WHITE PAPER

Securing Your IoT Solution Stack

A Proven, End-to-End Approach

sas
THE POWER TO KNOW.

Contents

IoT Is Everywhere – and Transforming Our World

The Internet of Things (IoT) creates tremendous opportunities and benefits for businesses like yours – from lower costs and higher efficiencies to innovations that are transforming customer experiences, service levels and value propositions. Its use is permeating everything at the core business of companies today. For example:

- A US municipality has implemented smart meter monitoring for all the town's residential and commercial water meters. The project involved placing water meter sensors on 66,000 devices that used to be manually read and recorded. Now management has real-time insights for faster, better decision making and reduced operational costs and can develop new, data-driven services for customers.
- A US oil and gas company is using IoT to optimize oil field production. In this case, the company is using sensors to measure oil extraction rates, temperatures, well pressure and other aspects for 21,000 wells. Now management can sense and respond to maintenance issues, take informed steps to optimize well performance and more.
- An international truck manufacturer recently created a new revenue stream by outfitting trucks with sensors for predictive maintenance. The system automatically schedules repairs when needed and orders the required parts for the repair. More than 100,000 trucks have been outfitted with devices that transmit more than 10,000 data points a day for each truck.

As these examples illustrate, IoT scenarios have expanded to link an ever-widening network of connected devices – cars, wells, trucks, refrigerators, machinery on shop floors and more.

Growing the Value – and Attack Surface – of IoT Scenarios

As these examples illustrate, the upshot of having connectivity across millions of devices is that businesses have access to new sources of data and compute environments, which they can use to support increasingly powerful and innovative IoT scenarios. The downside is that the expanded IT and OT infrastructure needed to support this connectivity and diverse communications also increases complexity and expands the potential attack surface. This not only leads to new and unforeseen vulnerabilities, but also makes data and analytics harder to secure.

Put simply, as IoT is becoming more important to business, security is harder to achieve.

Implications for SAS Customers

What does this mean for your IoT landscape and plans to use IoT and SAS® solutions for digital transformation? As a leading provider of analytics solutions used in many IoT scenarios, SAS has a vested interest in ensuring that our customers understand how to secure their IoT landscapes. Increasingly, SAS solutions are being run in the data center and cloud all the way out to the edge, where connected "things" live and generate data – lots of data.

We are committed to helping our customers fully harness the power of the IoT to drive innovation – but in a way that is safe and secure for them and their customers. To that end, this paper will explain security requirements for IoT solutions pushing analytic processing out to the edge, as well as how SAS secures the areas of IoT solutions that it touches.

Security best practice: Gateway as a front end to devices

One of the core security best practices focuses on managing the scope of connected devices. Not every device that performs some function needs to be connected to the external world. It is a great idea to shield sensors and other devices using an IoT gateway.

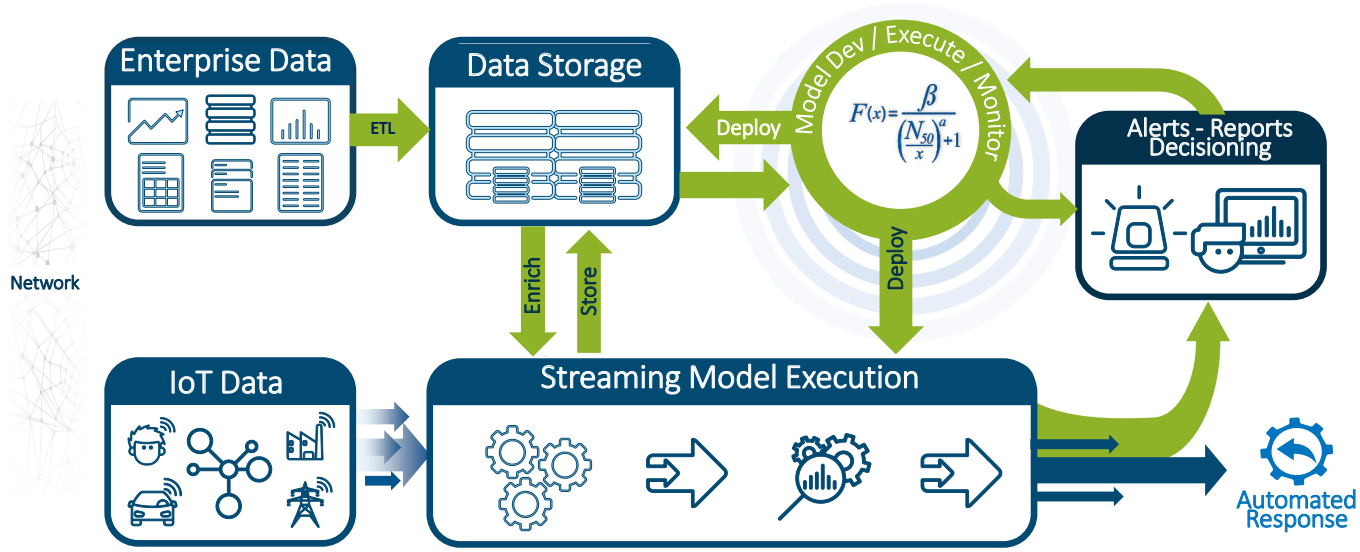

Figure 1: The IoT analytics life cycle

Understanding the Complexities of IoT Security

To understand why IoT security is so complex, you first need to understand the IoT analytics life cycle, illustrated in Figure 1.

The network edge means different things in different industries and organizations; it could refer to wells in oil and gas, factory machinery in manufacturing, trucks and drills in mining, home thermostats in energy, and so on. Devices and assets at the edge can stream massive volumes of high-frequency data – so much that it doesn't make sense to flow it back to a central data center or cloud solution for analysis.

Companies can distribute SAS Analytics throughout the network, all the way to the edge – for example, directly on a customer's industrial asset or a locomotive to monitor performance in real time, detect patterns indicating a pending failure or maintenance requirement, and automate the optimal response, such as scheduling maintenance. These analytics are more localized, as they are only applied to data streaming from one or several specific end points.

Analysis at the edge allows companies to truly operate in real time, detecting and responding proactively to issues that could hurt asset uptime, delay manufacturing or affect customers depending on timely deliveries. For example, SAS is currently working with railroad customers who are deploying SAS Analytics to analyze data streaming from their locomotives. The insights gained can be used to support use cases for improving efficiency, enhancing the operator experience and more. (http://www.ioti.com/analytics/ge-transportation-and-sas-team-iot-edge-computing-trains).

As shown in Figure 1, after data is processed locally, it needs to flow securely back to the data center or cloud for analysis that provides a global context. This analysis can support a cross-fleet analysis and trigger automatic responses to improve asset

performance, for example. Eventually, this streaming data must be consolidated with existing enterprise data and processed for more comprehensive, global insights. This analysis goes beyond providing traditional decision support to deliver next-generation analytical models, which can be deployed across various layers of the network.

What's the result? A feedback loop – in other words, the analytical life cycle for IoT.

While security threat models for the data center and cloud are mature – and thus able to support a robust security mechanism for data access, transport, storage and event processing – the models for securing from the data center to the edge are less established and far more complex. This is due largely to the incredible diversity of devices, protocols, data formats and operating environments involved. In the following sections, we will discuss this complexity and how to address it from a security perspective.

If securing a distributed IoT landscape is so complex, is it worth the effort?

The answer is a resounding yes – because security is only as strong as its weakest link. To illustrate, consider the US electric grid. Security breaches could have devastating effects, including long-term power outages. Grid operators have instrumented the grid with sensors to have real-time information about network events – and in recent years, connected billions of new devices attached to homes to create smart networks and cities. Individually, these devices may not represent a big risk, but because they are connected to the broader network, hackers could use just one sensor to access and bring down the entire network.

Securing the End-to-End IoT Landscape

Now that you understand what you're dealing with, let's delve into the technical realities of securing the entire IoT analytics life cycle.

IoT solution implementations are enabled by a combination of technologies, including:

- Sensors.
- Devices with sensors deployed on them (such as industrial machines).
- IoT gateways.
- Network.
- An on-site or cloud-based data center platform that supports a device management layer.
- A persistent data environment.
- Applications such as analytics software.

As we'll see, securing this stack is a collaborative effort that involves several players, each responsible for providing or managing one or more layers of the infrastructure and ensuring proper security (see Figure 2).

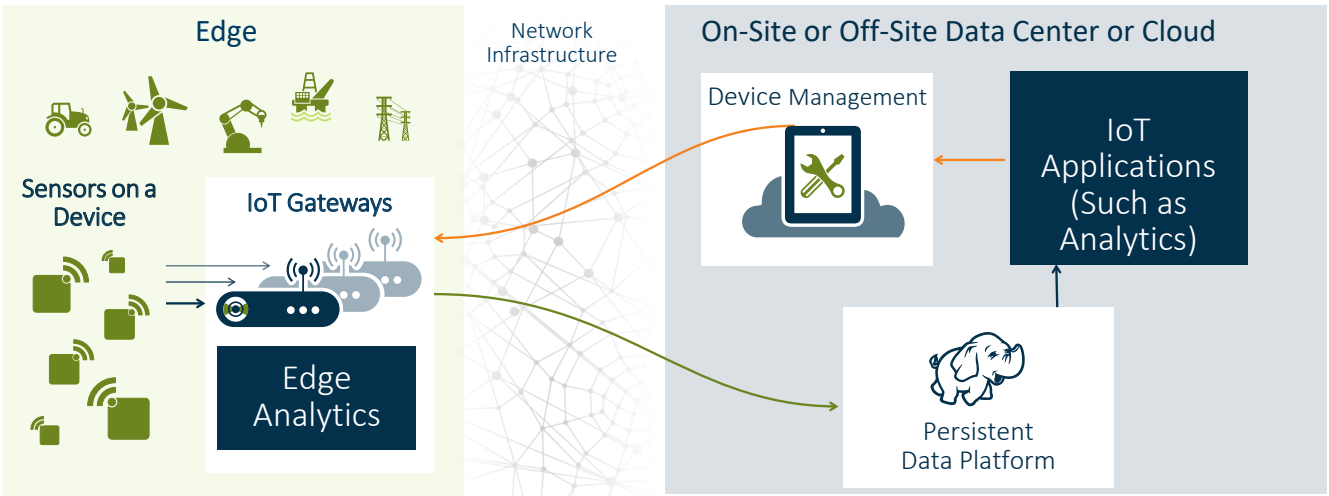

Figure 2: Layers of an IoT architecture

Security by Functional Layer

To illustrate a typical IoT stack at work, we're using a robust security stack supported by Intel (a SAS strategic IoT partner) that combines many security features that SAS IoT analytics solutions rely upon.

For our purposes, let's assume this stack is being used to support connected railroad operations, which can include the connected locomotives, security cameras at platforms, and sensors on tracks. In this scenario, the company is running SAS Analytics on both the data center and at the edge of the network to capture streaming data from these assets – for example, to detect potential failures, support predictive maintenance and more.

The Physical Layer at the Edge

While most of the potential attacks to the IoT stack will be focused on the enabling software, threats can also occur at the physical level. For a railroad, this would mean ensuring physical security on two levels:

- Securing the device or asset (in this case, a security camera on platforms) in a locked area with restricted access. This is a straightforward solution to avoid a number of attacks caused by physical intrusion and is highly recommended where possible. Adding an optional surveillance capability (such as video) provides an additional layer of hardening. This is a must-have requirement for devices part of a network supporting any mission-critical functionality.
- If securing a device in a locked area is not feasible, the next-best thing is to lock or cover any open physical ports (such as open USB ports, removable media, etc.) to avoid an intruder connecting via a physical device (like a USB) to compromise the device. IoT devices performing a critical service should be built with anti-tamper mechanisms to prevent compromise.

SAS is agnostic of the choice of the foundational layers that are involved in completing the IoT stack and works with a number of leading IoT partners to support a holistic architecture.

The Hardware and the OS Layer (for Devices or Assets at the Edge)

Hardware and the operating system is typically secured by the hardware manufacturer (in our example, Intel) and operating system vendors. Hardware is a particularly critical layer to secure due to the unique isolations it can provide for the entire IoT device stack. As we'll see, it's important to be able to isolate, or quarantine, a compromised asset so threats don't spread.

To provide solid security at this layer, the manufacturer must provide key features and capabilities at the hardware and OS level. These features and capabilities, which enable the critical "root of trust" for a secure IoT implementation, include:

- **Protected boot**: Protected boot ensures that the firmware and software used during boot of the platform, up to and including the kernel, are cryptographically verified and or measured against known, expected values (based on what was intended).
- **Protected data and keys**: Protected data and keys ensure that sensitive data, keys or credentials at rest or in transport are encrypted and stored to prevent misuse or disclosure.
- **Device and software identities**: Device identity can be used as a root of trust to attest to the validity of platform components, especially during initial device provisioning in the field. Software identities are layered onto the device by different ecosystem roles that need to authenticate the device against access control systems during the life cycle of the device.
- **Trusted execution environment (TEE)**: TEE supports the creation of isolated enclaves that protect sensitive data, code, I/O processes or keys at runtime to create a trusted application environment; in this way, they help protect against interference by other applications or malware on the platform. TEEs can be implemented different ways – for example, the most basic way is to use the operating system (OS) to provide process-level isolation. Containerization is another way to isolate applications from one another; however, its popularity has to do with the ease of transporting applications. Finally, virtualization can be used to separate OSs and support better security. However, it is more expensive to run compared to other options.

The Network Layer

Because IoT is all about connectivity, securing the network is critical to securing the IoT stack. Securing this layer is typically handled by the network provider.

Due to the wide range of communication protocols involved, securing the IoT network layer is more challenging than a traditional network, which can typically be locked down using endpoint security features such as antivirus and anti-malware. But in an IoT scenario, these technologies are no longer sufficient. Companies need to invest in firewalls and intrusion prevention and detection systems. In addition, network providers need to create zones that isolate IoT devices from the rest of the network. This allows IT to quarantine potentially compromised devices. In addition, IoT gateways enable network providers to overlay an IoT security proxy on top. This overlay can normalize and provide modern security communications for a group of devices, from edge to cloud.

Security best practice: Better data management at the edge

Moving large volumes of data over a large network is fraught with risk – even with the best possible security in place. Luckily, intelligent filtering techniques can be used to filter out redundant data, which can reduce volumes by more than 95 percent. This greatly reduces the cost of transferring data over the network to the data center or cloud for further analysis. In addition, it isolates the movement of a large chunk of data in a localized environment and avoids the transfer over wide-area networks, reducing the overall footprint of potential vulnerability.

The Device Management Layer

As a rule, an IoT device that cannot be updated is not secure. Supporting a full life cycle management, from onboarding a new device to decommissioning, it is a basic security requirement handled at the device management layer. This is typically secured by the overarching IoT software platform provider.

At all times, the provider must have the ability to send updates to IoT devices to keep up with the potential threats. To avoid man-in-the-middle attacks, the management layer itself must support encrypted channels between the platform and the device.

The Application Software Layer

SAS Analytics – along with any software from other providers – runs in this layer. These applications are ultimately what drive specific IoT use cases and what users interact with. To secure this layer, functionality supporting authentication, authorization, encryption and policy management must be provided for each IoT app. We will delve more deeply into how SAS meets these requirements later in this paper.

Securing Hardware OS and Management Layers

For the purposes of this discussion, the hardware manufacturer (in our example, Intel) plays a vital role in securing key aspects of this stack: the hardware, OS and management layers. Let's take a closer look at how Intel typically secures these layers.

Silicon Hardware Root of Trust

The broad selection of silicon that Intel offers ensures the highest level of security with capabilities like Unified Extensible Firmware Interface (UEFI) and Intel Platform Protection Technology, complete with Boot Guard that supports protected boot. Intel Platform Trust Technology provides a firmware Trusted Platform Module (TPM) implementation on its chip that supports protected storage for credentials or platform configuration registry values, which are used for remote device attestation.

Operating Systems

Intel's portfolio consists of several operating systems suited for IoT implementations, ranging from VxWorks (the leading real-time OS) to Wind River Linux, the yocto-based OS tailor-made for IoT devices such as gateways. Each of these OSs support the core security features listed earlier by supporting:

- **Identification and device onboarding**: Intel has supported Enhanced Privacy ID (EPID) as its IoT identity solution for several years. EPID meets the basic need to establish the hardware root of trust, which attests that a given device is a member of a valid group while preserving privacy and anonymity for a device. Additionally, Intel Secure Device Onboard is a service that can use EPID identities to securely provision devices to their IoT device management platforms. Software identification is enabled several ways – most commonly, with TPM (discussed previously).
- **Trusted execution environments (TEE)**: There are many solutions that support TEEs. In Intel's case, the Intel Trusted Platform Module supports TEE, and Intel Software guard extensions (or SGXs) provide TEE environments that support protected areas of execution in memory.

Security best practice: Use "learning"-based security solutions

Even when sophisticated security and prevention techniques are in place, networks can still be compromised. So during IoT implementations, companies must take an analytical approach to security that enables continuous detection and learning that allows the software to catch unanticipated security threats without being explicitly programmed to do so.

- **Management**: Using the Wind River Helix Device Cloud, which is available as a scalable, on-site or cloud-based solution, Intel has streamlined the task of managing IoT devices. Effectively managing these devices is extremely important to enabling IoT scenarios that include powerful software such as SAS IoT analytics.

Securing the Application Software Layer

SAS analytic software – and any other application software enabling the IoT scenario – runs on the application software layer of the stack. SAS is responsible for securing its own software. To this end, SAS provides a robust analytical platform that runs at the data center and at the edge: SAS Event Stream Processing. SAS Event Stream Processing is unique in the market because it can analyze data in motion as it flows through the IoT analytic life cycle. There's no need to capture and store data before analyzing it, which opens up a whole new world of real-time analytic scenarios. For example, it can enable low-latency responses that are highly desirable in use cases involving performance degradation and safety. And as a high-performance solution, it supports a flexible deployment anywhere on the network.

SAS Event Stream Processing (visualized in Figure 3) was developed from the ground up within the SAS R&D organization using the latest standards and best practices, ensuring security from both "inside out" and "outside in" perspectives. And its application security has been rigorously tested using static and dynamic techniques against industry standards (such as OWASP and SANS/Mitre) that look for common weaknesses and exposures.

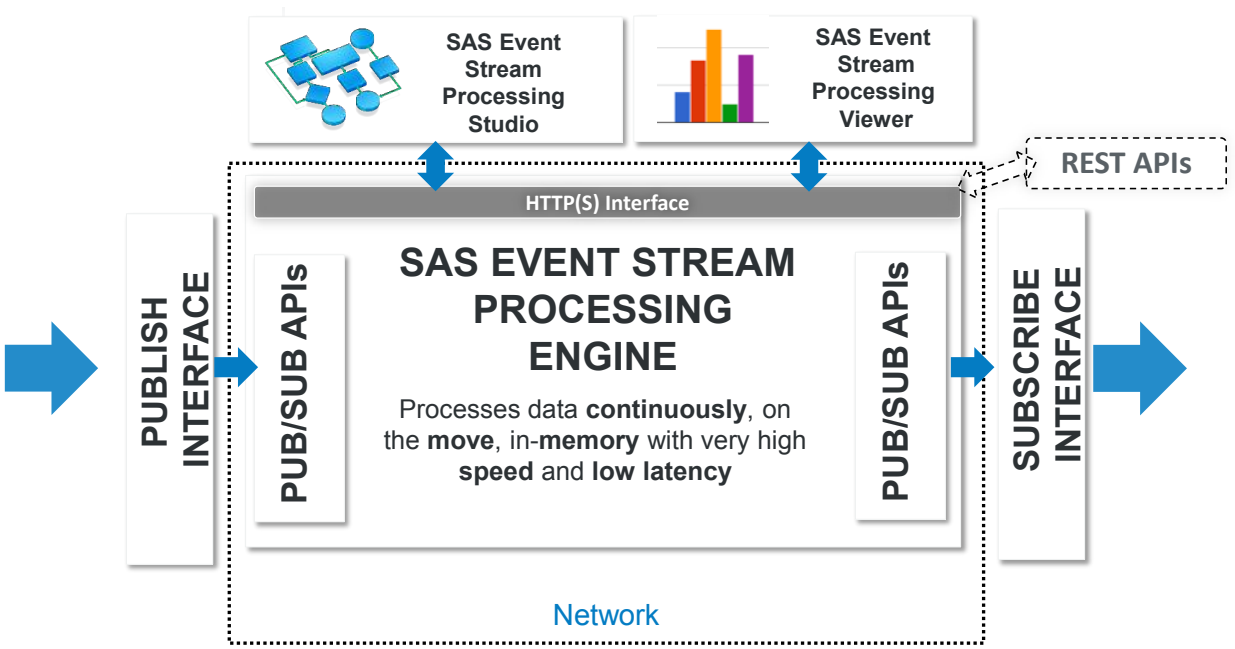

Figure 3: SAS Event Stream Processing delivers impressive security capabilities.

Before reviewing the security capabilities of SAS Event Stream Processing (encryption, authentication and user access control), let's look at the SAS Event Stream Processing components involved in the IoT analytical life cycle and what they do, including:

- **SAS Event Stream Processing engine**: This execution environment can be deployed on any compute environment to support streaming analytics. It includes small compute footprint edge devices (such as IoT gateways) and layers in between the edge and data center (such as fog).
- **SAS Event Stream Processing REST interface**: This REST interface, which is exposed by the execution environment, can support multiple tasks such as viewing the state of the streaming data, performing operations against the execution environment and more.
- **SAS Event Stream Processing Connectors and Adaptors and Pub/Sub APIs**: This mechanism is used to publish data into the SAS Event Stream Processing engine and to subscribe for processed data by consumer applications. Pub/Sub APIs are available for C, Java and Python. Many adaptors are available to support publish to – and subscribe to and from – multiple common data sources, protocols and message buses.
- **SAS Event Stream Processing Studio**: This graphical user interface environment is used to build and test "connected graph"-based IoT analytical data flows that are eventually deployed to the SAS Event Stream Processing engine environments.
- **SAS Event Stream Processing Streamviewer**: This streaming dashboard tool uses the pub/sub API to subscribe for data from the SAS Event Stream Processing engine environment.

Using SAS® Event Stream Processing to Support Encryption, Authentication and Access Control

All of these SAS Event Stream Processing components are used to support the three core tenets of application software security: encryption, authentication and user access control. Table 1 summarizes the core functionality and how it is delivered.

The SAS Technology Office maintains a commitment to maintaining a Software Security Framework that ensures continuous delivery and deployment (CI/CD) of secure software for the IoT analytics framework. As part of that commitment, SAS measures itself against industry software security maturity frameworks like BSIMM and OpenSAMM and improves its processes and products continually.

Table 1.

Security functionality	Scenarios Supported
Encryption: SAS Event Stream Processing secures data being transmitted throughout the IoT architecture using the latest TLSv1.2 encryption.	SAS Event Stream Processing supports SSL/TLS-based encryption on TCP/IP connections in the following transport scenarios: • Connections that are created by a client using the C, Java or Python publish/subscribe API to connect to an event stream processing server. • Connections that are created by an adapter connecting to a SAS Event Stream Processing server. • Connections via REST interface to the SAS Event Stream Processing engine environment. • Connections to SAS Event Stream Processing Studio over HTTP.
Authentication: Given the diversity of users involved (both machine and human), SAS Event Stream Processing authenticates users before granting access to data and functionality using: Auth2 tokens from designated third-party providers. Name and password via the SAS Logon service provided through the SAS® Viya® platform. Kerberos.	SAS Event Stream Processing supports the following: Authentication support for interacting with the SAS Event Stream Processing engine process for submitting definitions (using C++ modeling API or XML definition). • Authentication support for the REST interface to the SAS Event Stream Processing engine environment. • Authentication support for Pub/Sub APIs and Adaptors to the SAS Event Stream Processing engine environment. • Authentication support for SAS Event Stream Processing Studio's interaction with a SAS Event Stream Processing engine environment. These techniques are available to all the scenarios listed above except the SAS Event Stream Processing Studio authentication support, which only supports an OAuth2 based authentication.
User access control: SAS Event Stream Processing provides a more fine-grained user access controls.	SAS Event Stream Processing provides explicit read/write control over the SAS Event Stream Processing engine environment.

SAS® Event Stream Processing Architectural Constructs

SAS Event Stream Processing has also been designed to support flexible deployment models that improve overall security effectiveness. Key enabling architectural constructs include:

- **Process isolation**: SAS Event Stream Processing supports the separation of processes, which can have a significant bearing on security. For example, the pub/sub interface can be supported using in-process classes that publish or subscribe to and from SAS Event Stream Processing windows or with standalone executable files that publish and subscribe – potentially over a network.
- **Support for life cycle management**: SAS Event Stream Processing readily supports software life cycle management on edge devices via integration with centralized device management platforms like the Helix device cloud. The entire SAS Event Stream Processing environment deployed on an edge device can be compressed for a lean footprint. Updates require little bandwidth, and hence assets can be easily updated over the wide area network.

Conclusion

IoT is driving transformational changes across many industries. But to realize its full potential and value of IoT solutions, their underlying security risks must be addressed. These risks are real, as illustrated by a real-world DDOS attack that took down several internet services. And they are a constant concern of the US federal government.

As discussed in this paper, there is a clear path to securing your IoT solutions throughout the entire IoT analytics life cycle. To learn more about how SAS delivers secure, scalable IoT solutions, please visit sas.com/IoTsolutions.

To contact your local SAS office, please visit: sas.com/offices