Keywords: Access Control; Access Control Policy; Distributed Systems

Security

Access Control for Emerging Distributed Systems

Vincent C. Hu, D. Richard Kuhn, and David F. Ferraiolo, National Institute of Standards and Technology

Technologies such as BigData, Cloud, Grid, and IoT are reshaping current data systems and practices, and IT experts are just as keen on harnessing the power of distributed systems to boost security and prevent fraud. How can massive distributed system capabilities be used to improve processing, instead of inflating risk?

A recent report [1] estimates that the global data population will reach 44 zettabytes (44 billion terabytes) by 2020, figure that might have seemed inconceivable only a decade ago. This growth trend is influencing the way data is being managed for high-performance computing or operations and planning analysis. To address the challenge, distributed systems are constructed for large and/or dispersed data that is difficult to process with a single data processing unit. Current storage solutions are working, but massive data breaches show that big data security solutions are failing far too often. Big data means big risk, and we are no longer surprised by reports of 100 million or more accounts stolen in a single incident. Technologies such as BigData, Cloud, Grid, and IoT are reshaping current data systems and practices, and IT experts are just as keen on harnessing the power of distributed systems to boost security and prevent fraud. How can massive distributed system capabilities be used to improve processing, instead of inflating risk?

Distributed Systems

Distributed architecture (figure 1) aims to answer distributed system scaling issues such as capabilities for data storage, advanced analysis, and shared data services. Therefore, data processing systems for distributed architecture must collect, analyze, distribute, and secure data that requires cooperatively processing diverse data sets that defy non-distributed technologies. To maximize scalability and performance, some distributed systems apply massively parallel software running on many commodity computers that may include columnar databases and other distributed management solutions, in many varieties of configurations. In general, a distributed system application has multiple interconnected but independent computers coordinating to perform a joint computation. Different computers in the system are independent, in the sense that they do not directly share memory. Instead, they communicate with each other using messages, information transferred from one computer to another over a network.

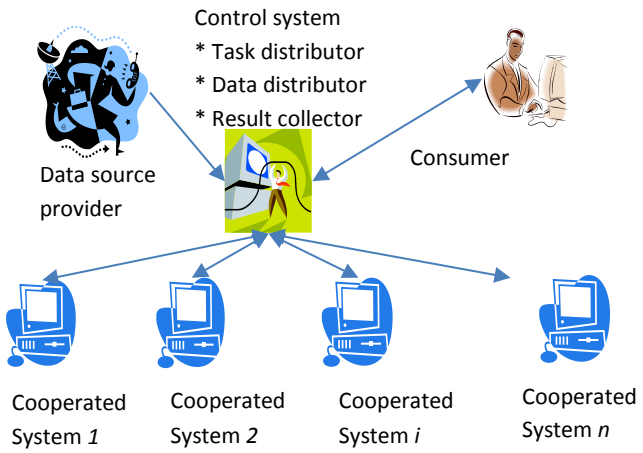

Figure 1 Distributed System

Enterprises want the same security capabilities for distributed systems as are in place for "non-distributed" information systems, including user authentication and authorization. However, the biggest challenge working with and leveraging technologies for distributed data is to maintain data security for sharing. The fundamental techniques to solve the sharing problem are access control policy enforcement and management that allows organizations to share their distributed data and processes while meeting security and privacy mandates.

Access Control Challenges of Distributed Systems

Existing distributed system models are usually overwhelmed by the processing requirements, which were not designed and built with access control capability in mind [2]. Thus, most of them cannot adequately manage the creation, use, and dissemination of distributed data and processes. As a result, they either introduce friction into collaboration through excessively strict rules, or risk serious data loss by sharing data too permissively [3]. Authentication is different from authorization, as distinguished in [4]; the authentication management function is not directly related to the data content. For distributed, as for non-distributed data systems, authentication is generally handled by coordinated systems independently. Thus, the focus of distributed system security schemes is on authorization, which is more complex than for non-distributed systems, because of the need to synchronize access privileges among the coordinated systems.

Support for the distributed system's features complicates its access control implementation, because the difficulties are in general handled by the following techniques, each with its own security challenges.

• Distributed computing – Distributed data is processed anywhere resources are available, enabling massively parallel computation between coordinated systems. This creates complicated environments that need multiple access control mechanisms and management, as opposed to centralized repositories that are monolithic and easier to implement.

• Fragmented/redundant data - Data within distributed clusters is fluid, with multiple copies moving to and from coordinated systems to ensure redundancy and resiliency. Data can become sliced into fragments that are shared across them. This fragmentation adds complexity to the data sharing as well as integrity and confidentiality.

• Node-to-node communication – Coordinated systems usually communicate through unsecure protocols such as RPC over TCP/IP [2], and data access might be compromised due to errors from communication.

The characteristics of distributed computing bring a unique set of challenges for distributed system access control, which requires a different set of concepts and considerations from traditional systems. A distributed system must not only enforce access control policies on data leaving the individual cooperated system but also control access to local resources. And depending on the sensitivity of the data, it needs to make certain that distributed applications on other coordinated systems have permission to access the data that they are processing, and deal with the access to the distributed processes and data from their local users [5].

ABAC for Distributed Systems

To answer the challenges, attribute-based access control (ABAC) [4] (Figure2) is well-adapted for distributed system access control because it provides granular and meta attributes capabilities, supporting privilege assignment in a distributed framework that requires federation and autonomy control between coordinated systems. We believe that ABAC is the future of access control. ABAC controls access to objects using rules that are evaluated with attributes of subject and object actions, and the environment relevant to a request. For example, a rule may state that access is allowed if the subject is an employee, the object is the employee's time sheet, and the environment is an office location during working hours.

Access control for distributed systems must rely on attributes to not only define access control policy rules, but also enforce the access control with collaboration among cooperating processing domains [6].

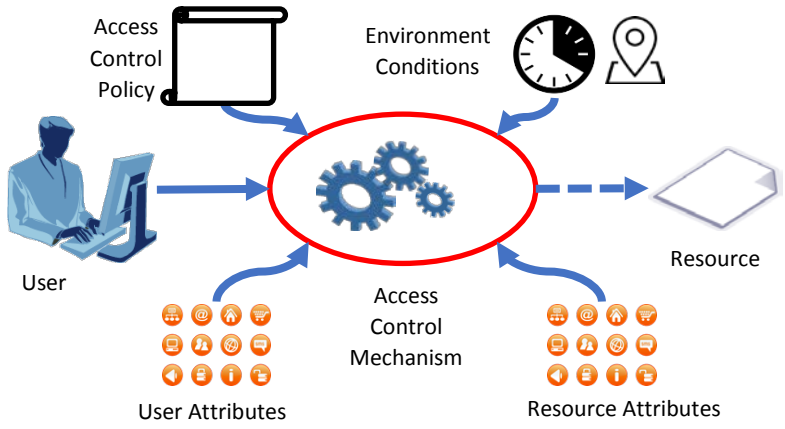

Figure 2 Access control system using attributes

In general, ABAC for distributed systems is composed of Access Control functions hosted in control and/or cooperated systems that must function together to provide access control decisions and policy enforcement. And ABAC attributes are provided by any system in the distributed environment are called Attribute Provider regardless of transmission method. An attribute provider may be the original authoritative source, or act as an intermediary between the authoritative source and the access control functions by receiving information from an authoritative source and then re-packaging the attributes for delivery/routing to storage repositories of access control functions or attribute provider.

Attributes are characteristics of the user (e.g., consumers), resource (i.e., protected resource/service), or environment conditions (e.g., time and location), which contain information given by a name-value pair (e.g. Department-Human Resource, Security level-5, Time-5:00). Attribute values may be human generated (e.g., an employee database), derived from formulas (e.g., a credit score), or system generated (e.g. environment conditions such as time, location, etc.). All must be defined and constrained by allowable values required by the appropriate scheme for the distributed environment [7]. Once attributes and their allowable values are defined, methods for provisioning attributes and appropriate attribute values to users and resources within a framework for storing, retrieving, updating, or revoking need to be devised. Therefore, attributes need to be established, issued, stored, and managed under authorities, which provide assurance schemes via location, retrieval, publication, validation, update, modification, security and revocation capabilities. As a result, it is important to ensure that the attributes obtained are secure and error-free regardless of the source, allowing risk-based decisions based on confidence in supplied attributes.

Attribute Considerations

Figure 3 illustrates the scope of attributes used in a local access control function and remote attribute providers from the perspective of a control or cooperated system unit in a distributed system. Note that the remote attributes are provisioned through remote networks. Interfaces and mechanisms must be developed or adopted to enable sharing of these attributes [4] in the distributed environment. Successful deployment of the scheme for attributes can be achieved through basic principles [8,9]: **Preparation** considers the establishing of subject, object, environment attributes as well as their granularity; **Veracity** considers the trustworthiness of attributes and their value's accuracy; **Security** considers the security of attribute-at-rest, attribute-in-transit; Readiness consider the attribute refresh, synchronization and cache mechanism; and **Management** considers the management attribute group, metadata, hierarchies, transformation, integration, minimization, and integration with authentication.

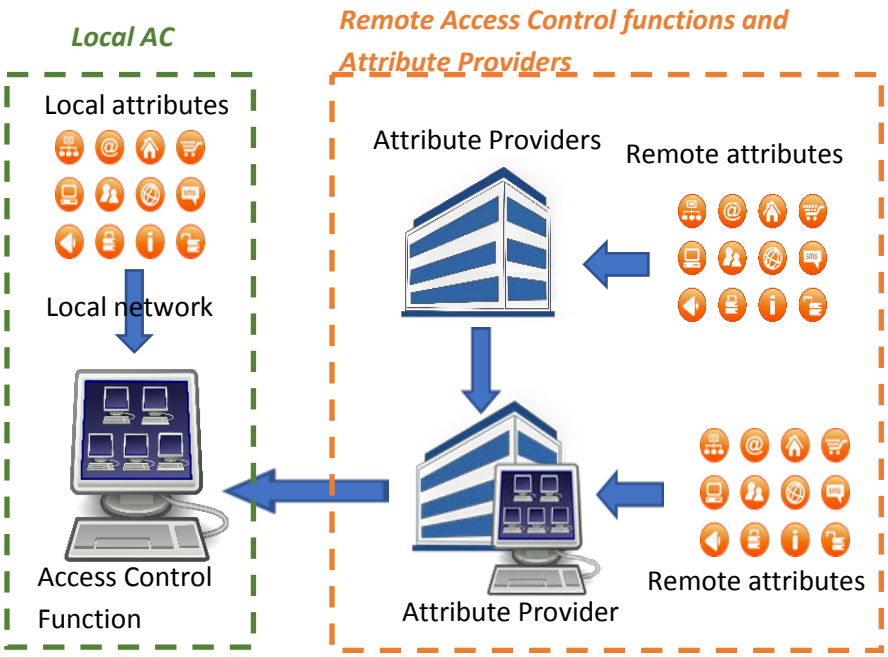

Figure 3 Scopes of attributes used in a distributed system

Privacy and security controls are also likely to be compromised due to the misconfiguration of access control policies [10], so it is important to ensure that access control policies for each computing unit in the distributed system are coordinated. For example, meta (global) policy distinguished from the local (cooperated) policy might be required depending on the configuration of the distributed system (e.g. control vs cooperated systems). Thus, synchronization and federation schemes between policies rules and attributes need to be established.

Conclusion

In summary, many distributed system designs have been proposed to address information availability challenges, but most of them were focused on processing capabilities. Considerations for security in protecting data are mostly ad hoc and patch efforts, which may not be well thought out as part of an overall security architecture. Attribute based authorization can be a critical architectural component for protecting distributed systems and their users from insider attacks. In addition, reliable attributes can provide the requisite granularity, flexibility and scalability for distributed access control configurations, making it finally possible to manage the risks that come with big data.

References

1. “Big data to turn ‘mega’ as capacity will hot 44 zettabytes by 2020”, DataIQ News, http://www.dataiq.co.uk/news/20140410/big-data-turn-mega-capacity-will-hit-44-zettabytes-2020, Oct. 2014.

2. “The Big Data Security Gap: Protecting the Hadoop Cluster,” White Paper, Zittaset, http://www.zettaset.com/wp-

content/uploads/2014/04/zettaset_wp_security_0413.pdf, 2014.

3. P. Miller, "Applying big data analytics to human-generated data," GIGAOM RESEARCH, http://research.gigaom.com/report/applying-big-data-analytics-to-human-generated-data/, Jan. 2014.

4. V. Hu, D. Ferraiolo, R. Kuhn, A. Schnitzer, K. Sandlin, R. Miller, and K. Scarfone ", Attribute Based Access Control Definition and Consideration," NIST Special Publication 800-162, Gaithersburg, MD, USA, 2013.

5. K. T. Smith, "Big Data Security: The Evolution of Hadoop's Security Model," InfoQ, http://www.infoq.com/articles/HadoopSecurityModel, Aug. 2014.

6. V. Hu, R. Kuhn, T. Xie, and J. Hwang, "Model Checking for Verification of Mandatory Access Control Models and Properties," International Journal of Software Engineering and Knowledge Engineering (IJSEKE) regular issue IJSEKE Vol. 21, No. 1., 2011.

7. NIST Internal Report 8112 -- Attribute Metadata - A Proposed Schema for Enhancing Confidence in Federated Attributes http://nvlpubs.nist.gov/nistpubs/ir/2018/NIST.IR.8112.pdf

8. NIST Interagency Report 7316, Assessment of Access Control System, September 2006, http://csrc.nist.gov/publications/nistir/7316/NISTIR-7316.pdf.

9. NIST Interagency Report 7874, Guidelines for Access Control System Evaluation Metrics, September, 2012, http://csrc.nist.gov/publications/nistir/ir7874/nistir7874.pdf.

10. V. C. Hu, T. Grance, D. F. Ferraiolo, and D. R. Kuhn, "An Access Control Scheme for BigData Processing", in Proceeding of 10th IEEE International Conference on Collaborative Computing: Networking, Applications and Worksharing, October 22–25, 2014 Miami, Florida, US.

Vincent C. Hu *is a project leader computer scientist in the Computer Security Division at the National Institute of Standards and Technology. Contact him at vhu@nist.gov.*

D. Richard Kuhn *is a project leader and computer scientist in the Computer Security Division at the National Institute of Standards and Technology. Contact him at kuhn@nist.gov.*

David F. Ferraiolo *is a computer scientist and manages the Secure Systems and Applications Group in the Computer Security Division at the National Institute of Standards and Technology. Contact him at dferraiolo@nist.gov.*

Editor: Jeffrey Voas, National Institute of Standards and Technology; jvoas@ieee.org