Unit 3 Lab – Identity and Access Management

Required Materials

Putty or other connection tool

Lab Server

Root or sudo command access

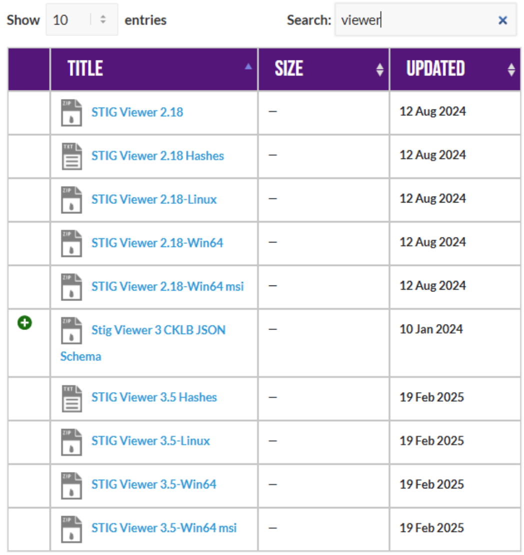
STIG Viewer 2.18 (download from https://public.cyber.mil/stigs/downloads/)

EXERCISES (Warmup to quickly run through your system and familiarize yourself)

1. ls -l /etc/pam.d/
    a. What are the permissions and names of files? Can everyone read them?
2. cat /etc/pam.d/sshd
    a. What information do you see in this file?
    b. Does any of it look familiar to you?
3.

PreLAB

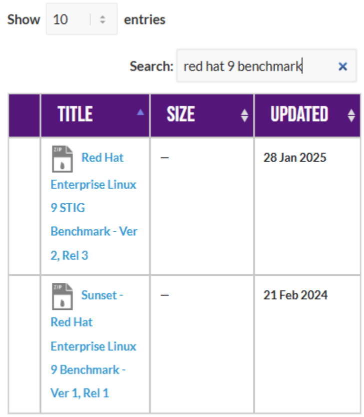
Download the STIG Viewer 2.18 from - https://public.cyber.mil/stigs/downloads/



Show 10 entries Search: viewer

	TITLE	SIZE	UPDATED
	STIG Viewer 2.18	–	12 Aug 2024
	STIG Viewer 2.18 Hashes	–	12 Aug 2024
	STIG Viewer 2.18-Linux	–	12 Aug 2024
	STIG Viewer 2.18-Win64	–	12 Aug 2024
	STIG Viewer 2.18-Win64 msi	–	12 Aug 2024
⊕	Stig Viewer 3 CKLB JSON Schema	–	10 Jan 2024
	STIG Viewer 3.5 Hashes	–	19 Feb 2025
	STIG Viewer 3.5-Linux	–	19 Feb 2025
	STIG Viewer 3.5-Win64	–	19 Feb 2025
	STIG Viewer 3.5-Win64 msi	–	19 Feb 2025

Download the STIG for RHEL 9 and the import it into your STIG viewer

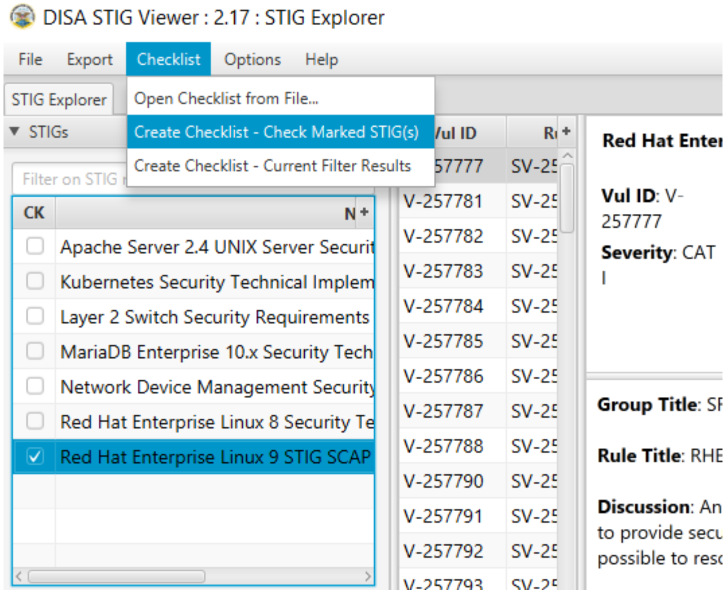


Show 10 entries

Search: red hat 9 benchmark

	TITLE	SIZE	UPDATED
	Red Hat Enterprise Linux 9 STIG Benchmark - Ver 2, Rel 3	–	28 Jan 2025
	Sunset - Red Hat Enterprise Linux 9 Benchmark - Ver 1, Rel 1	–	21 Feb 2024

Create a checklist from the opened STIG for RHEL 9


DISA STIG Viewer : 2.17 : STIG Explorer
File
Export
Checklist
Options
Help
STIG Explorer
Open Checklist from File...
Create Checklist - Check Marked STIG(s)
Create Checklist - Current Filter Results
STIGs
Filter on STIG

CK	N
☐	Apache Server 2.4 UNIX Server Securit
☐	Kubernetes Security Technical Implem
☐	Layer 2 Switch Security Requirements
☐	MariaDB Enterprise 10.x Security Tech
☐	Network Device Management Security
☐	Red Hat Enterprise Linux 8 Security Te
☑	Red Hat Enterprise Linux 9 STIG SCAP

Vul ID	Ru
V-257777	SV-25
V-257781	SV-25
V-257782	SV-25
V-257783	SV-25
V-257784	SV-25
V-257785	SV-25
V-257786	SV-25
V-257787	SV-25
V-257788	SV-25
V-257790	SV-25
V-257791	SV-25
V-257792	SV-25
V-257793	SV-25

Red Hat Enter

Vul ID: V-257777

Severity: CAT I

Group Title: SF

Rule Title: RHE

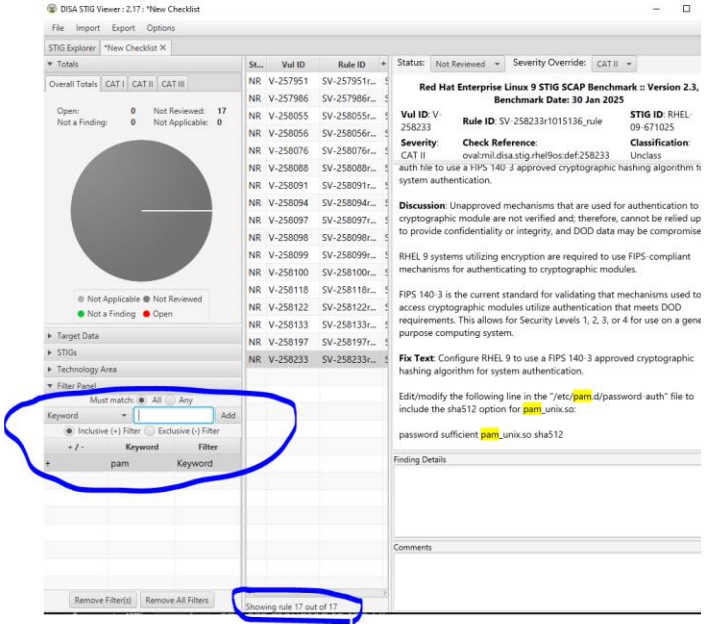
Discussion: An to provide secu possible to reso

LAB

This lab is designed to have the engineer practice securing a Linux server or service against a set of configuration standards. These standards are sometimes called benchmarks, checklists, or guidelines. The engineer will be using STIG Viewer 2.18 to complete this lab.

PAM configuration:

1. Connect to a hammer server
2. Filter by pam and see how many STIGS you have. (Why is it really only 16?)



Open: 0 Not Reviewed: 17
Not a Finding: 0 Not Applicable: 0

Must match: ● All ○ Any

● Inclusive (+) Filter ○ Exclusive (-) Filter

+ / -	Keyword	Filter
+	pam	Keyword

St...	Vul ID	Rule ID
NR	V-257951	SV-257951r...
NR	V-257986	SV-257986r...
NR	V-258055	SV-258055r...
NR	V-258056	SV-258056r...
NR	V-258076	SV-258076r...
NR	V-258088	SV-258088r...
NR	V-258091	SV-258091r...
NR	V-258094	SV-258094r...
NR	V-258097	SV-258097r...
NR	V-258098	SV-258098r...
NR	V-258099	SV-258099r...
NR	V-258100	SV-258100r...
NR	V-258118	SV-258118r...
NR	V-258122	SV-258122r...
NR	V-258133	SV-258133r...
NR	V-258197	SV-258197r...
NR	V-258233	SV-258233r...

Showing rule 17 out of 17

Status: Not Reviewed Severity Override: CAT II

Red Hat Enterprise Linux 9 STIG SCAP Benchmark :: Version 2.3, Benchmark Date: 30 Jan 2025

Vul ID: V-258233 **Rule ID**: SV-258233r1015136_rule **STIG ID**: RHEL-09-671025

Severity: CAT II **Check Reference**: oval:mil.disa.stig.rhel9os:def:258233 **Classification**: Unclass

auth file to use a FIPS 140-3 approved cryptographic hashing algorithm fo system authentication.

Discussion: Unapproved mechanisms that are used for authentication to cryptographic module are not verified and; therefore, cannot be relied up to provide confidentiality or integrity, and DOD data may be compromise

RHEL 9 systems utilizing encryption are required to use FIPS-compliant mechanisms for authenticating to cryptographic modules.

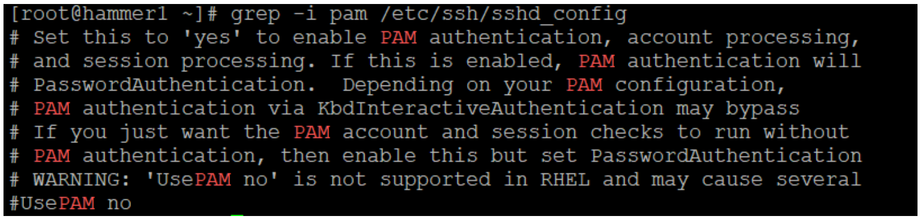
FIPS 140-3 is the current standard for validating that mechanisms used to access cryptographic modules utilize authentication that meets DOD requirements. This allows for Security Levels 1, 2, 3, or 4 for use on a gene purpose computing system.

Fix Text: Configure RHEL 9 to use a FIPS 140-3 approved cryptographic hashing algorithm for system authentication.

Edit/modify the following line in the "/etc/pam.d/password-auth" file to include the sha512 option for pam_unix.so:

password sufficient pam_unix.so sha512

3. Examine STIG V-257986
    a. What is the problem?
    b. What is the fix?
    c. What type of control is being implemented?
    d. Is it set properly on your system?
        i. grep -i pam /etc/ssh/sshd_config

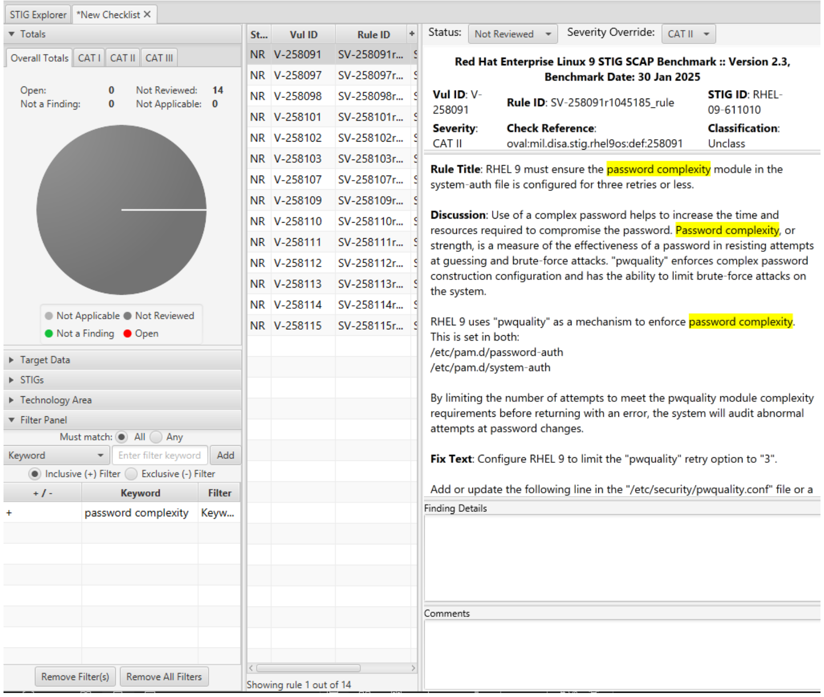


```
[root@hammer1 ~]# grep -i pam /etc/ssh/sshd_config
# Set this to 'yes' to enable PAM authentication, account processing,
# and session processing. If this is enabled, PAM authentication will
# PasswordAuthentication.  Depending on your PAM configuration,
# PAM authentication via KbdInteractiveAuthentication may bypass
# If you just want the PAM account and session checks to run without
# PAM authentication, then enable this but set PasswordAuthentication
# WARNING: 'UsePAM no' is not supported in RHEL and may cause several
#UsePAM no
```

        ii. Can you remediate this finding?

4. Check and remediate STIG V-258055
    a. What is the problem?
    b. What is the fix?
    c. What type of control is being implemented?
        i. Are there any major implications to think about with this change on your system? Why or why not?
    d. Is it set properly on your system?
    e. How would you go about remediating this on your system?

5. Check and remediate STIG V-258098
    a. What is the problem?
    b. What is the fix?
    c. What type of control is being implemented?
    d. Is it set properly on your system?
6. Filter by “password complexity”



Open:	0	Not Reviewed:	14
Not a Finding:	0	Not Applicable:	0

Must match: ● All ○ Any

Keyword ▾ | Enter filter keyword | Add

● Inclusive (+) Filter ○ Exclusive (-) Filter

+ / -	Keyword	Filter
+	password complexity	Keyw...

St...	Vul ID	Rule ID
NR	V-258091	SV-258091r...
NR	V-258097	SV-258097r...
NR	V-258098	SV-258098r...
NR	V-258101	SV-258101r...
NR	V-258102	SV-258102r...
NR	V-258103	SV-258103r...
NR	V-258107	SV-258107r...
NR	V-258109	SV-258109r...
NR	V-258110	SV-258110r...
NR	V-258111	SV-258111r...
NR	V-258112	SV-258112r...
NR	V-258113	SV-258113r...
NR	V-258114	SV-258114r...
NR	V-258115	SV-258115r...

Showing rule 1 out of 14

Status: Not Reviewed ▾ Severity Override: CAT II ▾

Red Hat Enterprise Linux 9 STIG SCAP Benchmark :: Version 2.3, Benchmark Date: 30 Jan 2025

Vul ID: V-258091	**Rule ID**: SV-258091r1045185_rule	**STIG ID**: RHEL-09-611010
Severity: CAT II	**Check Reference**: oval:mil.disa.stig.rhel9os:def:258091	**Classification**: Unclass

Rule Title: RHEL 9 must ensure the password complexity module in the system-auth file is configured for three retries or less.

Discussion: Use of a complex password helps to increase the time and resources required to compromise the password. Password complexity, or strength, is a measure of the effectiveness of a password in resisting attempts at guessing and brute-force attacks. "pwquality" enforces complex password construction configuration and has the ability to limit brute-force attacks on the system.

RHEL 9 uses "pwquality" as a mechanism to enforce password complexity.
This is set in both:
/etc/pam.d/password-auth
/etc/pam.d/system-auth

By limiting the number of attempts to meet the pwquality module complexity requirements before returning with an error, the system will audit abnormal attempts at password changes.

Fix Text: Configure RHEL 9 to limit the "pwquality" retry option to "3".

Add or update the following line in the "/etc/security/pwquality.conf" file or a

Finding Details

Comments

a. How many are there?

b. What are the password complexity rules?

i. Are there any you haven't seen before?

7. Filter by sssd

a. How many STIGS do you see?

b. What do these STIGS appear to be trying to do? What types of controls are they?

OpenLDAP Setup

You will likely not build an LDAP server in a real world environment. We are doing it for understanding and ability to complete the lab. In a normal corporate environment this is likely Active Directory.

To simplify some of the typing in this lab, there is a file located at /lab_work/identity_and_access_management.tar.gz that you can pull down to your system with the correct .ldif files.

8. Install and configure OpenLDAP

a. Stop the warewulf client

systemctl stop wwclient

b. Edit your /etc/hosts file #use your server line

# Entry for hammer1

192.168.200.151 hammer1 hammer1-default ldap.prolug.lan ldap

c. Setup dnf repo

dnf config-manager --set-enabled plus

```
dnf repolist
dnf -y install openldap-servers openldap-clients openldap
```

d. Start slapd

```
systemctl start slapd
ss -ntulp
```

e. Allow ldap through the firewall

```
firewall-cmd --add-service={ldap,ldaps} --permanent
firewall-cmd --reload
firewall-cmd --list-all
```

f. Generate a password #use testpassword

```
[root@hammer1 ~]# slappasswd
New password:
Re-enter new password:
{SSHA}wpRvODvIC/EPYf2GqHUlQMDdsFIW5yig
```

g. Change the password

```
vi changerootpass.ldif
    dn: olcDatabase={0}config,cn=config
    changetype: modify
    replace: olcRootPW
    olcRootPW: {SSHA}vKobSZO1HDGxp2OElzli/xfAzY4jSDMZ

[root@hammer1 ~]# ldapadd -Y EXTERNAL -H ldapi:/// -f changerootpass.ldif
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
modifying entry "olcDatabase={0}config,cn=config"
```

h. Generate basic schemas

```
ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/cosine.ldif
ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/nis.ldif
ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/inetorgperson.ldif
```

i. Set up the domain #USE THE PASSWORD YOU GENERATED EARLIER

```
vi setdomain.ldif

dn: olcDatabase={1}monitor,cn=config
changetype: modify
replace: olcAccess
olcAccess: {0}to * by
dn.base="gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth"
  read by dn.base="cn=Manager,dc=prolug,dc=lan" read by * none
```

```
dn: olcDatabase={2}mdb,cn=config
changetype: modify
replace: olcSuffix
olcSuffix: dc=prolug,dc=lan

dn: olcDatabase={2}mdb,cn=config
changetype: modify
replace: olcRootDN
olcRootDN: cn=Manager,dc=prolug,dc=lan

dn: olcDatabase={2}mdb,cn=config
changetype: modify
add: olcRootPW
olcRootPW: {SSHA}Uf13AbVHOcs/aDWJOvDxxfBSl3omExG2

dn: olcDatabase={2}mdb,cn=config
changetype: modify
add: olcAccess
olcAccess: {0}to attrs=userPassword,shadowLastChange by
  dn="cn=Manager,dc=prolug,dc=lan" write by anonymous auth by self write by *
none
olcAccess: {1}to dn.base="" by * read
olcAccess: {2}to * by dn="cn=Manager,dc=prolug,dc=lan" write by * read

###Run it
[root@hammer25 ~]# ldapmodify -Y EXTERNAL -H ldapi:/// -f setdomain.ldif
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
modifying entry "olcDatabase={1}monitor,cn=config"

modifying entry "olcDatabase={2}mdb,cn=config"

modifying entry "olcDatabase={2}mdb,cn=config"

modifying entry "olcDatabase={2}mdb,cn=config"

modifying entry "olcDatabase={2}mdb,cn=config"
```

j. Search and verify the domain is working.

```
[root@hammer25 ~]# ldapsearch -H ldap:// -x -s base -b "" -LLL "namingContexts"
```

```
dn:
namingContexts: dc=prolug,dc=lan
```

k. Add the base group and organization.

```
vi addou.ldif

dn: dc=prolug,dc=lan
objectClass: top
objectClass: dcObject
objectclass: organization
o: My prolug Organisation
dc: prolug

dn: cn=Manager,dc=prolug,dc=lan
objectClass: organizationalRole
cn: Manager
description: OpenLDAP Manager

dn: ou=People,dc=prolug,dc=lan
objectClass: organizationalUnit
ou: People

dn: ou=Group,dc=prolug,dc=lan
objectClass: organizationalUnit
ou: Group

ldapadd -x -D cn=Manager,dc=prolug,dc=lan -W -f addou.ldif
```

l. Verifying

```
ldapsearch -H ldap:// -x -s base -b "" -LLL "+"
ldapsearch -x -b "dc=prolug,dc=lan" ou
```

m. Add a user

Generate a password

```
slappasswd          #use testuser1234

vi adduser.ldif
dn: uid=testuser,ou=People,dc=prolug,dc=lan
objectClass: inetOrgPerson
objectClass: posixAccount
```

```
objectClass: shadowAccount
cn: testuser
sn: temp
userPassword: {SSHA}dk/Lks9078gfZQJ31ABvPpvKv3sHhr29
loginShell: /bin/bash
uidNumber: 15000
gidNumber: 15000
homeDirectory: /home/testuser
shadowLastChange: 0
shadowMax: 0
shadowWarning: 0

dn: cn=testuser,ou=Group,dc=prolug,dc=lan
objectClass: posixGroup
cn: testuser
gidNumber: 15000
memberUid: testuser
```

```
ldapadd -x -D cn=Manager,dc=prolug,dc=lan -W -f adduser.ldif
```

n. Verify that your user is in the system.

```
ldapsearch -x -b "ou=People,dc=prolug,dc=lan"
```

o. Secure the system with TLS   #accept all defaults

```
openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout
/etc/pki/tls/ldapserver.key -out /etc/pki/tls/ldapserver.crt

chown ldap:ldap /etc/pki/tls/{ldapserver.crt,ldapserver.key}

[root@hammer25 tls]# ls -l /etc/pki/tls/ldap*
-rw-r--r--. 1 ldap ldap 1224 Apr 12 18:23 /etc/pki/tls/ldapserver.crt
-rw-------. 1 ldap ldap 1704 Apr 12 18:22 /etc/pki/tls/ldapserver.key

vi tls.ldif
dn: cn=config
changetype: modify
add: olcTLSCACertificateFile
olcTLSCACertificateFile: /etc/pki/tls/ldapserver.crt

add: olcTLSCertificateKeyFile
olcTLSCertificateKeyFile: /etc/pki/tls/ldapserver.key
```

```
add: olcTLSCertificateFile
olcTLSCertificateFile: /etc/pki/tls/ldapserver.crt

[root@hammer25 ~]#  ldapadd -Y EXTERNAL -H ldapi:/// -f tls.ldif
```

p. Fix the /etc/openldap/ldap.conf to allow for certs
   vi /etc/openldap/ldap.conf

```
#
# LDAP Defaults
#

# See ldap.conf(5) for details
# This file should be world readable but not world writable.

#BASE   dc=example,dc=com
#URI    ldap://ldap.example.com ldap://ldap-master.example.com:666

#SIZELIMIT      12
#TIMELIMIT      15
#DEREF          never

# When no CA certificates are specified the Shared System Certificates
# are in use. In order to have these available along with the ones specified
# by TLS_CACERTDIR one has to include them explicitly:
TLS_CACERT      /etc/pki/tls/ldapserver.crt
TLS_REQCERT     never

# System-wide Crypto Policies provide up to date cipher suite which should
# be used unless one needs a finer grinded selection of ciphers. Hence, the
# PROFILE=SYSTEM value represents the default behavior which is in place
# when no explicit setting is used. (see openssl-ciphers(1) for more info)
#TLS_CIPHER_SUITE PROFILE=SYSTEM

# Turning this off breaks GSSAPI used with krb5 when rdns = false
SASL_NOCANON    on
```

q. systemctl restart slapd

SSSD Configuration and Realmd join to LDAP

SSSD can connect a server to a trusted LDAP system and authenticate users for access to local resources. You will likely do this during your career and it is a valuable skill to work with.

9. Install sssd, configure, and validate that the user is seen by the system

   a. ` dnf install openldap-clients sssd sssd-ldap oddjob-mkhomedir authselect`

   b. `authselect select sssd with-mkhomedir --force`

   c. `systemctl enable --now oddjobd.service`

   d. `systemctl status oddjobd.service`

   e. Uncomment and fix the lines in /etc/openldap/ldap.conf

      ```
      BASE    dc=prolug,dc=lan
      URI     ldap://ldap.ldap.lan/
      ```

   f. `vi /etc/sssd/sssd.conf`

      ```
      [domain/default]
      id_provider = ldap
      autofs_provider = ldap
      auth_provider = ldap
      chpass_provider = ldap
      ldap_uri = ldap://ldap.prolug.lan/
      ldap_search_base = dc=prolug,dc=lan
      #ldap_id_use_start_tls = True
      #ldap_tls_cacertdir = /etc/openldap/certs
      cache_credentials = True
      #ldap_tls_reqcert = allow

      [sssd]
      services = nss, pam, autofs
      domains = default

      [nss]
      homedir_substring = /home
      ```

   g. `chmod 0600 /etc/sssd/sssd.conf`

   h. `systemctl start sssd`

   i. `systemctl status sssd`

   j. validate that the user can be seen

      ```
      id testuser
      uid=15000(testuser) gid=15000 groups=15000
      ```