Методы автоматической аутентификации. Создание файла токена

Данный метод аутентификации предназначен для разработки и облегчения начала работы с StarVault Agent и StarVault Proxy. StarVault Agent и StarVault Proxy никогда не должны быть настроены на использование этого метода автоматической аутентификации в производственной среде.

Метод `token_file` считывает существующий, действительный токен StarVault из файла и использует его вместо собственной аутентификации. Хотя это первоклассный метод автоаутентификации, он, естественно, не аутентифицирует сам себя, поскольку ему требуется токен из другого места. Как и другие методы автоавторизации, этот метод будет пытаться обновить токен, если это необходимо.

Этот метод автоматической авторизации особенно полезен при тестировании StarVault Agent или StarVault Proxy без необходимости настраивать какие-либо методы аутентификации в StarVault. Для длительных процессов Agent или Proxy мы настоятельно рекомендуем использовать другой метод автоматической аутентификации, чтобы Agent и Proxy сами отправляли запросы на аутентификацию в StarVault.

1. Конфигурация

- `token_file_path` `(string: required)` - Путь к файлу с токеном внутри. Этот токен не может быть оберточным токеном.

2. Пример конфигурации

Ниже приведен пример конфигурации для StarVault Agent с использованием метода `token_file` для включения Автоматической аутентификации:

JSON

```
pid_file = "./pidfile"

vault {
  address = "https://127.0.0.1:8200"
}

auto_auth {
  method {
    type = "token_file"

    config = {
```

```
      token_file_path = "/home/username/.vault-token"
    }
  }
}
api_proxy {
  use_auto_auth_token = true
}

listener "tcp" {
  address = "127.0.0.1:8100"
  tls_disable = true
}
```

Агент. Совместимость версий

Нет необходимости запускать идентичные версии StarVault Agent и StarVault Server. Безопасно запускать разные версии, однако вы не сможете воспользоваться всеми новейшими функциями StarVault, если не обновитесь до последних версий Agent и Server. Мы понимаем, что это не всегда возможно, поэтому мы поддерживаем несоответствие версий как можно лучше.

Agent будет записывать примечание в свои журналы, когда он обнаруживает несоответствие между Agent и Server. Это чисто информативно, предназначено для помощи в отладке в случае, если несоответствие приводит к проблемам, например, потому что более новая версия Agent пытается использовать функциональность, которая отсутствует в версии Server, с которой он общается. Если Agent ведет себя приемлемо, сообщение может быть проигнорировано.

В этом документе описаны распространенные случаи. Могут быть случайные исключения, которые, если они намеренные, будут указаны в CHANGELOG в разделе `CHANGES`. Если они непреднамеренные/недокументированные, их следует рассматривать как ошибки и сообщать о них.

1. Версия агента старше версии сервера

Мы не ожидаем никаких проблем, возникающих из-за продолжения работы старой версии Agent после обновления серверных узлов до более поздней версии. Существующие развертывания с использованием Agent не должны быть затронуты, поскольку мы обычно не вносим обратно-несовместимые изменения в StarVault Server.

Автоматическая авторизация:

- новые методы аутентификации, которые были введены после создания Agent, будут недоступны
- существующие методы аутентификации должны продолжать нормально функционировать

Прокси:

- Поскольку Агент просто отражает входящие запросы, даже если входящий запрос использует конечную точку, которая не существовала на момент компиляции этой версии Агента, это не помешает Агенту проксировать запрос.

Шаблонизация:

- Функции языка шаблонов, взаимодействующие с сервером StarVault, используют стабильные API StarVault для извлечения и обновления секретов.
- Даже если в новых выпусках StarVault будут введены новые типы секретных движков, они не должны требовать обновления агента для доступа через шаблоны.

2. Более новая версия агента, чем сервер

Возможно, что Агент может зависеть от функций, которых нет в старых версиях Сервера.

Автоматическая аутентификация:

- Агент может заявлять о поддержке новых методов аутентификации, которые были введены с момента создания Сервера, но они не будут работать, поскольку Сервер их не поддерживает
- Агент может использовать новые функции для существующих методов аутентификации, которые недоступны в старом Сервере, который вы используете
    - Обычно мы стараемся сделать такое изменение добровольным или постепенно снижать его при подключении к старому экземпляру Сервера, если только нет очень веской причины (например, исправления серьезной уязвимости безопасности)

Прокси:

- поскольку Агент просто отражает входящие запросы, маловероятно, что несовместимости возникнут при проксировании, но новые функции могут быть недоступны
- **пример:** когда в Агент была добавлена поддержка согласованности, контролируемой клиентом, он начал искать заголовки X-Vault-Index в ответах и начал предоставлять заголовки X-Vault-Index в проксированных запросах. Более старые серверы StarVault, которые не используют эти заголовки, будут игнорировать новый заголовок запроса и также не будут их выдавать. Поведение прокси-агента останется неизменным, не сможет воспользоваться преимуществами новой функциональности, но и не будет ограничено в своем предыдущем поведении.

Шаблонизация:

- мы не ожидаем сценария, в котором изменения в шаблонизации агента приведут к несовместимости со старыми серверами StarVault, хотя, конечно, с любой версией агента можно писать шаблоны, которые выдают запросы, использующие функциональность, еще не представленную на вышестоящем сервере хранилища, например, {{ with secret "secret/my-secret?some-new-option" }}

- мы не будем намеренно вносить изменения в шаблонизацию, которые нарушат существующие развертывания

Совместимость версий

StarVault Proxy и StarVault Server не требуют полного совпадения версий. Допускается использование разных версий, и это безопасно. Однако для доступа ко всем последним возможностям рекомендуется использовать актуальные версии обоих компонентов. Понимая, что обновление не всегда возможно, мы обеспечиваем обратную совместимость между версиями.

При обнаружении несоответствия версий между `Proxy` и `Server`, `Proxy` записывает информационное сообщение в журнал. Эта запись носит диагностический характер и предназначена для упрощения отладки в случае, если расхождение версий вызывает ошибки — например, если более новая версия `Proxy` обращается к функциональности, отсутствующей в используемой версии Server. Если `Proxy` функционирует корректно, это сообщение можно игнорировать.

В этом документе описаны распространенные случаи. Возможны исключения. Если они намеренные, то они будут указаны в `CHANGELOG` в разделе `CHANGES`. Если они непреднамеренные/недокументированные, то они рассматриваются как ошибки.

1. Более старая версия прокси, чем сервер

Обновление серверных узлов до более новой версии не должно вызывать проблем при использовании предыдущей версии Proxy. Существующие развертывания, использующие Proxy, сохраняют работоспособность, поскольку в StarVault Server, как правило, не вносятся несовместимые изменения.

Автоматическая аутентификация:

- новые методы аутентификации, которые были введены после создания прокси, будут недоступны
- существующие методы аутентификации должны продолжать нормально работать

Прокси:

- Поскольку прокси просто отражает входящие запросы, даже если входящий запрос использует конечную точку, которая не существовала на момент компиляции этой версии прокси, это не помешает ему проксировать запрос.

2. Более новая версия прокси, чем у сервера

Возможно, что прокси может зависеть от функций, которые отсутствуют в старых версиях Сервера.

Автоматическая аутентификация:

- Прокси может заявлять о поддержке новых методов аутентификации, которые были введены с момента создания сервера, но они не будут работать, поскольку сервер их не поддерживает.
- Прокси может использовать новые функции для существующих методов аутентификации, которые недоступны в старом сервере, который вы используете.
    - Обычно мы стараемся сделать такое изменение добровольным или плавно снижать его при подключении к старому экземпляру сервера, если только на то нет очень веской причины (например, исправление серьезной уязвимости безопасности).

Прокси:

- поскольку прокси просто отражает входящие запросы, маловероятно, что несовместимости возникнут при проксировании, но новые функции могут быть недоступны