IoT Poses New Problems for Developers

From security to analytics to testing, developers have their hands full when it comes to the Internet of Things.

A Compass for the New World of IoT

As exciting as the Internet of Things world may be—from the promise of autonomous cars to the egalitarian idea of a robotic butler in every home—there are still some serious technical challenges that organizations dabbling in the IoT must be aware of.

The protocols of interconnectedness are still evolving, and the industry is miles away from a standard. Some devices require a fast and efficient protocol where reliability isn't important. Others prize reliability over speed. The flux state of IoT protocols has created new challenges for ensuring the security of the devices that rely on them. In the first article in this handbook, reporter George Lawton discusses the security issues devices may encounter and how an IoT gateway can be used to plug up some of those holes.

Another challenge organizations are dealing with is how to handle the massive amounts of data an army of interconnected devices generates. In the second article, software developer Swathija Raman discusses how IoT devices are changing the way analytics is done and what types of insights can be mined from the gathered data.

To close, we look at the challenges of testing IoT applications. When software gets deployed on components that can fly and accelerate, testing for safety and trustworthiness takes on new meaning.

If you're embarking on the path of IoT development, these are some of the key topics you'll want to be informed about. ■

Cameron McKenzie
Editor in Chief, TheServerSide.com

IoT Gateways Need Clear Security Framework

Developers face a host of challenges when implementing a robust security model for Internet of Things devices and gateways. Poor security can lead to denial-of-service attacks, corporate espionage, theft and brand damage. More serious problems, such as injury or death, might occur with products like automotive software or industrial equipment. For example, the 2009 and 2010 Stuxnet attack led to destruction of a large number of nuclear centrifuges in Iran.

At the JavaOne Conference in October 2015, Luca Dazi, senior software developer at Eurotech, offered best practices for implementing better security into an Internet of Things (IoT) gateway.

The IoT gateway is a device in the field responsible for gathering data from sensors and communicating with actuators when something needs to be controlled. It can be installed in a home, an industrial control system or an automobile. The gateway also provides developers with secure I/O access to individual devices.

A best practice is to create a security framework that uses public-key cryptography to authenticate communication between remote devices and gateways, according to Dazi. This can help prevent the possibility of hackers gaining access to data from IoT devices. It also can make it more difficult for hackers to send unauthorized control signals to IoT devices or use them to launch denial-of-service attacks on IoT infrastructure.

It is also important to think about implementing IoT software to reduce the risk of unauthorized software being side-loaded onto a device, Dazi explained. One good approach is to employ a framework that uses public-key cryptography to certify new software updates before installation.

Another security step, Dazi said, is to

generate unique passwords for each device. For example, developers can prime the key-generation algorithm with the unique media access control address to generate different public and private keys.

"You need different sources of variants that are combined to generate the master password," Dazi noted. "We want to make it more difficult to retrieve this password [through a] malicious deed."

Dazi said developers can also take advantage of Eclipse Kura and Eurotech's Everyware Software Framework (ESF) to implement a solid security model for an IoT gateway and device. These frameworks make it easier to instate cloud applications that securely communicate with IoT devices. Also, ESF includes a model for securely updating configurations and device applications in the field.

The Eclipse Foundation has baked a number of basic security mechanisms into the Eclipse Kura framework. It also provides an abstraction layer to allow developers to create basic application logic that can be deployed across different hardware models for devices and gateways. According to Dazi, Kura makes it easier for Java developers to put in place basic security.

The basic Kura framework is good at securing communications between IoT devices and gateways. But there is limited support for securely updating and configuring devices from a cloud application. To address this gap, Eurotech open-sourced ESF, which can be used with Kura. ESF adds support for advanced security, remote access via a virtual private network, diagnostics and bundles for specific vertical applications.

Use public-key cryptography to authenticate communication between remote devices and gateways, recommended Luca Dazi, a developer at Eurotech.

ESF uses the basic Kura security APIs to make it easier to write Java applications that ensure the integrity and security of new software bundles. A security manager component can check for environmental integrity, ensuring that no one has tampered with files before they're run. ESF enforces runtime policies to

deny execution of particular services or the import and export of specific packages. This makes it harder for hackers to access the service and retrieve the master password.

Another good strategy is to use a cloud service to simplify the update and management of remote devices. "If you have thousands of devices in the field, you don't want to update them one after the other, and you don't want to update them all at once," Dazi said. "With batch operations, you can set a batch job in the cloud, and the cloud solution will work through the batches until the job is complete."

—*George Lawton*

Analytics Proves Key to IoT

The wealth of conclusions and extrapolations that can be obtained by sifting through Internet of Things data demands that analytics be tightly integrated into all IoT applications and devices.

The basic principles of analysis apply equally well to a world of drones, embedded devices and driverless cars as they do to business applications running on an enterprise server. The only differences, and they are potentially game-changing, are the scope of the data that can be obtained and the significance of the conclusions that can be extrapolated from it.

IoT analytics can be broken down into the following main points:

Report. Reporting data is fairly straightforward in a controlled environment, where server-side software is running locally or an application is running on a user's desktop. Reliable network protocols like User Datagram Protocol or TCP/IP can be used by desktop applications to deliver data back to a central host. Server-side software makes reporting even easier, as analytics software can be plugged right into the Web server or applications server, grabbing metrics and triggering alerts in real time.

These types of luxuries do not exist in the world of embedded and IoT devices. Whether it's a driverless car being tested in the Arizona desert or an autonomous marine robot collecting scientific data in the middle of the ocean, the availability of reliable networks is nascent. Even when reliable networks are available, they lack the bandwidth necessary to upload all of the information they acquire.

Therefore, new strategies for metrics reporting need to be employed. Data is typically prioritized. So when network latency is high, only the most critical pieces of data are uploaded. When latency improves, secondary priority data is reported. Only when an IoT device

returns home will data with tertiary and quaternary priority be uploaded to the servers.

This, of course, presents problems. Real-time reporting is often not possible with IoT devices. When there is a delay in reporting, there will be a delay in aggregation and analysis. That postpones software adjustments and bug fixes, not to mention the stress it places on staff as their work schedules are dictated by the unpredictable reporting schedule of devices running live in the field. There is no question that IoT devices require an entirely new approach to data reporting and how organizations respond to that data.

Aggregate. The whole purpose of many IoT devices is to send data back to their home base about how they're functioning and what they're experiencing. The amount of data that an army of IoT devices might report back to a cloud server could easily overwhelm a relational database. Even if a traditional database system was sufficient for storing that information, querying or mining such giant clusters of data would be a frustratingly slow process.

"We're seeing a trend where there is just too much data and too many endpoints coming in," said Sean Bowen, CEO of Push Technology Ltd., a London-based company that develops software to improve the performance of Web and mobile apps.

As a result, new strategies are being employed in the aggregation and processing of IoT data. NoSQL databases that have looser transactional requirements but much better synchronization and storage capabilities are often a better choice for data storage and retrieval than traditional relational database management systems. Processing the large blocks of data that IoT applications collect is often a task better suited to a Hadoop cluster, a technology that was originally developed to help index the Internet, as opposed to standard query calls against a database.

Analyze. Tools that present aggregated data in a way that makes it easy to understand are paramount. Presenting meaningful data in graph-based structures or on dashboards that can be customized will always be a best practice, regardless of whether server-side or IoT data is being analyzed.

The big difference with IoT analysis is that the tools being used must be compatible with the underlying data aggregation tools. That means using analytics tools that work with technologies such as Hadoop, cloud data warehouses, Amazon Web Services-based big data tools and various NoSQL databases. Many organizations will need to move away from what they've used in the past and acquire new tools that can process their data effectively.

Conclude. Analytics doesn't end at analysis. If anything, the most important part only starts there. After all, the whole point is to look at your data and come to meaningful conclusions about your systems. Analysis tools help in this regard, and data mining tools that reveal trends or hidden correlations are vital to helping an organization come to conclusions and make decisions about how to update, change, adjust and improve a system. But, in the end, coming to conclusions about the data being analyzed will always remain a largely human endeavor.

Extrapolate. The truly exciting parts of IoT analytics are the extrapolation opportunities. Serendipity abounds, as information gathered from one application may inspire the development of a new or unrelated technology.

For example, Facebook check-ins were originally designed to help users share information about where they were and what they were doing. That data is now helping users figure out when the lines are longest at their favorite Starbucks or the local DMV. The exciting part of IoT analytics is the potential to discover unknown treasure in the data.

With the right analytics tools and the ability to report, aggregate, analyze, conclude and extrapolate, IoT opportunities abound.

—*Swathija Raman*

Testing IoT Applications Requires Focus, Care

Because of the vulnerability of network connections and the potential for misuse, IoT applications need to be subjected to more stringent testing requirements, criteria and documentation than traditional business software or desktop applications.

Three characteristics make IoT testing different from testing traditional computer software: autonomy, connectivity and momentum.

The concept of autonomy is intertwined with the idea of self-governance. The compelling nature of many IoT devices, whether a self-driving car or an implanted heart monitor, is the capability to immediately respond to the stimuli they're subjected to. But this ability also demands testing for safety, reliability and adaptability.

The first letter in the IoT acronym implies that a network connection is a given, but for most devices that are live in the field, bandwidth, latency and availability are key factors in their ability to communicate with other devices. More importantly, a network connection affects an IoT application's communication with a central server that might be waiting to push out an integral update or power down a device. Testing the reliability and fidelity of devices in connected, disconnected and intermittently connected states is an aspect that cannot be overlooked.

Some of the most interesting IoT devices, from flying drones to land robots, move. With mass and speed comes momentum, and with momentum comes the potential to damage property and people. When autonomous and disconnected devices have the potential for real, physical harm, testing standards and procedures need to be brought to an entirely new level.

The net effect is that IoT testing regimens must be much more rigid and pay far greater attention to things like safety, security and

overall trustworthiness. This is a significant departure from the way most modern software is developed.

SAFETY ASSURANCE

The safety of autonomous devices is a significant nonfunctional requirement that presents another departure from typical testing practices. Developers can't have their contraptions falling from the sky or accelerating uncontrollably into a crowd of people.

A number of international standards exist that describe how products should be tested at the system, hardware and software levels, along with requirements for documenting both potentially hazardous events and ways those events can be mitigated. For example, ISO 26262 is a standard for road vehicles, while DO-178C governs software in airborne systems. And standards can be useful even when they don't apply to your company or product.

"Even if an organization isn't bound by government regulations to abide by a standard, they are still great references," said Tyler Roscoe, a compliance officer with Toronto-based Imbico Tech, a consultancy that covers the banking industry. "They not only help to identify corner cases you may have missed, but they usually include documentation standards that will show that due diligence was followed just in case there ever is a problem."

SECURITY CONCERNS

The connectedness of devices makes security a paramount concern whether applications are online or offline. Online devices that aren't properly secured can be easily accessed by the outside world.

In February of 2014, the Federal Trade Commission settled a complaint against electronics manufacturer TRENDnet for "lax security practices [that] led to the exposure of the private lives of hundreds of consumers on the Internet for public viewing."

By not paying proper attention to security, neighbors could pick up live streams of active baby monitors and other products, compromising personal privacy and confidentiality. The connectivity of IoT devices that aren't properly secured creates a wealth of opportunities for

computer experts armed with port scanners and lacking personal ethics.

While connectedness presents a host of potential dangers, so does a lack of connection. Without connectivity, an autonomous IoT device could be captured, dismantled, hacked and reassembled, all without the original manufacturer's knowledge. As a result, it's important to ensure that devices have some type of mechanism to recognize a hack.

THE HUMAN ELEMENT

Finally, perhaps one of the biggest threats to the reliability, resilience and security of any system is the human element. Testing cycles may identify bugs, and software updates may provide valid patches, but none of those patches will work if a user puts up a firewall between his or her IoT application and the outside world.

"In many cases, you only have one chance to get it right. You may never get a chance to update the software your IoT device uses," said Serge Salerno, a software consultant with Leone Consulting. One way to eliminate the possibility of a defect is to make a given feature unavailable or at least disable it and make sure someone who wants to use that feature has to jump through some hoops to enable it. "The easiest feature to test is the one that doesn't exist. Sometimes IoT devices offer advanced features to users that just don't make sense. Eliminating a feature that rarely gets used is a great way to eliminate a potential problem with it," Salerno said.

As more software developers move into the world of IoT development, they should take into account the unique attributes of IoT devices and the reality that when they go rogue, they have the potential to do real harm. So before making IoT applications available for distribution, software developers and quality assurance managers need to work harder than ever to ensure that their products are safe and secure. —*Cameron McKenzie*

GEORGE LAWTON *is a journalist based near San Francisco, Calif. He has written over 2,000 stories for publications about computers, communications, knowledge management, business, health and other areas that interest him. Email him at glawton@gmail.com or visit his website.*

CAMERON MCKENZIE *is editor in chief of TheServerSide. He has more than a dozen years of development experience, having been a technical trainer and lead consultant for Perficient Inc. He is the author of several best-selling Java books, including* What is WebSphere?, Hibernate Made Easy *and* The SCJA Certification Guide. *Email him at cmckenzie@techtarget.com or follow him on Twitter: @potemcam.*

SWATHIJA RAMAN *is a senior software developer and solution architect specializing in customer-facing retail applications with extensive expertise in the international banking sector. A graduate from the University of Madras with a bachelor's in technology, Raman started working on applications in automotive and banking domains in India and the U.S. until coming to Toronto in 2010.*

STAY CONNECTED!

Follow @TSS_dotcom today.

TechTarget

IoT Poses New Problems for Developers is a TheServerSide.com e-publication.

Scott Wallask | *Editorial Director*

Ron Karjian | *Managing Editor*

Moriah Sargent | *Associate Managing Editor*

Jan Stafford | *Executive Editor*

Bree Matturro | *Site Managing Editor*

Linda Koury | *Director of Online Design*

Martha Moore | *Senior Production Editor*

Doug Olender | *Publisher* | dolender@techtarget.com

Annie Matthews | *Director of Sales*
amatthews@techtarget.com

TechTarget
275 Grove Street, Newton, MA 02466
www.techtarget.com

© 2016 TechTarget Inc. No part of this publication may be transmitted or reproduced in any form or by any means without written permission from the publisher. TechTarget reprints are available through The YGS Group.

About TechTarget: TechTarget publishes media for information technology professionals. More than 100 focused websites enable quick access to a deep store of news, advice and analysis about the technologies, products and processes crucial to your job. Our live and virtual events give you direct access to independent expert commentary and advice. At IT Knowledge Exchange, our social community, you can get advice and share solutions with peers and experts.

COVER ART: FOTOLIA