Настройка HTTPS и обратного прокси для веб-интерфейса JumpServer

Для чего нужен обратный прокси JumpServer?

Nginx отвечает на поддержку защищенных websockets (wss://), организовывая управление подключениями и защитой канала с помощью SSL-сертификата. Для того чтобы функция копирования и вставки в протоколе RDP работала, необходимо развернуть доверенный SSL-сертификат. Использование копирования и вставки в RDP-активах возможно при доступе через протокол HTTPS.

Установка SSL сертификата и настройка HTTPS для веб-интерфейса

Подготовьте SSL-сертификат (обратите внимание, что сертификат **должен быть в формате PEM**). Сертификаты необходимо разместить в директории **/opt/jumpserver/config/nginx/cert**

Остановите сервис JumpServer:

```
./jmsctl.sh stop
```

Откройте файл конфигурации JumpServer

```
vi /opt/jumpserver/config/config.txt
```

Найдите и измените параметры для конфигурации Nginx:

```
## Конфигурация Nginx
HTTP_PORT=80
SSH_PORT=2222
RDP_PORT=3389

## HTTPS Конфигурация
HTTPS_PORT=443              # Внешний порт для HTTPS, по умолчанию 443
SERVER_NAME=www.domain.com   # Ваш домен для HTTPS
SSL_CERTIFICATE=xxx.pem      # Имя вашего сертификата в /opt/jumpserver/config/nginx/cert
SSL_CERTIFICATE_KEY=xxx.key  # Имя файла ключа в /opt/jumpserver/config/nginx/cert
```

Сохраните изменения файла конфигурации и запустите JumpServer

```
./jmsctl.sh start
```

Если вам нужно дополнительно настроить файл конфигурации Nginx:

```
vi /opt/jumpserver/config/nginx/lb_http_server.conf
```

Многоуровневый обратный прокси на Nginx

Подсказка:

Эта конфигурация подходит для случаев, когда на верхнем уровне есть общий внешний прокси-сервер. Это пример многоуровневого обратного проксирования на Nginx. Каждая прокси-секция должна быть настроена для поддержки длительных соединений WebSocket.

Редактирование конфигурационного файла:

```
vi /etc/nginx/conf.d/jumpserver.conf
```

Пример конфигурации без SSL:

```
server {

    listen 80;
    server_name demo.jumpserver.org;  # Замените на ваш домен

    client_max_body_size 4096m;  # Ограничение на максимальный размер загружаемых файлов

    location / {
        # Здесь указывается IP-адрес Nginx сервера JumpServer
        proxy_pass http://192.168.244.144;
        proxy_http_version 1.1;
        proxy_buffering off;
        proxy_request_buffering off;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection "upgrade";
        proxy_set_header Host $host;
        proxy_set_header X-Forwarded-For $remote_addr;
    }
}
```

Рекомендация:

Для более безопасного доступа рекомендуется настроить SSL и использовать протокол HTTPS, следуя рекомендациям Mozilla SSL Configuration Generator.

Пример конфигурации с SSL:

Перенаправление HTTP на HTTPS:

```
server {
    listen 80;
    server_name demo.jumpserver.org;  # Замените на ваш домен
    return 301 https://$server_name$request_uri;  # Перенаправление всех HTTP-запросов на HTTPS
}
```

Настройка HTTPS:

```
server {
    listen 443 ssl http2;
    server_name demo.jumpserver.org;  # Замените на ваш домен
    ssl_certificate sslkey/1_jumpserver.org_bundle.crt;  # Укажите путь к вашему SSL-сертификату
    ssl_certificate_key sslkey/2_jumpserver.org_bundle.key;  # Укажите путь к ключевому файлу сертификата
    ssl_session_timeout 1d;
    ssl_session_cache shared:MozSSL:10m;
    ssl_ciphers ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-CHACHA20-POLY1305:ECDHE-RSA-CHACHA20-POLY1305:DHE-RSA-AES128-GCM-SHA256:DHE-RSA-AES256-GCM-SHA384;
    ssl_prefer_server_ciphers off;
    ssl_protocols TLSv1.1 TLSv1.2;
    add_header Strict-Transport-Security "max-age=63072000" always;

    client_max_body_size 4096m;  # Ограничение на размер загружаемых файлов и записей

    location / {
        # Здесь указывается IP-адрес Nginx сервера JumpServer
        proxy_pass http://192.168.244.144;
        proxy_http_version 1.1;
        proxy_buffering off;
        proxy_request_buffering off;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection "upgrade";
        proxy_set_header Host $host;
        proxy_set_header X-Forwarded-For $remote_addr;
    }
}
```

3. Другие балансировщики нагрузки (SLB)

Подсказка:

1. Необходимо правильно настроить поддержку длительных соединений WebSocket.
2. Важно учитывать вопросы, связанные с управлением сессиями.

Версия #3
Сергей Попцов создал 27 сентября 2024 11:38:39
Сергей Попцов обновил 29 октября 2024 12:18:10