Twenty Questions to Ask Yourself During a Red Hat Directory Server Deployment

by **Satish Chetty**

Technical Support Account Manager

Contributions by **Andy Fitzsimon**, **Orla Hegarty**, **Neil Kruse**, **Deon Lackey**, **Rich Megginson**, **Kevin Unthank**

Abstract

The key to having a reliable, flexible, and effective Red Hat Directory Server deployment is to plan the directory service in advance. The primary issues for a solid deployment are planning the masters, hubs, and suppliers for replication; deciding what machines will be involved in replication and where these machines are; determining what applications will access the directory services; identifying network considerations like firewalls; and estimating the load on the servers. This paper proposes and answers twenty of these basic questions as a guide to beginning to plan a Red Hat Directory Server implementation.

Introduction

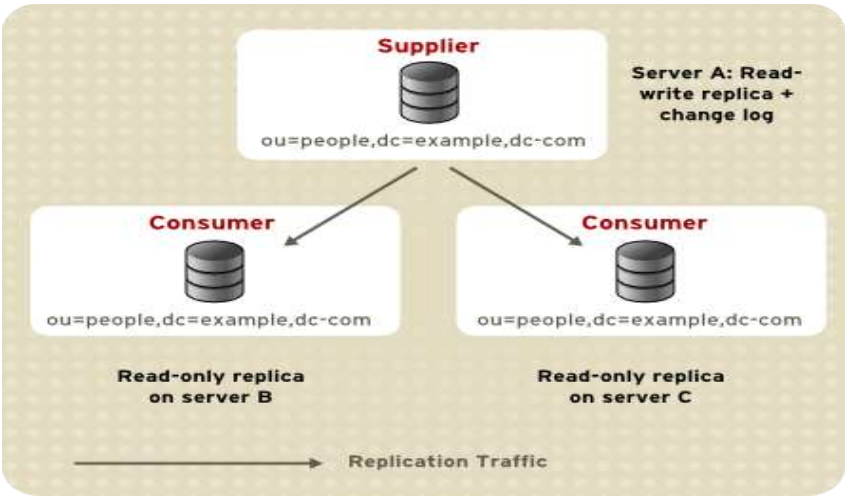
Now that you have decided to implement Red Hat Directory Server (Directory Server), you need to formalize a "how-to" plan. Below are twenty questions to get you started with planning for a massively scalable, reliable, highly available yet efficient deployment. This list of twenty questions is neither comprehensive nor complete, but it is a list of basic questions to ask while planning an enterprise deployment of Red Hat Directory Server. These questions and answers are based on a thoughts and text from Red Hat's Professional Services deployment experiences and product and training documentation.

1. What is the primary application of the Directory Server?

Some of the typical applications are NIS, Windows login, web-based phone book, authentication for Kerberos, FTP, or Samba. You could also have the Directory Server provide information to other applications like mail servers, calendar servers, and web servers. Depending on the type of application, you may need to modify or create your own schema. Finding out at this stage which applications will be using the Directory Server helps estimate the load the Directory Servers will handle. Knowing the load will help you decide how many masters and replicas you will need. Also, it will help you understand how to index data and how to distribute your data over several databases.

2. How many directory masters and replicas are required?

The Red Hat Directory Server that stores a master copy of the directory data and is an authoritative data source is called a **master**, or a **supplier**. Only a master server is write-able. A server that receives information from a master is called a **replica**. There are several configurations that you could choose for your deployment.



There are two common configuration examples:

- **Single server**: This server is the master and does not supply data to any replica. This configuration is quick and simple to set up. This is suitable only for very small deployments with no fail-over or redundancy requirements.

- **Single master with one or more replicas (cascading):** The single master sends updates to many replicas. This is an ideal configuration when the read load is heavy. Having multiple replicas allows the Directory Server data to be available. For more information on the deployment layout, see the *Red Hat Directory Server Deployment Guide*[1].

3. Where are the clients located?

Clients are LDAP users or applications that require information from the directory service. Knowing where the clients are physically located will help you decide if you need local masters or replicas. Local replicas help you distribute the read load. For example, if the master Directory Server is in San Jose, California, and clients log in from London, UK, there is a risk that the clients will not be able to log in if network connectivity is broken between the two locations. Having a local replica in London provides fail-over options.

4. How many users will the directory deployment handle?

Directory Server users can be applications (such as NIS or Kerberos) or people. Each of these clients generates requests. For example, an OS authentication client like `nss_ldap` will generate one request at a time (usually when the user logs in via the client);

1 *Red Hat Directory Server Deployment Guide.* http://www.redhat.com/docs/manuals/dir-server/

however, a web portal will generate multiple requests to the Directory Server, sometimes even simultaneously. Knowing how many requests will be made to the Directory Server will help you calculate the amount of memory that will be needed to set cache sizes effectively for optimal performance. Red Hat Directory Server performance increases when data are served from memory rather than from the disk.

Based on Red Hat testing results for Directory Server load-balancing, it is best to consider having another replica or master server if the Directory Server load is above 40% of the peak output. Again, keep in mind that the load generated by a phone book program is not the same as the load when an application looks up a CRL.

5. Are there any firewall configurations of which you should be aware?

The default ports to which the Directory Server listens and responds are port 389, for non-SSL requests, and port 636, for LDAPS requests. Replication information is also passed over these active ports. Firewalls should be configured to allow requests on these ports. Also, firewalls may be configured to allow requests only from certain domains or IPs to reach the Directory Server. This may help in preventing denial of service attacks on the Directory Server.

6. How many Red Hat Directory Server masters and replicas are required?

A directory master is an authoritative data source. A master server is also the server that accepts updates (writes). When users or applications try to update any information, they are referred by the

replicas to the masters. If the master is unreachable due to a network issue, then updates may not be possible. If updates are mission critical, having another master instead of a replica is recommended. The master server also initiates replication. If you have a configuration that has multiple replicas and a lot of updates, having multiple masters is helpful. This will help reduce both the replication load and the update load on a single master.

Replicas allow you to distribute information closer to where it is needed. In a large enterprise, having replicas may help reduce (read) load on the master Directory Servers. Depending on the number of requests, one or more replicas may be advisable.

7. How many hubs do I need to plan in my directory deployment?

In a large enterprise deployment, with large numbers of update requests, the master server may spend a considerable amount of its resources replicating data. Hubs help reduce replication load on the master. Depending on the scale of deployment, number of updates per minute and frequency of replication, a large deployment may require none or many hub servers. Keep in mind that hubs can become a single point of failure between master servers and multiple replicas. Consult with a Red Hat technical architect for the exact number of hubs needed for a deployment.

Note: Hubs can also serve as read-only replica servers.

8. What applications will use LDAP?

LDAP read and write requests (load) may be generated from user applications, like Outlook, or from other applications, like a phone book or web portal. Requests from applications usually are anonymous requests or proxy requests. Knowing how many applications and the type and volume of requests will help ensure an efficient deployment. For example, email address lookup from several clients will be a lot lighter than multiple CRL lookups of

similar volume. See the *Red Hat Directory Server Deployment Guide*[1] for more information.

9. What is the approximate load you expect?

The three main types of load are read/search load, write load, and replication load. Knowing the type of load will help you design the distribution topology. If the update load is high, having multiple masters can help in load balancing. Similarly, multiple read-only replicas balances the read load. A provisioning application may generate a lot more write load than read load. In such a case, additional master servers distribute the write load. See the *Red Hat Directory Server Deployment Guide*[2] for more information.

10. Do you plan to set up MMR?

Multi-master replication (MMR) is a popular Red Hat Directory Server feature. This feature enables multiple master Directory Servers to synchronize information among themselves and to other replicas or hubs. For example, a large enterprise uses Red Hat Directory Server to store and manage its employee information. This enterprise is head-quartered in Mountain View, California, and has offices in other parts of the world, such as Tokyo, London, and Paris. There is only a single master in the California office which acts as a central point for all updates. If the connectivity to California is lost or broken, updates from other parts of the world may not be possible until the connection is restored. Also, if the number of updates is large, even several password updates, this the load on the master is increased considerably. To overcome this bottleneck, Red Hat Directory Server lets you deploy multiple Directory Servers in each of these high-update load locations. All

1 *Red Hat Directory Server Deployment Guide.* http://www.redhat.com/docs/manuals/dir-server/

2 *Red Hat Directory Server Deployment Guide.* http://www.redhat.com/docs/manuals/dir-server/

of these servers can then be configured as masters, which hold a copy of information. When configured, the servers will synchronize among themselves and can act as local master servers.

11. Do you plan to synchronize information with Microsoft Active Directory?

Another interesting and popular feature in Red Hat Directory Server is **Windows Sync**. Windows Sync synchronizes user and group information between Red Hat Directory Server and Active Directory on Windows 2000/2003. Setting up Windows Sync is similar to setting up replication. However, if the `userpassword` attribute is part of the user information that is to be synchronized, then a separate module, called Password Sync, needs to be installed on the Active Directory host server. If multiple Active Directory servers are to be synchronized, the Password Sync module needs to be installed on each of the Windows host servers.

Password Sync intercepts password updates in Active Directory and sends a copy to the master Directory Server. Similarly, when password updates happen on the master Directory Server, the Password Sync module gets notified of the changes and, in turn, updates Active Directory. The Password Sync module requires that Directory Server be configured for SSL. Also, password synchronization happens only when the user changes the password.

12. How often do you want replication to happen (scheduled or instant)?

When setting up replication between master and replicas or between masters in MMR, information can be replicated in real time whenever the master gets updated or during a particular time of day. Scheduled replication is particularly useful for remote offices that periodically connect to the network and receive updates from the master server. Setting up scheduled replication between

masters is also possible. However, there is a greater chance of replication conflict in a timed MMR. If updates happen to the same entry, such as a password, on different master servers, there will be a conflict when the servers try to synchronize during timed replication. Directory Servers are capable of handling and resolving update conflicts that happen during MMR. See the *Red Hat Directory Server Deployment Guide*[1] for more information.

13. What is the network connectivity between the masters and replicas?

Understanding the bandwidth limitations between replicating servers is important for an efficient deployment. Bandwidth constraints may force you to set up **MMR**, use timed replication, or distribute the data across multiple databases and multiple servers. For more information, contact a Red Hat technical architect.

Note: Directory Servers can handle replication efficiently even over low-quality and slow networks. Directory Server also supports topologies that change due to traffic shaping.

14. What data get updated in the LDAP directory?

What data get updated and replicated will help you understand the load requirements of your Directory Server deployment. For example, the load generated by several clients doing an email look-up is different than the load generated by an **O**nline **C**ertificate **S**tatus **P**rotocol (**OCSP**) application. In most cases, Red Hat Directory Server can handle various types of requests effectively. Knowing what data get updated will help in planning the required number of master, replica, and hub servers.

1 Red Hat Directory Server Deployment Guide https://www.redhat.com/docs/manuals/dir-server/ag/7.1/replicat.html#1106141

15. Have you planned for fail-over or redundancy?

Red Hat Directory Server running on an enterprise-class machine (and running on Red Hat Enterprise Linux) can handle several thousand read requests and several hundred write requests per minute. However, you should not rely just on a single master machine for all your LDAP needs. Having multiple master or replica servers enables the load to be distributed, as well as having backup servers if the primary master server is unavailable due to hardware or network failure. Refer to the "Deployment Scenarios" section in *Red Hat Identity Management and Security Solutions*[1].

16. What ACIs do you plan to set up?

Access control instructions (ACI) are sets of rules placed on the directory or a subset of the directory. These rules are evaluated by the server and either allow or deny permissions to a request from a client. ACIs are part of the security subsystem offered by the Red Hat Directory Server. Knowing what data to protect during deployment is crucial in the overall security of the system. It is best to decide before deploying what data needs to be protected and how and when that data can be accessed. An ACI matrix will help you design ACIs.

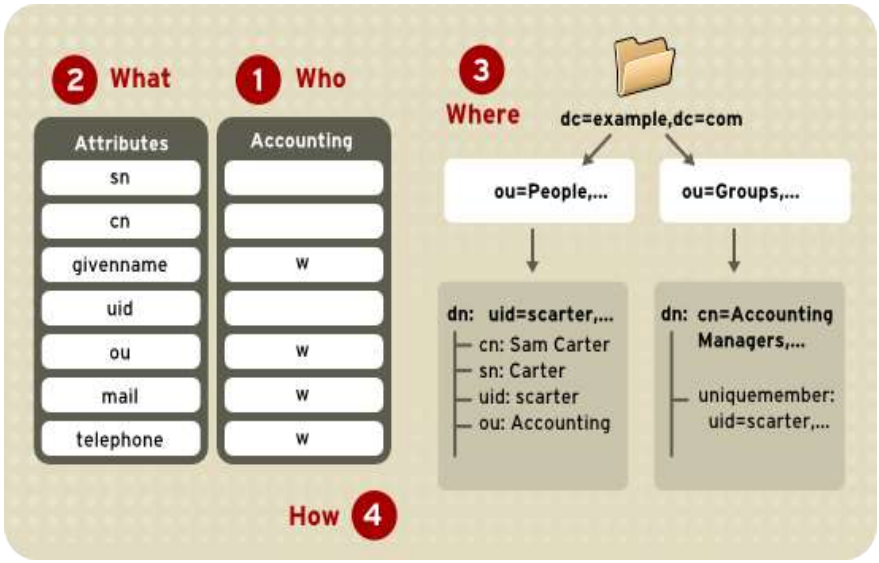
17. Have you planned your ACI matrix?

Red Hat Directory Server provides you with the ability to control access to your directory. While planning the directory deployment, you should define an access control strategy as an integral part of your overall security policy. An ACI matrix is a table of all the attributes with permissions associated by location, users, and groups. Setting up an ACI matrix helps you understand what

1 http://www.redhat.com/f/pdf/sec/rh-id-sec-solutions-wp.pdf

attributes need to be protected, how, and when. With an ACI matrix, you can set permissions for the following areas:

- The entire directory
- A particular subtree of the directory
- Specific entries in the directory
- A specific set of entry attributes
- Any entry that matches a given LDAP search filter



18. Do you want to turn on SSL for Red Hat Directory Server?[1]

The Directory Server provides security at three levels:

- At the database level (attribute encryption)
- At the content management level (ACI)
- At the network level (SSL)

To provide secure communications over the network, Red Hat Directory Server includes the LDAPS communications protocol. LDAPS is the standard LDAP protocol, but it runs on top of Secure Sockets Layer (SSL). Red Hat Directory Server also allows "spontaneous" secure connections over otherwise insecure LDAP ports, using **StartTLS** (**Transport Layer Security**). The Directory Server supports SSL/TLS to secure communications between LDAP clients and the Directory Server, between Directory Servers that are bound by a replication agreement, or between a database link and a remote database. You can use SSL/TLS with simple authentication (bind DN and password) or with certificate-based authentication. Using SSL with simple authentication ensures confidentiality and data integrity. The benefits of using a certificate to authenticate to the Directory Server instead of a bind DN and password include:

- **Improved efficiency** – With certificate-based authentication, an application will prompt once for your certificate database password and then uses that certificate for all subsequent bind or authentication operations. This makes authentication more efficient than continuously

1 Text from the *Red Hat Directory Server Administrator's Guide* at http://www.redhat.com/docs/manuals/dir-server/ag/7.1/ssl.html#1041472

providing a bind DN and password.

- **Improved security** - The use of certificate-based authentication is more secure than non-certificate bind operations. This is because certificate-based authentication uses public-key cryptography. As a result, bind credentials cannot be intercepted across the network.

The Directory Server is capable of simultaneous SSL and non-SSL communications. This means that you do not have to choose between SSL or non-SSL communications for your Directory Server; you can use both at the same time. You can also utilize the **StartTLS** extended operation to allow SSL/TLS secure communication over a regular (clear text) LDAP port.

19. Will replication be over SSL?

Directory Servers involved in replication can be configured for SSL so that all replication operations occur over an SSL connection. This helps in securing all replication data sent between master and replica servers. Digital certificates need to be installed on each of the master and replica servers. Certificates can be issued by Red Hat Certificate System, OpenSSL, or by a third party external digital certificates vendor. All SSL traffic, replication or operations, will generate additional overhead to the Red Hat Directory Server. Red Hat Directory Server is capable of using third party SSL hardware accelerators to enhance performance. Planning for these SSL overheads is crucial for an efficient deployment.

20. How will information be indexed? [1]

Proper indexing is the most important thing to improve read performance. Red Hat Directory Server uses index files to aid in searching the directory. Just like an index in a book aids in searches for words, indexes greatly improve the performance of searches in the directory databases, but they do so at the cost of slower database modification and creation operations. Indexes also cost more in terms of system resources, especially disk space. Indexes can be created on an attribute-by-attribute basis. The types of indexes created can also be configured to allow tuning based on the specific types of queries the directory is asked to perform.

The more indexes you maintain, the longer it takes the Directory Server to update the database. This is especially true for substring indexes that cause the Directory Server to generate multiple index file entries every time an attribute value is created or changed. For substring indexes, the number of index entries created is proportional to the length of the string being indexed.

One other cost to maintaining index files is the increased system resources they require. Although the impact to your system depends on how large a database you use and how many attributes exist within your database, consider these facts:

- **Index files use disk space**: The more attributes you index, the more files will be created. Also, creating approximate and substring indexes for attributes that contain long and complex strings can cause these files to quickly grow large.
- **Index files use memory**: To run more efficiently, the Directory Server tries to place as many index files in memory as possible. You can control the amount of

1 Index text from Red Hat Directory Server Admin Guide at *http://www.redhat.com/docs/manuals/dir-server/ag/7.1/index1.html#996824*

memory allowed per open index file using the maximum cache size parameter. Even so, a large number of index files will require more memory.

- **Managing index files uses CPU cycles**: Although index files will save CPU cycles during searches, maintaining indexes that are not frequently used can actually waste CPU cycles because of the need to create and manage unnecessary indexes. This is especially true for substring and approximate indexes that require parsing long, complex strings.

Conclusion

There are several phases to deploying a successful directory service using Red Hat Directory Server, and the first step is planning and analysis of the directory service deployment. These twenty questions should help to analyze the directory needs. These questions should eventually lead to more questions that should help you better understand your enterprise Red Hat Directory Server deployment. These questions are also not in any specific order. For further answers, consult with your Red Hat Directory technical architect.

Cited Works

Image, “Single Master Replication,” from “Managing Replication” in Red Hat Directory Server Administrator's Guide, version 7.1. *http://www.redhat.com/docs/manuals/dir-server/ag/7.1/replicat.html#1111921*

“Introduction to SSL” in Red Hat Directory Server Administrator's Guide, version 7.1. *http://www.redhat.com/docs/manuals/dir-server/ag/7.1/ssl.html#1041472*

"Managing Indexes" in Red Hat Directory Server Administrator's Guide, version 7.1. *http://www.redhat.com/docs/manuals/dir-server/ag/7.1/sync.html#2836267*

References

"Windows Sync," in Red Hat Directory Server Administrator's Guide, version 7.1. *http://www.redhat.com/docs/manuals/dir-server/ag/7.1/sync.html#2836267*

"Designing the Replication Process," in Red Hat Directory Deployment Guide, version 7.1. *http://www.redhat.com/docs/manuals/dir-server/deploy/7.1/rep.html#1013789*

"Designing a Secure Directory," in Red Hat Directory Deployment Guide, version 7.1. *http://www.redhat.com/docs/manuals/dir-server/deploy/7.1/aci.html#11284*

Red Hat Directory Server Schema Reference, version 7.1. *http://www.redhat.-com/docs/manuals/dir-server/schema/7.1/schemaTOC.html*