16. Criptografia

Visão geral

O Zabbix suporta comunicação criptografada entre os seus componentes (Zabbix server/Proxy/Agent/Sender/get) através de TLS v1.2. O suporte a criptografia começou no Zabbix 3.0. A criptografia baseada em PSK e certificado também é suportada.

A criptografia é opcional e configurável para cada componente (ex. alguns proxies e agentes podem estar configurados para utilizar criptografia com certificados ao falar com o Server, enquanto outros utilizam PSK, e outros não usam criptografia).

O proxy pode utilizar diferentes configurações de criptografia para diferentes hosts.

Os daemons do Zabbix utilizam uma porta para escutar comunicações criptografadas ou não criptografia. Adicionar a criptografia não exigirá a abertura de novas portas nos firewalls.

Compilando o Zabbix com o suporte a criptografia

Para suportar a criptografia do Zabbix você precisa compilar e associar com uma destas três bibliotecas:

- mbed TLS (antigamente PolarSSL)(a partir da versão 1.3.9, mas o mbed TLS 2.x não é suportado atualmente)
- GnuTLS (a partir da versão 3.1.18)
- OpenSSL (a partir da versão 1.0.1)

A biblioteca é selecionada através de parâmetro no script de configuração:

- `--with-mbedtls[=DIR]`
- `--with-gnutls[=DIR]`
- `--with-openssl[=DIR]`

Por exemplo, para configurar os fontes do servidor edo agente com OpenSSL você pode executar algo assim:

```
./configure --enable-server --enable-agent --with-mysql --enable-ipv6 --
with-net-snmp --with-libcurl --with-libxml2 --with-openssl
```

Diferentes componentes do Zabbix podem ser compilados com diferentes bibliotecas criptográficas (ex. um servidor com OpenSSL, um agente com GnuTLS).

Se você planejar utilizar PSK, considere a utilização das bibliotecas GnuTLS ou mbed TLS nos componentes Zabbix usando PSKs. As bibliotecas GnuTLS e mbed TLS / PSK suportam a suite de cifras com Perfeito encaminhamento de segredo. A biblioteca OpenSSL (versões 1.0.1, 1.0.2c) suporta o PSK mas não tem disponível as suítes de cifras que provejam o perfeito encaminhamento de segredo.

Gerenciamento de conexão criptografada

As conexões do Zabbix podem usar:

- nenhuma criptografia (default)
- criptografia baseada em PSK
- criptografia baseada em certificado

Existem dois parâmetros importantes utilizados para especificar a criptografia de conexões entre os componentes do Zabbix:

- TLSConnect
- TLSAccept

TLSConnect define qual criptografia a utilizar nas conexões de saída e pode pegar 1 de 3 valores (unencrypted, PSK, certificate). TLSConnect é utilizado em arquivos de configuração do Zabbix proxy (em modo ativo) e do Zabbix agentd (para verificações ativas). Na interface web o TLSConnect é equivalente ao campo "Conexões com o host" em "*Configuração → Hosts →<some host>" aba* ***Criptografia*** *e o campo "Conexões com o proxy" em* Administração→Proxies→<some proxy>" aba **Criptografia**. Se um tipo de conexão criptografada falhar, não será tentado outro tipo.

TLSAccept define quais tipos de conexão de entrada serão permitidos. Tipos possíveis: unencrypted, PSK, certificate. Podem ser definidos 1 ou mais valores. TLSAccept é utilizado em arquivos de configuração do proxy (em modo passivo) e do agente (em modo passivo). Na interface web do Zabbix o TLSAccept é equivalente ao campo "Conexões com o host" em "*Configuração → Hosts →<some host>" aba* ***Criptografia*** *e o campo "Conexões com o proxy" em* Administração→Proxies→<some proxy>" aba **Criptografia**.

Normalmente você configura somente um tipo de criptografia para as conexões de entrada. Mas você pode precisar alterar o tipo de criptografia, (ex. de não criptografado para criptografia baseada em certificados) com o mínimo de 'downtime' e possibilidade de rápido retorno. Para fazer isso defina TLSAccept=unencrypted,cert no arquivo de configuração do agente o o reinicie. Então você poderá testar a conexão com o zabbix_get com o agente usando certificado. Se funcionar, você reconfigura a criptografia daquele agente na interface web do Zabbix, configurando para o uso de certificado. Quando o cache de comunicação do servidor for atualizado (e a configuração do proxy for atualizada se o proxy estiver sendo monitorado por um) as conexões começarão a ocorrer de forma criptografa. Se tudo estiver funcionando como o esperado você pode configurar TLSAccept=cert na configuração do agente e reinicia-lo. Agora o agente vai aceitar apenas conexões criptografadas e com certificado. Comunicações sem criptografia ou baseadas em PSK serão rejeitadas.

A configuração funciona de forma similar entre o Zabbix Server/Proxy. Se o host estiver configurado para usar certificado, então apenas comunicações criptografadas com certificados serão aceitas pelo agente (verificações ativas) e pelo zabbix_sender (trapper items).

Provavelmente você irá configurar para que as comunicações de entrda e de saída ocorram com o mesmo tipo de criptografia ou sem criptografia para todos. Mas é tecnicamente possível configurar isso de forma assimétrica, ex. criptografia com certificados para entrada e com PSK para a saída.

Para uma visão geral, a configuração de criptografia de cada host será apresentada na interface do Zabbix no canto direito da listagem de hosts na coluna "Criptografia do agente".

O padrão sao conexões não criptografadas. A criptografia precisa ser configurada em cada host e

proxy individualmente.

Usando certificados

O Zabbix pode utilizar certificados no formato PEM, assinados por uma CA. A verificação de certificado é feita através de um certificado CA pré-instalado. Certificados auto-assinados não são suportados. Opcionalmente uma lista de certificados revogados poderá ser utilizada. Cada componente Zabbix deverá ter apenas um certificado configurado. A escolha entre múltiplos certificados não é suportada.

Para mais informações sobre como configurar e operação interna da CA, como gerar as requisições de certificados e assina-las, como revogar certificados você encontrará em inúmeros sites da internet, por exemplo: OpenSSL PKI Tutorial v1.1

Parâmetros de configuração do certificado

Parâmetro	Obrigatório	Descrição	
TLSCAFile	*	Caminho completo do arquivo contendo os certificados raiz (CA) para verificação dos certificados entre as partes, utilizado para comunicações criptografadas entre os componentes do Zabbix. Certificados de várias CAs deverão ser incluídos em uma única linha.	
TLSCertFile	*	Caminho completo para o arquivo contendo o certificado de agente ou a cadeia de certificados.	
TLSCRLFile	*	Caminho completo para o arquivo contendo os certificados revogados, utilizado para comunicações criptografadas entre os componentes do Zabbix. Se o arquivo definido em TLSCAFile contiver várias CAs e o componente Zabbix for compilado com OpenSSL e TLSCRLFile estiver definido, cada CA mencionada em TLSCAFile deverá ter um correspondente CRL (que pode ser um CRL vazio) no TLSCRLFile	
TLSKeyFile	não		Caminho completo para o arquivo contendo a chave privada do agente. Verifique se o permissionamento do arquivo permite que o usuário 'zabbix' o leia
TLSServerCertIssuer	*	Emissor de certificado autorizado do server (proxy)	
TLSServerCertSubject	*	Destino do certificado permitido pelo server (proxy)	

Usando pre-shared keys (PSK)

No Zabbix cada PSK atualmente é um par de:

- identidade não secreta PSK (texto),
- texto secreto (valor PSK).

A identidade PSK é um texto não vazio no formato UTF-8. Por exemplo, “PSK ID 001 Zabbix agentd”. É o nome único com o qual este PSK específico será referenciado pelos componentes do Zabbix. Não coloque informação sensível na identidade PSK - ela será transmitida de forma não criptografada pela rede.

O valor PSK é mais difícil de adivinhar por ser um texto hexadecimal, por exemplo, “e560cb0d918d26d31b4f642181f5f570ad89a390931102e5391d08327ba434e9”.

Aqui temos um tamanho máximo para a identidade e para o valor PSK no Zabbix, em alguns casos a biblioteca de criptografia tem um valor menor:

Componente	Tamannho da identidade PSK	Tamanho do valor PSK
Zabbix	128 UTF-8 characteres	2048-bit (256-byte PSK, informado como 512 digitos hexadecimais)
GnuTLS	128 bytes (may include UTF-8 characteres)	2048-bit (256-byte PSK, informado como 512 digitos hexadecimais)
mbed TLS (PolarSSL)	128 UTF-8 characters	256-bit (limite padrão) (32-byte PSK, informado como 64 digitos hexadecimais)
OpenSSL	127 bytes (pode incluri caracteres UTF-8)	2048-bit (256-byte PSK, informado como 512 digitos hexadecimais)

A interface web do Zabbix permite configurar identidades PSK de até 128 caracteres e até 2048-bit sem o uso das bibliotecas PSK. Se algum componente do Zabbix suportar valores menores é de responsabilidade do usuário configura ra identidade e valor PSK de forma que ambos o aceitem. Exceder o limite de tamanho resultará em falha de comunicação.

Antes do Zabbix Server se conectar com o agente usando PSK, o servidor analisa a identidade e valor PSK configurados para aquele agente no banco de dados (ou no cache de configuração). Após receber a conexão do agente, ele usa a identidade e valor PSK de sua configuração. Se ambas as partes tiverem o mesmo conjunto a conexão será estabelecida.

É de responsabilidade do usuário garantir que não existam duas chaves PSK com o mesmo conteúdo, mas com valores diferentes. Isso poderá causar interrupções imprevisíveis de comunicação entre os componentes.

Configurando PSK para a comunicação server-agent (exemplo básico)

No host do agente salve o valor PSK em um arquivo, por exemplo, `/home/zabbix/zabbix_agentd.psk`. O arquivo precisa conter o PSK em sua primeira linha, por exemplo:

```
1f87b595725ac58dd977beef14b97461a7c1045b9a1c963065002c5473194952
```

Defina o permissionamento do arquivo, somente o usuário Zabbix deverá conseguir lê-lo.

Edite os parâmetros de TLS no arquivo de configuração do agente `zabbix_agentd.conf`:

```
TLSConnect=psk
TLSAccept=psk
TLSPSKFile=/home/zabbix/zabbix_agentd.psk
TLSPSKIdentity=PSK 001
```

O agente irá se conectar com o servidor (verificação ativa) e aceitará do servidor e do `zabix_get` apenas conexões usando PSK. Neste caso a identidade PSK será “PSK 001”.

Reinicie o agente. Agora você pode fazer um teste de conexão utilizando o `zabbix_get`:

```
zabbix_get -s 127.0.0.1 -k "system.cpu.load[all,avg1]" --tls-connect=psk --
tls-psk-identity="PSK 001" --tls-psk-file=/home/zabbix/zabbix_agentd.psk
```

(Para minimizar o tempo de indisponibilidade veja como mudar o tipo de conexão em gerenciamento de conexão criptografada).

Configure o PSK para este host na interface web do Zabbix.

- Acesse *Configuração → Hosts*
- Selecione o host desejado, clique em seu nome
- Clique na aba **Criptografia**
- Defina o campo *Conexões com o host* para *PSK*
- Em *Conexões do host* marque a opção *PSK*
- No campo *Identidade PSK* Informe o valor “PSK 001”
- No campo *PSK* Informe o valor “1f87b595725ac58dd977beef14b97461a7c1045b9a1c963065002c5473194952”
- Clique no botão *Atualizar*

Após a atualização do cache de configuração do Zabbix Server/Proxy as comunicações começarão a ocorrer através de conexões criptografadas com PSK. Eventuais erros podem ser localizados tanto no log do agente quanto no log do servidor

Configurando o PSK para comunicação entre Zabbix Server e Zabbix Proxy (ativo) (exemplo básico)

Salve o arquivo de valor PSK do proxy em um arquivo, por exemplo, `/home/zabbix/zabbix_proxy.psk`. O arquivo precisa conter o valor em sua primeira linha:

```
e560cb0d918d26d31b4f642181f5f570ad89a390931102e5391d08327ba434e9
```

Defina o permissionamento do arquivo, somente o usuário Zabbix deverá conseguir lê-lo.

Edite os parâmetros de TLS no arquivo de configuração do agente `zabbix_proxy.conf`:

```
TLSConnect=psk
TLSPSKFile=/home/zabbix/zabbix_proxy.psk
TLSPSKIdentity=PSK 002
```

O proxy vai se conectar ao servidor usando PSK e a identidade PSK será “PSK 002”.

(Para minimizar o tempo de indisponibilidade veja como mudar o tipo de conexão em gerenciamento de conexão criptografada).

Configure o PSK para este proxy na interface web do Zabbix.

- Acesse *Administração → Proxies*
- Selecione o proxy desejado, clique em seu nome
- Clique na aba **Criptografia**
- Em *Conexões do proxy* marque a opção *PSK*
- No campo *Identidade PSK* Informe o valor “PSK 002”
- No campo *PSK* Informe o valor “e560cb0d918d26d31b4f642181f5f570ad89a390931102e5391d08327ba434e9”
- Clique no botão *Atualizar*

Reinicie o proxy. Ele irá se comunicar com o servidor usando conexão criptografada baseada em PSK. Verifique os logs do proxy e do server por mensagens de erro.

Para proxies passivos o processo é muito similar, a única diferença é que tem que se configurar o parâmetro `TLSAccept=psk` no arquivo de configuração do proxy e definr o campo *Conexões com o proxy* na interface web do Zabbix para PSK.

Limitações

- As chaves privadas são armazenadas em texto plano e os arquivos são possíveis de leitura pelos componentes do Zabbix durante sua inicialização.
- PSK informadas na interface web do Zabbix são salvas no banco de dados em texto plano.
- A criptografia não garante as comunicações:
  - entre o servidor web e o navegador do usuário,
  - entre o Zabbix Server (ou proxy) e seu banco de dados.

Soluções de problemas

`gnutls_handshake() failed: -110 The TLS connection was non-properly terminated.` in TLS client side log.
`gnutls_handshake() failed: -90 The SRP username supplied is illegal.` in TLS server side log.

Causa possível: Identidade PSK maior que 128 informada para a biblioteca GnuTLS.

`ssl_set_psk(): SSL - Bad input parameters to function`

Causa possível: PSK maior que 32 bytes informada para a biblioteca mbed TLS (PolarSSL)