Khimo, a secure access platform for endpoint smart grid connectivity

Juan Pechiar, MIEEE
Instituto de Ingeniería Eléctrica
Universidad de la República
Montevideo, Uruguay.
pechiar @ fing . edu . uy

Federico Núñez and Alberto Canabal
R&D Department
Ikatu
Montevideo, Uruguay.
fna,alberto @ ikatu . com

***Abstract*—A key requirement for a smart grid deployment is a robust and secure communications network to access all components. We present a platform that provides easy connectivity to smart grid and automation components from the Internet with security and availability as the main objectives. The system is already in production for home automation remote access.\**

***Index Terms*--Network security; remote access; home automation; information privacy; end-user; web-based user interface; Internet.**

I. Introduction

Communications is a key ingredient of smart grid technologies. There are communications at all levels, from telemetry inside the grid infrastructure, all the way to smart metering and control of appliances at the end user premises.

This work focuses on the end user and the interaction between the end user with the smart grid and home automation components in his home.

Typically, an end user residence within a smart grid will contain a combination of the following components:

- Smart meter, that provides comprehensive power consumption and generation reports to the energy supplier.
- Loads that can be controlled, such as lighting, shades, water boilers, heating and air conditioning, high wattage appliances such as washing machines or irrigation. Some of these loads may provide control interfaces such as lighting or HVAC controllers; other loads may be switched on or off by means of a controllable power outlets.
- Local power generation from wind or solar panels.
- Local energy storage (such as plug-in electric vehicles), which are both a load during charging, but may provide energy back to the system if needed.
- Power sensors on different circuits in order to provide the user detailed feedback and statistics on power usage.
- A controller, which integrates with all loads, sensors, keypads, and other devices and user interfaces. The controller has a programmable environment in which automation scenarios and conditions are defined.

The interconnection of all these components requires special care in order to provide both the features requested by the users, and at the same time, network security and privacy.

A. *User interface functionality and technologies*

The end users need a user interface (UI) in order to monitor energy usage and cost; to configure and control automatic energy policies; and to directly control selected appliances and loads.

The UI must be able to provide real-time monitoring of the main loads in the house, as well as information provided by the energy supplier on current grid conditions and incentives to use or save energy.

Automatic energy policies supported by the home controller may be based on several factors such as the time of day (with associated energy tariffs), grid conditions provided in real time by the energy supplier, house occupancy, availability of local energy (e.g. from residential solar panels, or energy stored in a parked electric vehicle or otherwise).

In any case, the user needs to be able to override any preconfigured load scenario if needed ("I need to use the washing machine now, no matter the energy tariff").

Several technologies exist for providing such UIs. Dedicated keypads or touch panels have been used extensively, but are either

This work was funded in part by ANII (Agencia Nacional de Investigación e Innovación).

978-1-4673-6605-2/15/$31.00 ©2015 IEEE

expensive or provide limited functionality. Moreover, users nowadays expect user interfaces to be available on general purpose computers, most notably smartphones and tablet or laptop computers.

Mobility is also a requirement; not only inside the home via wireless LAN, but also from anywhere on the Internet. The user should be able to access the UI no matter if his smartphone is within reach of the WiFi network, or connected directly to the Internet over the cellular data service.

B. Security risks for the end user

The connected nature of smart grid components carries security risks and concerns at all levels.

The home automation controller needs to be accessible from the Internet in order to provide a monitoring and control UI, and to interact with the smart grid operator.

Traditionally, controllers provide connectivity inside the LAN, with very basic or inexistent authentication and encryption. This is already a security risk given the relative ease of gaining access to a residential LAN via WiFi or otherwise.

With the requirement of connectivity from outside the house, usual solutions involve setting up port forwarding at the home router, thus risking attacks not only from neighbors, but from all over the world.

There are recommendations and warnings by security consultants [1] [2] and media [3], and organizations [4] are trying to expose the risks of deploying Internet connected devices without a comprehensive security evaluation.

II. Security risk case study

In order to illustrate the kind of security risks users can be exposed to, our team analyzed the ease of attack from the Internet to users of one of the main home automation controller brands. We call this brand "BrandX".

BrandX provides high-end home automation controllers, widely used in residences and office buildings. The controllers provide LAN connectivity for several purposes depending on the controler model: programming interface, debugging and diagnostic terminal, control and status interface for mobile applications and custom touch screens, and web-based user interfaces.

In order to provide remote access for installers and users, BrandX provides a dynamic DNS service. This way, the home's public IP address can be found out via a DNS query to some subdomain name such as "userID.brandx.com".

The installer must set up port forwarding in the home router so that incoming connections to a specific port are redirected to the controller.

A dictionary attack based on a database of common surnames immediately revealed 800 active subdomains under "brandx.com". This means that the IP address of 800 BrandX installations was readily available. Note that from the IP address it is also possible to obtain an approximate geographical location.

When trying to connect to the control interface of the controllers, one may authenticate giving a username / password combination. If this combination fails, the connection is closed.

However, we found out that if no authentication was carried out, the controller accepted and executed all commands.

So at this moment we have access, from the Internet, to many installations without the need for encryption or even authentication.

Moreover, in several cases, the controllers didn't even have authentication set up for the programming interface, revealing a very careless installation.

So, depending on the access options configured on the controller, and the laxity of home router setup, immediate access was available to the control interface, the configuration and diagnostics console, or even the whole home network.

Similar problems have been observed on other controller brands. In one case, the controller has an open SSH server, and the username and password are widely available, thus allowing immediate access to the Linux OS running inside the controller. Another case also left an SSH server running, but with an undisclosed username / password, for as long as an undisclosure can last.

In summary, our analysis revealed:

- Very bad system design: a dictionary attack provided direct access, with no encryption and no authentication, to the home automation control connection.
- Careless home network configuration: home routers set up to allow more incoming connection ports than necessary, exposing more services to the Internet.

- Controller misconfiguration: many controllers had no user/password authentication set up for the programming interface.

Similar precautions have been reported elsewhere [5].

III. Khimo system architecture

Our team has developed a system, called Khimo, which provides remote access to home automation and smart grid controllers inside the home.

The design objectives are:

- Security: as explained earlier, it is very common for Internet connected devices to be designed without network security in mind.
- Uniformity: the user interfaces should be independent of the actual devices being controlled and provide a comprehensive and intuitive control panel to the user. The UI must not be tied to a particular platform or screen size.
- Sold as a service: the business model behind Khimo is to provide secure remote access as a service.

The typical application for Khimo is for monitoring and controlling home and building automation and smart grid components. Alternative uses include infrastructure monitoring, like data centers or backup power stations.

Khimo has 3 components: a controller module, a cloud-based gateway server, and user interfaces (figure 1).

A. Controller module

The controller module is in charge of connecting to the cloud server, and interchanging all commands and status information. This connection is outgoing from the controller module to the server; this way, no configuration is necessary on the home router, and there is no need to set up dynamic DNS or port forwarding into the home LAN.

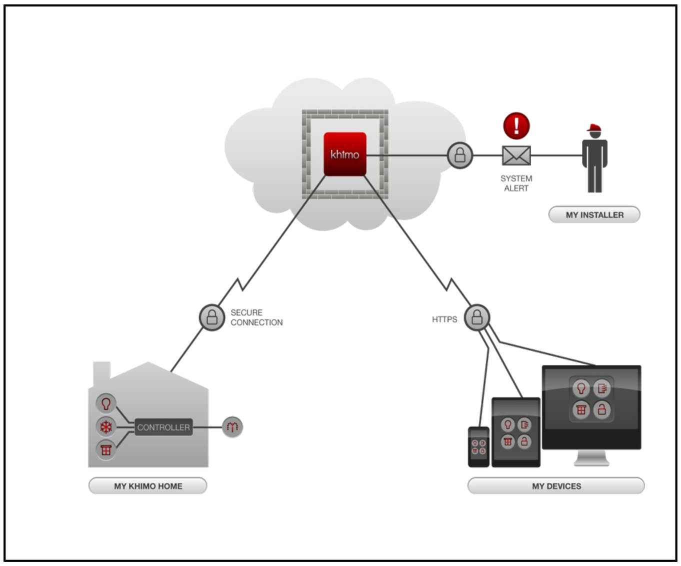

Figure 1. *Khimo architecture: controller with khimo module; khimo cloud server; user interfaces; and contractor / installer access.*

Each message on this connection is digitally signed and both ends of the connection are mutually authenticated, to avoid man-in-the-middle attacks. The connection is encrypted using HTTPS only if the controller platform supports it.

Note that most controller architectures are very limited in resources, and offer a very constrained programming environment, so implementing complex encryption frameworks is usually impossible. Thus, the only requirement for our protocol is access to TCP socket connections, and basic arithmetic and bitwise operations.

The controller module exists in 2 versions: a software module to include in home controllers, and a standalone hardware module for installations without a home controller.

The software module version is a plug-in module which currently is available for Crestron, Control4, and Bang & Olufsen Beolink Gateway controllers. The installer only needs to define which variables and signals in the current controller programming will be exposed on Khimo.

The standalone version is under development, and is intended for installations lacking a main home automation controller. The standalone Khimo controller can act as a home gateway, and connect directly to energy sensors, actuators and other home automation devices.

B. *User interface*

Users and installers use Khimo via a web-based user interface. This means that the system can be used on any platform without the need to install applications or executables. The web interface is adaptable and works on any screen size (figures 2 and 3). The main use of the user interface is from smartphones and tablet computers.

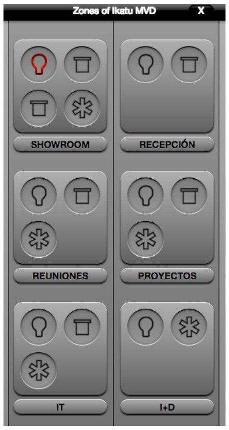

Figure 2. *Khimo user interface showing system overview.*

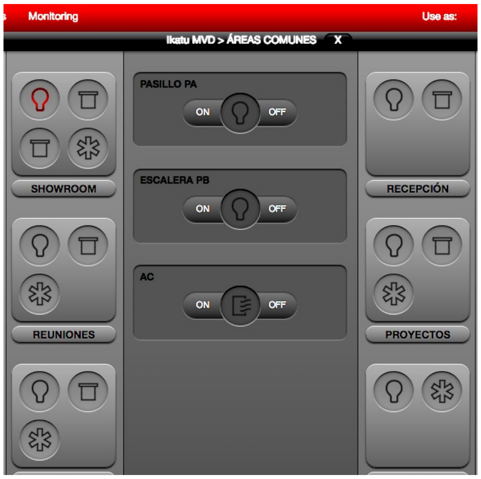

Figure 3. *Khimo user interface showing zone details*

The installer has access to a configuration user interface, where he can customize how the different signals will be grouped and displayed to the user.

C. *Khimo server*

The server component receives all connections and is in charge of authentication, user and installation management, and translating all signals, status and notifications from the controllers into the user interface for the users and installers.

IV. Security considerations

The connection and authentication for the user interfaces is based on industry standard HTTPS, for which the usual best practices for web-based applications are followed.

Therefore there remain two main areas to protect from attacks:

1. The communication between the controller and Khimo cloud gateway.
2. The cloud gateway itself.

A. *Securing controller to gateway communications*

The controller to Khimo communications, as already stated, cannot use sophisticated security frameworks due to the lack of resources and programming flexibility on the controller.

The attacks to avoid at this level are:

1. Impersonation of Khimo gateway by an attacker.
2. Impersonation of the controller by an attacker.
3. Man in the middle attacks where messages can be injected, removed, modified or replayed.

Mutual authenticity is based on a pre-shared secret key for each controller, which is generated or renewed during installation. This key never traverses the TCP connection, and is used to generate session keys upon each reconnection between controller and gateway service. The generation of the session key follows the methodology used for WiFi Protected Access (WPA), which involves random numbers generated on both ends (to avoid key reuse by an attacker), and extensive hash operations so that the pre-shared key cannot be deduced from gathering session initiation traffic.

All application messages in each direction are digitally signed, and if a signature verification fails for any message, the connection is dropped.

The message digest algorithm used for signing messages is based on FIPS PUB 198 standard: *The Keyed-Hash Message Authentication Code (HMAC)*. The message signing algorithm uses the message data, the session key, plus the previous signature. This way, message deletion or insertion will break the signature chain and cause a connection drop.

B. Securing the gateway server

The cloud-based gateway contains a database of all user information and configuration, and must contain all the pre-shared secrets for all controllers, plus authentication credentials for end users connecting from their mobile devices or web browsers. The server attends all controller connections, and generates all web interfaces and notifications for connected end users. Also, the installer interfaces are managed here, including the pre-shared key generation for new installations.

First, all the standard industry practices for an Internet-exposed service have been considered (network firewalling and protection, jailing of exposed processes, storage of end-user passwords only in hashed form, etc.).

The most sensitive information is the list of pre-shared keys for each controller. This list is *never stored on disk or in any database in the server*. Rather, there is a separate process that keeps the list in RAM and runs in a protected environment. This process will never expose a pre-shared key except when generating a new key for an installation. This key-keeping process then executes signature generation and verification for messages, but no longer exposes keys.

The keys reside elsewhere on a dedicated server, to which only the key-keeping process can authenticate, and which will transmit the key-list heavily encrypted.

So even in the case an attacker gains access to the gateway server, and gets hold of the whole database, there is no useful information on how to actually gain communication with the controller.

The design of the whole system is built around ensuring that an attacker won't be able to gain control to the user installation.

V. CONCLUSION

There is a historic lag with many electronic control and sensing devices in how they adapt to a more connected world. Once the sole domain of electrical engineering and electrical installers, these devices became programmable, then interconnected via dedicated buses, then interconnected via LAN, then included complex user interfaces, and now are getting accessible on the Internet and controllable via mobile devices.

This evolution requires new skills (both for development and installation) which are difficult to adopt. Products must go into market with all the new features built onto legacy technologies, and installed by people lacking the knowledge in telecommunications, user experience and network security needed nowadays.

The result is, in many cases, products and installations which are inherently insecure [6].

We presented a system to integrate all actors and devices designed from the top down, with all the above problems addressed from the beginning. Khimo has been on the market for more than 2 years now.

It is very important that all security algorithms and protocols are based on standard and tested industry standards. The temptation to innovate in this area can easily backfire [7].

The development of Khimo has been funded in part by ANII (Uruguayan innovation and research agency), and all protocols and procedures were reviewed and audited by an independent security consultant.

REFERENCES

[1] How to keep your smart home safe - F-Secure - https:// www . f-secure . com/weblog/archives/00002792.html accessed 2015-03-11

[2] [2] Open Web Application Security Project, “Internet of Things Top 10 for 2014”, URL:htt ps:// www . owasp . org/index.php /OWASP_ Internet_of_Things_ Top_Ten_Project #tab=OWASP_Internet_of_Things_ Top_10_for_2014 accessed 2015-03-11

[3] Search engine for Internet-connected devices - https:// www . shodan . io/

[4] Kashmir Hill - “When 'Smart Homes' Get Hacked: I Haunted A Complete Stranger's House Via The Internet” - URL: http:// www . forbes . com/sites/kashmirhill/2013/07/26/smart-homes-hack/ accessed 2015-03-11

[5] Veracode white paper: “The Internet of Things Poses Cybersecurity Risk” - URL: https:// info . veracode . com/whitepaper-the-internet-of-things-poses-cybersecurity-risk.html accessed 2015-03-11

[6] CNET: “Fridge caught sending spam emails” - URL: http:// www . cnet . com/news/fridge-caught-sending-spam-emails-in-botnet-attack/ accessed 2015-03-11

[7] Philipp Jovanovic and Samuel Neves: “Dumb Crypto in Smart Grids: Practical Cryptanalysis of the Open Smart Grid Protocol” - Cryptology ePrint Archive, Report 2015/428.