October - December, 2015

Trinity Tech Review

Vol. 1, No.4 (4)

Disclaimer: The views and opinions presented in the articles, case studies, research work and other contributions published in Trinity Tech Review (TTR) are solely attributable to the authors of respective contributions. If these are contradictory to any particular person or entity, TTR shall not be liable for the present opinions, inadequacy of the information, any mistakes or inaccuracies.

Trinity Institute of Professional Studies
An ISO 9001:2008 Certified Institution
(Affiliated to Guru Gobind Singh Indraprastha University, Delhi)

Sector-9, Dwarka, New Delhi-110075

Ph: 45636921/22/23/24, Telefax : 45636925
www.tips.edu.in, tips@tips.edu.in

TRINITY INSTITUTE OF PROFESSIONAL STUDIES

Sector-9, Dwarka Institutional Area, New Delhi-110075, Tel: 011-45636921/22/23/24

Certified as "A" Grade Institution by SFRC, Govt. of NCT of Delhi

ISO – 9001:2008 Certified
Affiliated to GGSIP University

STATEMENT ABOUT OWNERSHIP AND OTHER DETAILS OF TTR/TMR
FORM 5 (RULE 8)

1.	Printer's Name	:	Dr. R.K. Tandon
	Nationality	:	Indian
	Address	:	Trinity Institute of Professional Studies Sector-9, Dwarka, New Delhi 110075
2.	Place of Publication	:	Delhi
3.	Periodicity of Publication	:	Quarterly
4.	Publisher's Name	:	Dr. R.K. Tandon
	Nationality	:	Indian
	Address	:	Trinity Institute of Professional Studies Sector-9, Dwarka, New Delhi 110075
5.	Editor's Name	:	Dr. J.P. Singh/ Dr. L.D. Mago
	Nationality	:	Indian
	Address	:	Trinity Institute of Professional Studies Sector-9, Dwarka, New Delhi 110075
6.	Name and Address of the individual who owns the journal and partners or shareholders holding more than one per cent of the capital.	:	CHAIRMAN Trinity Institute of Professional Studies Sector-9, Dwarka, New Delhi 110075
7.	Hosted at (url)	:	www.tips.edu.in

I, Dr. R.K. Tandon, hereby declare that the particulars given above are true to the best of my knowledge and belief.

Dr. R.K. Tandon

Introduction

This issue of TTR focuses on "Emerging Technologies" and articles related to their usage, applications, benefits as well as rising security concerns with their increasing prevalence. Here we put forth a collection of highly informative articles that discuss advanced technologies that are expected to change the world and the way we live our lives in very near future. From omnipresent mobile computing devices elaborated in the article 'Mobile Computing is a modern Technology supporting M- Commerce' and cognitive networks in article 'Next Generation Networks – Cognitive Networks' to the big bang of IT industry called Big Data in the article 'Big Data: Challenges and Opportunities', the journal makes an interesting amalgamation of articles that not only emphasize on their significance but also raise a pertinent question of security of data. A technology that most of us are using on a regular basis is Cloud. Utilized most often for storage of data, lately there have been various security concerns associated with the technology like privacy and confidentiality of data. The issue is further elaborated in the article 'Security Risks involved with cloud Providers'. We have also included in the edition a proposed solution for the same, in the article 'Autonomic Security in Cloud Computing'. The concluding article portrays an interesting relationship between the most talked about technologies of Internet of Things, Cloud and Big Data and how all of them go hand in hand to formulate solutions with breakthrough technology.

Mobile Computing is a modern technology Supporting M-commerce

Dr. Brahampal Singh

Introduction

The most familiar aspect of mobile computing technology is the hand phone. About two decades ago, a hand phone was bulky and was only used for voice communication. It was merely an extension of the fixed line telephony that allowed users to keep in touch with colleagues. Now the hand phone is not only used for voice communication, it is also used to send text and multimedia messages. Future mobile devices will not only enable Internet access, but will also support high-speed data services.

In addition to the hand phone, various types of mobile devices are now available, for example, personal digital assistants (PDAs) and pocket personal computers (PCs). Road warriors use mobile devices to access up-to-date information from the corporate database. A police officer at a crime scene may send a fingerprint picked up there for matching with data in a central database through a wireless network, hence leading to faster identification and arrest of potential suspects. The global positioning system (GPS) is used in search and rescue missions, for monitoring and preservation of wildlife, and for vehicle theft prevention. Though many of us are unaware of when mobile computing technology is being used, it has permeated all aspects of our lives.

What is mobile computing? Simply defined, it is the use of a wireless network infrastructure to provide anytime, anywhere communications and access to information. There are many aspects of mobile computing and, sometimes, different terms are used to refer to them. This chapter gives an overview of what mobile computing has to offer and how it improves the quality of our lives. Later chapters discuss the underlying wireless networks and technologies that make mobile computing applications possible.

Mobile computing is distributed computing that involves elements whose location changes in the course of computation. Elements may be software components - such as mobile agents (*see* Agent-Based Computing) - data, hardware - such as palmtops and wireless phones -, or users. This being a very broad definition, the common underlying issue is *location* and its management.

The term mobile computing is very often used for *wireless mobile computing* - the use of portable devices capable of wireless networking. Wireless mobile computing faces additional constraints induced by the characteristics of wireless communications

and the demand for portability. Mobile wireless computing enables access to data at any time and from any place towards the vision of ubiquitous (*see* Ubiquitous Computing) or pervasive computing.

Managing Location in Mobile Computing

Since the location of distributed components is not fixed, identifying their current location is necessary to contact, use or invoke them. Solutions to the problem of *locating* or *tracking* mobile objects vary depending on the application domain. In general, such solutions rely on a combination of storing some information about the location of the objects at selected sites and on performing some form of searching [5]. To locate a mobile object, the stored information about its location is retrieved. Such information may be unavailable, out-of-date or approximate, thus to track the object, its actual location must be found by searching or performing appropriate estimations. Searching may take the form of selective broadcasting at all potential sites or gradually contacting sites from the one most possible to currently host the mobile object to the less possible one.

Several data structures have been proposed for storing the location of moving objects. One approach is to store the location of all moving objects in a single *centralized spatial database*. Every time the location of an object changes, this central database needs to be updated. To handle the high update rate in such databases, the location attribute is often represented as a function of time and thus is automatically updated with time without an explicit database update operation. Representing location as a function of time is possible, when objects follow pre-defined routes as is the case of vehicles moving in a highway. Such representations may also provide estimations for the future location of the objects.

The *home base* approach adds a degree of distribution. With this approach, a specific database is associated with each object called the home base of the object. The current location of the object is stored at its home base. To locate an object, the home base associated with the object is contacted. When the object moves, its home base is updated. An enhancement of the home base approach is to store the location of all objects currently located at a site in a database residing at the site, called the *visitor database*. In this case, an object *x* that wants to contact another mobile object *y*,first contacts the visitor database at its current location, to find out whether object *y* is in the same site. If so, *x* avoids contacting y's home base that possibly resides at a remote site. As an extension of the visitor database approach, a hierarchy of visitor databases may be built. In this approach, space is divided into regions. Each database at the lower level of the hierarchy stores the location of all objects at a single region. Databases at internal levels store information for all objects covered by the databases at their children nodes.

Finally, with the *forwarding pointer* approach, each time a mobile object changes location, a pointer to its new location is deposited at its old location. Thus to contact the object a chain of pointers is followed until the object is reached. Caching and

replication can be used in all cases to improve performance and availability.

Besides tracking mobile objects, there are several other interesting queries that relate to location. Examples of such queries include finding the nearest service when the service or the user is mobile, or geographical multicasting - sending a message to all objects within a specified geographical area for instance to support geographically targeted advertising.

Changing location also has important implications in distributed system design. Distributed systems have configurations that are no longer static. Thus, distributed algorithms and protocols cannot rely on a fixed topology. Moreover, the center of activity, the system load, and locality change dynamically.

Wireless Mobile Computing

The necessary networking infrastructure for wireless mobile computing combines various wireless networks including cellular, wireless LAN, private and public radio, satellite services, and paging. Wireless networks communicate by modulating radio waves or pulsing infrared light. Wireless communications add new challenges in several areas of distributed computing.

Disconnections and Low Connectivity

In general, wireless networks are more expensive, offer less bandwidth, and are less reliable than wire line networks. Consequently, network connectivity is often *intermittent*: there are short periods of burst connections followed by network disconnections. Such network disconnections are either forced by external factors, such as unavailability of the communication signal, or voluntary for example to save cost or energy.

Distributed software systems are usually built without taking into consideration disconnections; they fail to operate when a disconnection occurs. Coda [3] is a good example of a file system that handles disconnections. To support disconnections, either periodically or when a network disconnection is anticipated, data items are cached at the mobile device to allow its autonomous operation during disconnection. Preloading data to survive a forthcoming disconnection is called *hoarding*. A critical issue during hoarding is how to anticipate the future needs for data. While disconnected, the mobile unit can use only local data. All updates are locally maintained. Upon reconnection, any updates performed at the mobile host are reintegrated with updates performed at other sites, while any conflicting updates are somehow resolved.

Weak connectivity is the connectivity provided by networks in which connection is often lost for short periods of time, is slow or expensive, making prudent use of bandwidth necessary. To handle weak connectivity, various optimizations have been proposed such as selective servicing of cache misses, compression techniques, background re integration of local updates, as well as compromising the quality of data provided to the mobile client.

Asymmetric Communications

In the case of many wireless networks, such as in cellular or satellite networks, communication is asymmetric. In particular, server machines are provided with a relative high-bandwidth wireless broadcast channel to all clients located inside a specific geographical region. Furthermore, in general, it costs less to a client in terms of power consumption to receive than to send. These considerations favor *push-based* delivery. In traditional client/server systems (*see* Client-Server Computing), data are delivered on a demand basis. A client explicitly requests data items from the server. This is termed *pull-based* delivery. In contrast, with push-based data delivery, the server repetitively broadcasts data to a large client population without a specific request. Clients monitor the broadcast and retrieve the data items they need as they arrive on the broadcast channel.

Issues in terms of broadcast push include:

- Creating and broadcasting an index for the data on the broadcast, so that clients can estimate from the index when the item of interest will appear and tune in at the appropriate time instance, thus minimizing listening to the broadcast and conserving power.

- Determining the broadcast content so that frequently accessed data items are broadcast more often than less frequently accessed ones.

- Maintaining a local cache at the client and deriving appropriate cache replacement policies; handling updates of the broadcast data.

- Query processing that involves data on the broadcast channel.

- *Hybrid delivery*: efficiently combining both push and pull-based delivery.

Device Constraints

In wireless mobile computing, to be portable, devices must be small, light and operational under wide environmental conditions. Also, in the context of ubiquitous or pervasive computing, computational power is embedded in numerous small devices. In particular:

- Portable devices have small screens and small, multifunction keypads; a fact that necessitates the development of appropriate user interfaces.

- Portable or embedded devices have less resources than static elements, including memory, disk capacity and computational power than traditional computing devices.

- Portable devices rely for their operation on the finite energy provided by batteries. Even with advances in battery technology, this energy concern will not cease to exist. The concern for power consumption spans various levels in hardware and software design.

- There are higher risks to data in mobile devices, since it is easier for mobile devices to be accidentally damaged, stolen, or lost.

An additional issue is scalability. The number of portable computing devices is in the order of billions. Storing and managing

information in such systems is a formidable task.

Software Models

To deal with the characteristics of mobile computing, especially with wireless connectivity and small devices, various extensions of the client/server model have been proposed. Such extensions advocate the use of proxies or middleware components. Proxies of the mobile host residing at the fixed network, called server-*side* proxies, perform various optimizations to alleviate the effects of wireless connectivity such as message compression and re-ordering. Server-side proxies may also perform computations in lieu of their mobile client. Proxies at the mobile client undertake the part of the client protocol that relates to mobile computing thus providing transparent adaptation to mobility. They also support client caching and communication optimizations for the messages sent from the client to the fixed server. Finally, mobile agents have been used with client/server models and their extensions. Such agents are initiated at the mobile host, launched at the fixed network to perform a specified task, and return to the mobile host with the results.

Another concern in terms of software architectures is adaptability. The mobile environment is a dynamically changing one. Connectivity conditions vary from total disconnections to full connectivity. The resources available to mobile computers are not static either, for instance a "docked" mobile computer may have access to a larger display or memory. Furthermore, the location of mobile elements changes and so does the network configuration and the center of computational activity. Thus, a mobile system is presented with resources of varying number and quality. Consequently, a desired property of software systems for mobile computing is their ability to adapt to the constantly changing environmental conditions.

The Wireless Application Protocol (WAP)

With the increasing popularity of the Internet and mobile telephony, a need arises for a standard software model for developing applications that extend Internet services to the mobile telephony environment that includes mobile phones, pagers and PDAs. *WAP* is such a set of standards. Part of the protocol is WML (Wireless Markup Language) - WAP's equivalent to HTML. A standard web server, appropriately configured, can deliver WML files. WAP defines a micro browser that displays content pages in WML-format that get transmitted to the mobile device using the WAP communications protocol over a broad range of mobile data channels.

WAP addresses the low bandwidth, high latency and limited connection availability of wireless networks and the resource constraints of the mobile devices. The network issues are addressed in both the transport and application layers of the protocol. In the transport level, a WAP gateway is inserted between the wireless network and the client that acts as a proxy: encodes the WAP data into compact formats to reduce the size and number of packets traveling over the wireless network. In

addition, the WAP gateway typically takes over most of the computing tasks from the mobile device, permitting the device to be simple and inexpensive. The device-constraints issues are also dealt with directly by WML. WML provides a small (telephony aware) set of markup tags. WML documents are divided into a set of well-defined units of user interactions, called cards. A card is usually defined by a single action or operation, usually able to be displayed on a small screen. Services, called decks, are created by letting the user navigate back and forth between cards from one or several WML documents. A deck of cards providing a complete service is downloaded at the mobile device at one time, eliminating the need for a constant network connection.

Supporting M-commerce applications

Mobile applications are gaining popularity in mobile commerce or m-commerce, which is likely to become an important application of this technology. M-commerce application can be classified into ten types:

1.Mobile financial application (business-to-customer [B2C] and business-to-business [B2B]): The mobile device is used as a powerful financial medium.

2.Mobile advertising (B2C): It turns the wireless infrastructure and devices into a powerful marketing medium.

3.Mobile inventory management (B2C and B2B) or product locating and shopping (B2C and B2B): It is an attempt to reduce the amount of inventory needed by managing in-house and on-the-move inventory. It also includes applications that help to locate products and services that are needed.

4. Proactive service management (B2C and B2B): It attempts to locate products and services that are needed.

5. Wireless reengineering (B2C and B2B): It focuses on improving the quality of business services using mobile devices and wireless infrastructure.

6. Mobile auction or reverse auction (B2C and B2B): It allows users to buy or sell certain items using multicast support of wireless infrastructure.

7. Mobile entertainment services and games (B2C): It provides entertainment services to users on a per-event or subscription basis.

8. Mobile office (B2C): It provides the complete office environment to mobile users anywhere, anytime.

9. Mobile distance education (B2C): It extends distance or virtual education support for mobile uses everywhere.

10. Wireless data center (B2C and B2B): It supports large amounts of stored data to be made available to mobile users for making "intelligent" decisions.

References

[1] G. H. Forman and J. Zahorjan, The *Challenges of Mobile Computing*, IEEE Computer, 27(4), 38-42, 1994.

[2] T. Imielinski and B. R. Badrinath, *Wireless* Mobile *Computing: Challenges in Data Management*, Communications of the ACM, 37(10), 18-28, 1994.

[3] J. J. Kistler and M. Satyanarayanan. *Disconnected Operation in the Coda File System*, ACM Transactions in Computer Systems, 10(1), 3-25, Feb 1992.

[4] E. Pitoura and G. Samaras, *Data Management for Mobile Computing*, Kluwer Academic Publishers, 1998.

[5] E. Pitoura and G. Samaras, *Locating Objects in Mobile Computing*, IEEE Transactions on Knowledge and Data Engineering, To appear.

[6]http://www.ittoday.info/Articles/Introduction_to_Mobile.htm

[7] The Wireless Application Protocol (Pearson Education) written by Sandeep Singhal, Thomas Bridgman, Lalitha Suryanarayana, Daniel Mauney.

Dictionary Terms:

Agent-based Computing

An agent is any program that acts on behalf of a (human) user. A software mobile agent is a process capable of migrating from one computer node to another.

Ubiquitous computing

Ubiquitous computing enhances computer use by making many computers available throughout the physical environment, while making them effectively invisible to users.

Client-Server Computing

An architecture in which the client is the requesting machine and the server is the supplying machine. The client contains the user interface and may perform some or all of the application processing.

Next Generation Networks: Cognitive Networks

Natasha Maniktahla

Introduction

Next-generation networks are based on Internet technologies including Internet Protocol (IP) and multiprotocol label switching (MPLS). This new networking paradigm is referred to as Next Generation (xG) Networks as well as Dynamic Spectrum Access (DSA) and cognitive radio networks.

A Cognitive radio is a radio that can change its transmitter parameters based on interaction with the environment in which it operates. From this definition, two main characteristics of the cognitive radio can be defined as follows:

- **Cognitive Capability**: Cognitive capability refers to the ability of the radio technology to capture or sense the information from its radio environment. This

capability cannot simply be realized by monitoring the power in some frequency band of interest, but more sophisticated techniques such as autonomous learning and action decision are required to capture the temporal and spatial variations in the radio environment and avoid interference to other users

- **Reconfigurability**: The cognitive radio can be programmed to transmit and receive on a variety of frequencies and to use different transmission access technologies supported by its hardware design. The cognitive radio concept can be realized through cognitive capability and reconfigurability. First, the cognitive radio identifies radio information through observation and learning processes and makes proper decisions accordingly.

Based on these decisions, the cognitive radio reconfigures its software (e.g., communication protocols) and hardware (e.g., an radio frequency (RF) front-end and antenna).

Through cognitive capability and reconfigurability, the cognitive radio enables the usage of temporally unused spectrum, which is referred to as a spectrum hole or white space . If this band is further used by a licensed user, the cognitive radio moves to another spectrum hole to avoid interference to the licensed users. This new area of research foresees the development of cognitive radio (CR) networks to further improve spectrum efficiency. The components of the CR network architecture can be classified in two groups as the primary network and the cognitive radio network.

The primary network is referred to as an existing network, where the primary users have a license to operate in a certain spectrum band. If the primary network has an infrastructure, primary user (PU) activities are controlled through the primary base-stations. Because of their priority in spectrum access, the operations of primary users should not be affected by any other unlicensed users formatting will need to create these components, incorporating the applicable criteria that follow.

Proposed framework

This highlights the first problem of CNs: identifying their motivation and potential applications.

Across the field of computer networking, there is a need to implement network-level objectives in the face of increasing network complexity. Particularly in wireless networks, there has been a trend towards increasingly complicated, heterogeneous and dynamic environments.

Determine which portion of spectrum is available.

- Select the best available channel.
- Coordinate access to these channels with other users.
- Vacate the channel when an licensed user is detected.

Moreover the unique characteristics of spectrum management and proposed solutions can be addressed as under. We define various techniques to overcome drawbacks of existing system like there is no special attention given to either time or location varying spectrum availability or switching delay. Henceforth four

Models are:

MODEL1: Framework for spectrum management in Cognitive Radio Networks.

MODEL 2: Framework for optimal spectrum sensing for Cognitive Radio networks.

MODEL 3: Framework for Infrastructure-Based Cognitive Radio Networks.

MODEL 4: Framework for spectrum aware mobility management in cognitive radio cellular networks.

Model 1: Framework For Management Of Spectrum

In this paper, intrinsic properties and current research challenges of CR networks are presented .It includes functionalities like spectrum sensing ,spectrum sharing ,spectrum decision and spectrum mobility are introduced .Emphasized on cross layer design approaches from the viewpoint of both infrastructure based network requiring central network entities and ad-hoc networks based on distributed coordination .The main challenge in CR networks is to integrate these function in layer of protocol stack so CR users can communicate directly over dynamic spectrum environment. Henceforth the influence of these functions is on upper layer protocols such as network layer and transport layer and open research issues.

Model 2: Framework for Optimal Spectrum Sensing

Main objective of spectrum sensing is to provide more spectrum access opportunities without interfering with operations of licensed network .Hence this paper is focused on interference avoidance problems .Also current radio frequency (RF) front ends cannot perform sensing and transmission at same time which decreases their transmission opportunities leading so called sensing efficiency problem. This paper solves both interference problem and spectrum efficiency problem .For the same first strategy is to develop a framework to optimize sensing parameters in such a way as to maximize sensing efficiency subject to interference avoidance constraints. Second strategy is to exploit multiple spectrum bands spectrum selection and scheduling methods where the best spectrum band for sensing are selected to maximize the sensing capacity .Finally an adaptive and cooperative spectrum sensing method is proposed where the sensing parameters are optimized for number of operating users as in figure 1.

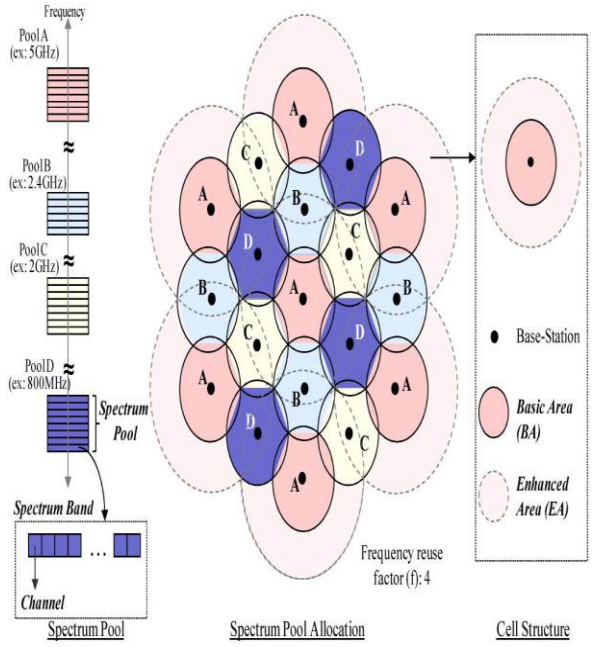

Fig 1: Architecture of Spectrum Framework

QoS - Aware Spectrum Decision Framework for Cognitive Radio Net-works Since CR networks can have multiple available spectrum bands with different channel

characteristics, they should be capable of selecting the proper spectrum bands according to the application requirements, called spectrum decision. In this thesis, a spectrum decision framework is proposed to determine a set of spectrum bands by considering the application requirements as well as the dynamic nature of the spectrum bands. For best effort applications, a maximum capacity-based spectrum decision is proposed where spectrum bands are decided to maximize the total network capacity. Moreover, a dynamic resource management scheme is developed to coordinate the spectrum decision adaptively dependent on the time-varying cognitive radio network capacity. Simulation results show that the proposed methods provide efficient bandwidth utilization while satisfying service requirements.

Model 3: Framework for Infrastructure Based Radio Networks

Since the spectrum availability varies over time and space, CR networks are required to have a dynamic spectrum sharing capability. This allows fair resource allocation as well as capacity maximization and avoids the starvation problems seen in the classical spectrum sharing approaches. In this thesis, a spectrum sharing framework for infrastructure-based CR networks is proposed that addresses these concerns by (i) opportunistically negotiating additional spectrum based on the licensed user activity (exclusive allocation), and (ii) having a share of reserved spectrum for each cell (common use sharing). Our algorithm consists of inter-cell and intra-cell spectrum sharing schemes, which account for the maximum cell capacity, minimize the interference caused to neighboring cells, and protect the licensed users through a sophisticated power allocation method. Simulation results reveal that the proposed spectrum sharing framework achieves better fairness and higher network capacity than the conventional spectrum sharing methods.

Model 4: Framework for Spectrum Aware Mobility

In CR cellular networks, CR users are traversing across multiple cells having different spectrum availability. Furthermore, they should switch to a new spectrum band when primary users appear in the spectrum, which is called spectrum mobility. Because of these heterogeneous and dynamic spectrum environments, it is challenging to provide reliable communication channels to mobile CR users. In this thesis, a spectrum-aware mobility management scheme is proposed for CR cellular networks to enable seamless mobile communications by considering both user mobility and PU activity. This can be achieved by an intelligent switching of mobile users to the best combination of a target cell and spectrum, which leads to reconfiguration of the network to maximize capacity with the minimum switching latency. More specifically, a novel network architecture is introduced to mitigate the heterogeneous spectrum availability. Based on this architecture, a unified mobility management framework is developed to support diverse mobility events in CR networks that consist of spectrum mobility management, user mobility management, and inter-cell resource allocation. The spectrum mobility management scheme increases cell capacity by allowing CR 8users to select target cells and spectrum's adaptively dependent on

current spectrum utilization. In the user mobility management scheme, a switching cost-based hand-off decision mechanism is developed to minimize quality degradation resulting from user mobility. Inter-cell resource allocation helps to improve the performance of both mobility management schemes by efficiently sharing spectrum's with Simulation results show that the proposed method can achieve better performance than conventional hand-off schemes in terms of both cell capacity as well as mobility support in communications multiple cells as in figure2.

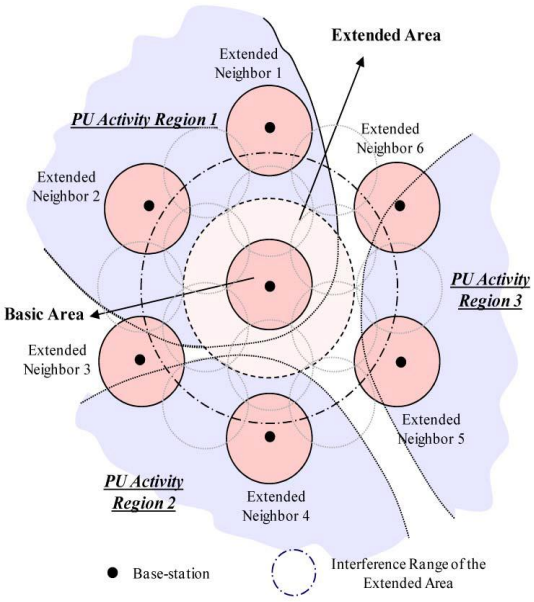

Fig 2: Framework of spectrum mobility management

Conclusion and Future Work

In this paper we proposed different strategies First, we propose the spectrum pool based network architecture, which mitigates the heterogeneous spectrum availability. Based on this architecture, a unified mobility management framework is defined so as to support diverse mobility events in CR networks, consisting of inter-cell resource allocation, and spectrum and user mobility management functions. Through inter-cell resource allocation, each cell determines its spectrum configuration to improve mobility as well as total capacity .For the PU activity, spectrum mobility management is developed where the network determines a proper spectrum band and target cell according to both current spectrum utilization and stochastic connectivity model. In user mobility management, the switching cost-based hand off decision mechanism is proposed so as to minimize quality degradation caused by user mobility. This new networking paradigm is referred to as Next Generation (xG) Networks as well as Dynamic Spectrum Access (DSA) and cognitive radio networks. As in this paper the spectrum mobility and user mobility are being jointly considered in designing a mobility management scheme for CR cellular networks. However, the probability evaluation equations can be still reduced while detecting their cell's respective probabilities. It can be done by substituting several enhanced techniques. As there is higher probability of formation of errors in calculating the probabilities concerned to Primary user activity region. This can be minimized by introducing some of smoothing techniques. All these constraints are rectified in future.

References

[1] C. Chou, S. Shankar, H. Kim, and K.G. Shin, "What and How Much to Gain by Spectrum Agility?" IEEE J. Selected Areas in Comm., vol. 25, no. 3, pp. 576-588, Apr. 2007.

[2] C. Peng, H. Zheng, and B.Y. Zhao, "Utilization and Fairness in Spectrum Assignment for Opportunistic Spectrum Access," ACM Mobile Networks and Applications, vol. 11, pp. 555-576, 2006.

[3] H. Kim and K.G. Shin," Fast Discovery of Spectrum Opportunities in Cognitive Radio Networks, "Proc. IEEE Symp. New Frontiers in Dynamic Spectrum Access Networks (DySPAN '08), Oct. 2008.

[4] I.F. Akyildiz, W.Y. Lee, M.C. Vuran, and S. Mohanty, "Next Generation/Dynamic Spectrum Access/Cognitive Radio Wireless Networks: A Survey," Computer Networks,vol. 50,pp. 2127-2159, Sept. 2006.

[5] K. Sriram and W. Whitt, "Characterizing Superposition Arrival Processes in Packet Multiplexers for Voice and Data," IEEE J. Selected Areas in Comm., vol. 4, no. 6, pp. 833-846, Sept. 1986.

[6] L. Yang, L. Cao, and H. Zheng, "Proactive Channel Access in Dynamic Spectrum Network,"

[7] Proc. Int'l ICST Conf. Cognitive Radio Oriented Wireless Networks (CROWNCOM '07), July 2007.

[8] Q. Zhao, L. Tong, A. Swami, and Y. Chen, "Decentralized Cognitive Mac Opportunistic Spectrum

[9] Access in Ad Hoc Networks: A Pomdp Framework," IEEE J.

[10] M.M. Buddhikot, I. Kennedy, F. Mullany, and H. Viswanathan, "Ultrabroadband Femtocells via Opportunistic Reuse of Multi-Operator and Multi-Service Spectrum," Bell Labs Technical J., Special Issue on 4G Networks, vol. 13, pp. 129-143, Feb. 2009.

[11] 3GPP TSG RAN WG4, "RF Requirements for Multicarrier and Multi-RAT BS (Release) 3GPP TR 37.900 V1.0.0, Sept. 2009.Selected Areas in Comm., vol. 25, no. 3, pp. 589-600, Apr. 2007

[12] R. Tandra, S.M. Mishra, and A. Sahai, "What Is a Spectrum Hole and What Does It Take to Recognize One?" Proc. IEEE, vol. 97, no. 5, pp. 824-848, May 2009.

[13] T.A. Weiss and F.K. Jondral, "Spectrum Pooling: An Innovative Strategy for the Enhancement of Spectrum Efficiency," IEEE Comm. Magazine, vol. 42, no. 3, pp. 8-14, Mar. 2004

[14] 3GPP TSG-RAN, "Requirements for Further Advancements for Evolved Universal Terrestrial Radio Access (E-UTRA)," 3GPP TR 36.913 V9.0.0, Dec. 2008.

Big data: Challenges and Opportunities

Roopal Kalra Priyanka Attri

Introduction

Big data is being generated by everything around us at all times. Every digital process and social media exchange produces it. Systems, sensors and mobile devices transmit it. Big data is arriving from multiple sources at an alarming velocity, volume and variety. In a broad range of application areas, data is being collected at unprecedented scale. Decisions that previously were based on guesswork, or on painstakingly constructed models of reality, can now be made based on the data itself. Such Big Data analysis now drives nearly every aspect of our modern society, including mobile services, retail, manufacturing, financial services, life sciences, and physical sciences. To extract meaningful value from big data, you need optimal processing power, analytics capabilities and skills.

Scientific research has been revolutionized by Big Data. The Sloan Digital Sky Survey has today become a central resource for astronomers the world over. The field of Astronomy is being transformed from one where taking pictures of the sky was a large part of an astronomer's job to one where the pictures are all in a database already and the astronomer's task is to find interesting objects and phenomena in the database. In the biological sciences, there is now a well established tradition of depositing scientific data into a public repository, and also of creating public databases for use by other scientists. As technology advances, particularly with the advent of Next Generation Sequencing, the size andnumber of experimental data sets available is increasing exponentially.

Big Data has the potential to revolutionize not just research, but also education. A recent detailed quantitative comparison of different approaches taken by 35 charter schools in NYC has found that one of the top five policies correlated with measurable academic effectiveness was the use of data to guide instruction [DF2011]. Imagine a world in which we have access to a huge database where we collect every detailed measure of every student's academic performance. This data could be used to design the most effective approaches to education, starting from reading, writing, and math, to advanced, college-level, courses. We are far from having access to such data, but there are powerful trends in this direction. In particular, there is a strong trend for massive Web deployment of educational activities, and this will generate an increasingly large amount of detailed data about students' performance.

It is widely believed that the use of information technology can reduce the cost of healthcare while improving its quality, by making care more preventive and personalized and basing it on more

extensive (home-based) continuous monitoring.

While the potential benefits of Big Data are real and significant, and some initial successes have already been achieved (such as the Sloan Digital Sky Survey), there remain many technical challenges that must be addressed to fully realize this potential. The sheer size of the data, of course, is a major challenge, and is the one that is most easily recognized. However, there are others. Industry analysis companies like to point out that there are challenges not just in Volume, but also in Variety and Velocity , and that companies should not focus on just the first of these. By Variety, they usually mean heterogeneity of data types, representation, and semantic interpretation. By Velocity, they mean both the rate at which data arrive and the time in which it must be acted upon. While these three are important, this short list fails to include additional important requirements such as privacy and usability.

The analysis of Big Data involves multiple distinct phases as shown in the figure below, each of which introduces challenges.

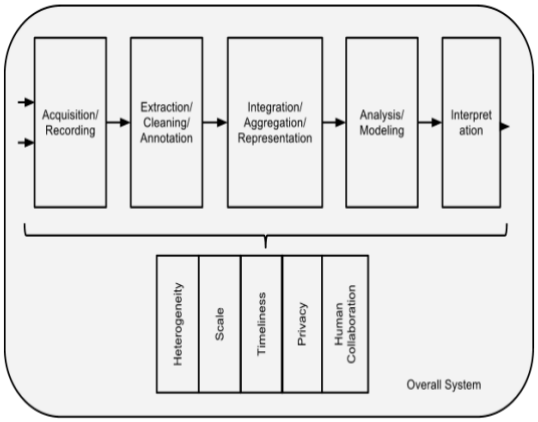

Figure 1: The Big Data Analysis Pipeline. Major steps in analysis of big data are shown in the flow at top. Below it are big data needs that make these tasks challenging.

Let us discuss both what challenges remain as we seek to exploit Big Data.

Challenges in Big Data Analysis

1. Heterogeneity and Incompleteness

When humans consume information, a great deal of heterogeneity is comfortably tolerated. In fact, the nuance and richness of natural language can provide valuable depth. However, machine analysis algorithms expect homogeneous data, and cannot understand nuance. In consequence, data must be carefully structured as a first step in (or prior to) data analysis. Efficient representation, access, and analysis of semi-structured data require further work.

Even after data cleaning and error correction, some incompleteness and some errors in data are likely to remain. This incompleteness and these errors must be managed during data analysis. Doing this correctly is a challenge. Recent work on managing probabilistic data suggests one way to make progress.

2. Scale

Of course, the first thing anyone thinks of with Big Data is its size. After all, the word "big" is there in the very name. Managing large and rapidly increasing volumes of data has been a challenging issue for many decades. In the past, this challenge was mitigated by processors getting faster, following Moore's law, to provide us with the resources needed to cope with increasing volumes of data. But, there is a fundamental shift underway now: data volume is scaling faster than compute resources, and CPU speeds are static.

3. Timeliness

The flip side of size is speed. The larger the data set to be processed, the longer it will take to analyze. The design of a system that effectively deals with size is likely also to result in a system that can process a given size of data set faster. However, it is not just this speed that is usually meant when one speaks of Velocity in the context of Big Data.

Given a large data set, it is often necessary to find elements in it that meet a specified criterion. In the course of data analysis, this sort of search is likely to occur repeatedly. Scanning the entire data set to find suitable elements is obviously impractical. Rather, index structures are created in advance to permit finding qualifying elements quickly. The problem is that each index structure is designed to support only some classes of criteria. Designing such structures becomes particularly challenging when the data volume is growing rapidly and the queries have tight response time limits.

4. Privacy

The privacy of data is another huge concern, and one that increases in the context of Big Data. For electronic health records, there are strict laws governing what can and cannot be done. Managing privacy is effectively both a technical and a sociological problem, which must be addressed jointly from both perspectives to realize the promise of big data.

5. Human Collaboration

In spite of the tremendous advances made in computational analysis, there remain many patterns that humans can easily detect but computer algorithms have a hard time finding. Indeed, CAPTCHAs exploit precisely this fact to tell human web users apart from computer programs. Ideally, analytics for Big Data will not be all computational – rather it will be designed explicitly to have a human in the loop. The new sub-field of visual analytics is attempting to do this, at least with respect to the modelling and analysis phase in the pipeline. There is similar value to human input at all stages of the analysis pipeline.

IoT: Big Data on Cloud

Ruchika Bajaj Bharti Dewani

1. IoT

IoT (Internet of things) is collection of "things" (objects) connected with internet. These objects collect and exchange data via internet. IoT allow objects to be sensed and controlled remotely. The Internet of Things (IoT) is an environment in which objects, animals or people are provided with unique identifiers and the ability to transfer data over a network without requiring human-to-human or human-to-computer interaction. IoT has evolved from the convergence of wireless technologies, micro -electromechanical systems (MEMS) and the Internet. The concept may also be referred to as the Internet of Everything.

2. Big Data

Big data includes data sets with sizes beyond the ability of commonly used software tools to capture, manage, and process data within a tolerable elapsed time. Big data "size" is a constantly moving target. Big data is a set of techniques and technologies that require new forms of integration to uncover large hidden values from large datasets that are diverse, complex, and of a massive scale.

Data growth challenges and opportunities being three – dimensional are increasing volume (increasing amount of data), velocity (speed of data in and out), and variety (range of data types and sources). These "3Vs" model is used for describing big data. "Big data is high volume, high velocity, and/or high variety information assets that require new forms of processing to enable enhanced decision making, insight discovery and process optimization."

3. Big Data Characteristics

- Volume: Big data doesn't sample. It just observes and tracks what happens
- Velocity: Big data is often available in real-time
- Variety: Big data draws from text, images, audio, video; plus it completes missing pieces through data fusion
- Machine Learning: Big data often doesn't ask why and simply detects pattern.
- Digital footprint: Big data is often a cost-free byproduct of digital interaction.

Cloud computing is a kind of internet-based computing, where shared resources and information are provided to computers and other devices on-demand. It is a model for enabling ubiquitous (available everywhere, at all times) to a shared pool of configurable computing resources. Cloud computing and storage solutions provide users and enterprises with various capabilities to store and process their data in third-party data centers. It relies on sharing of resources to achieve coherence and economies of scale over a network.

Cloud computing, or "the cloud", also focuses on maximizing the effectiveness of the shared resources. This can work for allocating resources to users. This approach helps maximize the use of computing power while reducing the overall cost of resources by using less power, air conditioning, rack space, etc. to maintain the system. With cloud computing, multiple users can access a single server to retrieve and update their data without purchasing licenses for different applications.

The term "moving to cloud" also refers to an organization moving away from a traditional CAPEX model (buy the dedicated hardware and depreciate it over a period of time) to the OPEX model (use a shared cloud infrastructure and pay as one uses it or " pay as you go " model).

4. Characteristics of Cloud Computing

- Agility improves with users' ability to re-provision technological infrastructure resources.
- Cost reductions claimed by cloud providers.
- Device and location independence enable users to access systems using a web browser regardless of their location or what device they use

- Maintenance of cloud computing applications is easier because they do not need to be installed on each user's computer and can be accessed from different places.
- Multitenancy enables sharing of resources and costs across a large pool of users thus allowing for:
- Performance is monitored and consistent and loosely coupled architectures are constructed using web services as the system interface.
- Productivity may be increased when multiple users can work on the same data simultaneously.
- Reliability improves with the use of multiple redundant sites, which makes well-designed cloud computing suitable for business continuity and disaster recovery.
- Scalability and elasticity via dynamic ("on-demand") provisioning of resources on a fine-grained, self-service basis in near real-time without users having to engineer for peak loads. This gives the ability to scale up when the usage need increases or down if resources are not being used.
- Security can improve due to centralization of data, increased security-focused resources, etc.

5. Conclusion: Convergence of IoT, Big data and Cloud

Cloud computing and the big-data analytics are the two new technologies that are evolving across the globe. IT organizations are moving towards the concept of seamless computing, and real-time processing of data with high degree of resource scalability. Moreover, cloud technology is continuously improving in security and data integration techniques. Business organizations are more concerned with the growing scope of data analytics, rather than selective storing of data from diverse resources. Thus, big-data and cloud technologies go hand-in-hand and as a result, most of the organizations are inclining towards cloud delivery models, in addition with the support of big-data analytics, especially for mission-critical workloads.

The collaboration of these two technologies enable the scope of another emerging technology – the Internet-of-Things (IoT). With the help of cloud and big-data networking, today it is possible to envision pervasive connectivity, storage, and computation, which, in turn, gives rise to different IoT solutions from environmental sensing to public safety. IoT-based applications such as innovative shopping system, infrastructure management in both urban and rural areas, remote health monitoring and emergency notification systems, and transportation systems, are gradually realizing the long-awaited dream of smart-cities. Hence, proper focus on Internet-of-Things, with the assistance of cloud and big-data technology, is of utmost importance in today's modern world.

Actually the data collected by IoT can be treated as big data and is stored in cloud. An IoT device generates continuous streams of data in a scalable way, this data is transferred and stored in cloud. Company uses big data methods to handle the high volume of stream data. The methods can be event correlation, metric calculation, statistics preparation, and analytics.

IoT gives us a dream of a home where all objects are inter-connected through internet and can be controlled and remotely accessed from anywhere in the world! This can only be possible with help of Big data and cloud.

References

[1] IBM Cloud Resource Centre

[2]Cloud Computing Implementation and Security, James W Riting and James F Rantom, CRC Press

[3]Impact of cloud, mobile and big data on IT,Williams Jones

[4]Impact of Internet of things on big data, Kaushik Pal

[5]The power of IoT and Big data.

Security Risks Involved with Cloud Providers

Akhil Kumar

Introduction

Using virtualized systems introduces many new risks, while maintaining many if not most of the risks inherent in using traditional systems. The publication by the Burton Group, "Attacking and Defending Virtual Environments," groups these risks as follows [1]:

- All existing attacks still work.
- As a separate system that must be protected, the hypervisor is risk additive.
- Aggregating separate systems into VMs increases risk.
- An untrusted hypervisor with a trusted VM has a higher risk than a trusted hypervisor with an untrusted VM.

Based on these parameters, we can identify several areas of risk to virtualized systems, including the following:

1. Complexity of configuration

Virtual systems add more layers of complexity to networks and systems, greatly increasing the possibility of improper configuration or the induction of heretofore-unseen vulnerabilities.

2. Privilege escalation

A hacker may be able to escalate his or her privileges on a system by leveraging a virtual machine using a lower level of access rights, and then attack a VM with a higher level of security controls through the hypervisor.

3. Inactive virtual machines

Virtual machines that are not active (i.e., are dormant), could store data that is sensitive. Monitoring access to that data in a dormant VM is virtually impossible, but provides a security risk through the loss of or access to the VM. Also, monitoring tools for VM systems are not as mature as traditional tools, but are expected to improve quickly.

4. Segregation of duties

A virtualized system poses risk to organizations through the improper definition of user access roles. Because the VM provides access to many type of components from many directions, proper segregation of duties may be difficult to maintain.

5. Poor access controls

The virtual machine's hypervisor facilitates hardware virtualization and mediates all hardware access for the running virtual machines. This creates a new attack vector into the VM, due to its single point of access. Therefore, the hypervisor can expose the trusted network through poorly designed access control systems, deficient patching, and lack of monitoring. This vulnerability also applies to virtualized databases.

Information Security Attacks

The number of information security attacks is increasing rapidly. Following is the list of information security attacks that IT professionals should take consideration while deploying a cloud[2].

1. Back-Door

A back-door attack takes place using dial-up modems or asynchronous external connections. The strategy is to gain access to a network through bypassing of control mechanisms, getting in through a "back door" such as a modem.

2. Spoofing

Intruders use IP spoofing to convince a system that it is communicating with a known, trusted entity in order to provide the intruder with access to the system. IP spoofing involves alteration of a packet at the TCP level, which is used to attack Internet-connected systems that provide various TCP/IP services. The attacker sends a packet with an IP source address of a known, trusted host instead of its own IP source address to a target host. The target host may accept the packet and act upon it.

3. Man-in-the-Middle

The man-in-the-middle attack involves an attacker, A, substituting his or her public key for that of another person, P. Then, anyone desiring to send an encrypted message to P using P's public key is unknowingly using A's public key. Therefore, A can read the message intended for P. A can then send the message on to P, encrypted in P's real public key, and P will never be the wiser. Obviously, A could modify the message before resending it to P.

4. Replay

The replay attack occurs when an attacker intercepts and saves old messages and then tries to send them later, impersonating one of the participants. One method of making this attack more difficult to accomplish is through the use of a random number or string called a nonce. For example, if Bob wants to communicate with Alice, he sends a nonce along with the first message to Alice. When Alice replies, she sends the nonce back to Bob, who verifies that it is

the one he sent with the first message. Anyone trying to use these same messages later will not be using the newer nonce. Another approach to countering the replay attack is for Bob to add a timestamp to his message. This timestamp indicates the time that the message was sent. Thus, if the message is used later, the timestamp will show that an old message is being used.

5. TCP Hijacking

In this type of attack, an attacker steals, or hijacks, a session between a trusted client and network server. The attacking computer substitutes its IP address for that of the trusted client, and the server continues the dialog believing it is communicating with the trusted client.

6. Social Engineering

This attack uses social skills to obtain information such as passwords or PIN numbers to be used against information systems. For example, an attacker may impersonate someone in an organization and make phone calls to employees of that organization requesting passwords for use in maintenance operations.

The following are additional examples of social engineering attacks:

- E-mails to employees from a cracker requesting their passwords to validate the organizational database after a network intrusion has occurred.
- E-mails to employees from a cracker requesting their passwords because work has to be done over the weekend on the system.
- E-mails or phone calls from a cracker impersonating an official who is conducting an investigation for the organization and requires passwords for the investigation.
- Improper release of medical information to individuals posing as doctors and requesting data from patients' records.
- A computer repair technician convinces a user that the hard disk on his or her PC is damaged and irreparable and installs a new hard disk. The technician then takes the hard disk, extracts the information, and sells the information to a competitor or foreign government.

7. Dumpster Diving

Dumpster diving involves the acquisition of information that is discarded by an individual or organization. In many cases, information found in trash can be very valuable to a cracker. Discarded information may include technical manuals, password lists, telephone numbers, credit card numbers, and organization charts. Note that in order for information to be treated as a trade secret, it must be adequately protected and not revealed to any unauthorized individuals. If a document containing an organization's trade secret information is inadvertently discarded and found in the trash by another person, the other person can use that information, as it was not adequately protected by the organization.

8. Password Guessing

Because passwords are the most commonly used mechanism to authenticate users to an

information system, obtaining passwords is a common and effective attack approach. Gaining access to a person's password can be obtained by physically looking around their desk for notes with the password, "sniffing" the connection to the network to acquire unencrypted passwords, social engineering, gaining access to a password database, or outright guessing. The last approach can be done in a random or systematic manner.

An effective means to prevent password guessing is to place a limit on the number of user attempts to enter a password. For example, a limit could be set such that a user is "locked out" of a system for a period of time after three unsuccessful tries at entering the password. This approach must be used carefully, however. For example, consider the consequences of employing this type of control in a critical application such as a Supervisory Control and Data Acquisition (SCADA) System. SCADA systems are used to run real-time processes such as oil refineries, nuclear power stations, and chemical plants. Consider the consequences of a panicked operator trying to respond to an emergency in the plant, improperly typing in his or her password a number of times, and then being locked out of the system. Clearly, the lock-out approach should be carefully evaluated before being applied to systems requiring rapid operator responses.

9. Trojan Hardware and Malware

Trojan horses hide malicious code inside a host program that seems to do something useful. Once these programs are executed, the virus, worm, or other type of malicious code hidden in the Trojan horse program is released to attack the workstation, server, or network, or to allow unauthorized access to those devices [3]. Trojans are common tools used to create back doors into the network for later exploitation by crackers. Trojan horses can be carried via Internet traffic such as FTP downloads or downloadable applets from websites, or distributed through e-mail.

Some Trojans are programmed to open specific ports to allow access for exploitation. If a Trojan is installed on a system, it often opens a high-numbered port. Then the open Trojan port could be scanned and located, enabling an attacker to compromise the system.

Summary

While the benefits of cloud computing are varied, the related risk issues are also just as varied. This article started by examining various security threats to the traditional systems and risks inherited to the virtual systems. The latter half of the article focus on the various information security threats that an IT professional/firm should consider while deploying a cloud from the vendor.

References

[1] Onur, E., Sfakianakis, E., Papagianni, C., Karagiannis, G., Kontos, T., Niemegeers, I., Chochliouros, I.P., de Groot, S.H., Sjodin, P., Hidell, M., Cinkler,T., Maliosz, M., Kaklamani, D.I., Carapinha, J., Belesioti, M., Fytros, E., "Intelligent End-To-End Resource"

[2] Virtualization Using Service Oriented Architecture", Delft Univ. of Technol., Delft, Netherlands, GLOBECOM Workshops, IEEE, 28 December 2009.

[3] Tim Mather, Subra Kumaraswamy, Shahed Latif, "Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance (Theory in Practice)", O'Reilly Media; 1 edition (4 September 2009), ISBN-10: 0596802765.

[4] Vic (J.R.) Winkler, "Securing the Cloud: Cloud Computer Security Techniques and Tactics", Syngress, 1st edition (April 29, 2011), ISBN-10: 1597495921.

Autonomic Security in Cloud Computing

Yugshakti

Introduction

Autonomic computing refers to a self-managing computing model in which computer systems reconfigure themselves in response to changing conditions and are self-healing. The promise of autonomic computing will take a number of years to fully materialize, but it offers capabilities that can improve the security of information systems and cloud computing in particular. The ability of autonomic systems to collect and interpret data and recommend or implement solutions can go a long way toward enhancing security and providing for recovery from harmful events.

Autonomic Systems

Autonomic systems are based on the human autonomic nervous system, which is self-managing, monitors changes that affect the body, and maintains internal balances. Therefore, an autonomic computing system has the goal of performing self-management to maintain correct operations despite perturbations to the system. Such a system requires sensory inputs, decision-making capability, and the ability to implement

remedial activities to maintain an equilibrium state of normal operation. Examples of events that would have to be handled autonomously include the following:

- Malicious attacks
- Hardware or software faults
- Excessive CPU utilization
- Power failures
- Organizational policies
- In-advertent operator errors
- Interaction with other systems
- Software updates

IBM introduced the concept of autonomic computing and its eight defining characteristics as follows:

1. Self-awareness

An autonomic application/system "knows itself" and is aware of its state and its behaviors.

2. Self-configuring

An autonomic application/system should be able configure and reconfigure itself under varying and unpredictable conditions.

3. Self-optimizing

An autonomic application/system should be able to detect sub-optimal behaviors and optimize itself to improve its execution.

4. Self-healing

An autonomic application/system should be able to detect and recover from potential problems and continue to function smoothly.

5. Self-protecting

An autonomic application/system should be capable of detecting and protecting its resources from both internal and external attack and maintaining overall system security and integrity.

6. Context-aware

An autonomic application/system should be aware of its execution environment and be able to react to changes in the environment.

7. Open

An autonomic application/system must function in a heterogeneous world and should be portable across multiple hardware and software architectures. Consequently, it must be built on standard and open protocols and interfaces.

8. Anticipatory

An autonomic application/system should be able to anticipate, to the extent possible, its needs and behaviors and those of its context, and be able to manage itself proactively.

The underlying concept of autonomic systems is self-management, whereby a computational system maintains proper operation in the face of changing external and internal conditions, evaluates the necessity for upgrades, installs software, conducts regression testing, performs performance tuning of middleware, and

detects and corrects problem situations in general.

Autonomic Protection

Autonomic self-protection involves detecting a harmful situation and taking actions that will mitigate the situation. These systems will also be designed to predict problems from analysis of sensory inputs and initiate corrective measures.

An autonomous system security response is based on network knowledge, capabilities of connected resources, information aggregation, the complexity of the situation, and the impact on affected applications.

The decision-making element of autonomic computing, taking into account the current security posture and security context of the system to be protected, can take actions such as changing the strength of required authentications or modifying encryption keys. The security context is derived from information acquired from network and system supervising elements and then collected into a higher-level representation of the system security status.

An oft-overlooked aspect of autonomic systems is that security vulnerabilities can be introduced by configuration changes and additional autonomous activities that are intended to address other computational areas.

Autonomous protection systems should, therefore, adhere to the following guidelines:

- Minimize overhead requirements.
- Be consistent with security policies.
- Optimize security-related parameters.
- Minimize impact on performance.
- Minimize potential for introducing new vulnerabilities.
- Conduct regression analysis and return to previous software versions if problems are introduced by changes.
- Ensure that reconfiguration processes are secure.

Autonomic Self-Healing

The process of diagnosing and repairing failures in IT systems can be difficult, time consuming, and usually requires intensive labor effort. Autonomic self- healing systems can provide the capability to detect and repair software problems and identify hardware faults without manual intervention.

The autonomic process would obtain logged and monitored information and perform an analysis to diagnose the problem area. This procedure is usually conducted by an autonomic manager that controls computing resource elements with well-defined interfaces that support the diagnostic and mitigation actions. The managed elements control their internal states and have defined performance characteristics and relationships with other computational elements.

The objective of the autonomous self-healing process is to keep the elements operating according to their design specifications.

Summary

Cloud computing security architecture is a critical element in establishing trust in the cloud computing paradigm. Confidence in using the cloud depends on trusted computing mechanisms, robust identity management and access control techniques, providing a secure execution environment, securing cloud communications, and supporting micro-architectures.

Autonomic computing can employ self-management, self-healing, and self-protection techniques to make cloud computing a more reliable, secure, and safe choice for the growing requirements for processing and storing large amounts of information in a cost-effective manner.

References

[1] Virtualization Using Service Oriented Architecture", Delft Univ. of Technol., Delft, Netherlands, GLOBECOM Workshops, IEEE, 28 December 2009.

[2] Vic (J.R.) Winkler, "Securing the Cloud: Cloud Computer Security Techniques and Tactics", Syngress, 1st edition (April 29, 2011), ISBN-10: 1597495921.

[3] Onur, E., Sfakianakis, E., Papagianni, C., Karagiannis, G., Kontos, T., Niemegeers, I., Chochliouros, I.P., de Groot, S.H., Sjodin, P., Hidell, M., Cinkler, T., Maliosz, M., Kaklamani, D.I., Carapinha, J., Belesioti, M., Fytros, E., "Intelligent End-To-End Resource