5 Deploying Identity Management in Red Hat Enterprise Linux

This section details the deployment tasks for configuring the IdM Server, Replica and IdM Admin client. The Replica server is deployed to increase the availability and resiliency of the services provided within IdM. Before proceeding with the IdM deployment tasks, each system must be configured as previously outlined in **Section 4 Staging the Infrastructure**:

- **4.2 Install Red Hat Enterprise Linux**
- **4.3 Configure Network**
- **4.4 Configure Domain Name System (DNS)**
- **4.5 Configure Time Service (NTP)**
- **4.6 Configure SELinux**
- **4.7 Update Hosts File**
- **4.8 Register Hosts and Run Updates**

Do not proceed until each of these steps has been fully completed.

5.1 Deploy IdM Server

Deploy the IdM server by configuring firewall ports, installing the required packages and configuring the IdM server. A set of user groups and accounts are also created for testing and verification purposes.

5.1.1 Configure Firewall Ports

On the IdM Server (**idm-srv1**) create a new chain (***ipa-server-chain***) and add the appropriate firewall rules for the ports required by IdM.

1. The default firewall service on Red Hat Enterprise Linux 7 uses **`firewalld`**. To avoid potential conflicts, stop and prevent the **`iptables`** (IPV4) and **`ip6tables`** (IPV6) services from running.

```
# systemctl stop iptables
# systemctl mask iptables
ln -s '/dev/null' '/etc/systemd/system/iptables.service'
# systemctl status iptables
iptables.service
   Loaded: masked (/dev/null)
   Active: inactive (dead)

Jun 20 19:06:55 idm-srv1.interop.example.com systemd[1]: Stopped IPv4
firewall with iptables.

# systemctl stop ip6tables
```

```
# systemctl mask ip6tables
ln -s '/dev/null' '/etc/systemd/system/ip6tables.service'
# systemctl status ip6tables
ip6tables.service
   Loaded: masked (/dev/null)
   Active: inactive (dead)

Jun 20 19:06:56 idm-srv1.interop.example.com systemd[1]: Stopped IPv6
firewall with ip6tables.
```

2. Start firewalld and enable it start on boot.

```
# systemctl start firewalld
# systemctl enable firewalld
# systemctl status firewalld
firewalld.service - firewalld - dynamic firewall daemon
   Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled)
   Active: active (running) since Fri 2014-06-20 15:00:05 EDT; 4h 7min ago
 Main PID: 678 (firewalld)
   CGroup: /system.slice/firewalld.service
           └─678 /usr/bin/python -Es /usr/sbin/firewalld --nofork --nopid
Jun 20 19:07:34 idm-srv1.interop.example.com systemd[1]: Started firewalld -
dynamic firewall daemon.
```

3. Create a new chain (***ipa-server-chain***) and add the appropriate firewall rules for the ports required by IdM**.**

```
# firewall-cmd --permanent --direct --add-chain ipv4 filter ipa-server-chain
success
# firewall-cmd --permanent --direct --add-rule ipv4 filter INPUT 0 -m conntrack
--ctstate NEW -j ipa-server-chain
success
# firewall-cmd --permanent --direct --add-rule ipv4 filter ipa-server-chain 0
--proto tcp --destination-port 80 --jump ACCEPT
success
# firewall-cmd --permanent --direct --add-rule ipv4 filter ipa-server-chain 0
--proto tcp --destination-port 80 --jump ACCEPT
success
# firewall-cmd --permanent --direct --add-rule ipv4 filter ipa-server-chain 0
--proto tcp --destination-port 443 --jump ACCEPT
success
# firewall-cmd --permanent --direct --add-rule ipv4 filter ipa-server-chain 0
--proto tcp --destination-port 389 --jump ACCEPT
success
# firewall-cmd --permanent --direct --add-rule ipv4 filter ipa-server-chain 0
--proto tcp --destination-port 636 --jump ACCEPT
success
# firewall-cmd --permanent --direct --add-rule ipv4 filter ipa-server-chain 0
--proto tcp --destination-port 88 --jump ACCEPT
success
# firewall-cmd --permanent --direct --add-rule ipv4 filter ipa-server-chain 0
--proto tcp --destination-port 464 --jump ACCEPT
success
# firewall-cmd --permanent --direct --add-rule ipv4 filter ipa-server-chain 0
--proto udp --destination-port 88 --jump ACCEPT
```

```
success
# firewall-cmd --permanent --direct --add-rule ipv4 filter ipa-server-chain 0
--proto udp --destination-port 464 --jump ACCEPT
success
# firewall-cmd --permanent --direct --add-rule ipv4 filter ipa-server-chain 0
--proto tcp --destination-port 53 --jump ACCEPT
success
# firewall-cmd --permanent --direct --add-rule ipv4 filter ipa-server-chain 0
--proto udp --destination-port 53 --jump ACCEPT
success
# firewall-cmd --permanent --direct --add-rule ipv4 filter ipa-server-chain 0
--proto udp --destination-port 123 --jump ACCEPT
success
# firewall-cmd --permanent --direct --add-rule ipv4 filter ipa-server-chain 0
--proto tcp --destination-port 7389 --jump ACCEPT
success
# firewall-cmd --reload
success
```

Each of the ports are described in further detail in **Table 3.3.4 Network Ports**.

4. Verify the entries.

```
# firewall-cmd --permanent --direct --get-all-rules
ipv4 filter INPUT 0 -m conntrack --ctstate NEW -j ipa-server-chain
ipv4 filter ipa-server-chain 0 --proto tcp --destination-port 80  --jump ACCEPT
ipv4 filter ipa-server-chain 0 --proto tcp --destination-port 443 --jump ACCEPT
ipv4 filter ipa-server-chain 0 --proto tcp --destination-port 389 --jump ACCEPT
ipv4 filter ipa-server-chain 0 --proto tcp --destination-port 636 --jump ACCEPT
ipv4 filter ipa-server-chain 0 --proto tcp --destination-port 88  --jump ACCEPT
ipv4 filter ipa-server-chain 0 --proto tcp --destination-port 464 --jump ACCEPT
ipv4 filter ipa-server-chain 0 --proto udp --destination-port 88  --jump ACCEPT
ipv4 filter ipa-server-chain 0 --proto udp --destination-port 464 --jump ACCEPT
ipv4 filter ipa-server-chain 0 --proto tcp --destination-port 53  --jump ACCEPT
ipv4 filter ipa-server-chain 0 --proto udp --destination-port 53  --jump ACCEPT
ipv4 filter ipa-server-chain 0 --proto udp --destination-port 123  --jump ACCEPT
ipv4 filter ipa-server-chain 0 --proto tcp --destination-port 7389 --jump ACCEPT
```

5.1.2 Install Packages

On the IdM Server (**idm-srv1**) install the IPA server, DNS and DDNS packages.

```
# yum install ipa-server bind bind-dyndb-ldap
```

5.1.3 Install/configure IdM Server

Next, run the server installation script by specifying options to configure DNS (***--setup-dns***), Kerberos realm (***--realm***) and DNS forwarder (***--forwarder***). It is also recommended to not configure the NTP server (***--no-ntp***) when IdM is configured on a virtual machine.

```
# hostname
idm-srv1.interop.example.com

# ipa-server-install --setup-dns --realm=INTEROP.EXAMPLE.COM
```