Controller Address Changes - Permanent and Temporary

Problem Description

There are times when a controller legitimately needs to tell an EVE device to go to a different URL. As of this writing, there are two such use cases:

1. Permanent Redirection: the controller address will cease to control the device
2. Temporary Redirection: the controller address will continue to control the device *in general*, but for this session, the device should go somewhere else

One of the key value propositions for EVE has been that it provides cloud-style operation for edge devices. Edge devices are significantly more expensive to update by hand than devices in data centres. In the cases of devices in hard to reach locations, e.g. cell towers, wide turbines, airplane engines, oil rigs, the cost can be orders of magnitude more. If the controller address is hard-coded into the device and cannot be changed without a "truck roll", one of the key benefits of EVE is lost.

The rest of this section describes each of the use cases.

Permanent Redirection

There are several use cases for permanent redirection.

- The controller is being retired, as the operator no longer will maintain it, e.g. an open-source controller that controls 20 devices will move to a commercial controller.
- The controller is being replaced, e.g. a commercial controller operator has several controllers and wish to consolidate
- The controller is being removed entirely, e.g. a commercial controller operator is going out of business and is to be replaced with an alternate operator.
- The controller is being consolidated, e.g. two controller operators merge and wish to combine their operations to a single controller address

Temporary Redirection

There are several use cases for temporary redirection.

- Geo-balancing: A controller operator may have multiple controllers balanced by region, where they wish to have a single global address to initiate, but send the device to a specific controller based on region.
- Maintenance: A controller operator may need to bring a specific named controller down for maintenance.
- Client direction: A controller operator may actually operate several controllers with real data, but one controller endpoint. Company X's data may reside at x.operator.com, while Company Y's may reside at y.operator.com, but all start globally at global.operator.com which directs it to the correct company-specific one.

Security Implications

The primary security implication is that a device would be able to be remotely directed to a different controller. As a device trusts its controller completely, this can be an avenue of attack. However, this should be a minor one. It is easier to hijack a DNS record or BGP announcement to get an address than steal a TLS private key. At heart, an EVE device trusts a controller not because of a specific hostname or FQDN, but rather because the TLS channel is validated via the certificate already loaded on the device.

Once a device trusts a controller because it has a valid certificate, it trusts it entirely, including telling it to go trust a different controller.

Nonetheless, we need to recognize that some implementors may *want* to make the name immutable. Thus, we should support a config that makes a device ignore redirection. This should be configurable in two places:

- a `/config` flag, in which the filesystem indicates, "this device may not change controllers"
- a config option, i.e. lock down device to controller, which adds that flag

Design Proposal

Since the communication between device and controller is over http/s, and http already has both permanent and temporary redirect codes, `301` and `302` respectively, we propose to use those codes to indicate a redirect from controller to device.

- If the device receives a `301 Permanent`, it **should** change the address in `/config` on the device.
- If the device receives a `302 Temporary`, it **should not** change the address in `/config` on the device. Instead, it keeps the new address in memory for a specified time. When that expires, or at device reboot, the next request goes to the original address. Of course, the original address can issue another `302 Temporary` redirect, and the cycle continues.

Valid Endpoints for Redirect

An open question is if the `301` and `302` should be supported at all stages, i.e. all requests from the controller, or just a limited subset. The limited subset would be:

- `GET /config`
- `GET /ping`
- `POST /register`

The reason to support only a subset, is that it would be indeterminate if a device registers to controller global.operator.com, gets its config, operates for a while, and then when it sends metrics, suddenly gets a redirect to foo.operator.com. Does this other operator have the device registered? Is the config accurate?

For this reason, we propose that all endpoints can return `301` or `302`. However, if a device receives a redirect at any address other than the above, it must return to its starting cycle, download config, and then continue operations. This ensures that the device is properly affiliated with the controller and functioning.

Deregistration

The above process assumes that the device either is registering for the first time, at which point a redirect simply restarts the registration process at a different controller address, or reloads its config from the newer controller address, implying that it already is registered there, i.e. registration has transferred, likely because the operator is the same.

What should be the process for transferring to a new controller under entirely new registration? For example, Company X is moving its device from Operator A at foo.com to Operator B at bar.com. Controller A has the device registered with all of its information; Controller B does not. Even if Company X somehow pre-registers its device with Controller B, there may be registration process steps that are required. Thus, the process needs to be:

1. Connect to Controller A as usual, e.g. to `GET /config`
2. Receive permanent redirect to Controller B
3. Connect to Controller B, `GET /config`
4. Receive notice that the device must register

This actually is part of a larger issue, forcing a device to re-register, and thus beyond the scope of this proposal.

Requirement

In order to support transfers between controllers, which, as stated above, is a key value of EVE, we propose that the redirect be an API requirement, and all controllers must support it.