使用GlobalSign IoT Edge注册自动证书更新

版本 1.0

8月11日, 2020

介绍

证书生命周期管理的一部分是决定如何处理即将过期的证书。在物联网(IoT)世界中，一旦设备被部署到现场，人工干预通常是昂贵或困难的。如果证书过期导致设备身份失效，那么与该设备的通信通常会停止，因为身份验证将失败。因此，部署一个强大的更新机制以确保设备继续运行，同时也维护有限的证书生命周期给物联网(IoT)生态系统带来的安全好处，这一点很重要。

本指南将演示如何使用无处不在且可靠的Cron实用程序以及GlobalSign的IoT Edge Enroll服务自动进行证书更新。Cron通常在大多数Linux发行版中都是默认可用的。需要注意的是，本指南应该作为自动证书更新的起点或演示，而不是在生产环境中依赖的临时解决方案。证书生命周期的自动化可以通过多种方式实现，下面的例子并不总是适合所有的物联网生态系统。

更新脚本

1. 将以下bash脚本复制到带有要监视的证书的设备上，更新
   highlighted 参数设置为正确的值。脚本会检查指定的证书是否超过过期阈值。
   如果是，则会从现有的文件中创建CSR文件
   证书的属性，并在IoT Edge Enroll EST reenroll端点下登记:

```
#!/bin/bash

# Set parameters applicable to your renewal conditions
renewIn=432000      # Seconds before expiration to attempt renewal
certToWatch="$1"    # The certificate file to watch
renewalEndpoint="https://opentest.est.edge.dev.globalsign.com:443/.well
-known/est/"        # Your Edge EST URI
key="$2"            # Private key for the certificate, will not be
rotated
# verify files
([ -f "$certToWatch" ] && [ -f "$key" ]) || exit 1
# check if certificate renewal period has been reached or exceeded
openssl x509 -checkend "$renewIn" -noout -in "$certToWatch" && exit 1
# Renew the certificate as renewal period has been reached
# Generate a CSR matching the previous certificate parameters
openssl x509 -in cert.pem -x509toreq -signkey "$key" | awk '/---/,0' >
"${certToWatch}".csr
# Reenroll for the certificate, sending the old certificate and
converting the new certificate
# to PEM format from pkcs7
curl -s -X POST --data-binary "@${certToWatch}.csr" --cert
"$certToWatch" --key "$key" -H "Content-Transfer-Encoding:base64" -H
"Content-Type:application/pkcs10" "${renewalEndpoint}"/simplereenroll |
openssl base64 -d -a | openssl pkcs7 -inform der -print_certs | tail -n
+5 > "${certToWatch}".new

# Test the new certificate before replacing the old one
openssl x509 -in cert.pem.new -noout -subject || exit 2
```

```
# Pull the CA certs, in case they have been updated since last renewal,
then append them to the new certificate file
curl -s  "${renewalEndpoint}"/cacerts | openssl base64 -d -a | openssl
pkcs7 -inform der -print_certs | tail -n +5 | sed
'/subject\|issuer\|^[[:space:]]*$/d' >> "${certToWatch}".new

#replace the old certificate with the new one then clean up
mv "${certToWatch}".new "$certToWatch"
rm "${certToWatch}".csr
```

2. 使脚本可执行:

```
chmod +x renewalCheck.sh
```

注意:为了更方便地处理API函数，可以考虑使用类似GlobalSign提供的EST客户机。上面的脚本使用了直接的API调用，以达到清晰和演示的目的。

使用Cron调度

3. 创建/编辑crontab文件，作为应该执行证书更新的用户(并且对必要的文件有访问权限):

```
crontab -e
```

4. 编辑crontab，添加一个新条目，以适当的时间间隔调度脚本。在这个例子中，脚本将为cert1运行。CRT每天上午9点。请确保指定了证书、脚本和私钥的完整路径:

```
00 09 * * * /home/user/scripts/renewalCheck.sh /var/www/certs/cert1.crt
/var/www/certs/key1.key
```

5. 保存文件，cron将根据计划自动开始运行脚本。

6. 对于同一系统上的其他证书，请在crontab中添加另一行:

```
30 14 * * Mon /home/user/scripts/renewalCheck.sh
/var/www/certs/cert2.crt /var/www/certs/key2.key
```

这个条目将针对cert2运行脚本。每周一下午二时三十分见。请确保选择适合特定用例的周期，在更新周期内有足够的重复次数，以应对连接丢失或停机可能导致设备在到期前错过更新的情况。

crontab条目的一般格式是:

```
[Minute] [Hour] [Day of Month] [Month] [Day of Week] [Script to
Execute] [Script arguments]
```

'*' 意味着 'every'.

7. 在依赖系统之前，请验证系统是否按预期工作。根据设备的预期可用性和对网络的访问，可能需要调整证书生命周期或更新周期。

关于GlobalSign

GlobalSign是全球领先的可信身份和安全解决方案提供商，使全球的企业、大型企业、云服务提供商和物联网创新者能够进行安全的在线通信，管理数百万已验证的数字身份以及自动化认证和加密。其大规模的PKI和身份解决方案支持数以亿计的服务、设备、人和物组成的万物互联。公司在美洲、欧洲和亚洲设有办事处。

了解GlobalSign的IoT Edge注册服务:

https://www.globalsign.cn/iot-edge-enroll