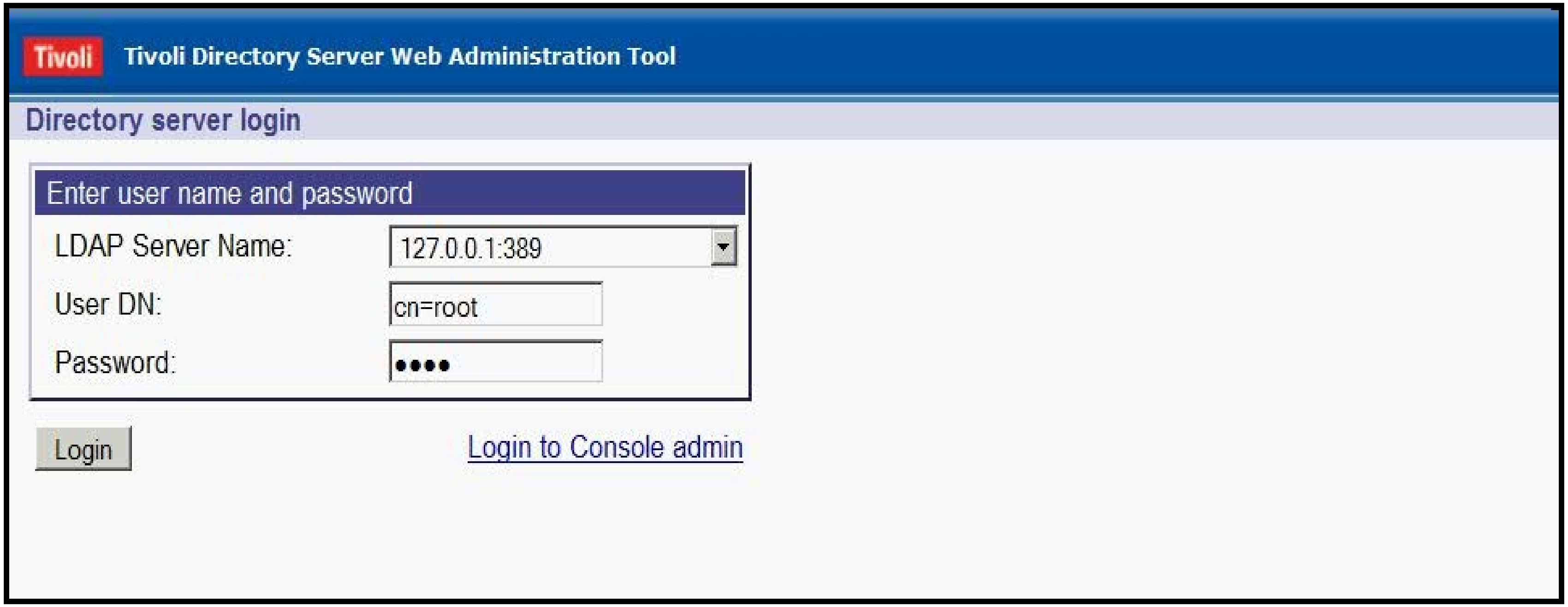
Configuration of the Web Admin Tool (Contd.)



The directory server 127.0.0.1 that listens on 389 was successfully added to the Web Administration Tool. The directory server is now available from the Web Administration Tool login panel.

Logging into directory instance via Web Admin Tool

You will notice now that in the LDAP server name dropdown menu, the server we added would be visible :



Tivoli
Tivoli Directory Server Web Administration Tool
Directory server login
Enter user name and password

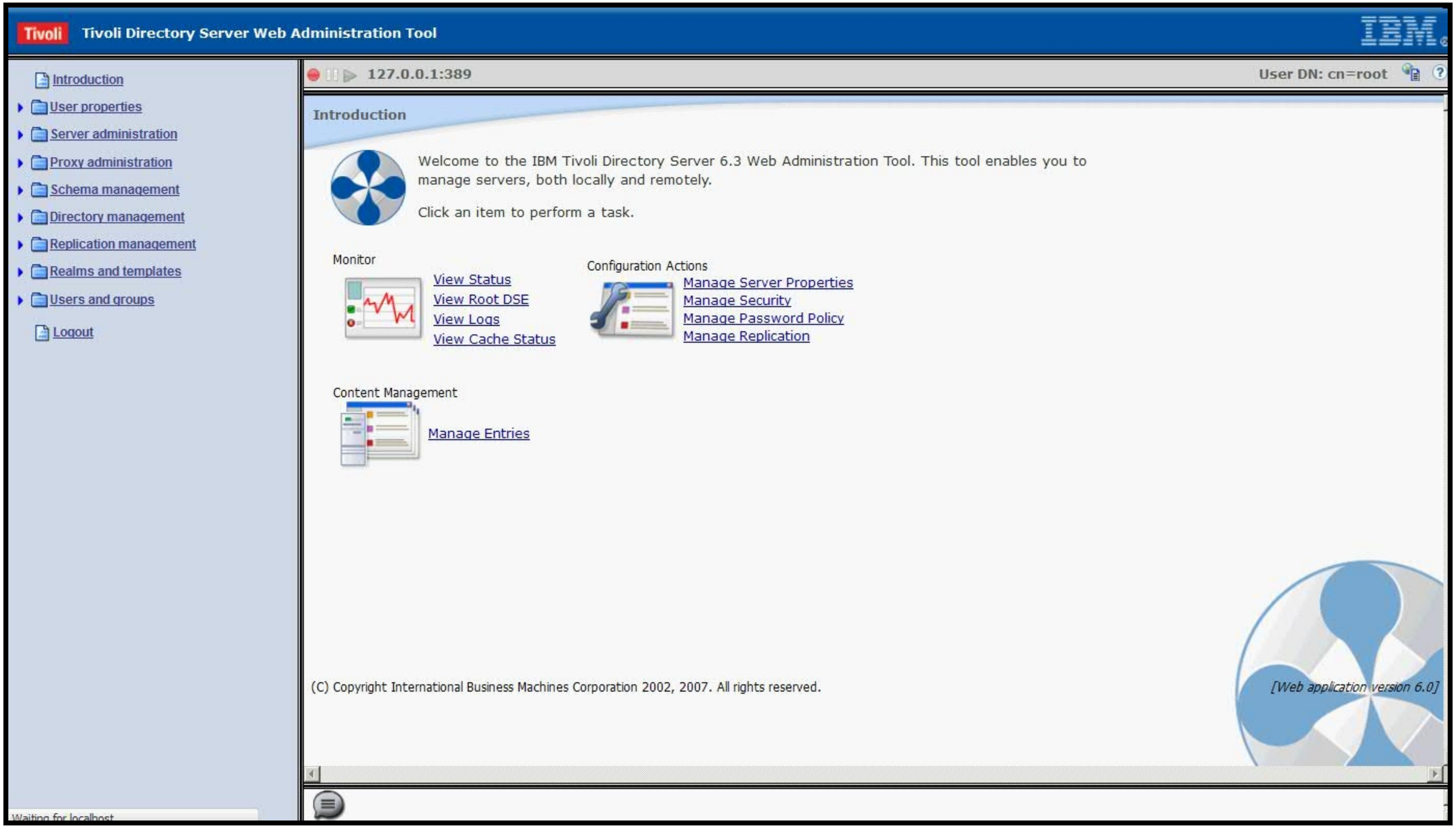
LDAP Server Name: 127.0.0.1:389

User DN: cn=root

Password: ●●●●

Login
Login to Console admin

What the ITDS Web Admin Tool looks like?



Welcome to the IBM Tivoli Directory Server 6.3 Web Administration Tool. This tool enables you to manage servers, both locally and remotely.

Click an item to perform a task.

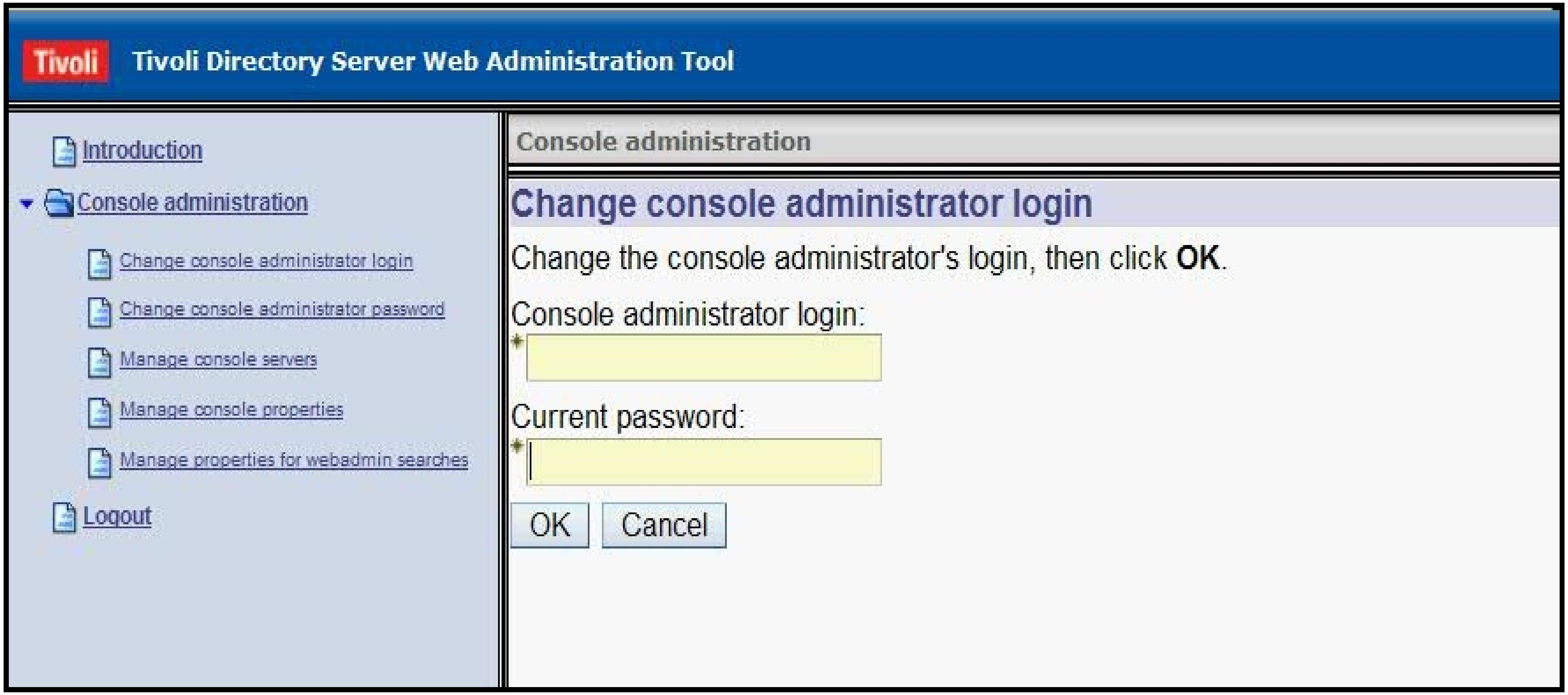
(C) Copyright International Business Machines Corporation 2002, 2007. All rights reserved.

[Web application version 6.0]

Waiting for localhost

Managing the console

Changing the console administrator login



Tivoli
Tivoli Directory Server Web Administration Tool
Introduction
Console administration
Change console administrator login
Change console administrator password
Manage console servers
Manage console properties
Manage properties for webadmin searches
Logout
Console administration

Change console administrator login

Change the console administrator's login, then click **OK**.

Console administrator login:

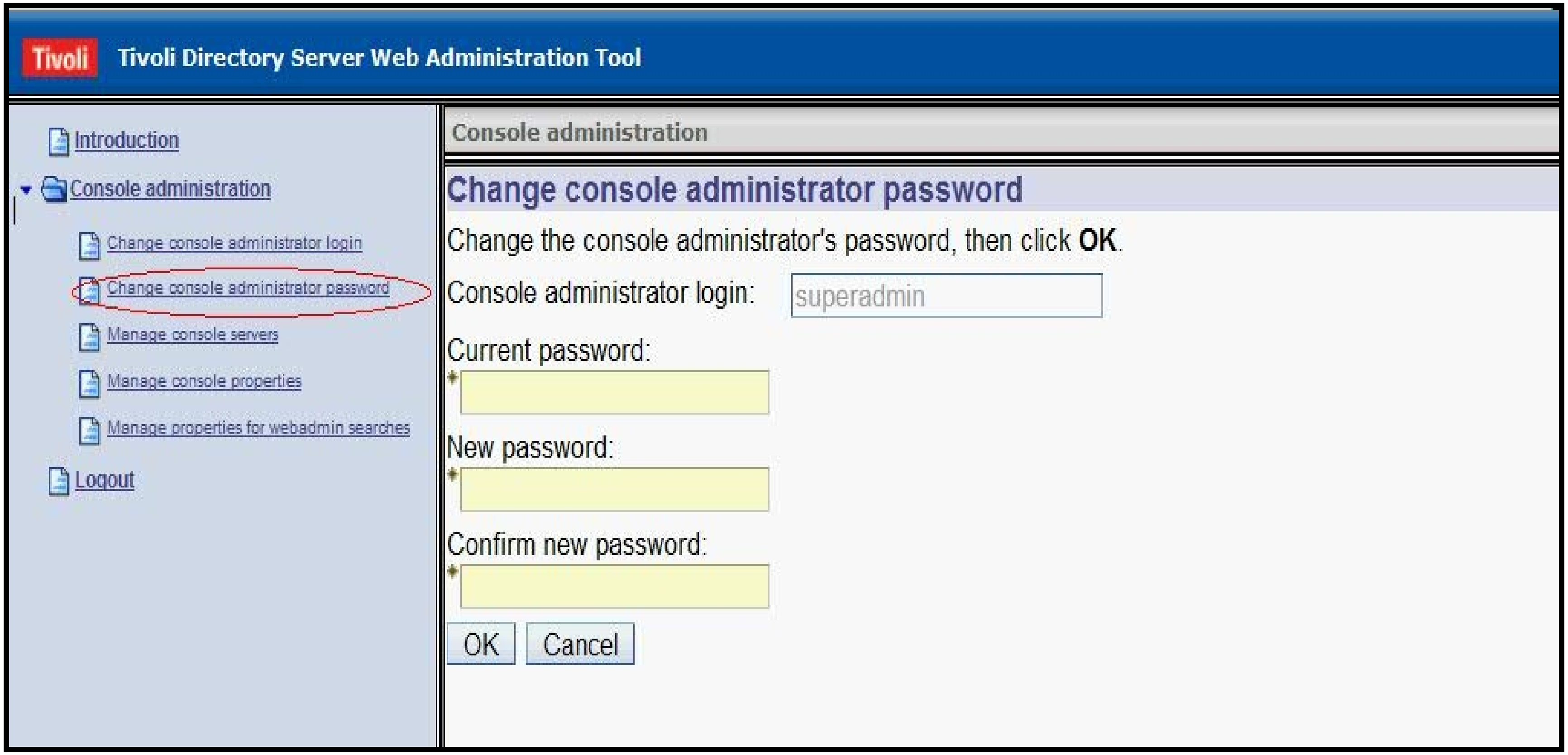
*

Current password:

*

Managing the console (Contd.)

Changing the console administrator password



Tivoli Tivoli Directory Server Web Administration Tool

- Introduction
- Console administration
  - Change console administrator login
  - Change console administrator password
  - Manage console servers
  - Manage console properties
  - Manage properties for webadmin searches
- Logout

Console administration

Change console administrator password

Change the console administrator's password, then click **OK**.

Console administrator login: superadmin

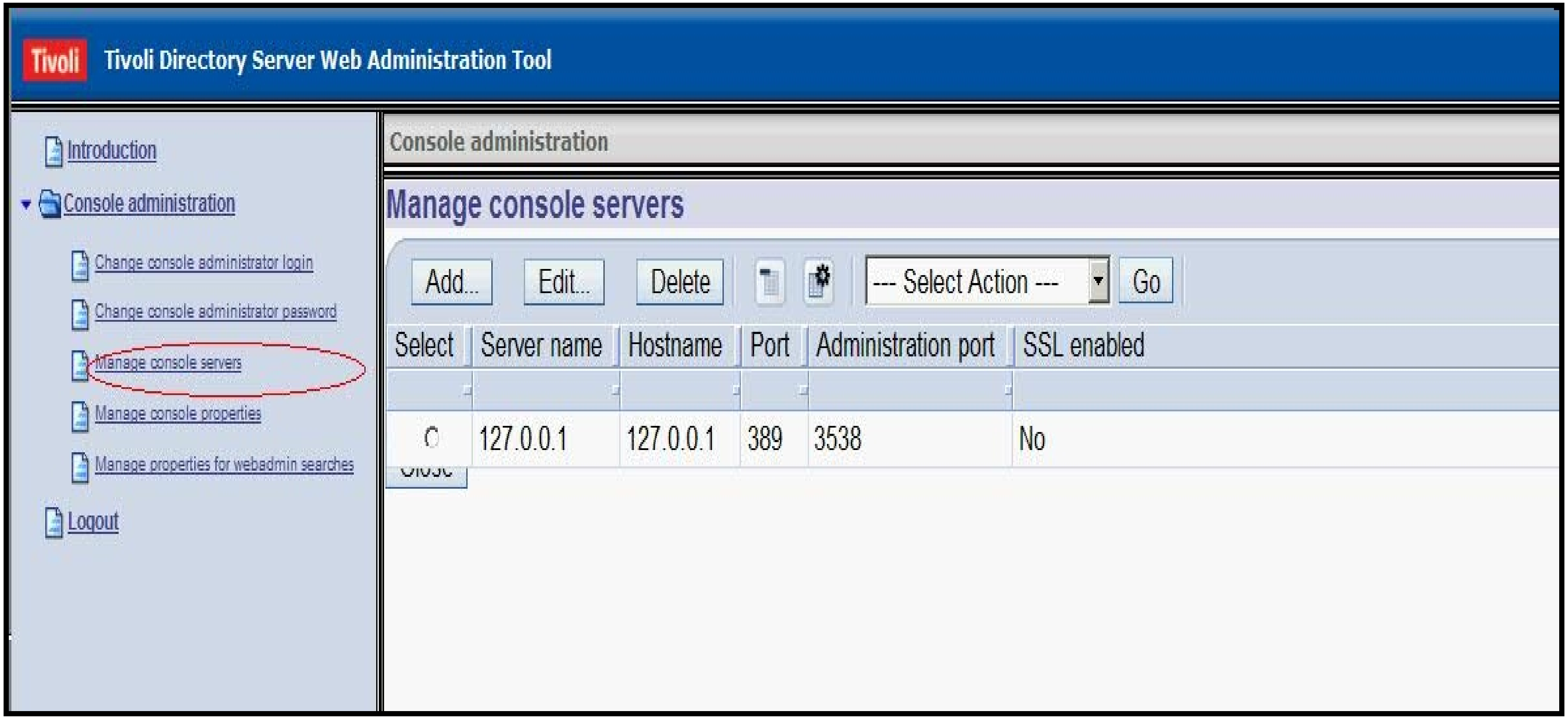
Current password: *

New password: *

Confirm new password: *

Managing the console (Contd.)

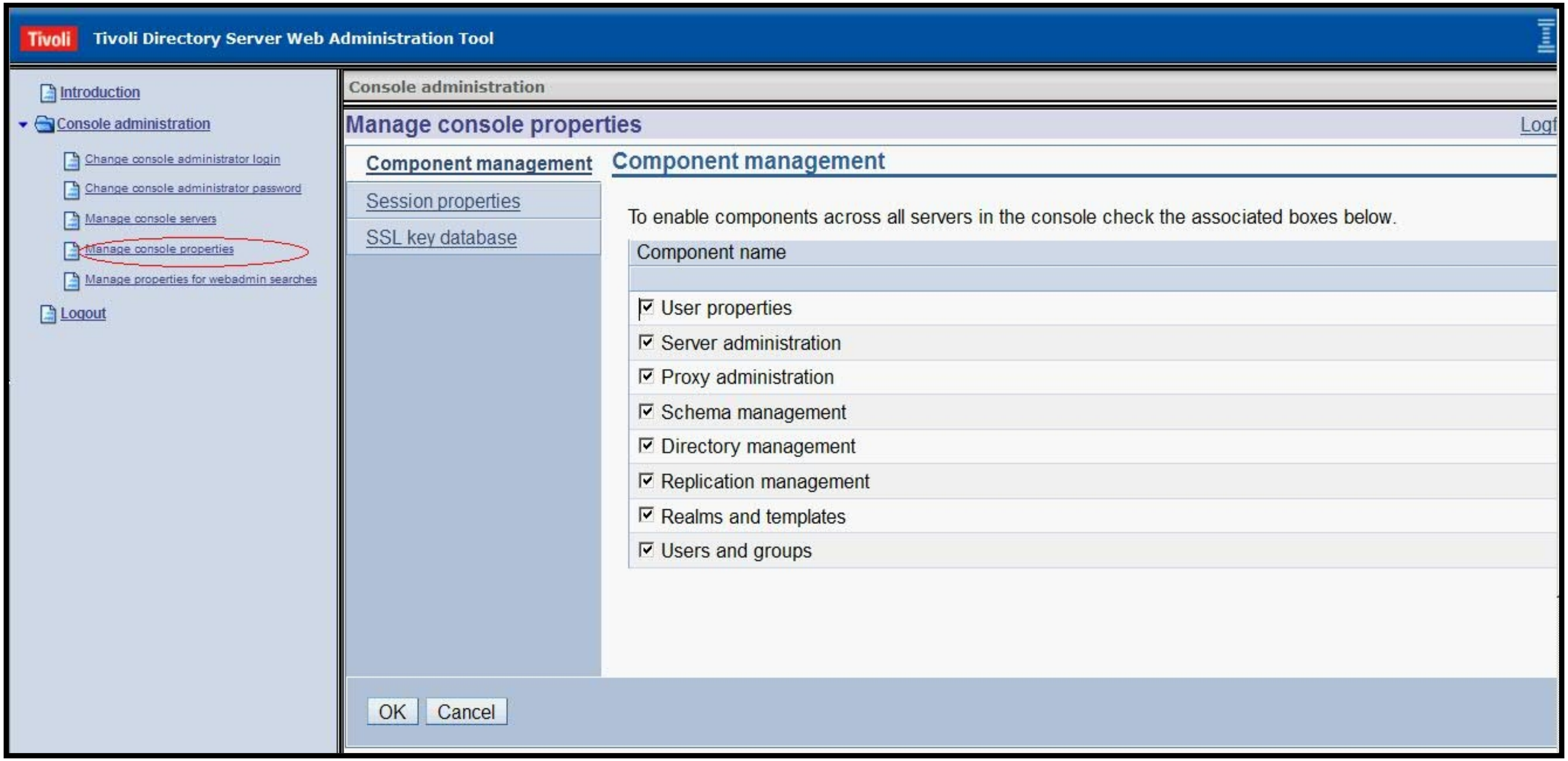
Manage console servers


Tivoli
Tivoli Directory Server Web Administration Tool
Introduction
Console administration
Change console administrator login
Change console administrator password
Manage console servers
Manage console properties
Manage properties for webadmin searches
Logout
Console administration
Manage console servers
Add...
Edit...
Delete
--- Select Action ---
Go

Select	Server name	Hostname	Port	Administration port	SSL enabled
○	127.0.0.1	127.0.0.1	389	3538	No

Managing the console (Contd.)

Manage console properties


Tivoli
Tivoli Directory Server Web Administration Tool
Introduction
Console administration
Change console administrator login
Change console administrator password
Manage console servers
Manage console properties
Manage properties for webadmin searches
Logout
Console administration
Manage console properties
Logf
Component management
Session properties
SSL key database
OK
Cancel

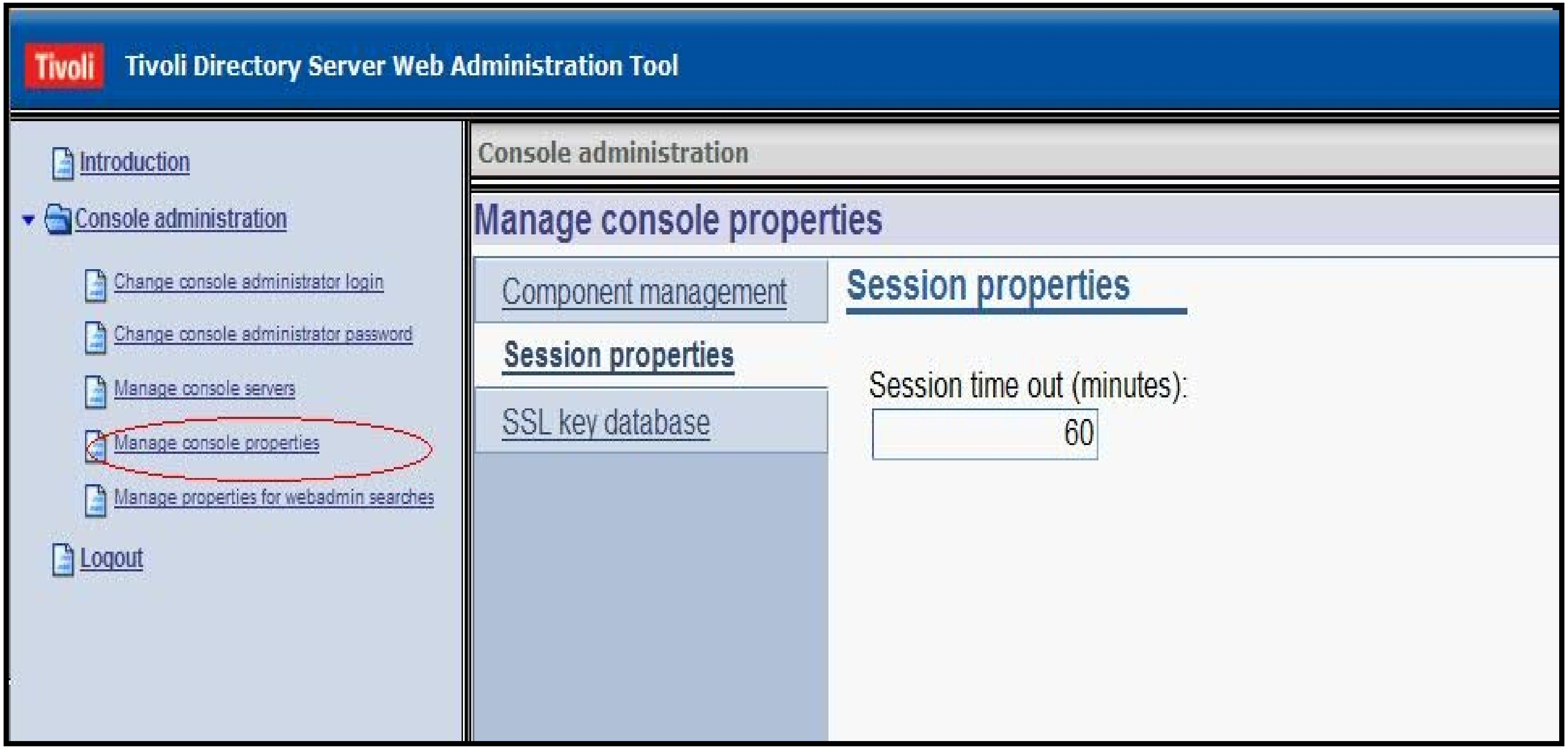
Component management

To enable components across all servers in the console check the associated boxes below.

Component name
☑ User properties
☑ Server administration
☑ Proxy administration
☑ Schema management
☑ Directory management
☑ Replication management
☑ Realms and templates
☑ Users and groups

Managing the console (Contd.)

Manage Console properties (Session properties)

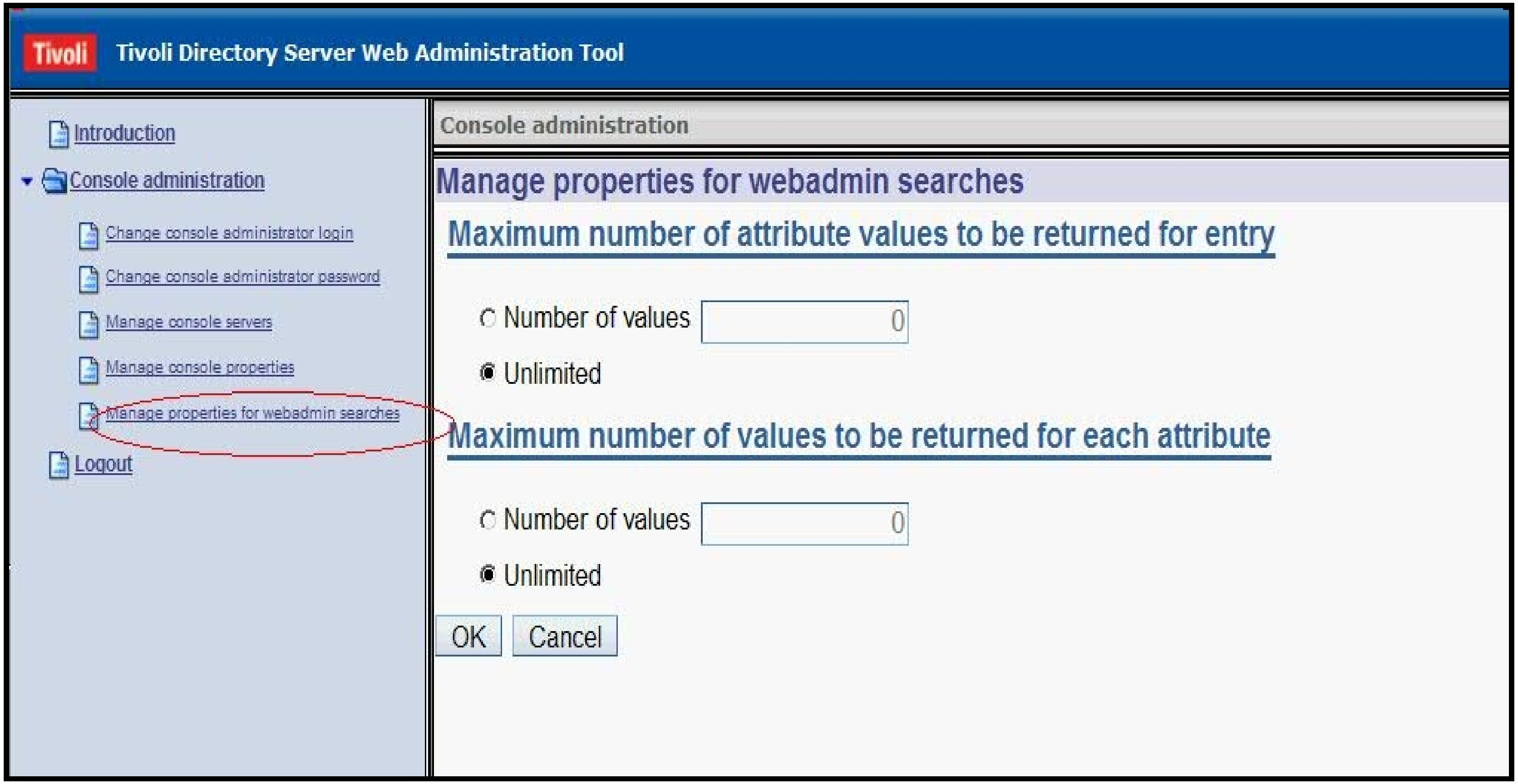


Tivoli
Tivoli Directory Server Web Administration Tool
Introduction
Console administration
Change console administrator login
Change console administrator password
Manage console servers
Manage console properties
Manage properties for webadmin searches
Logout
Console administration
Manage console properties
Component management
Session properties
SSL key database
Session properties

Session time out (minutes): 60

Managing the console (Contd.)

Manage properties for Web admin searches



Tivoli Directory Server Web Administration Tool

Tivoli
Introduction
Console administration
Change console administrator login
Change console administrator password
Manage console servers
Manage console properties
Manage properties for webadmin searches
Logout
Console administration

Manage properties for webadmin searches

Maximum number of attribute values to be returned for entry

○ Number of values 0

● Unlimited

Maximum number of values to be returned for each attribute

○ Number of values 0

● Unlimited

Overview of SSL

- The IBM Tivoli Directory Server has the ability to protect LDAP access by encrypting data with either Secure Sockets Layer (SSL) security or Transport Layer Security (TLS) or both.
- When using SSL or TLS to secure LDAP communications with the IBM Directory, both server authentication and client authentication are supported.
- To use SSL or TLS you must have GSKit installed on your system. Before you can use SSL or TLS you must first use GSKit to create the key database file and certificates.

Security Goals of SSL

As it relates to the actual data being communicated

- **Confidentiality**
  - Protection from disclosure to unauthorized recipients
- **Integrity**
  - Maintaining the message consistency
- **Authenticity**
  - Assurance of identity of originator of the message

GSKIT Commands

- **gsk8capicmd** - a tool that can be used to manage keys, certificates and certificate requests within a CMS key database. The tool has all of the functionality that GSKit's existing java command line tool has except that GSKCapiCmd supports CMS and PKCS11 key databases. If you are intending to manage key databases other than CMS or PKCS11 you will need to use the existing java tool.
- **gsk8cmd** – a command line java based tool that can be used to manage keys, certificates and certificate requests for various types of key stores. You must enable gskit to support CMS key databases before you can use this command to create a CMS key database.
- **gsk8ikm** – a java based gui tool that can be used to manage keys, certificates and certificate requests for various types of key stores. You must enable gskit to support CMS key databases before you can use this command to create a CMS key database.

Prerequisites

- ITDS version 6.3 must be installed and should be updated to the latest level
- GSKIT must be installed and should be updated to the latest supported level
- At minimum one server instance created and configured.

Required ITDS Packages on AIX for SSL

In order to configure SSL communication on AIX, you must have the max_crypto ITDS filesets installed.

Required GSKIT packages on AIX for SSL

You must install both the 64 bit GSKIT packages, gsksa.rte and gskta.rte.

Configuring Server Authentication

Configuring Server Authentication using command line utilities on AIX 6.1 with ITDS 6.3

- Create a key database using gsk8capicmd
- Create a self-signed certificate using gsk8capicmd
- Command line configuration of server Authentication
- Client configuration
- Verify ssl communication between the Client and Server using server authentication

Configuring Server Authentication

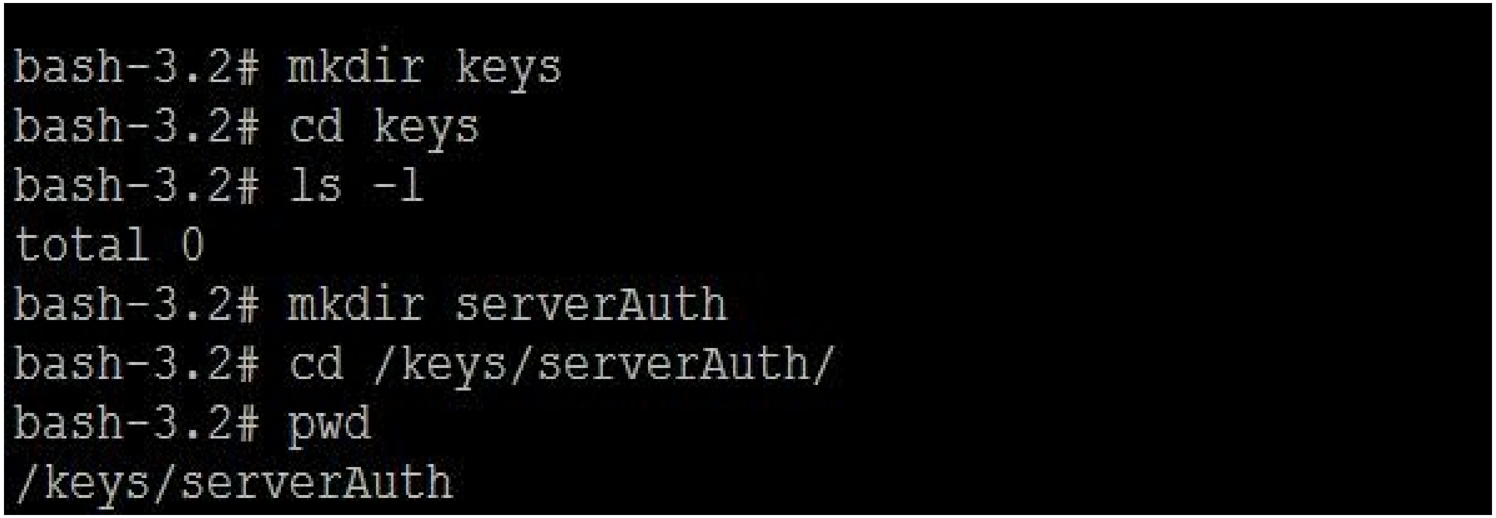
Server authentication

For server authentication the IBM Tivoli Directory Server supplies the client with the IBM Tivoli Directory Server's X.509 certificate during the initial SSL handshake. If the client validates the server's certificate, then a secure, encrypted communication channel is established between the IBM Tivoli Directory Server and the client application.

****Note:** For server authentication to work, the IBM Tivoli Directory Server must have a private key and associated server certificate in the server's key database file. The client must have the certificate of the signer of the server's certificate present in the client key database file, along with all certificates of the signer chain up to a trusted root.

Configuring Server Authentication

Create a subdirectory on your server system where you want to create and store the key database file.



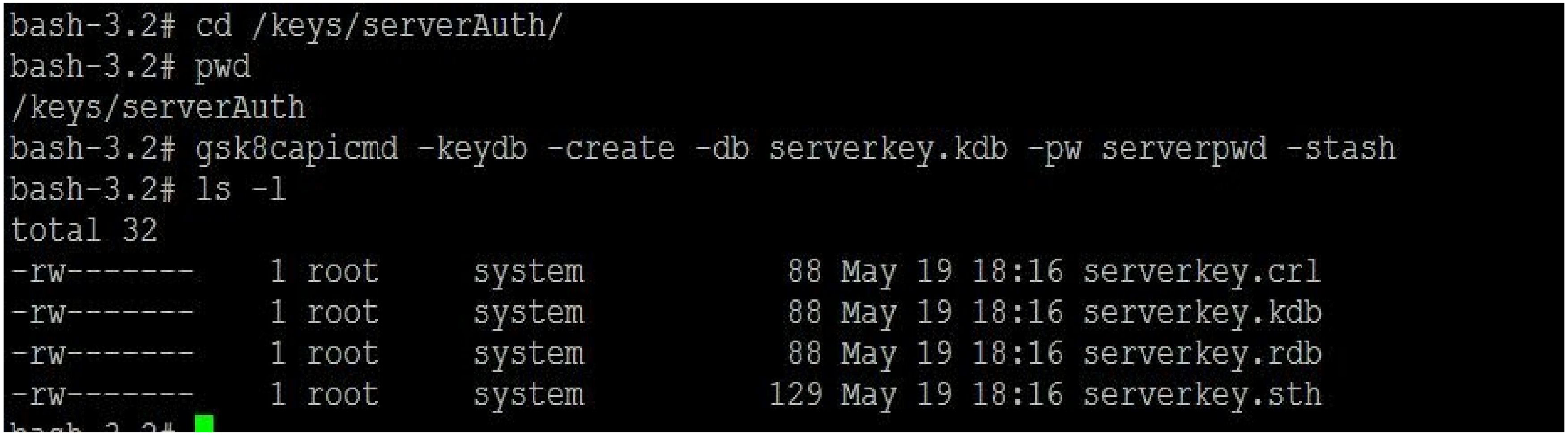
```
bash-3.2# mkdir keys
bash-3.2# cd keys
bash-3.2# ls -l
total 0
bash-3.2# mkdir serverAuth
bash-3.2# cd /keys/serverAuth/
bash-3.2# pwd
/keys/serverAuth
```

Configuring Server Authentication

Generate the CMS key database to be used by the ldap server:

ex: gsk8capicmd –keydb –create –db <keydb name> -pw <keydb password> - stash

For this example I created a key database called “serverkey.kdb” and gave it the password “serverpwd”.

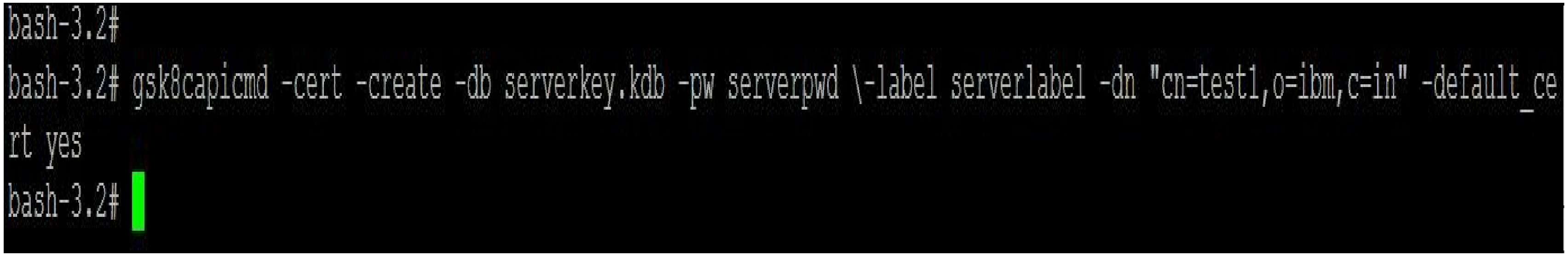


```
bash-3.2# cd /keys/serverAuth/
bash-3.2# pwd
/keys/serverAuth
bash-3.2# gsk8capicmd -keydb -create -db serverkey.kdb -pw serverpwd -stash
bash-3.2# ls -l
total 32
-rw-------    1 root     system           88 May 19 18:16 serverkey.crl
-rw-------    1 root     system           88 May 19 18:16 serverkey.kdb
-rw-------    1 root     system           88 May 19 18:16 serverkey.rdb
-rw-------    1 root     system          129 May 19 18:16 serverkey.sth
bash-3.2#
```

Configuring Server Authentication

Create a default self-signed certificate and add it to the serverkey.kdb key database.

ex: gsk8capicmd -cert -create -db <keydb name> -pw <keydb password> -label <certificate label> -dn <distinguished name> -default_cert <yes | no> -expire <# of days>

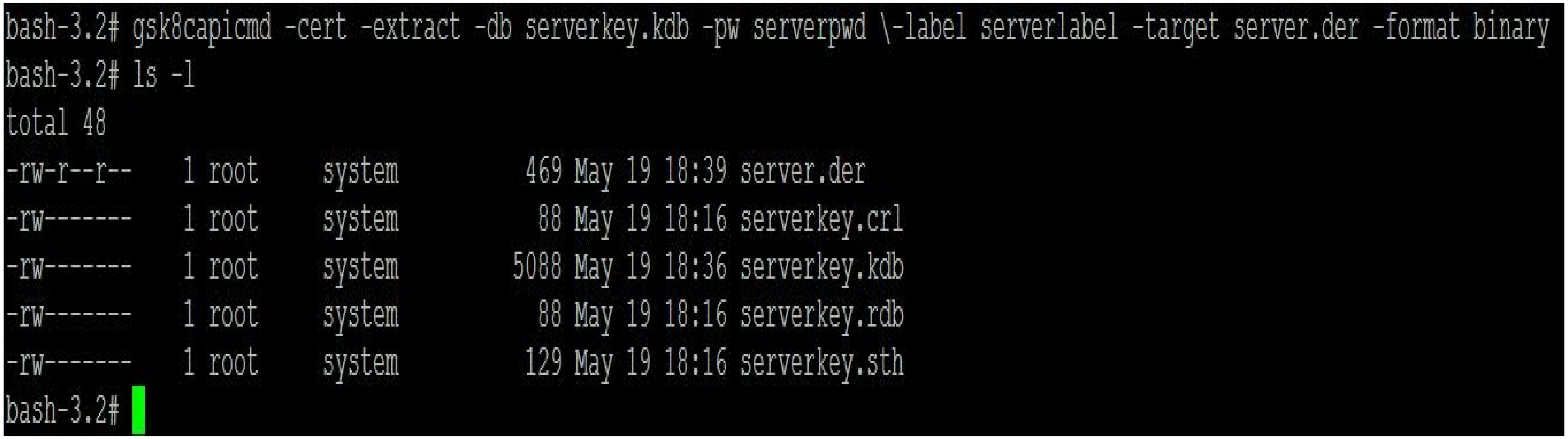


```
bash-3.2#
bash-3.2# gsk8capicmd -cert -create -db serverkey.kdb -pw serverpwd \-label serverlabel -dn "cn=test1,o=ibm,c=in" -default_ce
rt yes
bash-3.2#
```

Configuring Server Authentication

Extract the certificate from the key database in binary der format. We will use this extracted certificate during the client configuration later on.

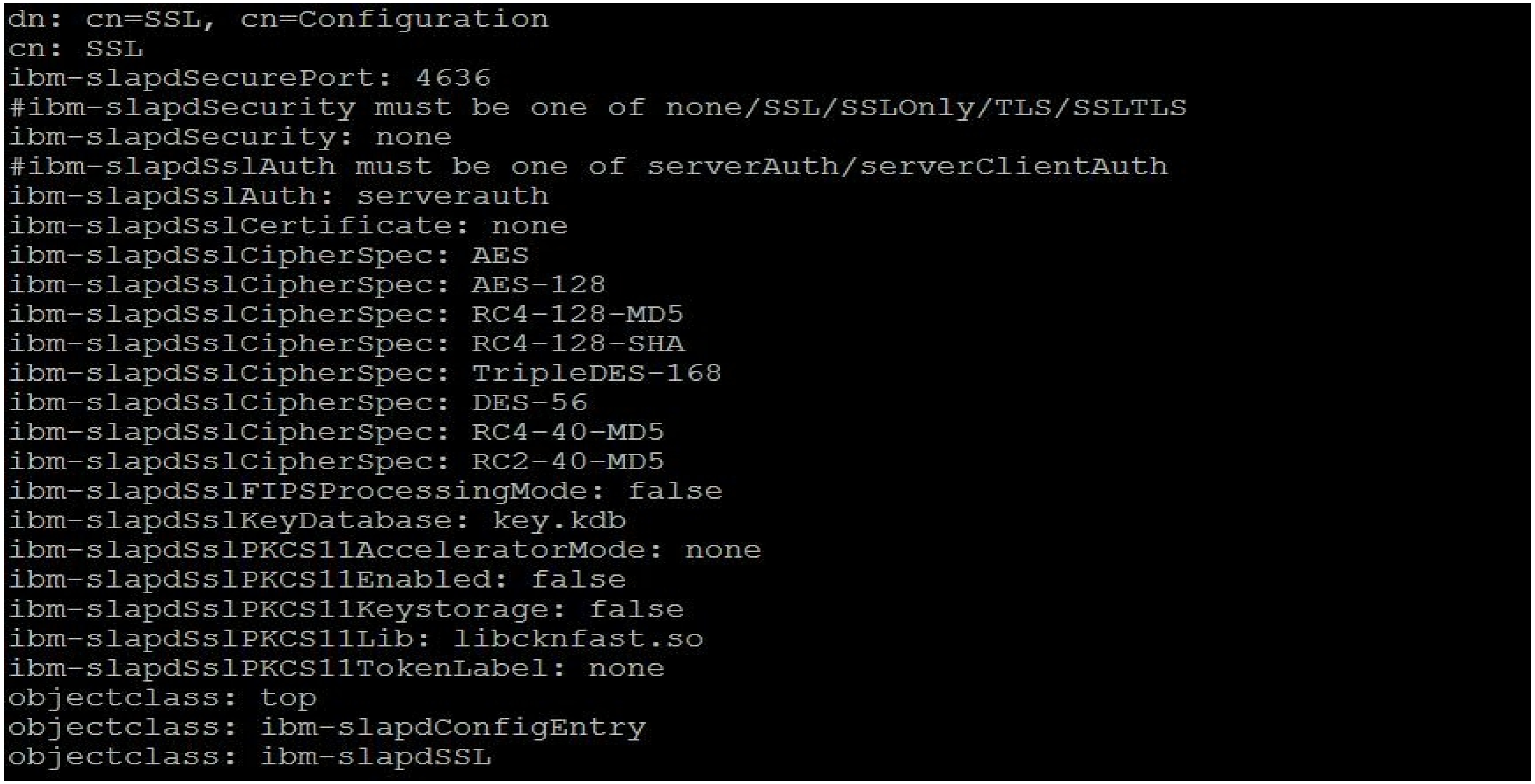
ex:gsk8capicmd -cert -extract -db <keydb name> -pw <keydb password> -label <certificate label> -target <destination file> -format <format of certificate>



```
bash-3.2# gsk8capicmd -cert -extract -db serverkey.kdb -pw serverpwd \-label serverlabel -target server.der -format binary
bash-3.2# ls -l
total 48
-rw-r--r--    1 root     system          469 May 19 18:39 server.der
-rw-------    1 root     system           88 May 19 18:16 serverkey.crl
-rw-------    1 root     system         5088 May 19 18:36 serverkey.kdb
-rw-------    1 root     system           88 May 19 18:16 serverkey.rdb
-rw-------    1 root     system          129 May 19 18:16 serverkey.sth
bash-3.2# 
```

Configuring Server Authentication

SSL stanza in the ibmslapd.conf file is by default configured as follows:



```
dn: cn=SSL, cn=Configuration
cn: SSL
ibm-slapdSecurePort: 4636
#ibm-slapdSecurity must be one of none/SSL/SSLOnly/TLS/SSLTLS
ibm-slapdSecurity: none
#ibm-slapdSslAuth must be one of serverAuth/serverClientAuth
ibm-slapdSslAuth: serverauth
ibm-slapdSslCertificate: none
ibm-slapdSslCipherSpec: AES
ibm-slapdSslCipherSpec: AES-128
ibm-slapdSslCipherSpec: RC4-128-MD5
ibm-slapdSslCipherSpec: RC4-128-SHA
ibm-slapdSslCipherSpec: TripleDES-168
ibm-slapdSslCipherSpec: DES-56
ibm-slapdSslCipherSpec: RC4-40-MD5
ibm-slapdSslCipherSpec: RC2-40-MD5
ibm-slapdSslFIPSProcessingMode: false
ibm-slapdSslKeyDatabase: key.kdb
ibm-slapdSslPKCS11AcceleratorMode: none
ibm-slapdSslPKCS11Enabled: false
ibm-slapdSslPKCS11Keystorage: false
ibm-slapdSslPKCS11Lib: libcknfast.so
ibm-slapdSslPKCS11TokenLabel: none
objectclass: top
objectclass: ibm-slapdConfigEntry
objectclass: ibm-slapdSSL
```

Configuring Server Authentication

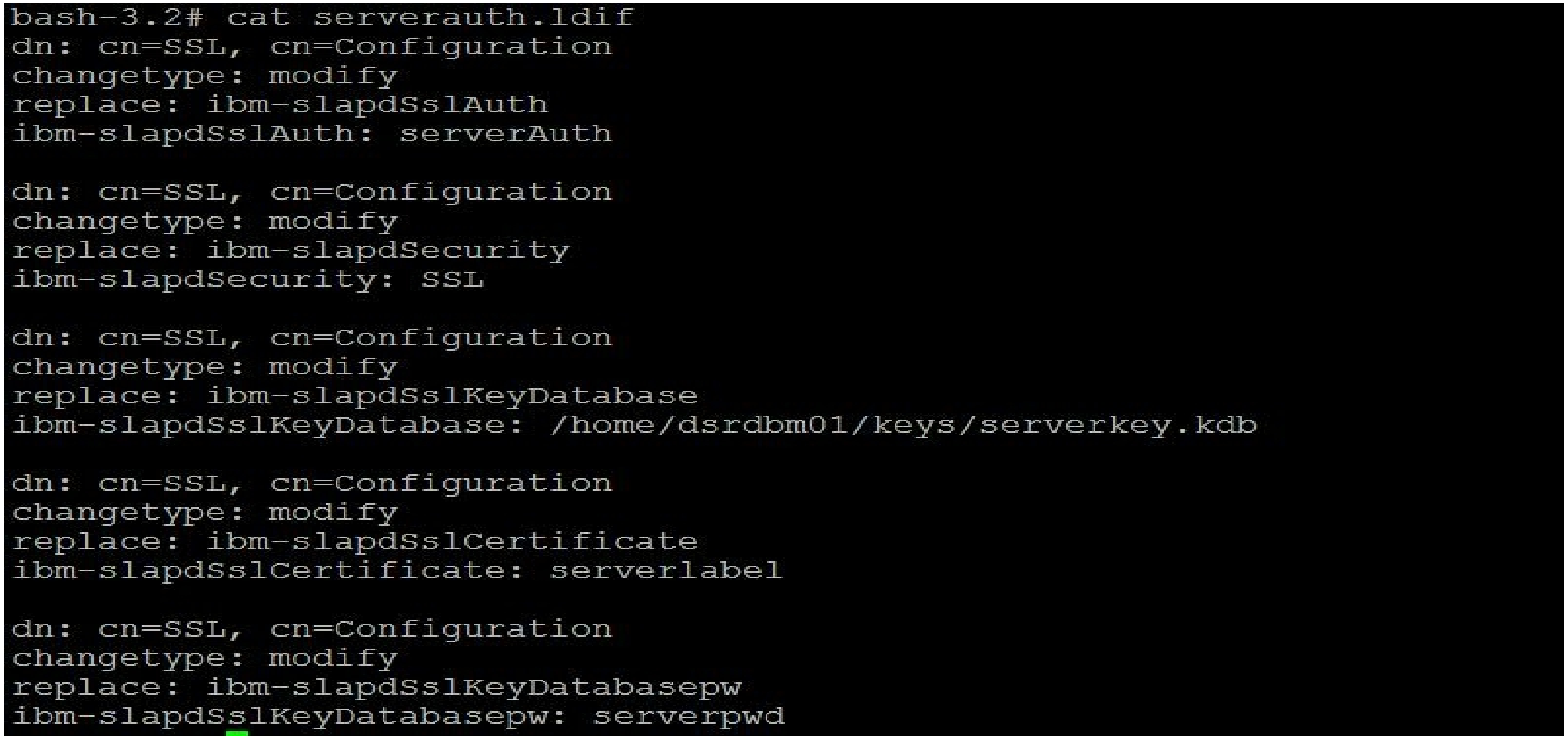
We have a few options for configuring the SSL portion of the ibmslapd.conf file.

- Manually edit the ibmslapd.conf
- Use an idsldapmodify command to update the conf file
- The Web Administration tool

For this example we will use the idsldapmodify option.

Configuring Server Authentication

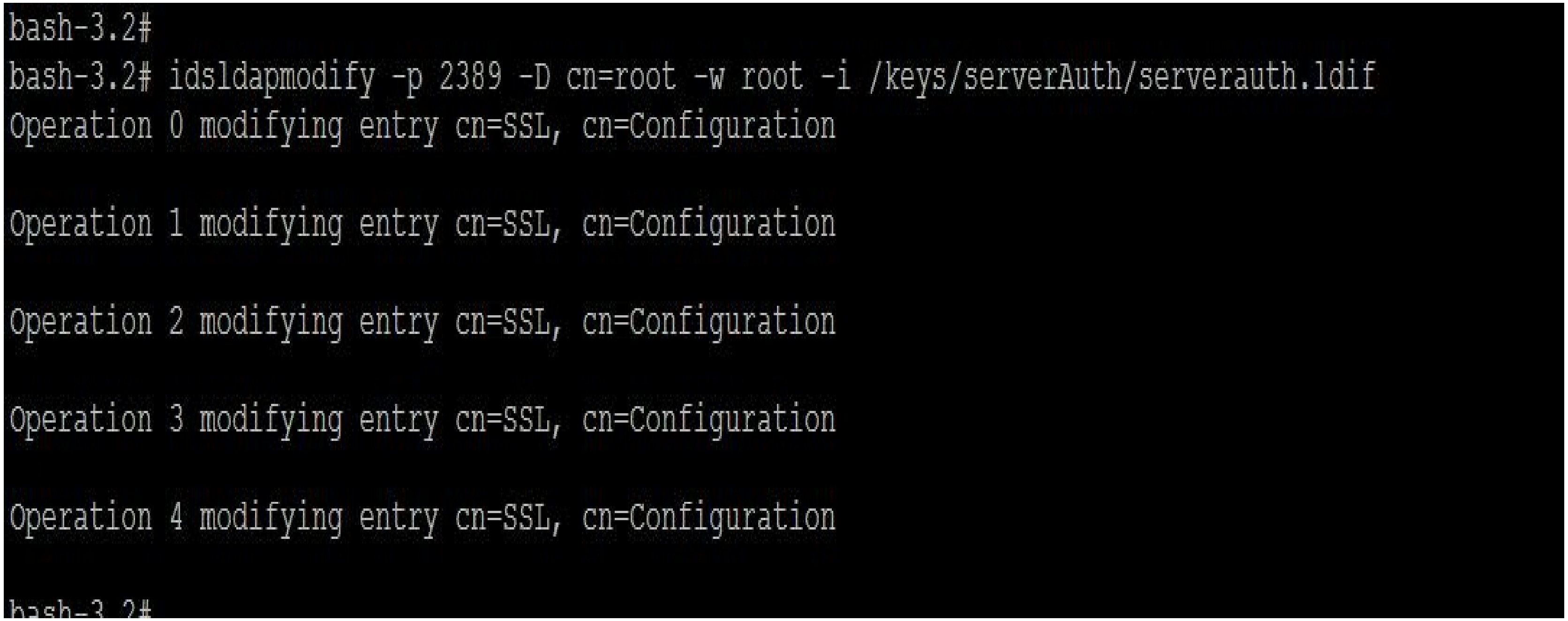
serverauth.ldif file that contains the update you need to make using the idsldapmodify command



```
bash-3.2# cat serverauth.ldif
dn: cn=SSL, cn=Configuration
changetype: modify
replace: ibm-slapdSslAuth
ibm-slapdSslAuth: serverAuth

dn: cn=SSL, cn=Configuration
changetype: modify
replace: ibm-slapdSecurity
ibm-slapdSecurity: SSL

dn: cn=SSL, cn=Configuration
changetype: modify
replace: ibm-slapdSslKeyDatabase
ibm-slapdSslKeyDatabase: /home/dsrdbm01/keys/serverkey.kdb

dn: cn=SSL, cn=Configuration
changetype: modify
replace: ibm-slapdSslCertificate
ibm-slapdSslCertificate: serverlabel

dn: cn=SSL, cn=Configuration
changetype: modify
replace: ibm-slapdSslKeyDatabasepw
ibm-slapdSslKeyDatabasepw: serverpwd
```

Configuring Server Authentication

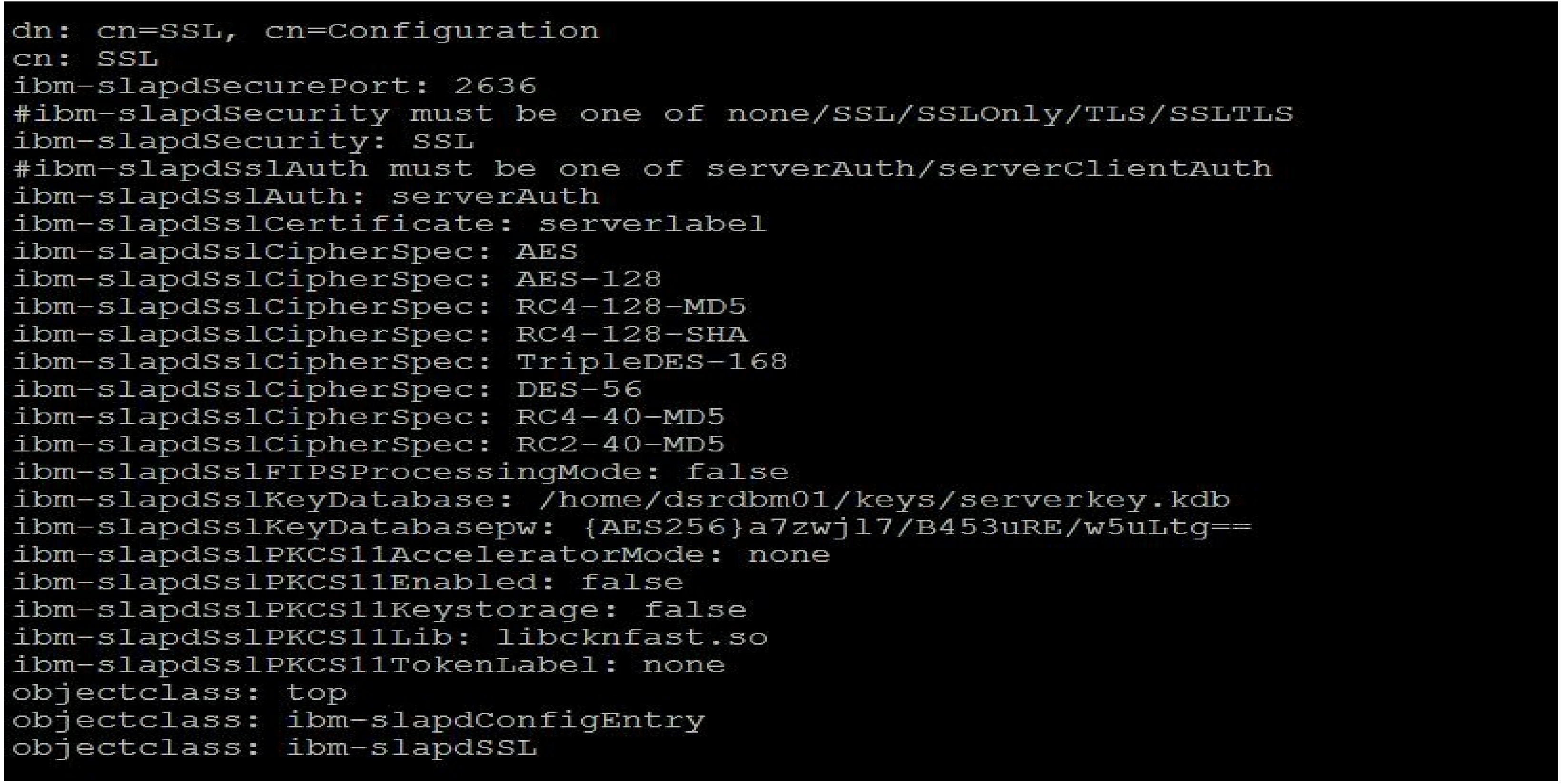
Using the idsldapmodify command to update the ibmslapd.conf file (server must be started to run this command)



```
bash-3.2#
bash-3.2# idsldapmodify -p 2389 -D cn=root -w root -i /keys/serverAuth/serverauth.ldif
Operation 0 modifying entry cn=SSL, cn=Configuration

Operation 1 modifying entry cn=SSL, cn=Configuration

Operation 2 modifying entry cn=SSL, cn=Configuration

Operation 3 modifying entry cn=SSL, cn=Configuration

Operation 4 modifying entry cn=SSL, cn=Configuration

bash-3.2#
```

Configuring Server Authentication

SSL stanza in the ibmslapd.conf file is now configured for ssl communication using server authentication.



```
dn: cn=SSL, cn=Configuration
cn: SSL
ibm-slapdSecurePort: 2636
#ibm-slapdSecurity must be one of none/SSL/SSLOnly/TLS/SSLTLS
ibm-slapdSecurity: SSL
#ibm-slapdSslAuth must be one of serverAuth/serverClientAuth
ibm-slapdSslAuth: serverAuth
ibm-slapdSslCertificate: serverlabel
ibm-slapdSslCipherSpec: AES
ibm-slapdSslCipherSpec: AES-128
ibm-slapdSslCipherSpec: RC4-128-MD5
ibm-slapdSslCipherSpec: RC4-128-SHA
ibm-slapdSslCipherSpec: TripleDES-168
ibm-slapdSslCipherSpec: DES-56
ibm-slapdSslCipherSpec: RC4-40-MD5
ibm-slapdSslCipherSpec: RC2-40-MD5
ibm-slapdSslFIPSProcessingMode: false
ibm-slapdSslKeyDatabase: /home/dsrdbm01/keys/serverkey.kdb
ibm-slapdSslKeyDatabasepw: {AES256}a7zwjl7/B453uRE/w5uLtg==
ibm-slapdSslPKCS11AcceleratorMode: none
ibm-slapdSslPKCS11Enabled: false
ibm-slapdSslPKCS11Keystorage: false
ibm-slapdSslPKCS11Lib: libcknfast.so
ibm-slapdSslPKCS11TokenLabel: none
objectclass: top
objectclass: ibm-slapdConfigEntry
objectclass: ibm-slapdSSL
```

Configuring Server Authentication

Now that the ibmslapd.conf file is updated with the SSL configuration, we must restart the server and the ibmdiradm.

- ibmslapd –I <instance name> -k
- ibmslapd –I <instance name> -n
- ibmdiradm –I <instance name> -k
- ibmdiradm –I <instance name>

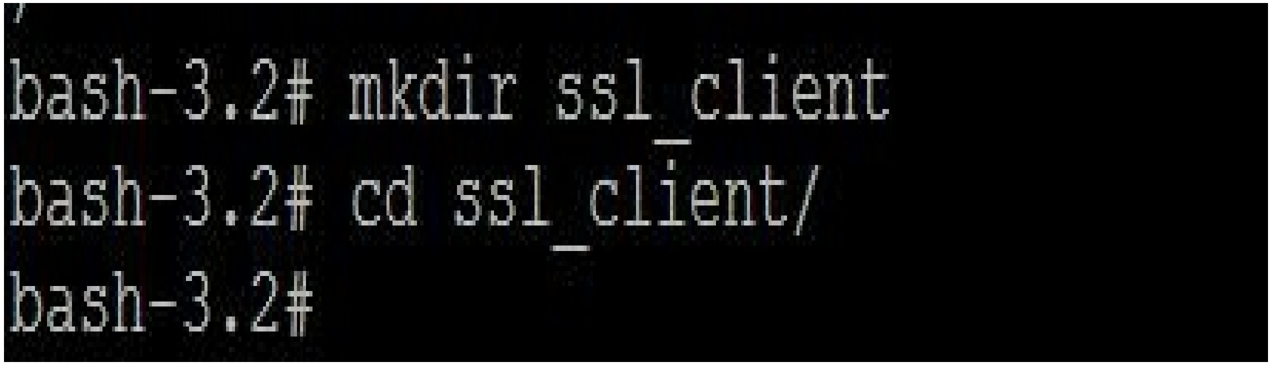
Configuring Server Authentication

The next step is to configure our ITDS client. In order to configure the client we must do the following:

- Create a CMS key database for the C based ldap client.
- Import the server certificate as a signer certificate into the client's key database.

Configuring Server Authentication

On the client system create a subdirectory where you will create and store the key database file.

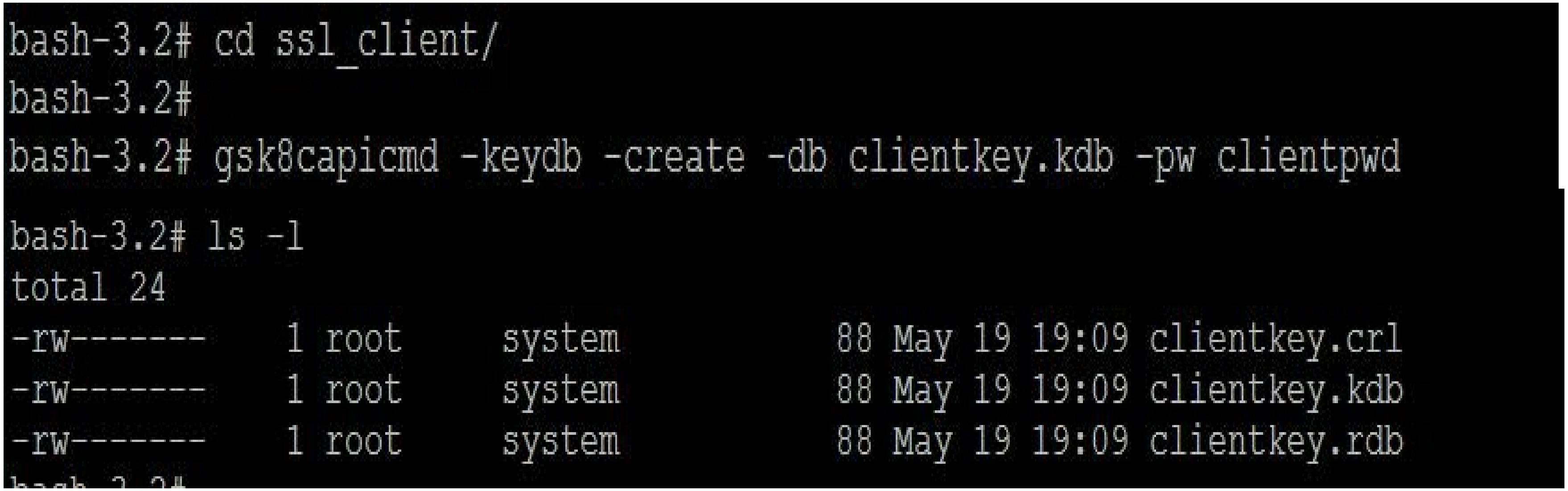


```
bash-3.2# mkdir ssl_client
bash-3.2# cd ssl_client/
bash-3.2#
```

Configuring Server Authentication

- Generate the key database to be used by C-based ldap client

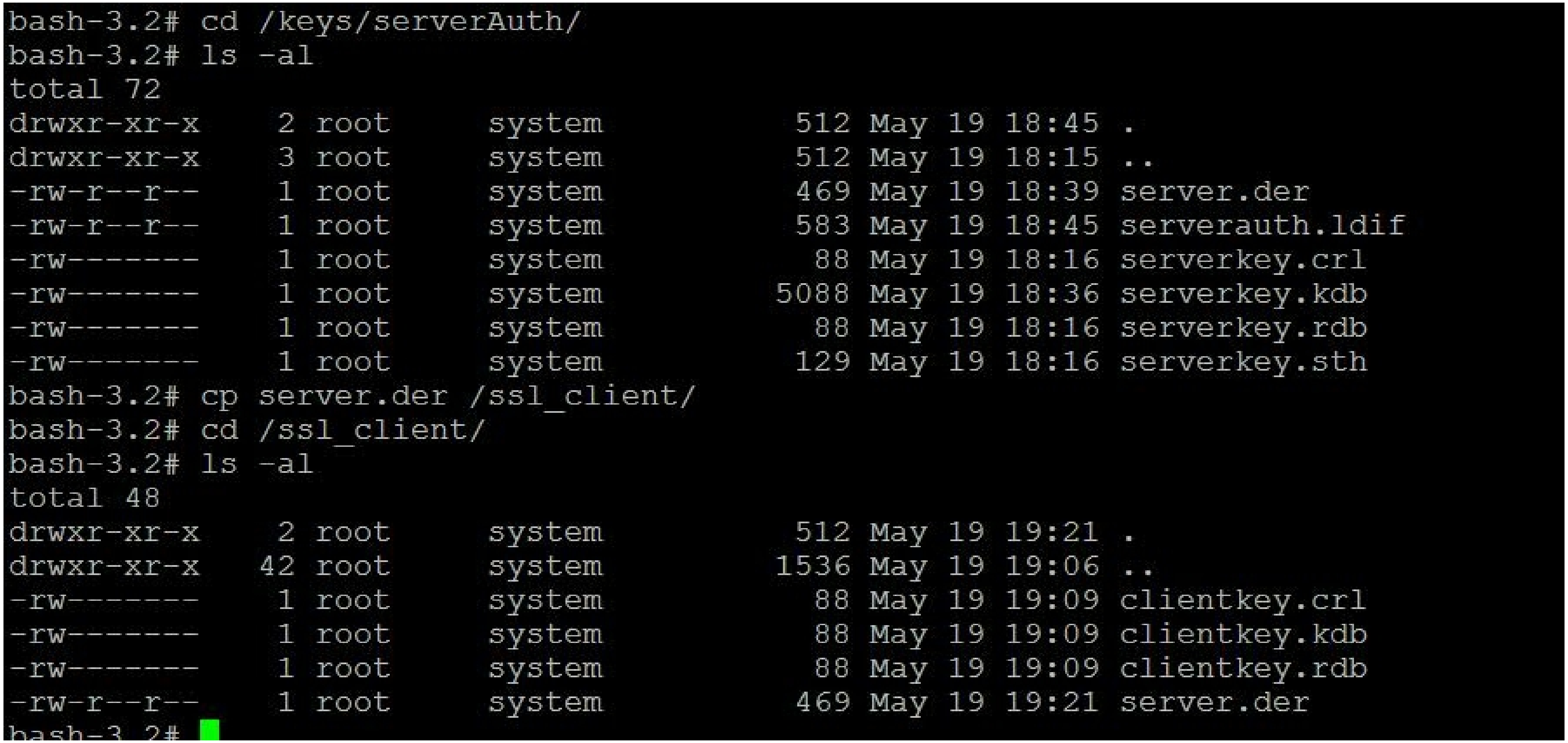
gsk7capicmd -keydb -create -db <client keydb name> -pw <client keydb password>



```
bash-3.2# cd ssl_client/
bash-3.2#
bash-3.2# gsk8capicmd -keydb -create -db clientkey.kdb -pw clientpwd

bash-3.2# ls -l
total 24
-rw-------    1 root     system          88 May 19 19:09 clientkey.crl
-rw-------    1 root     system          88 May 19 19:09 clientkey.kdb
-rw-------    1 root     system          88 May 19 19:09 clientkey.rdb
bash-3.2#
```

Configuring Server Authentication

Copy over the extracted server certificate from the server system to the client system.

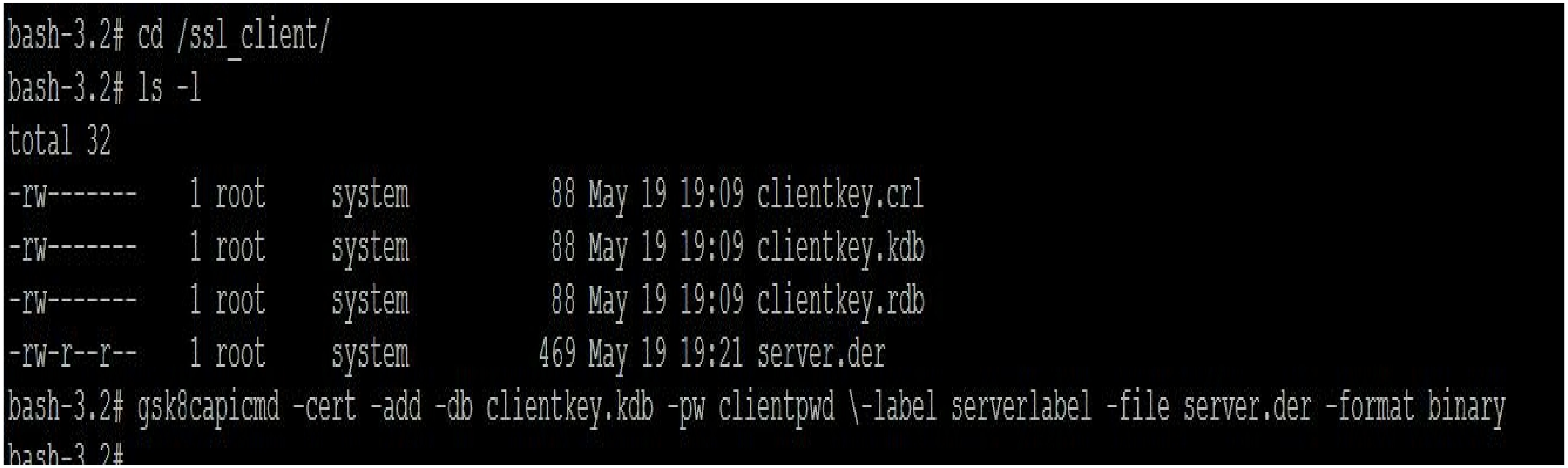


```
bash-3.2# cd /keys/serverAuth/
bash-3.2# ls -al
total 72
drwxr-xr-x    2 root     system          512 May 19 18:45 .
drwxr-xr-x    3 root     system          512 May 19 18:15 ..
-rw-r--r--    1 root     system          469 May 19 18:39 server.der
-rw-r--r--    1 root     system          583 May 19 18:45 serverauth.ldif
-rw-------    1 root     system           88 May 19 18:16 serverkey.crl
-rw-------    1 root     system         5088 May 19 18:36 serverkey.kdb
-rw-------    1 root     system           88 May 19 18:16 serverkey.rdb
-rw-------    1 root     system          129 May 19 18:16 serverkey.sth
bash-3.2# cp server.der /ssl_client/
bash-3.2# cd /ssl_client/
bash-3.2# ls -al
total 48
drwxr-xr-x    2 root     system          512 May 19 19:21 .
drwxr-xr-x   42 root     system         1536 May 19 19:06 ..
-rw-------    1 root     system           88 May 19 19:09 clientkey.crl
-rw-------    1 root     system           88 May 19 19:09 clientkey.kdb
-rw-------    1 root     system           88 May 19 19:09 clientkey.rdb
-rw-r--r--    1 root     system          469 May 19 19:21 server.der
bash-3.2#
```

Configuring Server Authentication

- Add the extracted server certificate into the client key database file

gsk8capicmd -cert -add -db <client keydb> -pw <client keydb password> -label <certificate label> -file <extracted server certificate> -format <format>

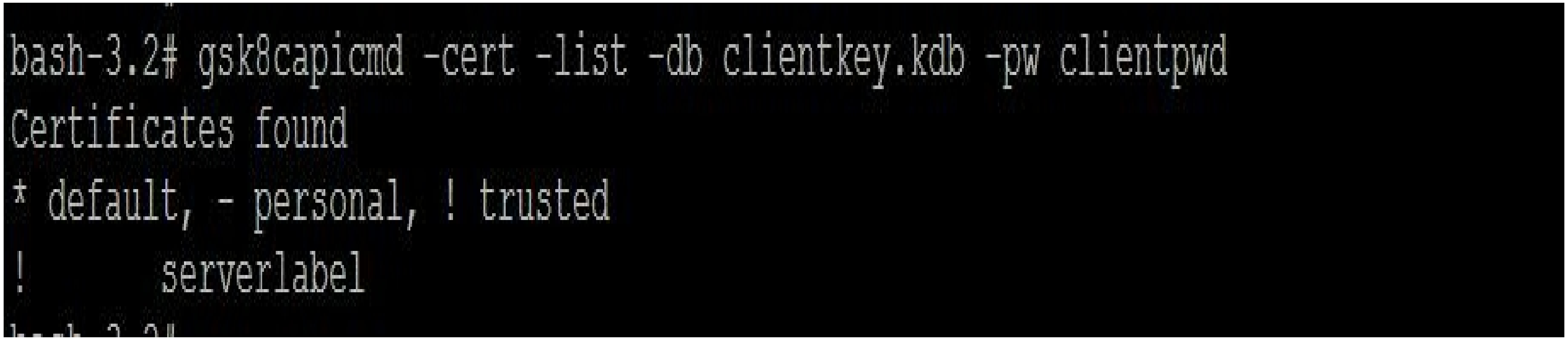


```
bash-3.2# cd /ssl_client/
bash-3.2# ls -l
total 32
-rw-------    1 root     system          88 May 19 19:09 clientkey.crl
-rw-------    1 root     system          88 May 19 19:09 clientkey.kdb
-rw-------    1 root     system          88 May 19 19:09 clientkey.rdb
-rw-r--r--    1 root     system         469 May 19 19:21 server.der
bash-3.2# gsk8capicmd -cert -add -db clientkey.kdb -pw clientpwd \-label serverlabel -file server.der -format binary
bash-3.2#
```

Configuring Server Authentication

- You can verify the certificate was added by using the “list” option of the gsk8capicmd command.

gsk8capicmd -cert -list -db <client keydb> -pw <client keydb password>



```
bash-3.2# gsk8capicmd -cert -list -db clientkey.kdb -pw clientpwd
Certificates found
* default, - personal, ! trusted
!       serverlabel
bash-3.2#
```

Configuring Server Authentication

- Now we are ready to test the ssl communication between the client and server.
- On the client system issue an idsldapsearch command.

```
idsldapsearch  -Z  -h <hostname> -p <port> -D <Bind dn> -w <Bind
password> -K <full path to the key database file> -P <keydb password>
-s <scope> -b <base dn> objectclass=*
```

where

-Z specifies to use a secure ssl connection

Known Issues :

- **idsldapmodify command puts Web Administration Tool into inconsistent state**

  If you are logged into the Web Administration Tool and you change your password using the command line (**idsldapmodify** command), the Web Administration Tool changes the server status to stopped. This occurs because the Web Administration Tool opens new connections to the server every time it launches a task. The Web Administration Tool tries to connect to the server with the old password because it is unaware that the password has been changed; consequently the connection fails. You must log out and log back in using the new password.

  To avoid this situation, if you have sufficient access authority, use the **User properties -> Change password** option to change your user password when working in the Web Administration Tool

Known Issues: (Contd.)

- **A new user might fail to logon to Web Administration Tool for the first time, if the password policy is enabled and pwdMustChange attribute is set .**

If the password policy is enabled and "User must change password after reset (pwdMustChange)" is set on the Password policy settings 1 panel in the Manage password policies wizard, user might not be able to logon to Web Administration Tool.

To resolve the problem, user can use the ldapchangepwd command line utility to reset the password and then use the new password to logon.

Thank
You

IBM®

Tivoli. software