Panasonic CST
CLOUD SERVICE TOOLKIT

Advantages of Panasonic's Enterprise IoT Platform over HTTP/REST in IoT Solutions

Panasonic Software and Analytics Solutions (PSAS) June 26, 2017

Overview

The Internet of Things (IoT) presents a unique set of challenges for system architects and software professionals. While HTTP/REST provides an attractive and familiar solution for many use cases, it falls short in many IoT scenarios. Inappropriately applying technology built for the web to IoT challenges will not unlock the full potential and benefits of IoT in the enterprise.

Panasonic North America's enterprise IoT platform, the Cloud Service Toolkit (CST), provides a ready solution for a wide variety of enterprise IoT use cases with software specifically designed for IoT devices, IoT gateways, and IoT clouds at unlimited scale. The CST deeply integrates industrial grade security including authentication, encryption and access control. It automatically enforces secure relationships between users and their IoT devices, and facilitates secure peer-to-peer communication between devices. The design is optimized for IoT, including support for inexpensive limited capability devices without compromising security. It is based on established standards and research and over a decade of experience within Panasonic R&D. It has been vetted both inside and outside of Panasonic and used in products around the world from securely monitoring solar farms across North America, managing fuel cells in Japan, and controlling campus-wide university air conditioning systems in Europe and Malaysia.

The CST provides several advantages over using HTTP/REST for IoT systems. These include device discovery, secure peer-to-peer IoT device communication, support for very small embedded devices, bandwidth savings, command and control for IoT devices including firewall traversal, device and user management, strong data model, and support for IoT gateways.

Device Discovery

The CST supports automated device discovery while HTTP/REST has no discovery capability. With device discovery, IoT systems become much more flexible and require much less configuration. For example, a switch can discover the lights it can control; devices can discover gateways; apps can discover all the IoT devices and services on the network; sensors can discover where to send their data (or vice-versa, services can discover sensors they are interested in).

Secure Peer-to-Peer Device Communication

IoT devices should talk to each other not just to the cloud**.** The CST is designed for peer-to-peer communication of IoT devices while HTTP/REST is designed as a client/server model. With the CST, the cloud becomes just another peer on the network. Intelligent routing rules keep local traffic local, without needing to go to the Internet. This keeps things working even when the Internet connection goes down, improving reliability while reducing latency. With an Internet connection available, device data can be easily collected in the cloud for analysis, and Internet services become available to devices.

With peer-to-peer communication, one device can directly control and/or notify other devices, which is key to the success of IoT. For example, sensors can talk to lights; train seats can notify trains of their availability; roads can talk to cars about ice or other conditions; water leakage systems can talk to shut-off valves; products can talk to shelves for logistics; parking spaces can talk to parking lots to show available spaces. The list goes on and on.

Unacceptable time delays can plague IoT systems without peer-to-peer capabilities.

Support for Very Small Embedded Devices

The CST supports embedded devices with low compute capability and as little as 64k memory. HTTP/REST requires much more memory, particularly if security is needed. And unlike HTTP/REST, all data is transmitted in its binary format so devices do not need to parse documents.

Authentication and key exchange is optimized for low capability IoT devices without comprising security. This results in a more appropriate security design as opposed to attempting to take security standards built for the web and apply them to IoT. It is based on established standards and research: IBM's KryptoKnight (a follow-up to Kerberos) and SRP.

Bandwidth Savings

The CST uses only about a third of the bandwidth of most HTTP/REST API's. Panasonic R&D spent two years optimizing to reduce network bandwidth and to support resource constrained devices on the network. The result is a significant bandwidth savings over REST and other text based protocols.

CST	HTTP/REST
• Optimized IoT binary format	• Text based web format, not optimized
• Integrated IoT security and authentication	• Transport Layer Security
• Metadata never retransmitted (unless requested)	• Metadata typically retransmitted with every request and response

We did a comparison of bandwidth required for a few requests using popular REST APIs versus the CST. In the examined cases, the CST results in a bandwidth savings of over 60% over REST APIs. And for mass data collection to the cloud, CST's data streaming service offers typical bandwidth savings of over 90%.

API	Bytes - REST API	Bytes - CST	Bandwidth Savings
Nest	6062	2241	63%
GitHub	6163	2506	60%

Command and Control and Firewall Traversal

The CST supports firewall traversal to allow commands to be sent from the cloud to IoT devices in homes, stores, factories, etc. without the commands being blocked by routers or firewalls. It also supports peer-to-peer device communication through the cloud. These scenarios are difficult or impossible with HTTP/REST which is designed for client/server communication, typically with the server in the cloud.

With HTTP/REST, it is simple to send commands to the cloud and hard to send commands to devices. **With the CST, it is easy to send commands anywhere:** from IoT devices to the cloud, from cloud services to IoT devices, or directly from one IoT device to another.

Device and User Management

Managing IoT devices and their users is a challenge unique to IoT. IoT devices must be provisioned for the systems in which they are deployed. This involves key management and setting up device and user permissions. The CST provides ready-made services to readily handle these challenges. HTTP/REST systems deal well with users and servers. And while a device can be treated as a user, it's not optimal, and doesn't work at all for peer-to-peer communications. This can be solved by issuing certificates to devices, but this is complicated, particularly at the scale of IoT. The CST, a solution built specifically for IoT, works much more smoothly.

Strong Data Model

A strong data model is right for IoT. The IoT is different from the web in that it is built into things that may be difficult to update frequently (sensors, buildings, cars, wearables, factory equipment), so the definition of data and commands must be stable for long periods of time to continue to provide compatibility between devices and IoT systems. A strong data model ensures backwards compatibility and system stability.

The CST uses a strong data model, which forms a contract between devices for how to communicate with and control each other. HTTP/REST systems typically use JSON or something similar, which has a weak data model—it can be changed at any time. While this is manageable with well-known versioning methods, it is up to the software professional to ensure backwards compatibility and stability. This is managed for you and enforced with the CST.

Support for IoT Gateways

The CST includes a ready-made IoT gateway. To manage the massive scale of IoT and to provide local device connectivity, most enterprise solutions will include the use of IoT gateways between end devices and the cloud. With the CST, an out-of-the-box IoT gateway solution is available which automatically enforces security and permissions and routes IoT commands and data where they should go—to other IoT devices or to the cloud. It also automatically works in conjunction with other gateways to provide infinite scale. These IoT gateways must be built from scratch for HTTP/REST systems.

Conclusions

The CST, an enterprise solutions purpose-built for IoT, is a better choice than HTTP/REST for most IoT use cases. It makes sense to use the right tool for the job. While web-based solutions like HTTP/REST are attractive due to their ubiquity and familiarity to software developers, they are not an ideal fit for IoT since IoT presents its own unique set of challenges. These include device discovery, secure peer-to-peer IoT device communication, support for very small embedded devices, bandwidth concerns, command and control for IoT devices including firewall traversal, device and user management, stable data models, and support for IoT gateways. The CST is a proven enterprise grade solution focused specifically on these challenges to unlock the full potential of the IoT to transform businesses by enabling new efficiencies, discovering insights, predicting outcomes, and offering new products and services.

Learn more at cstkit.com.