Implementar funcionalidades e integrações

BeeGFS on NetApp with E-Series Storage

NetApp
January 27, 2026

Índice

Implementar funcionalidades e integrações

Motorista CSI da BeeGFS

Configurar criptografia TLS para BeeGFS v8

Configurar criptografia TLS para proteger a comunicação entre os serviços de gerenciamento do BeeGFS v8 e os clientes.

Visão geral

O BeeGFS v8 introduz suporte a TLS para criptografar as comunicações de rede entre ferramentas administrativas (como o `beegfs` utilitário de linha de comando) e serviços do servidor BeeGFS, como Management ou Remote. Este guia aborda como configurar a criptografia TLS em seu cluster BeeGFS usando três métodos de configuração TLS:

- **Utilizando uma Autoridade Certificadora confiável:** Use certificados já assinados por uma Autoridade Certificadora em seu cluster BeeGFS.
- **Criação de uma Autoridade Certificadora local:** Criar uma Autoridade Certificadora local e usá-la para assinar certificados para seus serviços BeeGFS. Essa abordagem é adequada para ambientes em que você deseja gerenciar sua própria cadeia de confiança sem depender de uma CA externa.
- **TLS desativado:** Desative completamente o TLS em ambientes onde a criptografia não é necessária ou para fins de solução de problemas. Isso não é recomendado, pois expõe informações potencialmente sensíveis sobre a estrutura do sistema de arquivos interno e configuração como texto não criptografado.

Escolha o método que melhor se adapte ao seu ambiente e às políticas da sua organização. Consulte a "BeeGFS TLS" documentação para obter mais detalhes.

> As máquinas que executam o `beegfs-client` serviço não exigem TLS para montar o sistema de arquivos BeeGFS. TLS deve ser configurado para utilizar a CLI do BeeGFS e outros serviços BeeGFS, como remote e sync.

Usando uma Autoridade Certificadora Confiável

Se você tiver acesso a certificados emitidos por uma Autoridade Certificadora (CA) confiável—seja de uma CA interna da empresa ou de um provedor terceiro—você pode configurar o BeeGFS v8 para usar esses certificados assinados pela CA em vez de gerar certificados autoassinados.

Implantando um novo cluster BeeGFS v8

Para uma nova implementação de cluster BeeGFS v8, configure o arquivo de inventário do Ansible `user_defined_params.yml` para referenciar seus certificados assinados pela CA:

```
beegfs_ha_tls_enabled: true

beegfs_ha_ca_cert_src_path: files/beegfs/cert/ca_cert.pem

beegfs_ha_tls_cert_src_path: files/beegfs/cert/mgmtd_tls_cert.pem

beegfs_ha_tls_key_src_path: files/beegfs/cert/mgmtd_tls_key.pem
```

Se `beegfs_ha_tls_config_options.alt_names`não estiver vazio, o Ansible gerará automaticamente um certificado e uma chave TLS autoassinados, usando os alt_names fornecidos como Subject Alternative Names (SANs) no certificado. Para usar seu próprio certificado e chave TLS personalizados (conforme especificado por `beegfs_ha_tls_cert_src_path` e `beegfs_ha_tls_key_src_path`), você deve comentar ou remover toda a seção `beegfs_ha_tls_config_options`. Caso contrário, a geração do certificado autoassinado terá precedência e seu certificado e chave personalizados não serão usados.

Configurando um cluster BeeGFS v8 existente

Para um cluster BeeGFS v8 existente, defina os caminhos no arquivo de configuração dos serviços de gerenciamento do BeeGFS para os certificados assinados pela CA do nó de arquivos:

```
tls-cert-file = /path/to/cert.pem
tls-key-file = /path/to/key.pem
```

Configurando clientes BeeGFS v8 com certificados assinados por CA

Para configurar os clientes BeeGFS v8 para confiarem em certificados assinados por uma CA usando o pool de certificados do sistema, defina tls-cert-file = "" no arquivo de configuração de cada cliente. Se o pool de certificados do sistema não estiver sendo usado, forneça o caminho para um certificado local definindo tls-cert-file = <local cert>. Essa configuração permite que os clientes autentiquem os certificados apresentados pelos serviços de gerenciamento BeeGFS.

Criando uma Autoridade Certificadora local

Se a sua organização deseja criar sua própria infraestrutura de certificados para o cluster BeeGFS, você pode criar uma Autoridade Certificadora (CA) local para emitir e assinar certificados para o seu cluster BeeGFS. Essa abordagem envolve a criação de uma CA que assina certificados para os serviços de gerenciamento do BeeGFS, que são então distribuídos aos clientes para estabelecer uma cadeia de confiança. Siga estas instruções para configurar uma CA local e implantar certificados em seu cluster BeeGFS v8 existente ou novo.

Implantando um novo cluster BeeGFS v8

Para uma nova implementação do BeeGFS v8, a `beegfs_8` função Ansible será responsável por criar uma CA local no nó de controle e gerar os certificados necessários para os serviços de gerenciamento. Isso pode ser habilitado definindo os seguintes parâmetros no arquivo de inventário do Ansible `user_defined_params.yml`:

```
beegfs_ha_tls_enabled: true

beegfs_ha_ca_cert_src_path: files/beegfs/cert/local_ca_cert.pem

beegfs_ha_tls_cert_src_path: files/beegfs/cert/mgmtd_tls_cert.pem

beegfs_ha_tls_key_src_path: files/beegfs/cert/mgmtd_tls_key.pem

beegfs_ha_tls_config_options:
  alt_names: [<mgmt_service_ip>]
```

Se `beegfs_ha_tls_config_options.alt_names` não for fornecido, o Ansible tentará usar os certificados existentes nos caminhos de certificado/chave especificados.

Configurando um cluster BeeGFS v8 existente

Para um cluster BeeGFS existente, você pode integrar TLS criando uma Autoridade de Certificação local e gerando os certificados necessários para os serviços de gerenciamento. Atualize os caminhos no arquivo de configuração dos serviços de gerenciamento do BeeGFS para apontar para os certificados recém-criados.

As instruções desta seção devem ser usadas como referência. Devem ser tomadas as devidas precauções de segurança ao lidar com chaves privadas e certificados.

Criar a Certificate Authority

Em um computador confiável, crie uma Autoridade Certificadora local para assinar certificados para seus serviços de gerenciamento BeeGFS. O certificado da Autoridade Certificadora será distribuído aos clientes para estabelecer confiança e permitir a comunicação segura com os serviços BeeGFS.

As instruções a seguir servem de referência para criar uma Autoridade de Certificação local em um sistema baseado em RHEL.

1. Instale o OpenSSL se ainda não estiver instalado:

```
dnf install openssl
```

2. Crie um diretório de trabalho para armazenar arquivos de certificado:

```
mkdir -p ~/beegfs_tls && cd ~/beegfs_tls
```

3. Gerar a chave privada da CA:

```
openssl genrsa -out ca_key.pem 4096
```

4. Crie um arquivo de configuração de CA chamado `ca.cnf` e ajuste os campos de nome diferenciado para

corresponder à sua organização:

```
[ req ]
default_bits        = 4096
distinguished_name = req_distinguished_name
x509_extensions     = v3_ca
prompt              = no

[ req_distinguished_name ]
C  = <Country>
ST = <State>
L  = <City>
O  = <Organization>
OU = <OrganizationalUnit>
CN = BeeGFS-CA

[ v3_ca ]
basicConstraints = critical,CA:TRUE
subjectKeyIdentifier = hash
authorityKeyIdentifier = keyid:always,issuer:always
```

5. Gere o certificado da Autoridade Certificadora (CA). Este certificado deve ser válido por toda a vida útil do sistema, caso contrário, será necessário planejar a regeneração dos certificados antes que expirem. Após a expiração de um certificado, a comunicação entre alguns componentes ficará indisponível e a atualização dos certificados TLS geralmente exigirá reiniciar os serviços para concluir.

   O seguinte comando gera um certificado de CA válido por 1 ano:

```
openssl req -new -x509 -key ca_key.pem -out ca_cert.pem -days 365
-config ca.cnf
```

   Embora este exemplo utilize um período de validade de 1 ano por simplicidade, você deve ajustar o `-days` parâmetro de acordo com os requisitos de segurança da sua organização e estabelecer um processo de renovação de certificados.

Criar certificados de serviço de gerenciamento

Gere certificados para seus serviços de gerenciamento do BeeGFS e assine-os com a CA que você criou. Esses certificados serão instalados nos nós de arquivos que executam os serviços de gerenciamento do BeeGFS.

1. Gere a chave privada do serviço de gerenciamento:

```
openssl genrsa -out mgmtd_tls_key.pem 4096
```

2. Crie um arquivo de configuração de certificado chamado `tls_san.cnf`com Nomes Alternativos do Assunto (SANs) para todos os endereços IP do serviço de gerenciamento:

```
[ req ]
default_bits       = 4096
distinguished_name = req_distinguished_name
req_extensions     = req_ext
prompt             = no

[ req_distinguished_name ]
C  = <Country>
ST = <State>
L  = <City>
O  = <Organization>
OU = <OrganizationalUnit>
CN = beegfs-mgmt

[ req_ext ]
subjectAltName = @alt_names

[ v3_ca ]
subjectAltName = @alt_names
basicConstraints = CA:FALSE

[ alt_names ]
IP.1 = <beegfs_mgmt_service_ip_1>
IP.2 = <beegfs_mgmt_service_ip_2>
```

Atualize os campos de nome diferenciado para corresponder à sua configuração de CA e os `IP.1` e `IP.2` valores com os endereços IP do seu serviço de gerenciamento.

3. Gerar uma solicitação de assinatura de certificado (CSR):

```
openssl req -new -key mgmtd_tls_key.pem -out mgmtd_tls_csr.pem -config
tls_san.cnf
```

4. Assine o certificado com sua CA (válido por 1 ano):

```
openssl x509 -req -in mgmtd_tls_csr.pem -CA ca_cert.pem -CAkey
ca_key.pem -CAcreateserial -out mgmtd_tls_cert.pem -days 365 -sha256
-extensions v3_ca -extfile tls_san.cnf
```

Ajuste o período de validade do certificado (`-days 365`) de acordo com as políticas de segurança da sua organização. Muitas organizações exigem a renovação do certificado a cada 1-2 anos.

5. Verifique se o certificado foi criado corretamente:

```
openssl x509 -in mgmtd_tls_cert.pem -text -noout
```

Confirme que a seção Subject Alternative Name inclui todos os seus endereços IP de gerenciamento.

Distribuir certificados para nós de arquivo

Distribua o certificado da CA e os certificados do serviço de gerenciamento para os nós de arquivo e clientes apropriados.

1. Copie o certificado da CA, o certificado do serviço de gerenciamento e a chave para os nós de arquivo que executam os serviços de gerenciamento:

```
scp ca_cert.pem mgmtd_tls_cert.pem mgmtd_tls_key.pem
user@beegfs_01:/etc/beegfs/
scp ca_cert.pem mgmtd_tls_cert.pem mgmtd_tls_key.pem
user@beegfs_02:/etc/beegfs/
```

Direcione o serviço de gerenciamento para os certificados TLS

Atualize a configuração do serviço de gerenciamento BeeGFS para habilitar TLS e referenciar os certificados TLS criados.

1. A partir de um nó de arquivo que executa o serviço de gerenciamento BeeGFS, edite o arquivo de configuração do serviço de gerenciamento, por exemplo, em `/mnt/mgmt_tgt_mgmt01/mgmt_config/beegfs-mgmtd.toml`. Adicione ou atualize os seguintes parâmetros relacionados ao TLS:

```
tls-disable = false
tls-cert-file = "/etc/beegfs/mgmtd_tls_cert.pem"
tls-key-file = "/etc/beegfs/mgmtd_tls_key.pem"
```

2. Tome as medidas apropriadas para reiniciar com segurança o serviço de gerenciamento do BeeGFS para que as alterações entrem em vigor:

```
systemctl restart beegfs-mgmtd
```

3. Verifique se o serviço de gerenciamento foi iniciado com sucesso:

```
journalctl -xeu beegfs-mgmtd
```

Procure por entradas de log que indiquem inicialização TLS bem-sucedida e carregamento de certificado.

```
Successfully initialized certificate verification library.
Successfully loaded license certificate: TMP-XXXXXXXXXX
```

Configurar TLS para clientes BeeGFS v8

Crie e distribua certificados assinados pela CA local para todos os clientes BeeGFS que precisarão de comunicação com os serviços de gerenciamento BeeGFS.

1. Gere um certificado para o cliente usando o mesmo processo do certificado do serviço de gerenciamento acima, mas com o endereço IP ou nome do host do cliente no campo Subject Alternative Name (SAN).
2. Copie remotamente com segurança o certificado do cliente para o cliente e renomeie o certificado para `cert.pem` no cliente:

```
scp client_cert.pem user@client:/etc/beegfs/cert.pem
```

3. Reinicie o serviço do cliente BeeGFS em todos os clientes:

```
systemctl restart beegfs-client
```

4. Verifique a conectividade do cliente executando um `beegfs CLI` comando, como:

```
beegfs health check
```

Desativando TLS

O TLS pode ser desativado para fins de resolução de problemas ou se desejado pelos usuários. Isso não é recomendado, pois expõe informações potencialmente sensíveis sobre a estrutura interna do sistema de arquivos e configuração em texto não criptografado. Siga estas instruções para desativar o TLS em seu cluster BeeGFS v8 existente ou novo.

Implantando um novo cluster BeeGFS v8

Para uma nova implantação de cluster BeeGFS, o cluster pode ser implantado com o TLS desativado definindo o seguinte parâmetro no arquivo de inventário do Ansible `user_defined_params.yml`:

```
beegfs_ha_tls_enabled: false
```

Configurando um cluster BeeGFS v8 existente

Para um cluster BeeGFS v8 existente, edite o arquivo de configuração do serviço de gerenciamento. Por exemplo, edite o arquivo em `/mnt/mgmt_tgt_mgmt01/mgmt_config/beegfs-mgmtd.toml` e defina:

```
tls-disable = true
```

Tome as medidas adequadas para reiniciar com segurança o serviço de gerenciamento para que as alterações entrem em vigor.

Informações sobre direitos autorais

Informações sobre marcas comerciais