ARTICLE

More Secure, Less Costly IoT Edge Node Security Provisioning

November 9, 2015

By Eustace Asanghanwa

Imagine that you've purchased a fancy new connected home thermostat from some well-known company like Home Depot. It can communicate with the Cloud, specifically a network that Home Depot has built. But mostly, it's just a thermostat, a simple edge node device on a cloud-based system. Why would you need security in a thermostat? No one can get into your bank account with it; no one can spy on your children's phones with it; it's just a thermostat that happens to be able to phone home.

That is, until someone hacks through your thermostat into Home Depot's network, where they can then help themselves to credit card numbers and other tempting pieces of information that they're not supposed to be able to reach. And this isn't a hypothetical scenario: *it actually happened*.[1]

Now put yourself in Home Depot's position, after being burned by the hacker. Let's say you're approached by someone with an even niftier thermostat, and they'd like to partner up with you so that you'll sell their unit and connect it to your Cloud facility. You may be interested in working with them – *until* they get to the part where they tell you that they didn't build in security, because after all it's just a thermostat; what could possibly go wrong?

You know exactly what can go wrong, because you have the scars to prove it. You pass on the deal, and the thermostat maker learns the hard way, without security, they saved some cost, but do not sell many units.

This makes edge nodes the most vulnerable points in the Internet of Things (IoT). It's too easy to think of them as isolated embedded systems of no value to a hacker, and especially in the home market, cost is important. Every little opportunity to save is welcomed. Omitting security is not an option; all edge nodes just like any other node on the network, must be secure.

Two-Step Security

There are many components to security, but they all involve the exchange of certificates and other trusted artifacts that have an established chain of trust and rely on a unique device private key (from which a public key is derived). Implementing a security strategy on an edge node involves two separate steps: provisioning and commissioning.

[1] Banjo, Shelly. "Home Depot Hackers Exposed 53 Million Email Addresses." WSJ. November 6, 2014. www.wsj.com/articles/home-depot-hackers-used-password-stolen-from-vendor-1415309282.

Provisioning takes a raw device and gives it an identity for security purposes. This involves loading a unique private key and any other certificates or artifacts necessary for eventual use. Provisioning establishes the device as legitimate and trustworthy.

The trust inherent in a well-provisioned device originates with a “root of trust” backing up all other trusted entities in a chain of trust. The root can be generated by anyone. What’s important is that it be, by definition, trusted unquestioningly. Buyers and network managers would need to understand who that root is in order for the trust to be credible.

A large, well-known company with an established reputation at stake can probably make its own root. A small start-up or maker organization, by contrast, probably cannot. It’s not because the smaller company is any less trustworthy; it’s simply because they’re not well known, they must find someone else to vouch for them.

This is where Certificate Authorities (CAs) like Verisign fit in; they are in the business of being trusted and of vouching for other companies. So, while a large company might sign its own keys, small companies are likely to have keys signed by a CA.

Provisioning prepares a device for use, but it leaves the device in a generic state. *Commissioning* (or “on-boarding” or “taking ownership”) is the second security step performed after the device is purchased. It’s the act of letting the pre-provisioned device join a network, and it usually involves an account with the network provider and validation steps like passwords delivered by cell phone. It transforms the device from generic to one associated with a specific user on a specific network.

Provisioning is done during the device (or component) manufacturing or testing process; commissioning is managed by the network provider and end user. Successful commissioning relies on proper provisioning, but traditional provisioning techniques incur significant costs and leave security holes. A new approach to provisioning is emerging that provides better security at lower cost.

The Provisioning Problem

A signed key is the minimal requirement for provisioning a device. This key will eventually need to be recognized and accepted by a network during commissioning. Provisioning is done with the assistance of a Hardware Secure Module (HSM) that protects the loading of keys in an auditable process.

Not just any key will do when provisioning. If that were the case, then manufacturers could easily make more devices than ordered (so-called “over-building”) and sell the excess on the gray market. In order to thwart this, the manufacturing process must account for all of the keys assigned. The network that will accept a device must be sure that the device is legitimate, so the network must know which keys were assigned during manufacturing. Only those keys will be allowed to join the network.

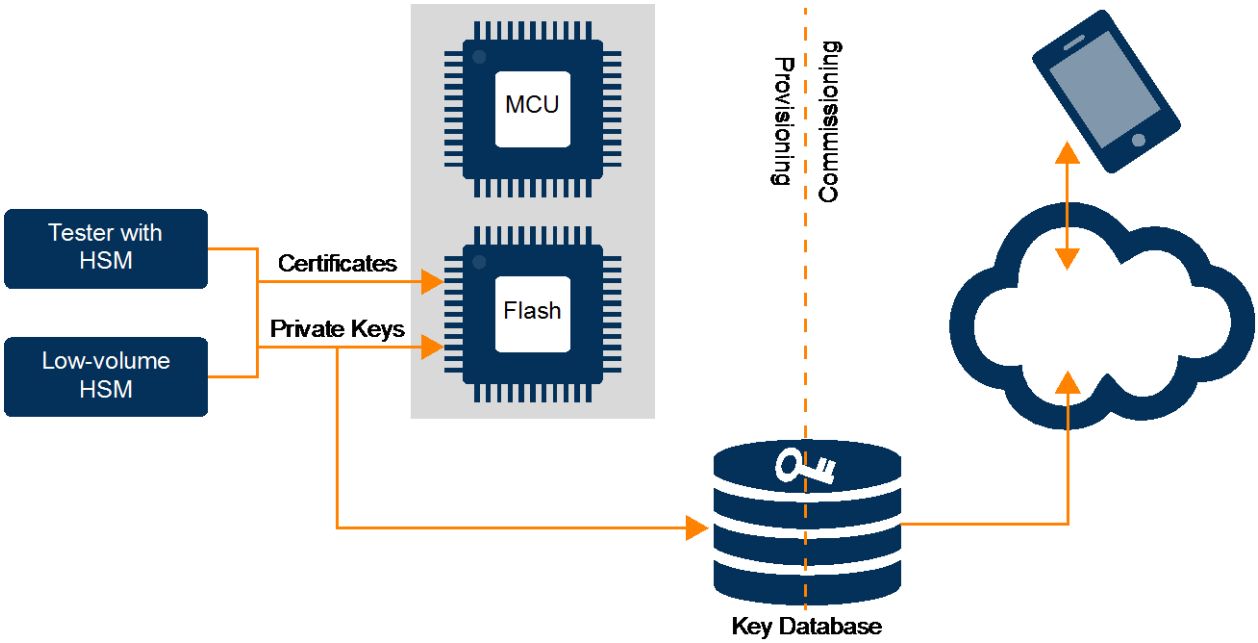
This means that a database of keys must be maintained. The database is built when devices are manufactured, and is used by the network operator to validate keys when they’re commissioned. Maintaining and managing this database is cumbersome and adds cost, but, worse yet, it becomes a single point of attack for the network. If someone can hack into the database, then all of the keys are compromised.

While this process prevents outright over-building, it doesn’t prevent cloning. In the case of cloning, more than one device is built with the same key. One is sold on the legitimate market; the clones are sold on the gray market.

Small companies need to turn to CAs for signed keys, and those CAs charge for each key. That adds to the bill of materials of each device manufactured. And, while large companies can amortize HSM costs over large quantities of devices or systems, those costs can be substantial for smaller companies provisioning modest numbers of devices.

Finally, if the keys are stored in Flash memory in the edge node device, it becomes possible for a determined hacker to find and decrypt the key. While not a result of the provisioning process per se, it adds to the downsides of the way provisioning has traditionally been done.

Figure 1. Provisioning and Commissioning



Note: Traditionally, devices or systems are provisioned either on high-volume testers or low-volume HSMs. Keys are recorded in a database that the network consults during commissioning

Cheaper Provisioning for Better Security

An alternative approach to provisioning is now available, based on newer devices whose sole purpose is to act as a trusted guardian of keys, certificates, and other security secrets. Such a device is provisioned with a private key (ideally, generating the key internally), but that key is never divulged by the device. Instead, when a key is needed, the device generates a public key, or any other secure artifact, derived from, but not the same as, the private key.

Primarily, this prevents cloning. The key in the type of system is controlled by a single, separate device – not other commodity chips such as Flash memory. If 1,000 systems are built, then exactly 1000 units of the security device is provided, leaving no extras for cloned versions.

These devices permit storage of other security elements like certificates. They may provide key and certificate generation and storage, as well as, authentication support in an opaque manner; thereby offloading the system processor.

The Trusted Platform Module (TPM) is one well-known example of hardware designed to support these security functions. But devices based on the TPM typically can't meet the cost expectations of low-cost IoT edge nodes.

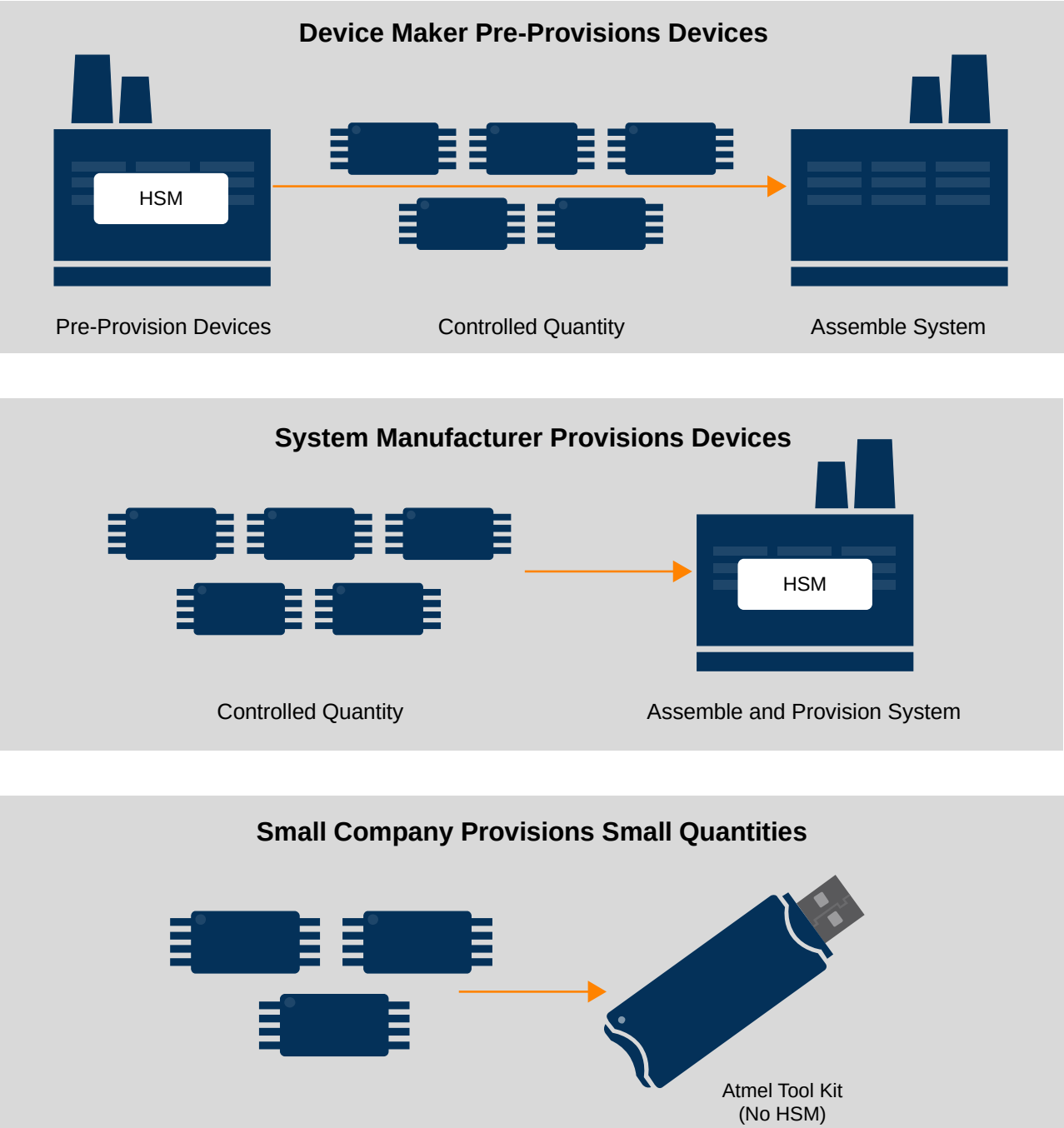
A lower-cost example is the Atmel® ATECC508 CryptoAuthentication™ device. As with Atmel TPM devices, this device creates its own private key internally when triggered during provisioning. That private key can never be read by anyone, making cloning completely impossible. It also eliminates the need for a database, saving time and effort, as well as, taking away a single point of attack for the network.

At higher volumes, Atmel can pre-provision devices during testing as a service. Alternatively, a system manufacturer can do the provisioning using its tester. For lower volumes, Atmel provides a toolkit that allows system design houses to provision prototypes and first-run devices in-house. This saves the cost of investing in an HSM. The toolkit provides the option of either self-signing or acquiring a single CA certificate to use as the basis for the chain of trust for each device, making it unnecessary to buy a separate CA certificate for each device.

Using a dedicated security device can help close security holes and reduce provisioning cost. Security is improved by eliminating cloning and removing the key database as a hacking target. Costs are reduced by eliminating the need to maintain a key database and getting rid of per-unit key charges.

This combination of higher security and lower cost make it more attractive and affordable to add security to IoT edge nodes. Given the increased scrutiny of edge nodes, especially by network managers, such security will be mandatory for any edge node to achieve significant sales and for the IoT to build out as promised.

Figure 2. Provisioning Options



Note: Devices in large quantities can be pre-provisioned by a device maker like Atmel or by an authorized assembly house during system test. Low quantities can be provisioned using an Atmel toolkit without an HSM investment.

Atmel Corporation 1600 Technology Drive, San Jose, CA 95110 USA **T:** (+1)(408) 441.0311 **F:** (+1)(408) 436.4200 | **www.atmel.com**