Red Hat Enterprise Linux 9

调优身份管理中的性能

优化 IdM 服务，如目录服务器、KDC 和 SSSD，以提高性能

Last Updated: 2024-09-29

Red Hat Enterprise Linux 9 调优身份管理中的性能

优化 IdM 服务，如目录服务器、KDC 和 SSSD，以提高性能

法律通告

摘要

红帽调整身份管理(IdM)，以便在大多数部署中表现良好。然而，在特定场景中，调整 IdM 组件（如复制协议、目录服务器、Kerberos 密钥分发中心(KDC)或系统安全服务守护进程(SSSD)）非常有用。

目录

对红帽文档提供反馈

我们感谢您对我们文档的反馈。让我们了解如何改进它。

通过 Jira 提交反馈（需要帐户）

1. 登录到 Jira 网站。
2. 在顶部导航栏中点 **Create**
3. 在 **Summary** 字段中输入描述性标题。
4. 在 **Description** 字段中输入您的改进建议。包括文档相关部分的链接。
5. 点对话框底部的 **Create**。

第 1 章 调优 IDM 时的重要注意事项

对身份管理组件服务进行微调，以适用于大多数部署的最佳方式。作为系统管理员，您可能想要调整 IdM 服务的性能，以适应特定环境的需求。

重要注意事项

- 每个 IdM 部署可能有不同的硬件、软件、联网、数据、工作负载和其他因素，因此每个部署都可能有其唯一性。适合一个环境的调整可能并不适合于另一个环境。
- 性能升级是一个迭代的实验过程。红帽建议一次仅调整一个变量，并监控其在环境中的影响。在通过一个变量达到预期的结果后，调整下一个变量，同时继续监控之前调整的性能。

第 2 章 硬件建议

对于性能调整，RAM 是最重要的硬件。请确定您的系统有足够可用 RAM。典型的 RAM 要求是：

- 对于 10,000 个用户和 100 个组：至少 4 GB RAM 和 4 GB 交换（swap）空间
- 对于 100,000 个用户和 50,000 个组：至少 16 GB RAM 和 4 GB swap 空间

对于较大的部署，增加 RAM 比增加磁盘空间更为有效，因为许多数据都存储在缓存中。通常，对于大型部署，添加更多 RAM 会因为有更多的缓存使系统具有更好的性能。

注意

基本用户条目或带有证书的简单主机条目大约是 5-10 kB 大小。

第 3 章 IDM 服务器性能建议

下表显示了您可以同时向 IdM 环境添加或注册的最大用户和客户端数，以确保 IdM 服务器的稳定性能。

表 3.1. 每个 IdM 服务器的最大数

操作	描述	Number
客户端注册	在注册开始失败前，您可以同时注册到 IdM 服务器的最大 IdM 客户端数。	130
添加用户	在无法添加用户前，您可以使用 **ipa user-add[]** 命令从不同的 IdM 客户端同时添加的最大用户数。 您可以使用 IdM API **batch** 命令同时添加更多用户。我们建议以 100 个用户为一批添加用户。有关 **batch** 命令的详情，请参阅 使用批处理来执行 IdM API 命令。	325
客户端身份验证	身份验证开始失败前可以同时验证的 IdM 客户端的最大数。	800
将成员添加到用户组	推荐的您可以在组中添加的成员数，而不超过向组添加新成员的时间。IdM 有一个两秒规则，来作为将成员添加到组中的正常时间段。您可以添加更多成员，但操作的时间将逐渐延长。	1500

第 4 章 IDM 中的故障转移、负载平衡和高可用性

身份管理（IdM）为 IdM 客户端提供了内置的故障转移机制，为 IdM 服务器提供了负载平衡和高可用性功能。

4.1. 客户端故障转移功能

- 默认情况下，IdM 客户端中的 **SSSD** 服务被配置为使用 DNS 中的服务（SRV）资源记录来自动决定要连接的最佳 IdM 服务器。此行为由 **/etc/sssd/sssd.conf** 文件的 **ipa_server** 参数中的 **_srv_** 选项控制：

```
[root@client ~]# cat /etc/sssd/sssd.conf

[domain/example.com]
id_provider = ipa
ipa_server = _srv_, server.example.com
...
```

  如果 IdM 服务器离线，IdM 客户端中的 SSSD 服务会自动连接到另一个 IdM 服务器。

- 如果您希望因为性能原因绕过 DNS 查找，请从 **ipa_server** 参数中删除 **_srv_** 条目，并指定客户端应该连接的 IdM 服务器，按首选顺序排列：

```
[root@client ~]# cat /etc/sssd/sssd.conf

[domain/example.com]
id_provider = ipa
ipa_server = server1.example.com, server2.example.com
...
```

4.2. 服务器端负载平衡和服务可用性

您可以通过安装多个 IdM 副本在 IdM 中实现负载平衡和高可用性：

- 如果您的网络分布在不同的地理位置，可以通过为每个数据中心配置多个 IdM 副本来缩短 IdM 客户端和最快的服务器间的路径。
- 红帽支持最多有 60 个副本的环境。
- IdM 复制机制提供主动/主动服务可用性：所有 IdM 副本的服务都同时可用。

注意

红帽建议不要将 IdM 和其它负载均衡或高可用性（HA）软件合并。

许多第三方高可用性解决方案假定使用主动/被动模式，并可能导致 IdM 服务出现不必要的中断。其他解决方案使用虚拟 IP 或每个集群服务使用一个主机名。所有这些方法通常不适用于 IdM 所提供的服务。另外，它们与 Kerberos 的集成效果也不好，从而降低了部署的整体安全性和稳定性。

第 5 章 优化副本拓扑

一个良好的副本拓扑可以对工作负载进行分散，并减少复制延迟。按照以下步骤优化副本拓扑布局。

5.1. 决定拓扑中合适数量的 IDM 副本的指南

规划 IdM 拓扑，以匹配您所在机构的要求，并确保最佳性能和服务可用性。

在每个数据中心中设置至少两个副本

在每个数据中心中至少部署两个副本，以确保一个服务器出现故障时，副本可以接管并处理请求。

为您的客户端设置足够数量的服务器

一个 IdM 服务器可为 2000 - 3000 个客户端提供服务。这假设客户端每天会多次查询服务器，但不会每分钟都查询一次。如果您期望频繁的查询，请计划更多的服务器。

设置足够数量的证书颁发机构（CA）副本

只有安装了 CA 角色的副本才能复制证书数据。如果使用 IdM CA，请确保您的环境至少有两个带有证书复制协议的 CA 副本。

在单个 IdM 域中设置最多 60 个副本

红帽支持最多有 60 个副本的环境。

5.2. 拓扑中连接 IDM 副本的指南

将每个副本连接到至少两个其他副本

这确保信息不仅在安装的初始副本和第一个服务器之间复制，而且还在其他副本之间复制。

将副本连接到最多四个其他副本（这并不是硬要求）

每个服务器有大量的复制协议不会带来很大的好处。接收副本一次只能被另外一个副本更新，而其他复制协议则处于闲置状态。每个副本有超过四个复制协议通常意味着资源不足。

注意

本建议适用于证书复制协议和域复制协议。

每个副本有四个复制协议的限制有两个例外：

- 如果某些副本没有在线或没有响应时，您需要使用故障切换路径。
- 在大型部署中，您需要特定节点间的其他直接链接。

配置大量复制协议可能会对整体性能造成负面影响：当拓扑中的多个复制协议正在发送更新时，某些副本可能会在进入更新和传出更新之间在更改日志数据库文件出现高竞争。

如果您决定每个副本使用更多复制协议，请确保您没有遇到复制问题和延迟。但请注意，但距离大及存在大量中间节点时也可能造成延迟问题。

相互连接数据中心中的副本

这样可保证数据中心中的域复制。

将每个数据中心连接到至少两个其他数据中心

这样可确保数据中心间的域复制。

至少使用一对复制协议连接数据中心

如果数据中心 A 和 B 有从 A1 到 B1 的复制协议，当存在从 A2 到 B2 的复制协议时，可确保其中一个服务器停止工作时复制可在两个数据中心之间继续。

5.3. 副本拓扑示例

您可以使用以下示例之一创建一个可靠的副本拓扑。

图 5.1. 具有四个数据中心的副本拓扑，每个数据中心具有与复制协议连接的四个服务器

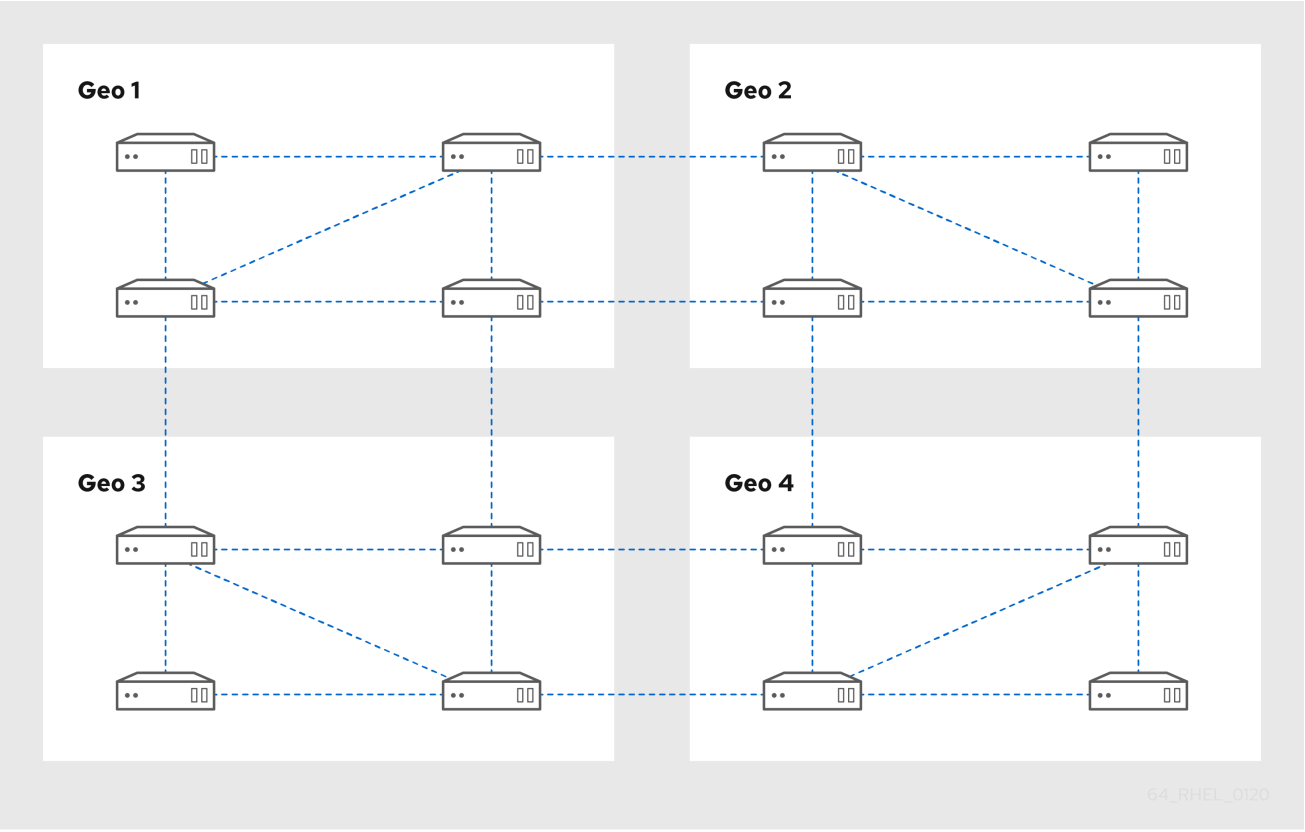
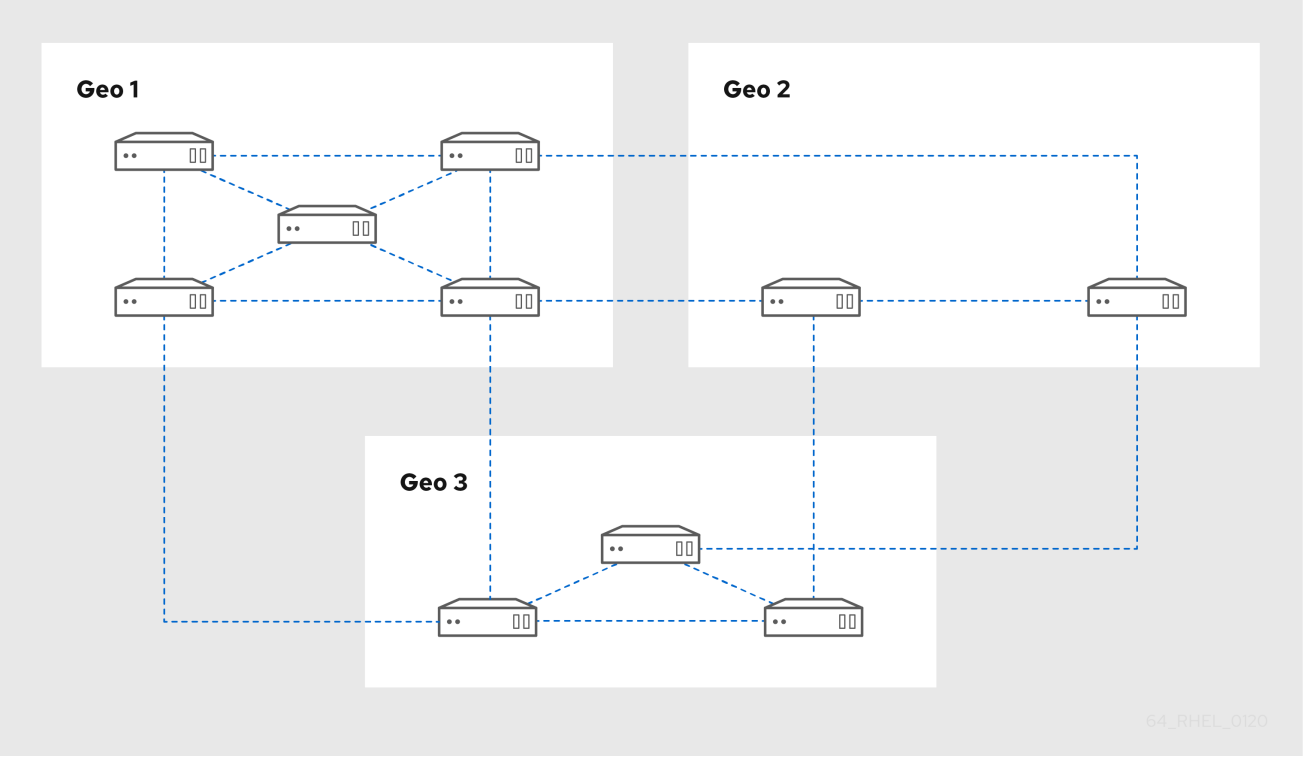
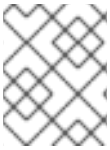


图 5.2. 具有三个数据中心的副本拓扑，每个数据中心都有不同数量的服务器，它们通过复制协议互连



5.4. 从 IDM 服务器卸载 IDM CA 服务

如果在您的拓扑中有超过四个具有 **CA 角色** 的身份管理(IdM)副本，并且由于冗余的证书复制而遇到性能问题，红帽建议您从 IdM 副本中删除冗余的 CA 服务实例。为此，您必须首先在其上重新安装 IdM 之前完全停用受影响的 IdM 副本，这一次没有 CA 服务。



注意

虽然您可以将 CA **角色添加到** IdM 副本中，但 IdM 没有提供一种只从 IdM 副本中 **删除** CA 角色的方法：**ipa-ca-install** 命令没有 **--uninstall** 选项。

先决条件

- 您已在拓扑中超过四个 IdM 服务器上安装了 IdM CA 服务。

步骤

1. 识别冗余 CA 服务，并按照在托管此服务的 IdM 副本上 卸载 IdM 服务器 中的流程操作。
2. 在同一台主机上，按照 安装 IdM 服务器：带有集成 DNS，没有 CA 中的流程操作。

5.5. 其他资源

- 规划副本拓扑。
- 管理复制拓扑.

第 6 章 调整搜索大小和时间限制

有些查询（比如请求 IdM 用户列表）可能会返回大量条目。通过调优这些搜索操作，您可以在运行 **ipa *-find** 命令时提高服务器的总体性能，例如 **ipa user-find**，并在 Web UI 中显示相应的列表。

搜索大小限制

定义从客户端 CLI 发送发送到服务器的请求或从访问 IdM Web UI 的浏览器返回的最大条目数。
默认： 100 条目。

搜索时间限制

定义服务器等待搜索运行的最长时间（以秒为单位）。搜索达到这个限制后，服务器将停止搜索并返回该时间里发现的条目。
默认： 2 秒。

如果您将值设为 **-1**，IdM 在搜索时不会应用任何限制。

重要

如果设置的搜索大小或时间限制太大，则可能会对服务器性能造成负面影响。

6.1. 在命令行中调整搜索大小和时间限制

以下流程描述了在命令行中调整搜索大小和时间限制：

- 全局
- 对于一个特定条目

步骤

1. 要在 CLI 中显示当前搜索时间和大小限制，请使用 **ipa config-show** 命令：

```
$ ipa config-show

Search time limit: 2
Search size limit: 100
```

2. 要为所有查询调整 **全局** 限制，请使用 **ipa config-mod** 命令，并添加 **--searchrecordslimit** 和 **--searchtimelimit** 选项。例如：

```
$ ipa config-mod --searchrecordslimit=500 --searchtimelimit=5
```

3. 要仅为特定查询 **暂时** 调整限制，请在命令中添加 **--sizelimit** 或 **--timelimit** 选项。例如：

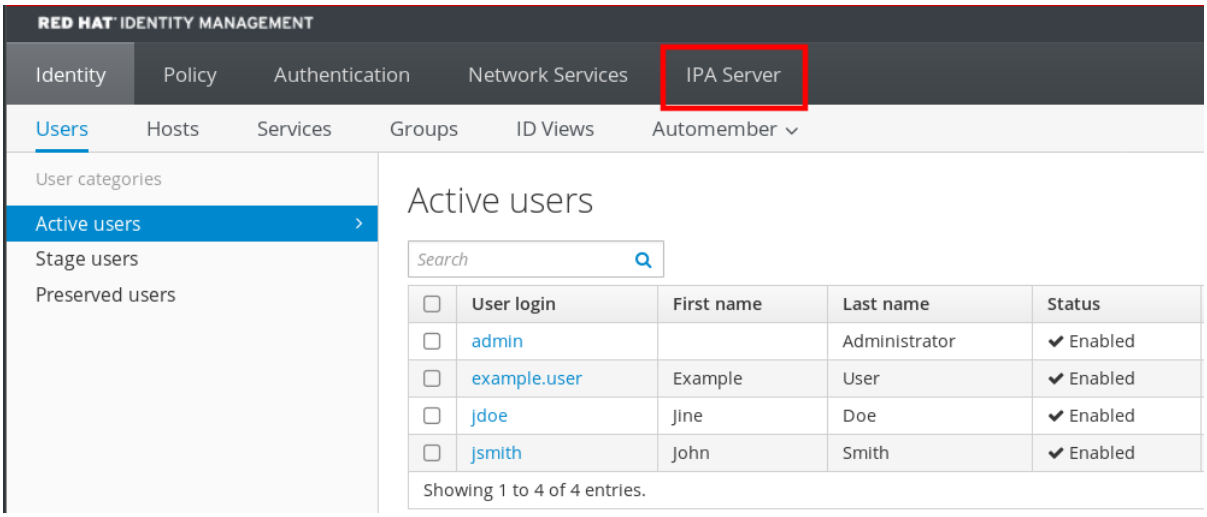
```
$ ipa user-find --sizelimit=200 --timelimit=120
```

6.2. 在 WEB UI 中调整搜索大小和时间限制

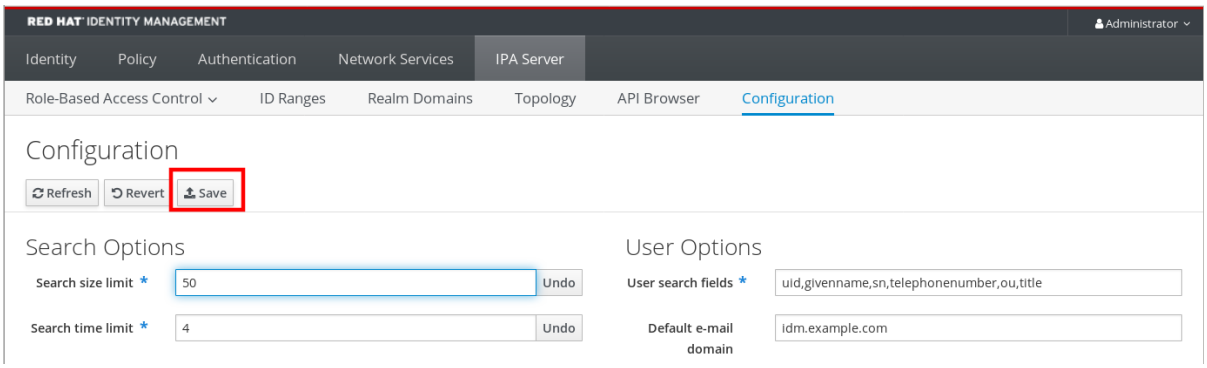
以下流程描述了在 IdM Web UI 中调整全局搜索大小和时间限制。

步骤

1. 登录到 IdM Web UI。
2. 点 **IPA Server**。



3. 在 **IPA Server** 选项卡中点 **Configuration**。
4. 在**搜索**选项区域中设置所需的值。
   默认值为：
   - 搜索大小限制：100 个条目
   - 搜索时间限值：2 秒
5. 点页面顶部的 **Save**。



第 7 章 调整 IDM 目录服务器性能

您可以通过调整 LDAP 属性来控制目录服务器的资源和行为来调整身份管理数据库的性能。

要调整目录服务器缓**存数据**的方式，请参阅以下步骤：

- 调整条目缓存大小
- 调整数据库索引缓存大小
- 重新启用条目和数据库缓存自动大小
- 调整 DN 缓存大小
- 调整规范化 DN 缓存大小

要调整 Directory 服务器的**资源限值**，请参阅以下步骤：

- 调整最大消息大小
- 调整文件描述符的最大数量
- 调整连接数据的大小
- 调整数据库锁定的最大数量
- 禁用透明巨页功能

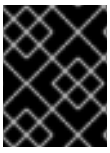
要调整对性能有最大影响的**超时**设置，请参阅以下步骤：

- 调整输入/输出块超时
- 调整闲置连接超时
- 调整复制发行超时

要安装一个带有来自 LDIF 文件中的自定义 Directory 服务器设置的 IdM 服务器或副本，请参阅以下步骤：

- 使用 LDIF 文件中的自定义数据库设置安装 IdM 服务器或副本

7.1. 调整条目缓存大小



重要

红帽建议您使用内置缓存自动缩放功能来优化性能。只有在需要与自动调整的值分离时才会更改这个值。

nsslapd-cachememsize 属性指定条目缓存的可用内存空间大小（以字节为单位）。这个属性是控制目录服务器使用的物理 RAM 最重要的值之一。

如果条目缓存太小，您可能在 Directory Server 错误日志中看到以下错误：
/var/log/dirsrv/slapd-*INSTANCE-NAME*/errors：

> REASON: entry too large (*83886080* bytes) for the import buffer size (*67108864* bytes). **Try increasing nsslapd-cachememsize.**

红帽建议在内存中安装条目缓存和数据库索引条目缓存。

默认值	**209715200** *(200 MiB)*
有效范围	**500000 - 18446744073709551615** *(500 kB - (2^{64}-1))*
条目 DN 位置	**cn=*database-name*,cn=ldbm database,cn=plugins,cn=config**

先决条件

- LDAP Directory Manager 密码

流程

1. 禁用自动缓存调整。

```
[root@server ~]# dsconf -D "cn=Directory Manager" ldap://server.example.com
backend config set --cache-autosize=0
```

2. 显示数据库后缀及其对应的后端。

```
[root@server ~]# dsconf -D "cn=Directory Manager" ldap://server.example.com
backend suffix list
cn=changelog (changelog)
dc=example,dc=com (userroot)
o=ipaca (ipaca)
```

   这个命令显示每个后缀旁的后端数据库名称。在下一步中使用后缀的数据库名称。

3. 为数据库设置条目缓存大小。这个示例将 userroot 数据库的条目缓存设置为 2GB。

```
[root@server ~]# dsconf -D "cn=Directory Manager" ldap://server.example.com
backend suffix set --cache-memsize=2147483648 userroot
```

4. 重启 Directory 服务器。

```
[root@server ~]# systemctl restart dirsrv.target
```

5. 监控 IdM 目录服务器的性能。如果它没有以理想的方式改变，请重复这个过程并将 **cache-memsize** 调整为不同的值，或者重新启用缓存自动大小。

验证

- 显示 **nsslapd-cachememsize** 属性的值，并将其设置为您所需的值。

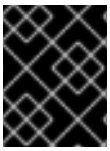
```
[root@server ~]# ldapsearch -D "cn=directory manager" -w DirectoryManagerPassword
```

```
-b "cn=userroot,cn=ldbm database,cn=plugins,cn=config" | grep nsslapd-
cachememsize
nsslapd-cachememsize: 2147483648
```

其他资源

- Directory Server 11 文档中的 nsslapd-cachememsize
- 重新启用条目和数据库缓存自动大小

7.2. 调整数据库索引缓存大小



重要

红帽建议您使用内置缓存自动缩放功能来优化性能。只有在需要与自动调整的值分离时才会更改这个值。

nsslapd-dbcachesize 属性控制数据库索引使用的内存量。这个缓存大小对 Directory 服务器性能的影响比条目缓存大小的影响要小。但是如果在设定了条目缓存大小后有可用的 RAM，红帽建议增加分配给数据库缓存的内存量。

数据库缓存限制为 1.5 GB RAM，因为更高的值并不会提高性能。

默认值	**10000000** *(10 MB)*
有效范围	**500000 - 1610611911** *(500 kB - 1.5GB)*
条目 DN 位置	**cn=config,cn=ldbm database,cn=plugins,cn=config**

先决条件

- LDAP Directory Manager 密码

流程

1. 禁用自动缓存调整，并设置数据库缓存大小。这个示例将数据库缓存设置为 256MB。

```
[root@server ~]# dsconf -D "cn=Directory Manager" ldap://server.example.com
backend config set --cache-autosize=0 --dbcachesize=268435456
```

2. 重启 Directory 服务器。

```
[root@server ~]# systemctl restart dirsrv.target
```

3. 监控 IdM 目录服务器的性能。如果它没有以理想的方式改变，请重复这个过程并将 **dbcachesize** 调整为不同的值，或者重新启用缓存自动大小。

验证

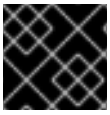
- 显示 **nsslapd-dbcachesize** 属性的值，并将其设置为您所需的值。

```
[root@server ~]# ldapsearch -D "cn=directory manager" -w DirectoryManagerPassword
-b "cn=config,cn=ldbm database,cn=plugins,cn=config" | grep nsslapd-dbcachesize
nsslapd-dbcachesize: 2147483648
```

其他资源

- Directory Server 11 文档中的 nsslapd-dbcachesize
- 重新启用条目和数据库缓存自动大小

7.3. 重新启用数据库和条目缓存自动大小



重要

红帽建议您使用内置缓存自动缩放功能来优化性能。红帽不推荐手动设置缓存大小。

默认情况下，IdM Directory 服务器会自动决定数据库缓存和条目缓存的最佳大小。自动的设置会忽略一部分可用 RAM，并在实例启动时根据服务器的硬件资源优化这两个缓存的大小。

使用这个流程取消自定义数据库缓存和条目缓存值，并将缓存自动大小功能恢复到默认值。

nsslapd-cache-autosize	这个设置控制为自动分配数据库和条目缓存而分配的可用 RAM 量。**0** 代表禁用自动大小。
默认值	**10** *（10% 的可用 RAM）*
有效范围	**0 - 100**
条目 DN 位置	**cn=config,cn=ldbm database,cn=plugins,cn=config**

nsslapd-cache-autosize-split	这个值设定由 **nsslapd-cache-autosize** 决定用于数据库缓存的可用内存百分比。剩余百分比用于条目缓存。
默认值	**25** *（25% 用于数据库缓存，60% 用于条目缓存 ）*
有效范围	**0 - 100**
条目 DN 位置	**cn=config,cn=ldbm database,cn=plugins,cn=config**

先决条件

- 之前您已经禁用了数据库和条目缓存自动扩展。

流程

1. 停止 Directory 服务器。

```
[root@server ~]# systemctl stop dirsrv.target
```

2. 在进行任何进一步的修改前，请备份 **/etc/dirsrv/*slapd-instance_name*/dse.ldif** 文件。

```
[root@server ~]# *cp /etc/dirsrv/slapd-instance_name/dse.ldif \
    /etc/dirsrv/slapd-instance_name/dse.ldif.bak.$(date "+%F_%H-%M-%S")
```

3. 编辑 **/etc/dirsrv/*slapd-instance_name*/dse.ldif** 文件：

   a. 设置用于数据库的可用系统 RAM 百分比，恢复默认的 10% 可用 RAM。

```
nsslapd-cache-autosize: 10
```

   b. 将可用系统 RAM 中数据库缓存使用的百分比设置为默认的 25%：

```
nsslapd-cache-autosize-split: 25
```

4. 将更改保存到 **/etc/dirsrv/*slapd-instance_name*/dse.ldif** 文件。

5. 启动 Directory 服务器。

```
[root@server ~]# systemctl start dirsrv.target
```

验证

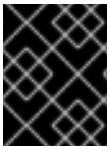
- 显示 **nsslapd-cache-autosize** 和 **nsslapd-cache-autosize-split** 属性的值，并验证它们已设置为您所需的值。

```
[root@server ~]# ldapsearch -D "cn=directory manager" -w DirectoryManagerPassword
-b "cn=config,cn=ldbm database,cn=plugins,cn=config" | grep nsslapd-cache-autosize
nsslapd-cache-autosize: *10
nsslapd-cache-autosize-split: 25
```

其他资源

- Directory Server 11 文档中的 nsslapd-cache-autosize

7.4. 调整 DN 缓存大小



重要

红帽建议您使用内置缓存自动缩放功能来优化性能。只有在需要与自动调整的值分离时才会更改这个值。

nsslapd-dncachememsize 属性指定可辨识名称（DN）缓存的可用内存空间大小（以字节为单位）。DN 缓存与数据库的条目缓存类似，但其表只存储条目 ID 和条目 DN，这样可加快查找 **rename** 和 **moddn** 操作。

默认值	**10485760** *(10 MB)*
有效范围	**500000 - 18446744073709551615** *(500 kB - (2^{64}-1))*
条目 DN 位置	**cn=*database-name*,cn=ldbm database,cn=plugins,cn=config**

先决条件

- LDAP Directory Manager 密码

流程

1. 可选：显示数据库后缀，及其相应的数据库名称。

```
[root@server ~]# dsconf -D "cn=Directory Manager" ldap://server.example.com backend suffix list
dc=example,dc=com (userroot)
```

这个命令显示每个后缀旁的后端数据库名称。在下一步中使用后缀的数据库名称。

2. 为数据库设置 DN 缓存大小。这个示例将 DN 缓存设置为 20MB。

```
[root@server ~]# dsconf -D "cn=Directory Manager" ldap://server.example.com backend suffix set --dncache-memsize=20971520 userroot
```

3. 重启 Directory 服务器。

```
[root@server ~]# systemctl restart dirsrv.target
```

4. 监控 IdM 目录服务器的性能。如果它没有以理想的方式改变，请重复这个过程并将 **dncache-memsize** 调整为不同的值，或者返回到默认值 10MB。

验证

- 显示 **nsslapd-dncachememsize** 属性的新值，并将其设置为您所需的值。

```
[root@server ~]# ldapsearch -D "cn=directory manager" -w DirectoryManagerPassword -b "cn=userroot,cn=ldbm database,cn=plugins,cn=config" | grep nsslapd-dncachememsize
nsslapd-dncachememsize: 20971520
```

其他资源

- Directory Server 11 文档中的 nsslapd-dncachememsize

7.5. 调整规范化 DN 缓存大小

重要

红帽建议您使用内置缓存自动缩放功能来优化性能。只有在需要与自动调整的值分离时才会更改这个值。

nsslapd-ndn-cache-max-size 属性控制存储规范化可分辨名称(NDN)的缓存的大小（以字节为单位）。增加这个值将在内存中保留更频繁使用的 DN。

默认值	**20971520** *(20 MB)*
有效范围	**0 - 2147483647**
条目 DN 位置	**cn=config**

先决条件

- LDAP Directory Manager 密码

流程

1. 确保启用了 NDN 缓存。

```
[root@server ~]# dsconf -D "cn=Directory Manager" ldap://server.example.com config
get nsslapd-ndn-cache-enabled
Enter password for cn=Directory Manager on ldap://server.example.com:
nsslapd-ndn-cache-enabled: on
```

如果缓存**关闭**，使用以下命令启用它。

```
[root@server ~]# dsconf -D "cn=Directory Manager" ldap://server.example.com config
replace nsslapd-ndn-cache-enabled=on
Enter password for cn=Directory Manager on ldap://server.example.com:
Successfully replaced "nsslapd-ndn-cache-enabled"
```

2. 检索 **nsslapd-ndn-cache-max-size** 参数的当前值，并在需要恢复任何调整前记录它。提示时输入 Directory Manager 密码。

```
[root@server ~]# dsconf -D "cn=Directory Manager" ldap://server.example.com config
get nsslapd-ndn-cache-max-size
Enter password for cn=Directory Manager on ldap://server.example.com:
nsslapd-ndn-cache-max-size: 20971520
```

3. 修改 **nsslapd-ndn-cache-max-size** 属性的值。这个示例将值增加到 **41943040** (40 MB)。

```
[root@server ~]# dsconf -D "cn=Directory Manager" ldap://server.example.com config
replace nsslapd-ndn-cache-max-size=41943040
```

4. 监控 IdM 目录服务器的性能。如果它没有以理想的方式改变，请重复这个过程并将 **nsslapd-ndn-cache-max-size** 调整为不同的值，或者重新启用缓存自动大小。

验证

- 显示 **nsslapd-ndn-cache-max-size** 属性的新值，并将其设置为您所需的值。

```
[root@server ~]# dsconf -D "cn=Directory Manager" ldap://server.example.com config
get nsslapd-ndn-cache-max-size
Enter password for cn=Directory Manager on ldap://server.example.com:
nsslapd-ndn-cache-max-size: 41943040
```

其他资源

- Directory Server 11 文档中的 nsslapd-ndn-cache-max-size。

7.6. 调整最大的消息大小

nsslapd-maxbersize 属性以字节为单位设定传入消息或 LDAP 请求的最大值。限制请求大小可防止某种形式拒绝服务攻击。

如果最大消息的大小太小，您可能会在 Directory Server 错误日志中看到以下错误：**/var/log/dirsrv/slapd-*INSTANCE-NAME*/errors**：

> Incoming BER Element was too long, **max allowable is 2097152 bytes. Change the nsslapd-maxbersize attribute in cn=config to increase.**

限制适用于 LDAP 请求的总大小。例如，如果请求要添加条目，并且请求中的条目大于配置的值或默认值，则拒绝添加请求。但是，这个限制不适用于复制进程。在更改此属性前请小心。

默认值	**2097152** *(2 MB)*
有效范围	**0 - 2147483647** *(0 到 2 GB)*
条目 DN 位置	**cn=config**

先决条件

- LDAP Directory Manager 密码

流程

1. 检索 **nsslapd-maxbersize** 参数的当前值，并在需要恢复任何调整前记录它。提示时输入 Directory Manager 密码。

```
[root@server ~]# dsconf -D "cn=Directory Manager" ldap://server.example.com config
get nsslapd-maxbersize
Enter password for cn=Directory Manager on ldap://server.example.com:
nsslapd-maxbersize: 2097152
```

2. 修改 **nsslapd-maxbersize** 属性的值。这个示例将值增加到 **4194304**，4MB。

```
[root@server ~]# dsconf -D "cn=Directory Manager" ldap://server.example.com config
replace nsslapd-maxbersize=4194304
```

3. 身份验证为 Directory Manager 以进行配置更改。

```
Enter password for cn=Directory Manager on ldap://server.example.com:
Successfully replaced "nsslapd-maxbersize"
```

4. 监控 IdM 目录服务器的性能。如果它没有以希望的方式改变，请重复这个过程并将 **nsslapd-maxbersize** 调整为不同的值，或者回到默认值 **2097152**。

验证

- 显示 **nsslapd-maxbersize** 属性的值，并将其设置为您所需的值。

```
[root@server ~]# dsconf -D "cn=Directory Manager" ldap://server.example.com config get nsslapd-maxbersize
Enter password for cn=Directory Manager on ldap://server.example.com:
nsslapd-maxbersize: 4194304
```

其他资源

- Directory Server 11 文档中的 nsslapd-maxbersize (Maximum Message Size)

7.7. 调整文件描述符的最大数量

可以为 **/etc/systemd/system.conf** 文件中的 **DefaultLimitNOFILE** 参数定义一个值。具有 **root** 特权的管理员可使用 **setrlimit** 命令将 **ns-slapd** 进程的 **DefaultLimitNOFILE** 参数设置为较低的值。然后，这个值优先于 **/etc/systemd/system.conf** 中的值，并作为 **nsslapd-maxdescriptors** 属性的值被身份管理(IdM)目录服务器(DS)所接受。

nsslapd-maxdescriptors 属性设置 IdM LDAP 使用的独立于平台的文件描述符的最大值。文件描述符用于客户端连接、日志文件、套接字和其他资源。

如果没有在 **/etc/systemd/system.conf** 中或被 **setrlimit** 定义值，则 IdM DS 将 **nsslapd-maxdescriptors** 属性设置为 1048576。

如果 IdM DS 管理员稍后决定为 **nsslapd-maxdescriptors** 手动设置新值，则 IdM DS 将 **setrlimit** 或 **/etc/systemd/system.conf** 中定义的值与新值进行比较：

- 如果 **nsslapd-maxdescriptors** 的新值比本地定义的值大，则服务器拒绝新值设置，并继续将本地限制值强制为高水位线值。
- 如果新值低于本地定义的值，则将使用新值。

这个流程描述了如何为 **nsslapd-maxdescriptors** 设置新值。

先决条件

- LDAP Directory Manager 密码

流程

1. 检索 **nsslapd-maxdescriptors** 参数的当前值，并在需要恢复任何调整前记录它。提示时输入 Directory Manager 密码。

```
[root@server ~]# dsconf -D "cn=Directory Manager" ldap://server.example.com config get nsslapd-maxdescriptors
Enter password for cn=Directory Manager on ldap://server.example.com:
```

```
nsslapd-maxdescriptors: 4096
```

2. 修改 **nsslapd-maxdescriptors** 属性的值。这个示例将值增加到 **8192**。

```
[root@server ~]# dsconf -D "cn=Directory Manager" ldap://server.example.com config
replace nsslapd-maxdescriptors=8192
```

3. 身份验证为 Directory Manager 以进行配置更改。

```
Enter password for cn=Directory Manager on ldap://server.example.com:
Successfully replaced "nsslapd-maxdescriptors"
```

4. 监控 IdM 目录服务器的性能。如果它没有以理想的方式改变，请重复这个过程将 **nsslapd-maxdescriptors** 调整为不同的值，或者重新使用默认的 **4096**。

验证

- 显示 **nsslapd-maxdescriptors** 属性的值，并验证它已设置为您所需的值。

```
[root@server ~]# dsconf -D "cn=Directory Manager" ldap://server.example.com config
get nsslapd-maxdescriptors
Enter password for cn=Directory Manager on ldap://server.example.com:
nsslapd-maxdescriptors: 8192
```

其他资源

- 目录服务器 12 文档中的 nsslapd-maxdescriptors (文件描述符的最大值)

7.8. 调整连接数据的大小

侦听服务设定可用于接收进入的连接的套接字的数量。**nsslapd-listen-backlog-size** 值代表，在拒绝连接前 **sockfd** socket 队列的最大长度。

如果您的 IdM 环境处理大量连接，请考虑增加 **nsslapd-listen-backlog-size** 的值。

默认值	**128** 个队列插槽
有效范围	**0 - 9223372036854775807**
条目 DN 位置	**cn=config**

先决条件

- LDAP Directory Manager 密码

流程

1. 检索 **nsslapd-listen-backlog-size** 参数的当前值，并在进行任何调整前记录它，以备需要恢复时使用。提示时输入 Directory Manager 密码。

```
[root@server ~]# dsconf -D "cn=Directory Manager" ldap://server.example.com config
```

```
get nsslapd-listen-backlog-size
Enter password for cn=Directory Manager on ldap://server.example.com:
nsslapd-listen-backlog-size: 128
```

2. 修改 **nsslapd-listen-backlog-size** 属性的值。这个示例将值增加到 **192**。

```
[root@server ~]# dsconf -D "cn=Directory Manager" ldap://server.example.com config
replace nsslapd-listen-backlog-size=192
```

3. 身份验证为 Directory Manager 以进行配置更改。

```
Enter password for cn=Directory Manager on ldap://server.example.com:
Successfully replaced "nsslapd-listen-backlog-size"
```

验证

- 显示 **nsslapd-listen-backlog-size** 属性的值，并验证它已设置为您所需的值。

```
[root@server ~]# dsconf -D "cn=Directory Manager" ldap://server.example.com config
get nsslapd-listen-backlog-size
Enter password for cn=Directory Manager on ldap://server.example.com:
nsslapd-listen-backlog-size: 192
```

其他资源

- Directory Server 11 文档中的 nsslapd-listen-backlog-size

7.9. 调整数据库锁定的最大数量

锁定机制控制目录服务器进程可以同时运行多少个副本，**nsslapd-db-locks** 参数设置最大锁定数。

如果您在 **/var/log/dirsrv/slapd-*instance_name*/errors** 日志文件中看到以下错误信息，请增加最大锁定数：

```
libdb: Lock table is out of available locks
```

默认值	**50000** 个锁定
有效范围	**0 - 2147483647**
条目 DN 位置	**cn=bdb,cn=config,cn=ldbm database,cn=plugins,cn=config**

先决条件

- LDAP Directory Manager 密码

流程

1. 检索 **nsslapd-db-locks** 参数的当前值，并在需要恢复任何调整前记录它。

```
[root@server ~]# ldapsearch -D "cn=directory manager" -w DirectoryManagerPassword
-b "cn=bdb,cn=config,cn=ldbm database,cn=plugins,cn=config" | grep nsslapd-db-
locks
nsslapd-db-locks: 50000
```

2. 修改 **locks** 属性的值。这个示例将值加倍到 **100000** 个锁定。

```
[root@server ~]# dsconf -D "cn=Directory Manager" ldap://server.example.com
backend config set --locks=100000
```

3. 身份验证为 Directory Manager 以进行配置更改。

```
Enter password for cn=Directory Manager on ldap://server.example.com:
Successfully updated database configuration
```

4. 重启 Directory 服务器。

```
[root@server ~]# systemctl restart dirsrv.target
```

验证

- 显示 **nsslapd-db-locks** 属性的值，并将其设置为您所需的值。

```
[root@server ~]# ldapsearch -D "cn=directory manager" -w DirectoryManagerPassword
-b "cn=bdb,cn=config,cn=ldbm database,cn=plugins,cn=config" | grep nsslapd-db-
locks
nsslapd-db-locks: 100000
```

其他资源

- Directory Server 11 文档中的 nsslapd-db-locks

7.10. 禁用透明巨页功能

默认在 RHEL 上启用了透明巨页(THP) Linux 内存管理功能。THP 功能会降低 IdM 目录服务器(DS)性能，因为 DS 有稀疏内存访问模式。

如何禁用此功能，请参阅 Red Hat Directory Server 文档中的 禁用透明巨页功能。

其他资源

- RHDS 上透明巨页(THP)的负面影响

7.11. 调整输入/输出块超时

nsslapd-ioblocktimeout 属性代表一个时间（以毫秒为单位），在经过这个时间后到停滞的 LDAP 客户端的连接将关闭。当 LDAP 客户端没有为读或写操作进行任何 I/O 处理时，它被视为已停止工作。

降低 **nsslapd-ioblocktimeout** 属性的值可以更早地释放连接。

默认值	**10000** 毫秒

有效范围	**0 - 2147483647**
条目 DN 位置	**cn=config**

先决条件

- LDAP Directory Manager 密码

流程

1. 检索 **nsslapd-ioblocktimeout** 参数的当前值，并在需要恢复任何调整前记录它。提示时输入 Directory Manager 密码。

```
[root@server ~]# dsconf -D "cn=Directory Manager" ldap://server.example.com config get nsslapd-ioblocktimeout
Enter password for cn=Directory Manager on ldap://server.example.com:
nsslapd-ioblocktimeout: 10000
```

2. 修改 **nsslapd-ioblocktimeout** 属性的值。这个示例将值降低为 **8000**。

```
[root@server ~]# dsconf -D "cn=Directory Manager" ldap://server.example.com config replace nsslapd-ioblocktimeout=8000
```

3. 身份验证为 Directory Manager 以进行配置更改。

```
Enter password for cn=Directory Manager on ldap://server.example.com:
Successfully replaced "nsslapd-ioblocktimeout"
```

4. 监控 IdM 目录服务器的性能。如果它没有以理想的方式改变，请重复这个过程将 **nsslapd-ioblocktimeout** 调整为不同的值，或者重新使用默认的 **10000**。

验证

- 显示 **nsslapd-ioblocktimeout** 属性的值，并将其设置为您所需的值。

```
[root@server ~]# dsconf -D "cn=Directory Manager" ldap://server.example.com config get nsslapd-ioblocktimeout
Enter password for cn=Directory Manager on ldap://server.example.com:
nsslapd-idletimeout: 8000
```

其他资源

- Directory Server 11 文档中的 nsslapd-ioblocktimeout (IO Block Time Out)

7.12. 调整闲置连接超时

nsslapd-idletimeout 属性以秒为单位设置闲置 LDAP 客户端连接被 IdM 服务器关闭的时间长度（以秒为单位）。**0** 表示服务器永远不会关闭闲置连接。

红帽建议调整这个值，从而使停滞的连接关闭，但活跃的连接不会在不适当的情况下被关闭。

默认值	**3600** 秒 *(1 小时)*
有效范围	**0 - 2147483647**
条目 DN 位置	**cn=config**

先决条件

- LDAP Directory Manager 密码

流程

1. 检索 **nsslapd-idletimeout** 参数的当前值，并在需要恢复任何调整前记录它。提示时输入 Directory Manager 密码。

```
[root@server ~]# dsconf -D "cn=Directory Manager" ldap://server.example.com config
get nsslapd-idletimeout
Enter password for cn=Directory Manager on ldap://server.example.com:
nsslapd-idletimeout: 3600
```

2. 修改 **nsslapd-idletimeout** 属性的值。这个示例将值降低为 **1800** （30 分钟）。

```
[root@server ~]# dsconf -D "cn=Directory Manager" ldap://server.example.com config
replace nsslapd-idletimeout=1800
```

3. 身份验证为 Directory Manager 以进行配置更改。

```
Enter password for cn=Directory Manager on ldap://server.example.com:
Successfully replaced "nsslapd-idletimeout"
```

4. 监控 IdM 目录服务器的性能。如果它没有以理想的方式改变，请重复这个过程将 **nsslapd-idletimeout** 调整为不同的值，或者重新使用默认的 **3600**。

验证

- 显示 **nsslapd-idletimeout** 属性的值，并将其设置为您所需的值。

```
[root@server ~]# dsconf -D "cn=Directory Manager" ldap://server.example.com config
get nsslapd-idletimeout
Enter password for cn=Directory Manager on ldap://server.example.com:
nsslapd-idletimeout: 3600
```

其他资源

- Directory Server 11 文档中的 nsslapd-idletimeout (Default Idle Timeout)

7.13. 调整复制发行超时

IdM 副本在带有另一个副本的复制会话中被锁定。在一些环境中，因为大型更新或网络拥塞导致副本长时间锁定，这会增加复制延迟。

您可以通过调整 **repl-release-timeout** 参数在固定时间后发布副本。红帽建议将此值设置为 **30** 到 **120** 之间：

- 如果值设置过低，则副本会持续重新分配另一个，并且副本无法发送较大的更新。
- 较长的超时可以改进高流量的情况。在高流量的情况中，一个服务器可以在较长的时间内独家访问一个副本是最佳的，但如果高于 **120** 秒则会减慢复制速度。

默认值	**60** 秒
有效范围	**0 - 2147483647**
推荐的范围	**30 - 120**

先决条件

- LDAP Directory Manager 密码

流程

1. 显示数据库后缀及其对应的后端。

```
[root@server ~]# dsconf -D "cn=Directory Manager" ldap://server.example.com
backend suffix list
cn=changelog (changelog)
dc=example,dc=com (userroot)
o=ipaca (ipaca)
```

   这个命令会在后缀旁的显示后端数据库名称。在下一步中使用后缀名称。

2. 修改主 userroot 数据库的 **repl-release-timeout** 属性的值。这个示例将值增加到 **90** 秒。

```
[root@server ~]# dsconf -D "cn=Directory Manager" ldap://server.example.com
replication set --suffix="dc=example,dc=com" --repl-release-timeout=90
```

3. 身份验证为 Directory Manager 以进行配置更改。

```
Enter password for cn=Directory Manager on ldap://server.example.com:
Successfully replaced "repl-release-timeout"
```

4. 可选：如果您的 IdM 环境使用 IdM 证书颁发机构(CA)，您可以修改 CA 数据库的 **repl-release-timeout** 属性的值。这个示例将值增加到 **90** 秒。

```
[root@server ~]# dsconf -D "cn=Directory Manager" ldap://server.example.com replication
set --suffix="o=ipaca" --repl-release-timeout=90
Enter password for cn=Directory Manager on ldap://server.example.com:
Successfully replaced "repl-release-timeout"
```

5. 重启 Directory 服务器。

```
[root@server ~]# systemctl restart dirsrv.target
```

6. 监控 IdM 目录服务器的性能。如果它没有以理想的方式改变，请重复这个过程并将 **repl-release-timeout** 调整为不同的值，或者返回使用默认值 **60** 秒。

验证

- 显示 **nsds5ReplicaReleaseTimeout** 属性的值，并验证它已设置为您的所需值。

```
[root@server ~]# ldapsearch -D "cn=directory manager" -w DirectoryManagerPassword
-b "cn=replica,cn=dc\3Dexample\2Cdc\3Dcom,cn=mapping tree,cn=config" | grep
nsds5ReplicaReleaseTimeout
nsds5ReplicaReleaseTimeout: 90
```

注意

本例中后缀的可辨识名称为 **dc=example,dc=com**，但等号(=)和逗号(,)必须在 **ldapsearch** 命令中进行转义。

使用以下转义字符将后缀 DN 转换为 **cn=dc\3Dexample\2Cdc\3Dcom**：

- **\3D** 替换 **=**
- **\2C** 替换 **,**

其他资源

- Directory Server 11 文档中的 nsDS5ReplicaReleaseTimeout

7.14. 使用 LDIF 文件中的自定义数据库设置安装 IDM 服务器或副本

您可以使用活动目录数据库的自定义设置安装 IdM 服务器和 IdM 副本。以下流程演示了如何使用数据库设置创建 LDAP 数据交换格式(LDIF)文件，以及如何将这些设置传递给 IdM 服务器和副本安装命令。

先决条件

- 您已确定了可改进 IdM 环境性能的自定义目录服务器设置。请参阅 调整 IdM 目录服务器性能。

流程

1. 使用自定义数据库设置，创建一个 LDIF 格式的文本文件。使用短划线(-)分隔 LDAP 属性修改。这个示例为空闲超时和最大文件描述符设置了非默认值。

```
dn: cn=config
changetype: modify
replace: nsslapd-idletimeout
nsslapd-idletimeout=1800
-
replace: nsslapd-maxdescriptors
nsslapd-maxdescriptors=8192
```

2. 使用 **--dirsrv-config-file** 参数将 LDIF 文件传递给安装脚本。

   a. 要安装 IdM 服务器：

```
# ipa-server-install --dirsrv-config-file filename.ldif
```

-

b. 要安装 IdM 副本：

```
# ipa-replica-install --dirsrv-config-file filename.ldif
```

其他资源

- **ipa-server-install** 和 **ipa-replica-install** 命令的选项

7.15. 其他资源

- Directory Server 11 性能调优指南

第 8 章 调整 KDC 的性能

以下小节介绍了如何调整 Kerberos 密钥分发中心(KDC)的性能，它负责验证用户、主机和服务。

8.1. 调整 KDC 侦听队列的长度

您可以通过在 **/var/kerberos/krb5kdc/kdc.conf** 文件的 **[kdcdefaults]** 部分中设置 **kdc_tcp_listen_backlog** 选项，来调整 KDC 守护进程的监听队列长度的大小。对于某些有大量 Kerberos 流量的 IdM 部署，默认值 **5** 可能太低，但如果设置的值太高会降低性能。

默认值	**5**
有效范围	**1 - 10**

流程

1. 在文本编辑器中打开 **/var/kerberos/krb5kdc/kdc.conf** 文件。
2. 将 TCP 侦听功能设置为所需值，如 **7**。

```
[kdcdefaults]
 ...
 kdc_tcp_listen_backlog = 7
```

3. 保存并关闭 **/var/kerberos/krb5kdc/kdc.conf** 文件。
4. 重启 KDC 来加载新设置。

8.2. 每个域控制 KDC 行为的选项

为了跟踪每个 Kerberos 域的锁定和解锁用户帐户，KDC 会在每个成功和身份验证失败后写入其数据库。通过调整 **/etc/krb5.conf** 文件的 **[dbmodules]** 部分中的以下选项，您可以最大程度减少 KDC 写入信息的频率来提高性能。

disable_last_success

如果设置为 **true**，这个选项会阻止 KDC 更新到需要预身份验证的主条目的 **Last successful authentication** 字段。

默认值	**false**
有效范围	**true** 或 **false**

disable_lockout

如果设置为 **true**，这个选项会阻止 KDC 更新到需要预身份验证的主条目的 **Last failed authentication** 和 **Failed password attempts** 字段。设置此标志可能会提高性能，但禁用帐户锁定可能会被视为安全风险。

默认值	**false**
有效范围	**true** 或 **false**

其他资源

- 根据每个域（realm）调整 KDC 设置

8.3. 根据每个域（REALM）调整 KDC 设置

这个过程调整每个 Kerberos 域的 KDC 行为。

流程

1. 在文本编辑器中打开 **/etc/krb5.conf** 文件。
2. 在 **[dbmodules]** 部分中指定任意选项及其所需值，并在相应的 Kerberos 域中指定。在本例中，您要为 **EXAMPLE.COM** Kerberos 域设置 **disable_last_success** 变量。

```
[dbmodules]
   EXAMPLE.COM = {
      disable_last_success = true
   }
```

3. 保存并关闭 **/etc/krb5.conf** 文件。
4. 重启 KDC 来加载新设置。

其他资源

- 每个域控制 KDC 行为的选项

8.4. 调整 KRB5KDC 进程的数量

按照以下流程手动调整密钥分发中心(KDC)开始处理传入连接的进程的数量。

默认情况下，IdM 安装程序会检测 CPU 内核数，并在 **/etc/sysconfig/krb5kdc** 文件中定义值。例如，该文件可能包含以下条目：

```
KRB5KDC_ARGS='-w 2'
[...]
```

在这个示例中，**KRB5KDC_ARGS** 参数设为 **-w 2**，KDC 会启动两个独立的进程来处理来自主进程的进入连接。您可能想要调整这个值，特别是在虚拟环境中，您可以根据您的要求轻松添加或删除虚拟 CPU 的数量。为了防止性能问题，甚至 IdM 服务器因为端口 88 上的不断增加 TCP/IP 队列而变得没有响应，请通过手动将 **KRB5KDC_ARGS** 参数设置为更高值来模拟更多进程。

流程

1. 在文本编辑器中打开 **/etc/sysconfig/krb5kdc** 文件。
2. 指定 **KRB5KDC_ARGS** 参数的值。在本例中，您要将进程数设置为 10：

```
KRB5KDC_ARGS='-w 10'
[...]
```

3. 保存并关闭 **/etc/sysconfig/krb5kdc** 文件。

4. 重新载入 systemd 配置：

```
# systemctl daemon-reload
```

5. 重启 **krb5kdc** 服务：

```
# systemctl restart krb5kdc.service
```

注意

您可以使用 IdM Healthcheck 工具来验证 KDC 是否已配置为使用 worker 进程的最佳数量。请参阅 使用 IdM Healthcheck 验证 KDC worker 进程的最佳数量 。

8.5. 其他资源

- MIT Kerberos 文档 - kdc.conf.

第 9 章 为大型 IDM-AD 信任部署调整 SSSD 性能

对于系统安全服务守护进程 (SSSD)，检索用户和组群信息会涉及大量数据操作，特别是在带有信任到大型 Active Directory (AD) 域的 IdM 部署中。提高此性能的方法是，调整 SSSD 从身份提供程序检索哪些信息，以及进行多久。

9.1. 为大型 IDM-AD 信任部署在 IDM 服务器中调整 SSSD

此流程对 IdM 服务器中的 SSSD 服务配置应用调整选项，以改进从大型 AD 环境检索信息时的响应时间。

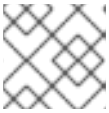
先决条件

- 您需要 **root** 权限来编辑 **/etc/sssd/sssd.conf** 配置文件。

流程

1. 在文本编辑器中打开 **/etc/sssd/sssd.conf** 配置文件。
2. 在您的身份管理(IdM)域的 **[domain]** 部分中添加以下选项：

```
[domain/idm.example.com]
ignore_group_members = true
subdomain_inherit = ignore_group_members
...
```



注意

subdomain_inherit 选项中列出的设置适用于主(IdM)域和可信 AD 域。

3. 保存并关闭服务器上的 **/etc/sssd/sssd.conf** 文件。
4. 重启 SSSD 服务以载入配置更改。

```
[root@client ~]# systemctl restart sssd
```

其它资源

- 为大型 IdM-AD 信任部署在 IdM 服务器和客户端中调整 SSSD 的选项

9.2. 在 IDM 服务器中调整 IPA-EXTDOM 插件的配置超时

IdM 客户端无法直接从 Active Directory(AD)接收用户和组的信息，因此 IdM 服务器使用 **ipa-extdom** 插件接收 AD 用户和组的信息，并将这些信息转发到请求的客户端。

ipa-extdom 插件向 SSSD 发送有关 AD 用户的数据的请求。如果信息不在 SSSD 缓存中，SSSD 会从 AD 域控制器(DC)请求数据。您可以调整 config 超时值，它定义 **ipa-extdom** 插件在插件取消连接前等待 SSSD 的回复，并将超时错误返回给调用者。默认值为 10000 毫秒（10 秒）。

以下示例将配置超时调整为 20 秒（20000 毫秒）。

警告

调整配置超时时要非常谨慎：

- 如果您设置了太小的值（如 500 毫秒），SSSD 可能没有足够的时间来回复，请求始终会返回超时。
- 如果您设置了太大的值，如 30000 毫秒（30 秒），则单个请求可能会阻止到 SSSD 的连接。因为一个线程一次只能连接到 SSSD，所以来自插件的所有其他请求都必须等待。
- 如果 IdM 客户端发送了多个请求，它们可以阻止为 IdM 服务器上的 Directory 服务器配置的所有可用 worker。因此，服务器可能无法在一段时间内回复任何类型的请求。

只在以下情况下更改配置超时：

- 在请求 AD 用户和组的信息时，如果 IdM 客户端会在达到自己的搜索超时前频繁收到超时错误，这代表配置超时值**太小**。
- 如果 IdM 服务器上的 Directory Server 经常会锁定，**pstack** 程序报告有很多或所有 worker 线程在处理 **ipa-extdom** 请求，这代表这个值 **太大**。

先决条件

- LDAP Directory Manager 密码

流程

- 使用以下命令将配置超时调整为 20000 毫秒：

```
# ldapmodify -D "cn=directory manager" -W
dn: cn=ipa_extdom_extop,cn=plugins,cn=config
changetype: modify
replace: ipaExtdomMaxNssTimeout
ipaExtdomMaxNssTimeout: 20000
```

9.3. 在 IDM 服务器中调整 IPA-EXTDOM 插件的最大缓冲区大小

IdM 客户端无法直接从 Active Directory(AD)接收用户和组的信息，因此 IdM 服务器使用 **ipa-extdom** 插件接收 AD 用户和组的信息，并将这些信息转发到请求的客户端。

您可以调整 **ipa-extdom** 插件的最大缓冲区大小，它调整 SSSD 可以存储它接收数据的缓冲区的大小。如果缓冲区太小，SSSD 会返回 **ERANGE** 错误，并且插件会以更大的缓冲区重试请求。默认缓冲区大小为 134217728 字节(128 MB)。

以下示例将最大缓冲区大小调整为 256 MB（268435456 字节）。

先决条件

- LDAP Directory Manager 密码

流程

- 使用以下命令将最大缓冲区大小设置为 268435456 字节：

```
# ldapmodify -D "cn=directory manager" -W
dn: cn=ipa_extdom_extop,cn=plugins,cn=config
changetype: modify
replace: ipaExtdomMaxNssBufSize
ipaExtdomMaxNssBufSize: 268435456
```

9.4. 为 IDM 服务器上的 IPA-EXTDOM 插件调整实例的最大数量

因为 IdM 客户端无法直接从活动目录(AD)接收用户和组的信息，因此 IdM 服务器使用 **ipa-extdom** 插件接收 AD 用户和组的信息，然后将此信息转发到请求的客户端。

默认情况下，**ipa-extdom** 插件被配置为使用最多 80% 的 LDAP worker 线程来处理 IdM 客户端的请求。如果 IdM 客户端上的 SSSD 服务已请求了大量 AD 信任用户和组的信息，则此操作可在使用大多数 LDAP 线程时停止 LDAP 服务。如果您遇到这些问题，您可能会在 AD 域的 SSSD 日志文件中看到类似的错误，即 **/var/log/sssd/sssd__your-ad-domain-name.com_.log**:

```
(2022-05-22  5:00:13): [be[ad.example.com]] [ipa_s2n_get_user_done] (0x0040): s2n exop request failed.
(2022-05-22  5:00:13): [be[ad.example.com]] [ipa_s2n_get_user_done] (0x0040): s2n exop request failed.
(2022-05-22  5:00:13): [be[ad.example.com]] [ipa_s2n_exop_done] (0x0040): ldap_extended_operation result: Server is busy(51), Too many extdom instances running.
```

您可以通过为 **ipaExtdomMaxInstances** 选项设置值来调整 **ipa-extdom** 实例的最大数量，它必须是大于 0 的整数，并小于 worker 线程的总数。

先决条件

- LDAP Directory Manager 密码

流程

1. 检索 worker 线程的总数：

```
# ldapsearch -xLLLD cn=directory\ manager -W -b cn=config -s base nsslapd-threadnumber
Enter LDAP Password:
dn: cn=config
nsslapd-threadnumber: 16
```

   这意味着 **ipaExtdomMaxInstances** 的当前值是 13。

2. 调整实例的最大数量。本例将值改为 14：

```
# ldapmodify -D "cn=directory manager" -W
dn: cn=ipa_extdom_extop,cn=plugins,cn=config
changetype: modify
replace: ipaExtdomMaxInstances
ipaExtdomMaxInstances: 14
```

3. 检索 **ipaExtdomMaxInstances** 的当前值：

```
# ldapsearch -xLLLD "cn=directory manager" -W -b
"cn=ipa_extdom_extop,cn=plugins,cn=config" |grep ipaextdommaxinstances

Enter LDAP Password:

ipaextdommaxinstances: 14
```

4. 监控 IdM 目录服务器的性能，如果没有提高，请重复这个过程，并调整 **ipaExtdomMaxInstances** 变量的值。

9.5. 为大型 IDM-AD 信任部署在 IDM 客户端中调整 SSSD

此流程对 IdM 客户端中的 SSSD 服务配置应用调整选项，以便在从大型 AD 环境检索信息时提高其响应时间。

先决条件

- 您需要 **root** 权限来编辑 **/etc/sssd/sssd.conf** 配置文件。

流程

1. 确定单个未缓存登录所需的秒数。

   a. 清除 IdM 客户端 **client.example.com** 上的 SSSD 缓存。

   ```
   [root@client ~]# sss_cache -E
   ```

   b. 使用 **time** 命令测量以 AD 用户身份登录所需的时间。在本例中，从 IdM 客户端 **client.example.com** 中与 **ad.example.com** AD 域中的用户 **ad-user** 身份登录同一主机。

   ```
   [root@client ~]# time ssh ad-user@ad.example.com@client.example.com
   ```

   c. 尽快输入密码。

   ```
   Password:
   Last login: Sat Jan 23 06:29:54 2021 from 10.0.2.15
   [ad-user@ad.example.com@client ~]$
   ```

   d. 尽快注销以显示已经过的时间。在本例中，单个未缓存的登录大约需要 **9** 秒。

   ```
   [ad-user@ad.example.com@client /]$ exit
   logout
   Connection to client.example.com closed.

   real 0m8.755s
   user    0m0.017s
   sys     0m0.013s
   ```

2. 在文本编辑器中打开 **/etc/sssd/sssd.conf** 配置文件。

3. 在您的 Active Directory 域的 **[domain]** 部分添加以下选项。将 **pam_id_timeout** 和 **krb5_auth_timeout** 选项设置为未缓存登录所需的秒数。如果您还没有 AD 域的 domain 部分，请创建一个。

```
[domain/example.com/ad.example.com]
krb5_auth_timeout = 9
ldap_deref_threshold = 0
...
```

4. 在 **[pam]** 部分添加以下选项：

```
[pam]
pam_id_timeout = 9
```

5. 保存并关闭服务器上的 **/etc/sssd/sssd.conf** 文件。

6. 重启 SSSD 服务以载入配置更改。

```
[root@client ~]# systemctl restart sssd
```

其它资源

- 为大型 IdM-AD 信任部署在 IdM 服务器和客户端中调整 SSSD 的选项

9.6. 在 TMPFS 中挂载 SSSD 缓存

系统安全服务守护进程(SSSD)持续将 LDAP 对象写入其缓存中。这些内部 SSSD 事务将数据写入磁盘，它的速度比从 Random-Access Memory(RAM)进行读取和写入要慢。

要提高此性能，请在 RAM 中挂载 SSSD 缓存。

注意事项

- 如果 SSSD 缓存位于 RAM，则缓存的信息不会在重启后保留。
- 在 IdM 服务器上执行此更改是安全的，因为 IdM 服务器中的 SSSD 实例不会丢失与同一主机上 Directory 服务器的连接。
- 如果您在 IdM 客户端中执行此调整，且丢失了与 IdM 服务器的连接，则用户重启后将无法进行身份验证，直到重新建立连接为止。

先决条件

- 您需要 **root** 权限来编辑 **/etc/fstab** 配置文件。

流程

1. 创建 **tmpfs** 临时文件系统：

   a. 确认 SSSD 用户拥有 **config.ldb** 文件：

```
# ls -al /var/lib/sss/db/config.ldb
-rw-------. 1 sssd sssd 1286144 Jun  8 16:41 /var/lib/sss/db/config.ldb
```

b. 将以下条目作为一行添加到 **/etc/fstab** 文件中：

```
tmpfs /var/lib/sss/db/ tmpfs
size=300M,mode=0700,uid=sssd,gid=sssd,rootcontext=system_u:object_r:sssd_var_lib_
t:s0 0 0
```

这个示例创建了一个 300MB 缓存。根据您的 IdM 和 AD 目录大小调整 **size** 参数，每个 10,000 LDAP 条目大约为 100 MBs。

2. 挂载新的 SSSD 缓存目录。

```
[root@host ~]# mount /var/lib/sss/db/
```

3. 重启 SSSD 以反应这个配置更改。

```
[root@host ~]# systemctl restart sssd
```

9.7. SSSD.CONF 中用于为大型 IDM-AD 信任部署调整 IDM 服务器和客户端中的选项

在具有大型 IdM-AD 信任部署时，您可以使用 **/etc/sssd/sssd.conf** 配置文件中的 SSSD 的性能调整 IdM 服务器和客户端中的性能。

9.7.1. IdM 服务器的调整选项

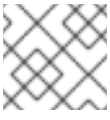
ignore_group_members

在验证和授权用户时，了解用户所属的组而不是属于组的所有用户是非常重要地。当将 **ignore_group_members** 设为 **true** 时，SSSD 只检索关于组对象本身而不是其成员的信息，从而显著提高性能。

注意

id user@ad-domain.com 命令仍然会返回正确的组列表，但 **getent group ad-group@ad-domain.com** 会返回一个空列表。

默认值	**false**
推荐的值	**true**



注意

当部署涉及带有 compat 树的 IdM 服务器时，您不应该将这个选项设置为 **true**。

subdomain_inherit

使用 **subdomain_inherit** 选项，您可以将 **ignore_group_members** 设置应用到可信 AD 域配置。**subdomain_inherit** 选项中列出的设置适用于主(IdM)域以及 AD 子域。

默认值	**none**

推荐的值	**subdomain_inherit = ignore_group_members**

9.7.2. IdM 客户端的调优选项

pam_id_timeout

此参数控制 PAM 会话的结果被缓存多长时间，以避免在身份查找期间对身份提供商过度的往返。在 IdM 服务器和 IdM 客户端中填充复杂组成员资格的环境中，默认值 **5** 秒可能不足。红帽建议将 **pam_id_timeout** 设置为一个未缓存的单个登录所需的秒数。

默认值	**5**
推荐的值	**单个未缓存登录所需的秒数**

krb5_auth_timeout

对于存在用户是大量组的成员的环境，增加 **krb5_auth_timeout** 可以允许更多的时间来处理复杂的组信息。红帽建议把这个值设置为一个未缓存的登录所花的秒数。

默认值	**6**
推荐的值	**单个未缓存登录所需的秒数**

ldap_deref_threshold

解引用查找是在单个 LDAP 调用中获取所有组成员的方法。**ldap_deref_threshold** 值指定必须是内部缓存中缺少的组成员的数量，以触发解引用查找。如果缺少的成员较少，则会单独查找。在大型环境中，解引用查找可能需要很长时间，并降低性能。要禁用解引用查找，将此选项设置为 **0**。

默认值	**10**
推荐的值	**0**

9.8. 其他资源

- 大型 IdM-AD 信任部署的性能调优 SSSD

第 10 章 调优 WSGI 进程

如果您因为长时间运行 API 进程而看到请求失败，则这些 API 进程可能会从调优中受益。

默认情况下，IPA 为 64 位系统上的 API 服务分配 4 个 Web 服务器网关接口(WSGI)进程。这个 4 个进程的默认限制是为内存保留而实施的。增加 WSGI 进程的数量允许接受更多的请求，代价是更高的 CPU 使用率和内存消耗。默认情况下，IPA 将大约 100 到 110MB 驻留内存用于每个 WSGI 进程的 API 。将此调优到 16 个进程（这是推荐的限制）后，内存量大约为 1.3GB。

流程

- 修改 **/etc/httpd/conf.d/ipa.conf** 文件中的进程值：

```
WSGIDaemonProcess ipa processes=<4> threads=1 maximum-requests=500 \
```

任何长时间运行的 API 端点都可以从调优中受益。这个调优决定是由用户来做的。

例如，OpenStack 安装由几个包含多个服务的控制器组成。每个服务都请求一个证书，以便所有内部通信都通过Transport Layer Security(TLS)发生。安装或刷新控制器或计算节点时，可以请求或刷新这些证书。在涉及多个控制器或计算节点的场景，证书请求的数量可能变得相当大。这些请求是自动的，因此它们几乎在同时发生。增加 WSGI 线程的数量允许安装完成。

10.1. 优化 CPU 使用率，以提高 IPA 服务器性能

在大量证书颁发任务过程中遇到性能限制时，调优 CPU 和 Web 服务器网关接口(WSGI)进程数可能会显著增强 IPA 服务器处理同步请求的能力。

如果服务器配置了 4 个 CPU 和 70 个客户端，每个客户端请求 7 个证书（总共 490 个证书），会因为请求卷超过服务器的处理能力而发生服务器超时。

将 CPU 数量增加到 8，并将 WSGI 进程数增加到 8，来将证书处理能力增加到 630 个证书，比 4 个 CPU 配置增加了 28%，尽管 CPU 数增加了 100%。将 CPU 数进一步增加到 16 ，只有 8 个 WSGI 进程没有获得额外的性能提升。但是，通过将 WSGI 进程数增加到 16，服务器处理了 110 个客户端的 770 个证书，与 8 个 CPU 设置相比，反映了 22% 的提高。

平均而言，将 CPU 数翻一倍会使证书颁发能力提高 25%，只要也相应地调整 WSGI 进程。这强调了需要将 CPU 和 WSGI 进程一起扩展，以防止瓶颈并优化服务器性能。