Using Let's Encrypt certificates with Brocade vADC

by Baptiste Assmann 01-19-2017 10:05 AM - edited 01-24-2017 12:27 AM (719 Views)

Letsencrypt.org is a free and automated Certificate Authority that makes it easy for organizations to secure websites. It can set up TLS certificates very easily, limited to one domain name (i.e., www.domain.com) and has the advantage that it supports both RSA and ECC certificates.

In this article, we show an example of how to configure **Let's Encrypt** to work with Brocade vADC, including:

- Issue new certificates
- Automated renewal of certificates
- Install certificates and tools
- Use both RSA and ECC for performance and maximum compatibility
- Enable automatic OCSP stapling

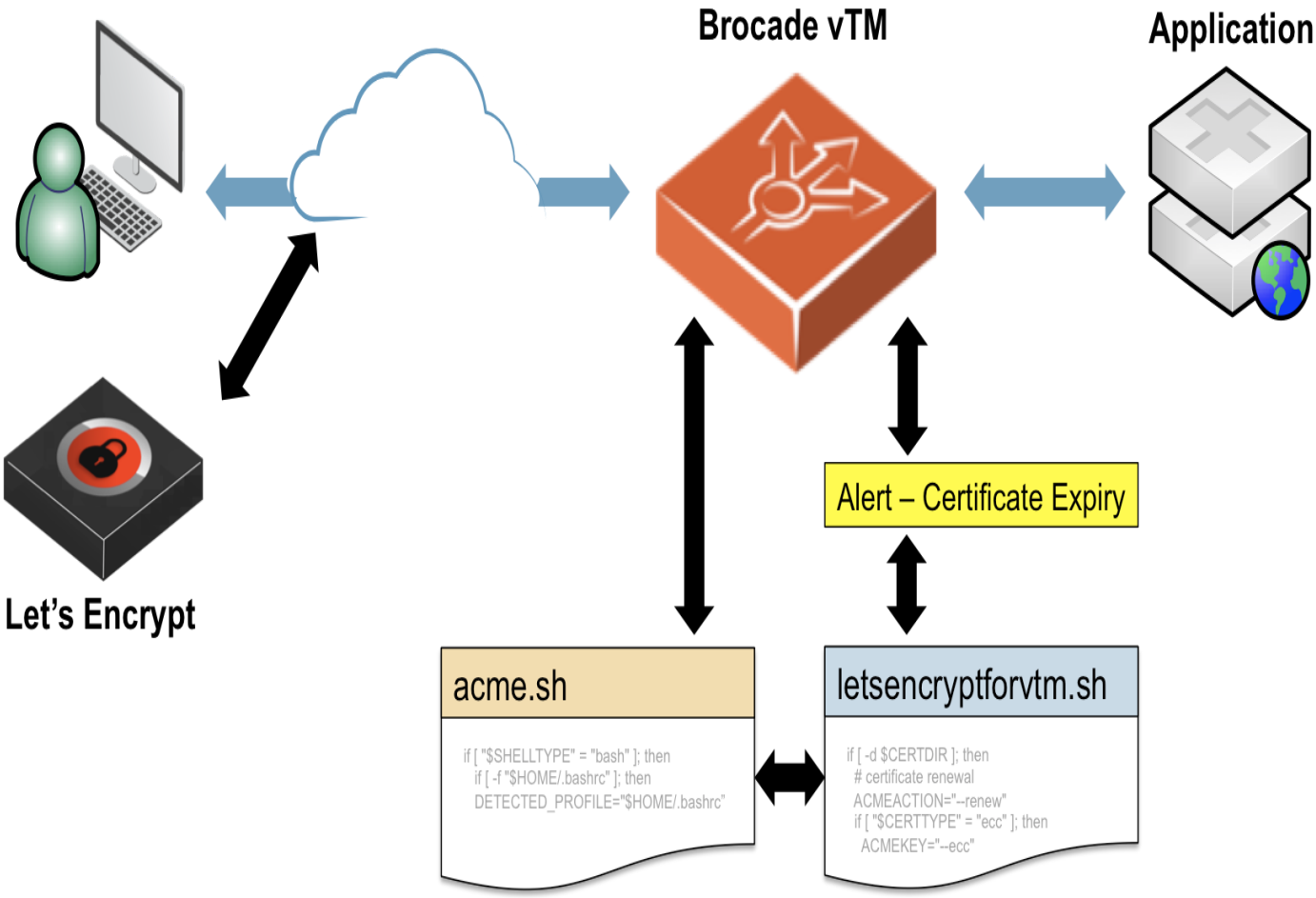
Introduction

Let's Encrypt relies on the ACME protocol for Automated Certificate Management Environment, which was developed with automation in mind:

- The software client connects to **Let's Encrypt** to issue or renew a certificate
- An authorization/signature happens to verify you are the owner of the domain
- The certificate is generated and sent back to the software client

We'll use an open source client (acme.sh) to manage communications with **Let's Encrypt** and we install a short script (letsencryptforvtm.sh) into Brocade vTM, which is used to issue and renew certificates. Then, we use the Brocade vTM alerting and scripting to trigger certificate renewal automatically through the open source script (acme.sh) to talk to Letsencrypt.org. The steps are as follows:

1. Install the open-source acme.sh client
2. Install our action script (letsencryptforvtm.sh)
3. Create a new action type for Brocade vTM
4. Create a new alert for Brocade vTM
5. Create a resource pool to manage responses from Letsencrypt.org
6. Create a TrafficScript rule to manage responses from Letsencrypt.org



Installation procedure

This installation procedure can be applied the same way if you're running the Brocade vTM appliance or the Brocade vTM software on your own Linux distribution.

Install acme.sh

First, we need to download the open source component to manage the ACME protocol. Login as admin into your vTM through SSH, then run the following commands:

```
curl -LO https://raw.githubusercontent.com/Neilpang/acme.sh/master/acme.sh
chmod +x acme.sh
./acme.sh --install --nocron
```

More options are available when installing acme.sh. For more information, please read https://github.com/Neilpang/acme.sh/wiki/How-to-install.

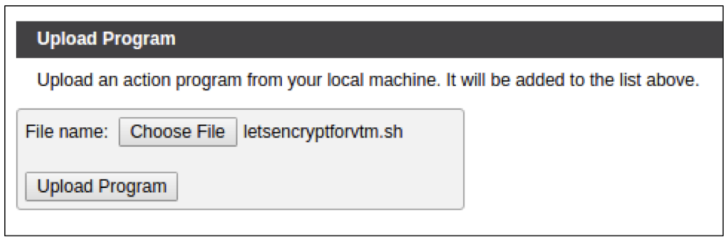
Install letsencryptforvtm.sh

Next, we need to download and install the short script that is used to issue and renew certificates and communicate with the acme.sh component.

- Download the **letsencryptforvtm.sh** script (Available on github: https://github.com/bedis/letsencryptforvtm)
- Upload the script into the admin's home directory on Brocade vTM

This script can be run manually to issue a new certificate and insert into Brocade vTM, but we need to load it into the Brocade vTM Catalog, so that it can be called automatically to renew certificates.

- Connect to the Brocade vTM Web UI, then navigate to **Catalogs** > **Extra Files** > **Action Programs**
- Click on the **Choose File** button and point to the letsencryptforvtm.sh script
- Click on the **Upload Program** button

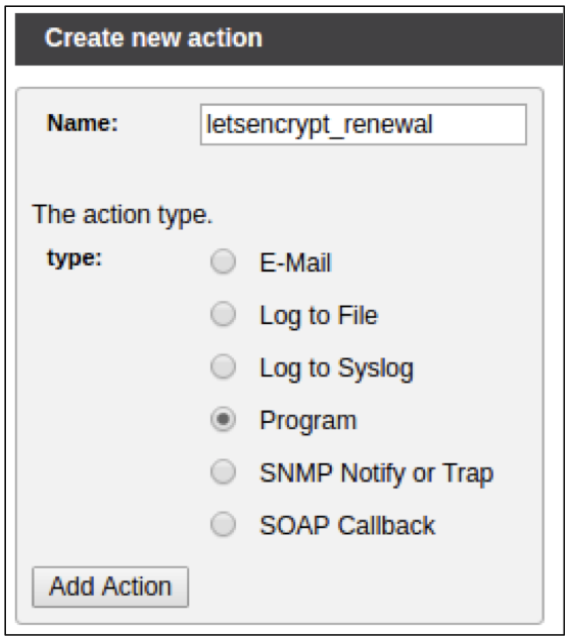


Set up a new Action

Now, we need to create a new action, which will be called when a certificate is about to expire.

- Connect to the Web UI, then navigate to **System** > **Alerting**
- Click on the **Manage Actions** link
- Complete the "Create new action" form:

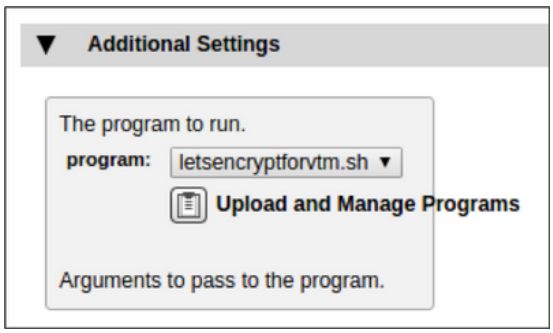
Name: letsencrypt_renewal
Type: Program



- Click on **Add action**
- On the next page, in **Additional Settings**:

Program: choose **letsencryptforvtm.sh**

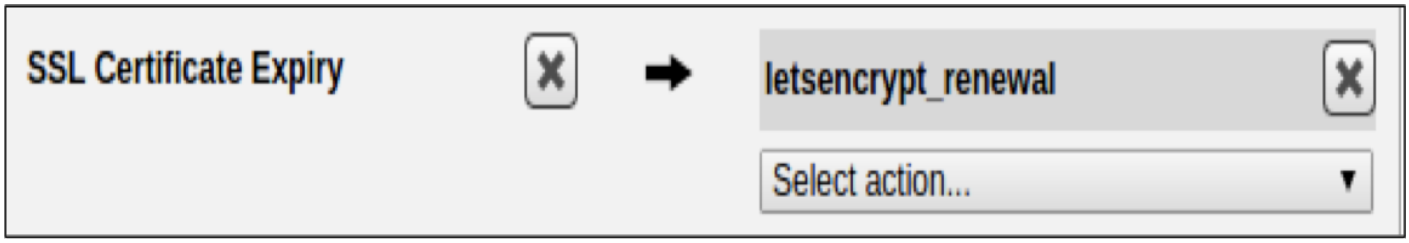
- Click on the **Update**button at the bottom of the page



Create a new Alert mapping

In Brocade vTM, an alert maps an event to an action. In our case, we'll match the event "Certificate is about to expire" to the action we've created at the step before.

- Connect to the Web UI, then browse **System**> **Alerting**
- In **Select Event Type**, choose **SSL Certificate Expiry**
- As an action, choose **letsencrypt_renewal**



Note: the **SSL Certificate Expiry** will match for all certificates configured into Brocade vTM, whether or not they were issued by **Let's Encrypt**. It is possible to create a copy of this event, named **SSL Certificate Expiry Let's Encrypt** and match only the vservers where your **Let's Encrypt** certificates are enabled.

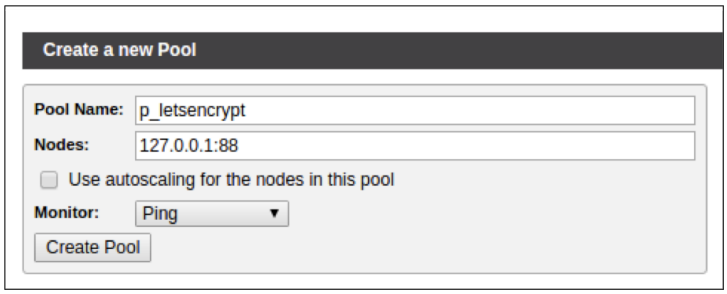
Create a pool for the acme.sh script

The purpose of this pool is to send ping back from **Let's Encrypt**. While we are running **acme.sh**, we will need to make it listen for http requests on port 88.
Note that once the certificate has been renewed or issued, then acme.sh will shutdown the port.

- Connect to the Web UI, then browse **Services** > **Pools**
- Complete the **Create a new pool**form as below:

Pool name: p_letsencrypt
Nodes: 127.0.0.1:88
Monitor: Ping

- Click on the **Create Pool** button



Create a new TrafficScript rule

The purpose of this rule is to route ping back from letsencrypt.org to the pool which we created (**p_letsencrypt)** - which itself will route the request to the **acme.sh** script.

- Connect to the Web UI, then browse **Catalogs** > **Rules**
- Complete the **Create a new rule** form with the following information:

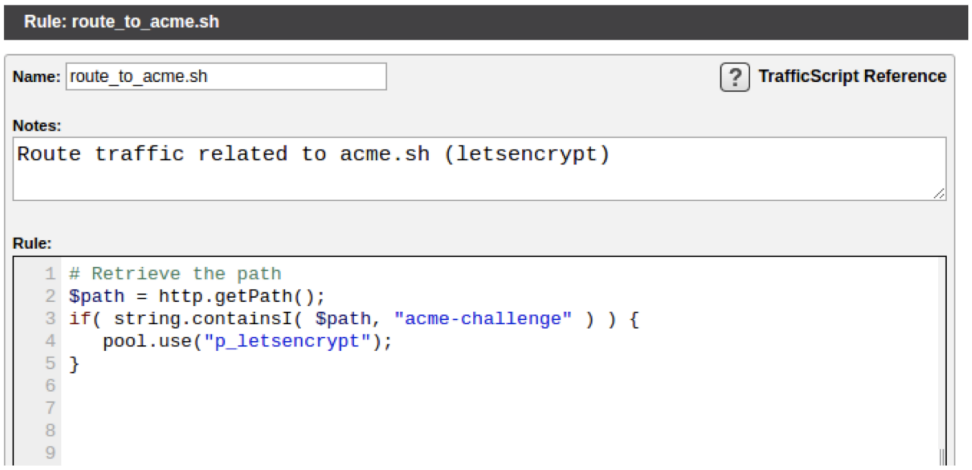
Name: route_to_acme.sh
Check the **Use TrafficScript Language** option
Click on the **Create Rule** button

- In the next page, complete the form as below:

Notes: Route traffic related to acme.sh (letsencrypt)
Rule: As shown here:

```
$path = http.getPath();
if( string.containsI( $path, "acme-challenge" ) ) {
   pool.use("p_letsencrypt");
}
```

- Click on the **Update** button
- Later, you will add this rule to the **vserver** for your application



Complete your environment

If you have not already set up your application with a **vserver**, you will need to create a **vserver** listening on port 80 on the IP address pointed by the domain for which you are issuing the certificate.

Now you can enable the TrafficScript rule **route_to_acme.sh** into the **vserver** which is managing the **domain**. *This should be one of the first rules in the list*.

Generate a new certificate

In order to generate a new certificate for our application, we need to run the script to request a new certificate from **Let's Encrypt**:

- Connect to vTM using ssh
- Run the following command for an ECC certificate:

```
./letsencryptforvtm.sh --issue c_www.domain.com_ecc
```

- Alternatively, run this command to request an RSA certificate:

```
./letsencryptforvtm.sh --issue c_www.domain.com_rsa
```

The new certificate is automatically inserted into Brocade vTM, which you can confirm by navigating to **Catalogs** > **SSL**. You can now navigate to your **vserver**, enable SSL offloading and select the new certificate.

Let's Encrypt certificate chain

The **letsencryptforvtm.sh** script takes care of this task for you. When inserting the certificate into Brocade vTM, the script uses the full chain, including the certificate for the domain and the required intermediaries.

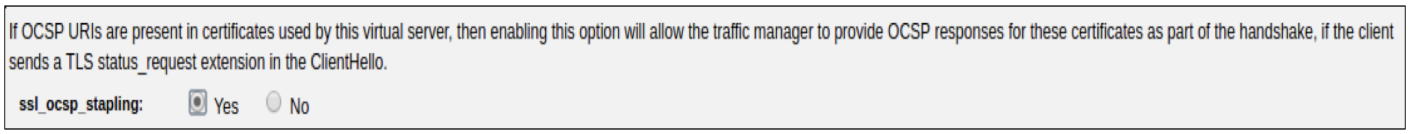
Renew a Let's Encrypt certificate

When the certificate is due for renewal, our script should take care of the certificate renewal. Seven days before expiration, the alert mapping will run the **letsencryptforvtm.sh** script with the name of the certificate as an argument.
If, for some reasons, **Let's Encrypt** is not available at the first execution, Brocade vTM will attempt to call **letsencryptforvtm.sh** every hour until the certificate is renewed.

OCSP stapling

Brocade vTM can use information available in the certificate to process OCSP stapling automatically. This feature works out of the box with **Let's Encrypt** certificates.

All you need to do is to enable **ssl_ocsp_stapling** in your **vserver** when configuring **SSL Decryption**.



Using the TEST environment variable

It is highly recommended to use the **Let's Encrypt** test / staging environment during the installation phase. Otherwise, **Let's Encrypt** may blacklist your domain if you generate too many certificates.

In order to use the test environment, edit **letsencryptforvtm.sh** script and search for the TEST variable: uncomment the TEST variable and re-upload the script into **Catalogs** > **Extra Files** > **Actions**.

Once the full procedure is validated and you want to move to production, simply comment out the TEST variable line and re-upload the script into **Catalogs** > **Extra Files** > **Actions**.

Labels:

Software Networking