AXIOMATICS

FACTSHEET

5 misconceptions about a **policy-based** approach to access control

Even for senior software developers, wading into the world of identity and access management (IAM) and access control can be daunting.

The terrain is complex and the acronyms alone are a chore (RBAC, ABAC, PBAC, XACML, SAML, etc.) while also navigating the complexities of aligning roles, permissions, and attributes to achieve the business processes required to maintain productivity and business objectives while also adhering to multiple industry and global regulations..and so on. Add to this the explosion of end points, remote and/or hybrid working environments and the volume of employees and customers accessing a variety of applications and data sets and this quickly becomes overwhelming.

At its core a policy-based access control model (also referred to as Attribute Based Access Control or "ABAC") is a concept any developer can understand. The phrase "access control" refers to application mechanisms that govern what each user can (or can't) see and do. And a "policy" is a principle, rule, or guideline formulated or adopted by an organization.

While learning the fundamentals of access control and how it helps development teams secure their applications, you may be exposed to some conflicting ideas or even misinformation about policy-based access control and the value of an ABAC-based solution.

Here are five common misconceptions about a policy-based access control model and the value you may be missing.

1

Adopting external authorization will impact performance

System performance is a major concern for most teams. As a result, when developers are introduced to the concept of externalized authorization via "a centralized server," the conversation quickly turns to performance and concerns about further slowing processes. In reality, there is no impact on process efficiency and flow. **The Axiomatics centralized decision engine is purpose-built to scale and typically adds a single digit millisecond of latency. Yes, really.**

ABAC streamlines decision processes so your application code is not overwhelmed with security rules. For the developer, the interface is very simple: send a package of attributes to the authorization service, then process the permit/deny response.

2

Dynamic Authorization requires a customer to consolidate their authentication Externalized authorization is a necessary complement to authentication and can be added even if you are already using multiple login credentials. Think about it this way: you first authenticate into an application, then once you are in the application, authorization grants access to data based on a series of context-driven attributes that define what you can access and how you can access it within that application. Our customers found by pairing authentication and authorization they can enforce the use of stronger credentials and requirements when accessing critical or sensitive resources and transactions.

3

Developers can code flexible code that achieves Authorization The overall benefits of an ABAC implementation go beyond just technology. It introduces more effective work methodology for enterprise security. The dynamic authorization model is focused on centralizing policies that can be easily managed, governed and updated across the enterprise. Often when developers develop their own authorization code it is maintained within the application they work in. This is not transferable to other applications which does not scale to consistent policy adoption and a governance model in a global organization. An ABAC program investment generates many returns for future projects.

4

ABAC is just a fad ABAC is here to stay. Authorization and especially runtime dynamic authorization is gaining momentum, supported by increasing adoption of the Zero Trust (ZT) methodology. NIST recently highlighted the need for robust externalized authorization in conjunction with ZT implementation. Based on the caliber of our client list and the rapid rate of adoption by global enterprises and government agencies, ABAC is becoming the mainstream method for managing access control.

5

Roles based access control (RBAC) is good enough, I don't need attributed based access (ABAC). Unlike other access control models (RBAC), Dynamic Authorization with ABAC provides a multi-dimensional system that through its use of attributes and policies prevents role explosion, increases scalability, enables relationships, and externalizes authorization for ease of management control. It allows organizations to comply with complex regulations in a changing and demanding regulatory environment. Lastly, ABAC bridges the gap between business and IT. ABAC uses natural language policies that can be quickly analyzed and shared with auditors and compliance managers closing the loop on access reviews. Put simply, this means you can ensure that your organization's data is only available to the right people, at the right time, for the right reasons, and from the right location and device. If you have business critical or sensitive data that needs to be protected, you likely have a case for dynamic authorization.

About Axiomatics:

Axiomatics is the originator and leading provider of runtime, fine-grained, dynamic authorization delivered with Attribute Based Access Control (ABAC) for applications, data, APIs and microservices. The world's largest enterprises and government agencies use the Axiomatics Dynamic Authorization Suite to enable digital transformation, share and safeguard sensitive information, meet compliance requirements and minimize data fraud. Our innovative solutions enable enterprises to share sensitive, valuable and regulated digital assets – but only to authorized users and in the right context. To learn more, please visit <http://www.axiomatics.com> or follow us on <insert LinkedIN>

Want to learn more about our solution for Dynamic Authorization?

axiomatics.com