Threats to Networking Cloud and Edge Datacenters in the Internet of Things

Deepak Puthal
University of Technology Sydney

Surya Nepal
CSIRO Data 61

Rajiv Ranjan
Newcastle University

Jinjun Chen
University of Technology Sydney

Many new applications in the smart healthcare, smart city, and precision agriculture domains are collecting data using Internet of Things (IoT) sensing devices and shipping it to remote cloud datacenters for analysis (fusion, storage, and processing). The big data analytics lifecycle, which starts with raw data collection and moves to data analytics and decision making, requires intelligent coordination of activities between tiny IoT sensors, IoT gateways, and in-transit network devices in an edge datacenter (EDC), with the big data processing frameworks and hardware resources hosted in large cloud datacenter (CDC) farms.[1] Such coordination of data analytics activities raises a new set of technical challenges from the perspective of ensuring end-to-end security and privacy of data as it travels from EDC to CDC (or vice versa). Although a number of research activities have addressed securing data in the cloud,[2] the area of securing data in an EDC while it's being collected has only recently begun to receive significant attention.[3–7]

Applications in risk-critical domains (such as flood prediction and response), which need near real-time data processing and decision making, demand that the data exchange between EDC and CDC is always secured because critical decisions (where to put temporary flood defenses, where to send rescue teams, and so on) are made based on analysis of these datasets.[1] Traditional perimeter defense strategies won't work in such applications since sensing devices are normally deployed outside the standard CDC perimeter defense (in an EDC). This raises the question of how to securely network the devices in the EDC with the software frameworks and hardware resources in the CDC. Here, we outline the security challenges and provide future research directions.

2325-6095/16/$33.00 © 2016 IEEE

Internet of Things and Edge Datacenter

Currently, more than 20 billion IoT sensors are deployed on the Internet, and this number is poised to increase in scale over the next five to 10 years. The US Federal Trade Commission estimates that there will be 50 billion IoT devices by 2020.[8] Gartner Inc.'s "Hype Cycle for Emerging Technologies, 2015" forecasts that IoT will take about five to 10 years to reach full market adoption.[9]

As noted in the previous installment of "Blue Skies," the IoT comprises billions of Internet-connected devices (ICDs) or "things," each of which can sense, communicate, compute, and potentially actuate, and can have intelligence, multimodal interfaces, physical/virtual identities, and attributes.[10] ICDs can be sensors, gateways, mobile phones, RFIDs, actuators (such as machines/equipment fitted with sensors and deployed for mining, oil exploration, or manufacturing operations), lab instruments (such as a high energy physics synchrotron), and smart consumer appliances (TV, phone, and so on). Social media, clickstreams, and business transactions are also instances of data sources in the IoT.

On the other hand, an EDC can be defined as a collection of smart IoT sensors, IoT gateways (Raspberry Pi 3, UDOO board, ESP8266, and so on), and software-defined networking devices solutions (for example, Cisco IOx, HP OpenFlow, and Middlebox Technologies) at the network edge that can offer computing and storage capabilities on a much smaller scale than CDCs.

IoT Applications

Data collected by IoT devices is being analyzed for decision making in several application domains, such as smart healthcare, smart home, precision agriculture, and disaster management. Jayavardhana Gubi and his colleagues categorized IoT applications into four groups[11]:

- *personal and home* IoT applications are at the level of an individual or home;
- *enterprise* IoT applications are at a community scale;
- *utility* IoT applications are at the local or regional scale; and
- *mobile* IoT applications spread across domains because of the network structure and the nature of connectivity.

Luigi Atzori and his colleagues categorized IoT applications into different categories[12]:

- the *transportation and logistics* domain deals with road and vehicle safety;
- the *healthcare* domain deals with object tracking, identification/authentication of people, and automatic data collection/sensing;
- the *smart environment* domain deals with the intelligence of contained objects at a home or office; and
- the *personal and social* domain deals with social networking and object intercommunications.

All of these applications deal with different types of data, sensors, and gateways, and consequently require different data analytics models for real-time decision making. We therefore need a new approach that can network IoT devices in an EDC with software and hardware resources in a CDC while ensuring end-to-end security and privacy.[1,4,7,13]

Networking CDC with EDC

IoT devices in EDCs are often used to measure physical quantities, which are then converted into understandable digital signals for analytics. Sensed data can be categorized as sound wave, vibration, voice, chemical, automobile, current, pressure, weather, and temperature.[8,11,12] Sensed IoT data or information is transferred to a CDC through different gateways and in-transit networking devices (such as wired or wireless networks) available within an EDC. Major IoT applications use wireless communication because most sensors are deployed in remote locations.[14] Individual source sensors transmit data to the CDC either in a single hop or multiple hops based on the EDC's location. Data routing is a key responsibility of the network layer and there exist several protocols to find and maintain optimal routing paths between EDCs and CDCs. IoT sensor data is aggregated on an EDC gateway device before being shipped to a CDC for further handling. Hence, the hard challenge is designing and developing cross-device (sensor-to-gateway to in-transit-networking-devices to CDC) networking protocols for ensuring end-to-end security and privacy.

The data is normally transmitted from an EDC to a CDC through three layers—perception layer,

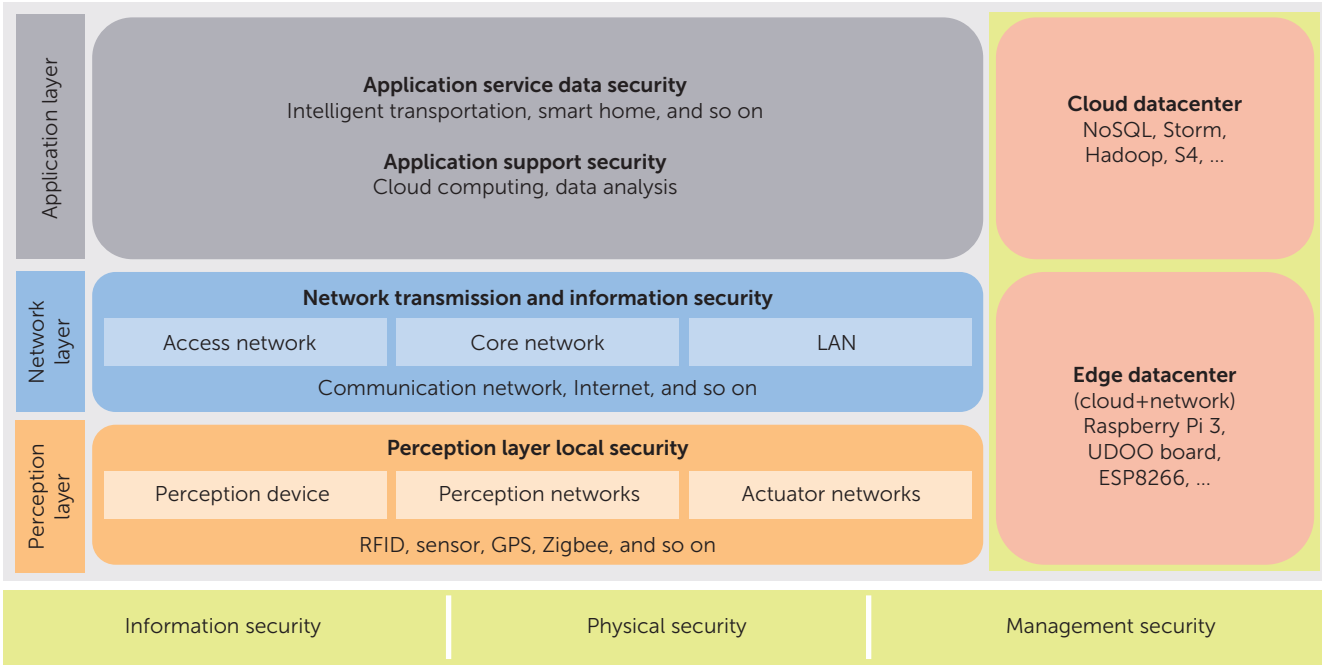

FIGURE 1. Layer-wise security architecture of cloud datacenters (CDCs) and edge datacenters (EDCs) in the context of the Internet of Things.

network layer, and application layer.[3,5] The perception layer deals with low-level data transmission (device level), comparable to the TCP/IP model's network access layer, whereas the network layer deals with Internet-level data transmission, comparable to the TCP/IP model's Internet and transport layers. The application layer works like the TCP/IP application layer. In our conceptual architecture, the perception and network layers belong to the EDC whereas the application layer belongs to the CDC.

Security Issues

Security challenges of IoT applications come from the following complexities:

- How trustworthy is the IoT sensor?
- How do we verify that that sender is a sensor device, not a robot or malicious device hosted in the EDC?
- How do we ensure that data wasn't corrupted in the EDC while in transit to the CDC?

Addressing these issues requires enforcing the security and privacy in IoT applications across layers. Figure 1 shows the layer-wise IoT security architecture and its associated properties, which extends our previous architecture.[2] Security issues at the perception layer (within the EDC) are physical device level security, such as node tempering, node capturing, and jamming. EDC network layer security issues include local area network security, Internet security, spoofing, and selective forwarding. Security issues within the CDC application layer depend on the type of software and hardware resources being provisioned for data analytics.[5]

Perception Layer

The perception layer in an EDC deals with information collection, node perception, and node control. Its main functions are perceiving and gathering information. We can divide this layer into two parts: the perception node or device (sensors or controllers) and the perception network, which communicates with the devices (gateways) in the upstream layer

(that is, the network layer).[5] The actuator network also plays a vital role at the perception layer, providing localized, on-the-edge data processing. The perception node controls node and data acquisition, whereas the perception network forwards data to the gateway and forwards messages to the IoT sensor device sent by upstream devices (in an EDC) and services (in a CDC). Sensor nodes and RFID tags are popular types of perception IoT nodes or sensors/controllers. The perception network medium is usually a wireless or mobile channel (3G/4G/5G). Both perception nodes and perception networks within an EDC are vulnerable to different types of cyber and physical security attacks.

Perception device. One of the most common types of perception device in an EDC is a sensor node and sensor gateway. RFID is an automatic identification technology to tag and track (send and receive data) remote physical objects.[5,6] Potential security attacks in the RFID sensor device context include uniform coding, conflict collision, and privacy disclosure.[6]

Depending on the IoT application type (smart home, disaster management, smart grid, and so on), we can use a large variety of hardwired IoT sensor devices—rain sensors, rain radars, pH sensors, smart meters, temperature sensors, humidity sensors, and so on. Similar to RFID sensors, these sensors are deployed at unattended remote locations. As a result, they're prone to attacks such as node capture, physical tampering, and eavesdropping. The following are security threats relevant to perception sensing devices.

- *Tampering*. An adversary can physically tamper with (for example, switch off, restart) the node and steal keys, code, and data.
- *Sensor device capture*. An adversary can collect information about the EDC by eavesdropping on the wireless medium and use this information to hack sensor devices.
- *Fake device and malicious data*. Attackers can introduce a new malicious sensor device into the system and input fake code or data. The malicious sensor device can flood the data transmission link to the EDC gateway.
- *Sybil attack*. An attacker can forge the identities of more than one sensor device, resulting in multiple malicious data sources with the same identity (IP address). In this kind of attack, the data's integrity can be compromised.
- *Source device authentication problem*. Implementing a robust sensor device authentication protocol and system is a major challenge due to resource and energy constraints.
- *Implicit deduction from sensing behaviors*. This type of attack would deduce malicious behaviors or conditions even if traffic is anonymized or encrypted. For example, a lower chatter rate might mean there are no occupants at home so it's safe to rob it.
- *Encryption leakage*. Another type of attack could stem from the fact that actual sensed data is encrypted but the sensor device ID is sent unencrypted. This could trigger a chain of new attacks based on some well-known vulnerability of the device type in question.

Several solutions applicable to the perception layer have been proposed in the literature. These solutions include intrusion detection technologies, physical security design, RFID security measures, authentication, and access control.

Perception networks. The second attack point in an EDC is the perception network layer, which tends to comprise wireless sensor networks (WSNs). In general, WSNs are dynamic networks and realize multihop routing. Examples of potential attacks in this part of the EDC include eavesdropping, malicious routing, and message tampering. These kinds of attacks can impact the security of the entire IoT ecosystem.[5,6] Since perception networks also include sensed data, security issues are also related to data confidentiality, authenticity, integrity, and freshness.[7]

The following are possible attacks at the perception network layer in an EDC.

- *Jamming* occurs if the malicious device broadcasts radio signals on the same frequency as the source sensor device, overpowering the original signal. The jam signal results in additional collisions to other frames and leads to excessive wait times for nontransmitting devices.
- In a *timing attack*, an attacker can obtain the secret and shared key information by analyzing

the encryption algorithm. An attacker can predict how much time he or she needs to get all possible secret keys and use each key to decrypt the encrypted data packets.
- A *replay attack*, which is mainly employed during authentication, destroys the certificate's validity. In this type of attack, the intruder can provide a false response on behalf of the destination node to get access to the trusted properties of the source-sensing device.
- In *routing threats*, an attacker can create routing loops by tampering with and resending or blocking the routing information. This type of attack blocks data packet transmission via the network layer, leading to aggravated delays.

Solutions that address these security threats include spectrum communication, jamming reports, cryptography technology scheme, and IPSec security channel.

Actuator networks. Perception networks gather data through sensors and pass it upward, actuator networks interact with the physical world based on local or remote data analysis and decision making. Threats from attacks on perception networks might involve loss of privacy or incorrect or delayed data; however, attacks on actuator networks can lead to real-world physical catastrophes, as can incorrect analytics or decision making upstream due to perception network attacks. For example, hackers destroyed a steel mill in Germany and made Iranian centrifuges spin too fast.[15] Actuator networks are common in applications such as environmental monitoring, healthcare, position and animal tracking, and transportation. The following are possible attacks on actuator networks within an EDC.

- A *hardware Trojan* is a malicious and deliberately stealthy modification made to electronic devices (actuators). It could cause bugs within the error-detection module, which could lead to erroneous decision making.
- *Illegal hardware clones* are the source of hardware-based exploitation. Illegal hardware cloning increases the chances of instrumenting illegally counterfeited hardware that might contain malicious backdoor or hardware Trojans.
- A *denial-of-service (DoS) attack* is produced by the unintentional failure of nodes or malicious action and can severely limit a wireless sensor network's value.
- In a *collision*, an intruder alters the transmission octets to disrupt packets.

Solutions for dealing with actuator network layer security threats include secure group management, intrusion detection, secure data aggregation, secure routing, and resilience to node capture.

Integration of perception sensor devices and networks. The third attack point is introduced by the integration of the perception sensor device and perception networks with the software frameworks and hardware resources hosted in the CDC. For example, an RFID sensor network (RSN) is an integration of an RFID (perception sensor device) and a WSN (perception network). IoT applications involve a large amount of widely distributed data that needs to be collected from heterogeneous sources. However, data gathered using different protocols might exist in heterogeneous formats. Hence, analyzing the collected data in an EDC and/or a CDC demands an effective data fusion technology. Data fusion and integration always raise the question of privacy exposure.

In addition to privacy, there are many security issues related to data integration. For example, RSNs, WSNs, and RFID use different communication protocols and different data formats, making it difficult to securely integrate and analyze data in a heterogeneous format. The problem is further complicated because RSNs, WSNs, and RFID apply different data processing methods for data filtering, aggregation, and processing.

The *side channel attack* (SCA) is another possible attack across the perception sensor device and perception network layers of an EDC. Here, attacks on sensing devices exploit side channel leakage information, such as time consumption, power consumption, and electromagnetic radiation. Intrusion detection and access control have evolved as possible mechanisms for thwarting SCAs.

Network Layer Security Problems

The EDC network layer deals with data transmission through the Internet or mobile networks. This

layer needs certain information processing and management ability for handling data available from the perception layer. Because of the heterogeneity of devices, identity authentication is an open security problem for EDCs. In the literature, approaches based on IPv6 over the low-power WPAN (6LoWPAN) architecture address network layer security problems.[16]

Data confidentiality and integrity are the most common security problems in the EDC's network layer. Some common threats to this include illegal access to networks, eavesdropping, confidentiality compromise, violation of data integrity, man-in-the-middle attacks, viruses, and exploit attacks. Furthermore, given the large number of devices, security issues such as network congestion, DoS attacks, and authentication failure are also common. The following are possible security attacks in the perception network layer of the EDC.

- In *selective forwarding*, malicious nodes refuse to forward certain data packets (messages) and/or simply drop the specific packets during data communication.
- In a *sinkhole* attack, the adversary attracts the surrounding nodes with unfaithful routing information, which will affect the data communication process.
- *Wormhole* attacks involve an adversary maliciously tunneling the incoming traffic to the wrong receivers.
- In a *HELLO flood*, a laptop-class attacker broadcasts information with enough transmission power to convince every node in the network that it's a neighbor. Because the transmission medium is wireless, the affected node selects the attacker as its neighbor for future data transmission.
- *Spoofing and alternating routing information* involve an adversary node successfully creating routing loops, attracting or repelling network traffic, extending or shortening source routes, generating false error messages, partitioning the network, increasing end-to-end latency, and so on.
- In *man-in-the-middle attacks*, intruders secretly introduce themselves and alter the communication link between source sensor device, gateways, in-transit network devices, and the CDC.
- In an *exploit attack*, an intruder takes advantage of an existing vulnerability and introduces a surprising behavior to confuse data senders and receivers.

Several solutions have been proposed to overcome network layer security threats in an EDC. These solutions include network encryption technologies, authentication and key management, ad hoc network routing protocols, multipath routing, identity verification, authenticated broadcast, data encryption (symmetric, asymmetric), and digest algorithm.

Application Layer Security Problems

Different applications such as smart city, smart health, and smart farm require different levels of security.[5,17] For example, smart health applications deal with highly sensitive data and require high-level security and privacy assurance. Hence, application layer security in a CDC is more complex and burdensome. The following are some security issues at the application layer.

- *Authentication.* Different applications have different kind of users. Effective user authentication is necessary to prevent unauthenticated access.
- *Data protection and recovery.* Existing data protection mechanisms have limitations, and they can cause catastrophic damage to data in the event of unexpected malicious attacks.
- *Ability to deal with big data.* A huge amount of data is collected and transferred to a CDC from an EDC, and data could get lost due to buffer overflow or network congestion issues.
- *Application layer software vulnerabilities.* Attackers can use software development tools to exploit vulnerabilities at the software level, such as buffer overflow, SQL injection, and cross-site request forgery.
- *Data Availability.* Only authenticated users have privileges to access the data, so unauthorized accesses are prevented from tampering with the data.

Several existing solutions can tackle above issues, including private information protection (maintaining confidentiality), data security protection, access control, heterogeneous network authentication, and key agreement and management.

The solutions we've discussed work well under the assumption that the CDC/EDC is fully protected by traditional perimeter defense mechanisms. However, this protection isn't possible since EDCs are often deployed in hostile environments.[18] We need a completely new approach to designing security solutions in such circumstances. Recently, industries have advocated three promising approaches: zero trust, deperimeterization and software-defined perimeter (SDP).

The Zero Trust architecture aims to address security problems by following a "never trust, always verify" principle.[19] Zero Trust allows for no default trust for any entity (users, devices, applications, packets, and so on) regardless of its type or whether it's on or related to the corporate network. Hence, this approach is suitable for securing EDCs and CDCs.

Paul Simmonds of the Jericho Forum (www.opengroup.org/jericho) coined the term "deperimeterization." A hardened perimeter security strategy is impossible to sustain within an agile business model. Deperimeterization protects user data on multiple levels using encryption and dynamic data-level authentication. This multilevel approach fits naturally to the multilayered architecture of EDCs to CDCs in IoT systems.

The Cloud Security Alliance (https://cloudsecurityalliance.org) launched the SDP research initiative in December 2013 with the goal of stopping network attacks against the application infrastructure. Recently, SDP, a secure integration of CDC and EDC, has gained a lot of research interest.

SDP creates a cryptographic perimeter from an EDC to a CDC. Only data from authenticated and authorized entities whose software and location have been prevalidated and deemed acceptable are transferred from the EDC to the CDC. Such entities are given a distinctive and transient cryptographic association with an obscured location that ensures the secured connection from EDC to CDC. There would be no identification of the initiator of the process when any device scans the network for details. Hence, it's useless for an attacker to scan the network to access the secured application's visibility and accessibility. Because an attacker can't see secret information, such as IP address (including DNS entries), responses to pinging, SYN/ACK, and open ports,[18] SDP can easily avoid numerous attacks, such as denial of service, man-in-middle attacks, and server/application vulnerability attacks.

We expect these approaches will be actively investigated to provide security from EDC to CDC for IoT applications. We plan to explore them further in future columns.

Acknowledgments

This research is funded by an Australia India strategic research grant, titled "Innovative Solutions for Big Data and Disaster Management Applications on Clouds (AISRF-08140)," from the Department of Industry, Australia. We also thank IEEE Cloud Computing editorial board members Raymond Choo and Joe Wienman for their valuable feedback on the paper.

References

1. R. Ranjan, "Streaming Big Data Processing in Datacenter Clouds," *IEEE Cloud Computing*, vol. 1, no. 1, 2014, pp. 80–83.
2. D. Puthal et al., "Cloud Computing Features, Issues, and Challenges: A Big Picture," *Proc. Int'l Conf. Computational Intelligence and Networks* (CINE), 2015, pp. 116–123.
3. K. Zhao and L. Ge, "A Survey on the Internet of Things Security," *Proc. 9th Int'l Conf. Computational Intelligence and Security* (CIS), 2013, pp. 663–667.
4. D. Puthal et al., "A Dynamic Key Length Based Approach for RealTime Security Verification of Big Sensing Data Stream," *Proc. Web Information Systems Engineering* (WISE), 2015, pp. 93–108.
5. Q. Jing et al., "Security of the Internet of Things: Perspectives and Challenges," *Wireless Networks*, vol. 20, no. 8, 2014, pp. 2481–2501.
6. H. Feng and W. Fu, "Study of Recent Development about Privacy and Security of the Internet of Things," *Proc. Int'l Conf. Web Information Systems and Mining* (WISM), 2010, pp. 91–95.
7. D. Puthal et al., "A Dynamic Prime Number Based Efficient Security Mechanism for Big Sensing Data Streams," to be published in *Computer and System Sciences*, 2016; http://dx.doi.org/10.1016/j.jcss.2016.02.005.

8. V. Cerf and M. Ohlhausen, "Internet of Things," lecture, Federal Trade Commission: Internet of Things Workshop, 19 Nov. 2013; www.ftc.gov/sites/default/files/documents/public_events/internet-things-privacy-security-connected-world/final_transcript.pdf.
9. Gartner, Inc., "Gartner's 2015 Hype Cycle for Emerging Technologies Identifies the Computing Innovations," Aug. 2015; www.gartner.com/newsroom/id/3114217.
10. L. Wang and R. Ranjan, "Processing Distributed Internet of Things Data in Clouds," *IEEE Cloud Computing*, vol. 2, no. 1, 2015, pp. 76–80.
11. J. Gubbi et al., "Internet of Things (IoT): A Vision, Architectural Elements, and Future Directions," *Future Generation Computer Systems*, vol. 29, no. 7, 2013, pp. 1645–1660.
12. L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A Survey," *Computer Networks*, vol. 54, no. 15, 2010, pp. 2787–2805.
13. D. Puthal, S. Nepal, R. Ranjan, and J. Chen. "DPBSV—An Efficient and Secure Scheme for Big Sensing Data Stream," *Trustcom/BigDataSE/ISPA, IEEE*, vol. 1, 2015, pp. 246–253.
14. W. Heinzelman, J. Kulik, and H. Balakrishnan, "Adaptive Protocols for Information Dissemination in Wireless Sensor Networks," *Proc. 5th Ann. ACM/IEEE Int'l Conf. Mobile Computing and Networking*, 1999, pp. 174–185.
15. K. Zetter, "A Cyberattack Has Caused Confirmed Physical Damage for the Second Time Ever," *Wired*, 8 Jan. 2015; www.wired.com/2015/01/german-steel-mill-hack-destruction.
16. A.J. Jara, M.A. Zamora, and A.F.G. Skarmeta, "HWSN6 Hospital Wireless Sensor Networks Based on 6LoWPAN Technology: Mobility and Fault Tolerance Management," *Proc. Int'l Conf. Computational Science and Eng.*, 2009, pp. 879–884.
17. X. Yang et al., "Design of Security and Defense System for Home Based on Internet of Things," *J. Computer Applications*, vol. 30, no. 12, 2010, pp. 300–318.
18. Vidder, "Software Defined Perimeter," 2016; www.vidder.com/why-vidder/software-defined-perimeter.html.
19. Nat'l Inst. of Standards and Technology, "Developing a Framework to Improve Critical Infrastructure Cybersecurity," NIST RFI 130208119-3119-01, submitted 4 Aug. 2013.

DEEPAK PUTHAL *is a PhD student in the School of Computing and Communications at the University of Technology Sydney. His research interests include big data analytics, cloud computing, information security, and wireless communication. Puthal has a MTech in computer science and engineering from NIT Rourkela, India. Contact him at deepak.puthal@gmail.com.*

SURYA NEPAL *is a principal research scientist at CSIRO Data 61, Australia. His research interests include cloud computing, Big Data, and cybersecurity. Nepal has a PhD in computer science from Royal Melbourne Institute of Technology, Australia. Contact him at surya.nepal@csiro.au.*

RAJIV RANJAN *is an associate professor (reader) in the School of Computing Science at Newcastle University, UK, and a visiting scientist at Data61, Australia. His research interests include cloud computing, content delivery networks, and big data analytics for Internet of Things (IoT) and multimedia applications. Ranjan has a PhD in computer science and software engineering from the University of Melbourne (2009). He has published about 170 scientific papers. Contact him at raj.ranjan@ncl.ac.uk or http://rajivranjan.net.*

JINJUN CHEN *is a professor in the School of Computing and Communications at the University of Technology Sydney. His research interests include cloud computing, big data, and data intensive systems. Chen has a PhD in computer science and software engineering from the Swinburne University of Technology, Melbourne, Australia. Contact him at jinjun.chen@gmail.com.*