了解 SnapDrive for UNIX 守护进程

Snapdrive for Unix

NetApp
October 04, 2023

目录

了解 SnapDrive for UNIX 守护进程

在运行任何 SnapDrive for UNIX 命令之前，您必须了解 Web 服务和守护进程以及如何使用它们。所有 SnapDrive for UNIX 命令均使用守护进程服务运行。在 AIX 主机上使用 SnapDrive for UNIX 之前，必须启动守护进程，以便 SnapDrive for UNIX 能够与其他 NetApp 和非 NetApp 产品无缝安全地集成。

什么是 Web 服务和守护进程

SnapDrive for UNIX Web 服务为所有 NetApp SnapManager 和第三方产品提供了一个统一的界面，可与 SnapDrive for UNIX 无缝集成。要在适用于 UNIX 的 SnapDrive 中使用命令行界面（ CLI ）命令，您需要启动守护进程。

各种 NetApp SnapManager 产品都使用命令行界面（ CLI ）与 SnapDrive for UNIX 进行通信。使用命令行界面会限制 SnapManager 和 SnapDrive for UNIX 的性能和易管理性。使用 SnapDrive for UNIX 守护进程时，所有命令都是一个唯一的过程。守护进程服务不会影响 SnapDrive for UNIX 命令的使用方式。

SnapDrive for UNIX Web 服务允许第三方应用程序与 SnapDrive for UNIX 无缝集成。它们使用 API 与 SnapDrive for UNIX 进行交互。

启动守护进程时， SnapDrive for UNIX 守护进程会首先检查该守护进程是否正在运行。如果守护进程未运行，则启动守护进程。如果守护进程已在运行，而您尝试启动它，则 SnapDrive for UNIX 将显示以下消息：

SnapDrive 守护进程已在运行`

您可以检查守护进程的状态以查看 SnapDrive for UNIX 是否正在运行。在决定启动守护进程之前，应先检查状态。如果非 root 用户尝试检查状态，则 SnapDrive for UNIX 将检查该用户的凭据并显示消息：

只有 root 用户` 才能查看 SnapDrive 守护进程状态

当您尝试停止守护进程时， SnapDrive for UNIX 会检查您的凭据。如果您不是 root 用户，则 SnapDrive for UNIX 将显示此消息

SnapDrive 守护进程只能由 root 用户` 停止

停止守护进程后，必须重新启动 SnapDrive for UNIX 守护进程，对配置文件或任何模块所做的任何更改才能生效。如果非 root 用户尝试重新启动 SnapDrive for UNIX 守护进程，则 SnapDrive for UNIX 将检查该用户的凭据并显示消息

SnapDrive 守护进程只能由 root 用户` 重新启动

正在检查守护进程的状态

您可以检查守护进程的状态以查看守护进程是否正在运行。如果守护进程已在运行，则在更新 SnapDrive for UNIX 配置文件之前，无需重新启动它。

您必须以 root 用户身份登录。

步骤

1. 检查守护进程的状态：

   ` * 快照状态 *`

启动 SnapDrive for UNIX 守护进程

您必须先启动并运行 SnapDrive for UNIX 守护进程，然后才能使用任何 SnapDrive for UNIX 命令。

您必须以 root 用户身份登录。

步骤

1. 启动守护进程：

   ` * 快照启动 *`

更改默认守护进程密码

SnapDrive for UNIX 会分配一个默认守护进程密码，您可以稍后更改该密码。此密码存储在一个加密文件中，其中只会为 root 用户分配读取和写入权限。更改密码后，必须手动通知所有客户端应用程序。

您必须以 root 用户身份登录。

步骤

1. 更改默认密码：

   ` * 快照密码 *`

2. 输入密码。
3. 确认密码。

正在停止守护进程

如果更改 SnapDrive for UNIX 配置文件，则必须停止并重新启动守护进程。您可以非强制或强制停止守护进程。

不强制停止守护进程

如果 SnapDrive for UNIX 配置文件发生更改，则必须停止守护进程，配置文件更改才能生效。停止并重新启动守护进程后，配置文件中的更改将生效。如果不强制停止守护进程，则所有排队的命令都可以完成执行。收到停止请求后，不会执行任何新命令。

您必须以 root 用户身份登录。

1. 输入以下命令以非强制停止守护进程：

   ` * 快照停止 *`

强制停止守护进程

如果您不想等待所有命令完成执行，则可以强制停止守护进程。收到强制停止守护进程的请求后， SnapDrive for UNIX 守护进程将取消正在执行或正在排队的所有命令。强制停止守护进程时，系统的状态可能未定义。不建议使用此方法。

您必须以 root 用户身份登录。

步骤

1. 强制停止守护进程：

   ` * snapdrived -force stop*`

正在重新启动守护进程

您必须在停止守护进程后重新启动它，以便对配置文件或其他模块所做的更改生效。只有在完成正在执行且处于队列中的所有命令后， SnapDrive for UNIX 守护进程才会重新启动。收到重新启动请求后，不会执行任何新命令。

- 确保以 root 用户身份登录。
- 确保同一主机上未并行运行任何其他会话。在这种情况下， `snapdrived restart` 命令将挂起系统。

步骤

1. 输入以下命令重新启动守护进程：

   ` * 快照重新启动 *`

强制守护进程重新启动

您可以强制此守护进程重新启动。强制重新启动守护进程将停止执行所有正在运行的命令。

确保以 root 用户身份登录。

步骤

1. 输入以下命令强制重新启动守护进程：

   ` * snapdrived -force restart*`

   收到强制重新启动请求后，守护进程将停止所有正在执行和正在排队的命令。只有在取消执行所有正在运行的命令后，守护进程才会重新启动。

使用 HTTPS 安全守护进程通信

您可以使用 HTTPS 进行安全的 Web 服务和守护进程通信。通过在 `snapdrive.conf` 文件中设置一些配置变量，并生成和安装自签名证书或 CA 签名证书，可以启用安全通信。

您必须在 `snapdrive.conf` 文件中指定的路径处提供自签名或 CA 签名证书。要使用 HTTPS 进行通信，必须在 `snapdrive.conf` 文件中设置以下参数：

- `use-https-to-sdU-daemon=on`
- `contact-https-port-sdU-daemon = 4095`
- `sdu-daemon-certificate-path=/opt/NetApp/SnapDrive/SnapDrive.pem`

> SnapDrive 5.0 for UNIX 及更高版本支持 HTTPS 用于守护进程通信。默认情况下，此选项设置为 `off` 。

生成自签名证书

SnapDrive for UNIX 守护进程服务要求您生成用于身份验证的自签名证书。与命令行界面通信时需要进行此身份验证。

步骤

1. 生成 RSA 密钥：

   ` * $ openssl genrsa 1024 > host.key $ chmod 400 host.key*``

   ```
   # openssl genrsa 1024 > host.key Generating
   RSA private key, 1024 bit long modulus
   ..............................++++++ ...++++++ e is 65537(0x10001)
   # chmod 400 host.key
   ```

2. 创建证书：

   ` * $ openssl req -new -x509 -nodes -SHA1 -days 365 -key host.key > host.cert*`

   使用 ` -new` ， ` -x509` 和 ` -nodes` 选项创建未加密证书。` - 天` 选项指定证书保持有效的天数。

3. 当系统要求您填写证书的 x509 数据时，请输入您的本地数据：

```
# openssl req -new -x509 -nodes -sha1 -days 365 -key host.key >
host.cert
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a
DN. There are quite a few fields
but you can leave some blank For some fields there will be a default
value, If you enter '.', the field will be left blank.

Country Name (2 letter code) [AU]:US
State or Province Name (full name) [Some-State]:California
Locality Name (eg, city) []:Sunnyvale
Organization Name (eg, company) [Internet Widgits Pty Ltd]:abc.com
Organizational Unit Name (eg, section) []:
Common Name (eg, YOUR name) []:localhost
Email Address []:postmaster@example.org
```

`Common Name` 值必须为 *localhost* 。

4. 提取元数据（可选）。

```
$ openssl x509 -noout -fingerprint -text < host.cert > host.info
```

您可以保存证书元数据以供日后快速参考。

5. 将密钥和证书数据结合使用。

SnapDrive for UNIX 要求密钥和证书数据位于同一文件中。组合文件必须作为密钥文件进行保护。

` * $ cat host.cert host.key > host.pem \*`

` * 和 rm host.key*`

` * $ chmod 400 host.pem*`

```
# cat host.cert host.key >  /opt/NetApp/snapdrive.pem
# rm host.key rm: remove regular file `host.key'? y
# chmod 400 /opt/NetApp/snapdrive.pem
```

6. 将守护进程证书的完整路径添加到 `snapdrive.conf` 文件的 ` *sdU-daemon-certificate-path*` 变量。

生成 CA 签名的证书

SnapDrive for UNIX 守护进程服务要求您生成 CA 签名的证书，才能成功进行守护进程通

信。您必须在 `snapdrive.conf` 文件中指定的路径处提供 CA 签名证书。

- 您必须以 root 用户身份登录。
- 您必须已在 `snapdrive.conf` 文件中设置以下参数，才能使用 HTTPS 进行通信：
  - use-https-to-sdU-daemon=on
  - contact-https-port-sdU-daemon = 4095
  - sdU-daemon-certificate-path=` /opt/NetApp/SnapDrive/SnapDrive.pem`

步骤

1. 以 pem 格式生成新的未加密 RSA 私钥：

   ` * $ openssl genrsa -out privkey.pem 1024*`

   ```
   Generating RSA private key, 1024 bit long modulus
    ....................++++++ ....................................++++++
   e is 65537 (0x10001)
   ```

2. 配置 ` /etc/ssl/openssl.cnf` 以创建 CA 私钥和证书 vi ` /etc/ssl/openssl.cnf` 。
3. 使用 RSA 私钥创建未签名的证书：

   ` * $ openssl req -new -x509 -key privkey.pem -out cert.pem*`

   ```
   You are about to be asked to enter information that will be
   incorporated into your certificate request.
   What you are about to enter is what is called a Distinguished Name or
   a DN.
   There are quite a few fields but you can leave some blank For some
   fields there will be a default value, If you enter '.', the field
   will be left blank.
   -----
   Country Name (2 letter code) [XX]:NY
   State or Province Name (full name) []:Nebraska Locality Name (eg,
   city) [Default City]:Omaha Organization Name (eg, company) [Default
   Company Ltd]:abc.com Organizational Unit Name (eg, section) []:
   Common Name (eg, your name or your server's hostname) []:localhost
   Email Address []:abc@example.org
   ```

4. 使用私钥和证书创建 CSR：

   ` * cat cert.pem privkey.pem | openssl x509 -x509 tor均衡 器 -signkey privkey.pem -out certreq.csd*`

   ```
   Getting request Private Key Generating certificate request
   ```

5. 使用刚刚创建的 CSR 使用 CA 专用密钥对证书进行签名：

` * $ openssl ca -in certreq.csr -out newcert.pem*`

```
Using configuration from /etc/pki/tls/openssl.cnf Check that the
request matches the signature Signature ok Certificate Details:
         Serial Number: 4096 (0x1000)
         Validity
            Not Before: May 17 06:02:51 2015 GMT
             Not After : May 16 06:02:51 2016 GMT
             Subject:
             countryName               = NY
             stateOrProvinceName       = Nebraska
             organizationName          = abc.com
             commonName                = localhost
             emailAddress              = abc@example.org
             X509v3 extensions:
             X509v3 Basic Constraints:
                 CA:FALSE
             X509v3 Key Usage:
                 Digital Signature, Non Repudiation, Key Encipherment
             Netscape Comment:
                 OpenSSL Generated Certificate
             X509v3 Subject Key Identifier:

FB:B0:F6:A0:9B:F2:C2:BC:50:BF:45:B2:9D:DB:AA:3B:C5:07:5B:7F
             X509v3 Authority Key Identifier:

 keyid:FB:B0:F6:A0:9B:F2:C2:BC:50:BF:45:B2:9D:DB:AA:3B:C5:07:5B:7F

 Certificate is to be certified until May 16 06:02:51 2016 GMT (365
 days) Sign the certificate? [y/n]:y


 1 out of 1 certificate requests certified, commit? [y/n]y Write out
 database with 1 new entries Data Base Updated
```

6. 安装 SSL 服务器要使用的签名证书和专用密钥。

```
The newcert.pem is the certificate signed by your local CA that you can
then use in an
ssl server:
( openssl x509 -in newcert.pem; cat privkey.pem ) > server.pem
ln -s server.pem `openssl x509 -hash -noout -in server.pem`.0 # dot-zero
( server.pem refers to location of https server certificate)
```