WHITE PAPER

Context-Based Access Control using Graph Databases for IoT

EnerNOC, an Enel Group Company: Per Gyllstrom, James Lombardi
Nulli - Identity Management: Seyed Hossein Ahmadinejad, Hadi Ahmadi, Derek Small

Abstract

EnerNOC, an Enel Group Company, a division of Enel e-solutions, is an energy company that is capitalizing on the power industry transformation, aimed at understanding and servicing the needs of Enel's global customer base by exploring opportunities in areas of new technologies to develop customer-centric, innovative products and both non-commodity and digital solutions. EnerNOC services need to store and retrieve complex data about consumer organizations. The data fundamentally includes definitions of users, roles, sites, spaces, and equipment as well as various relationships between these entities.

The variety of data objects and the complexity of relationships between them in a multi-tenant service platform will demand highly granular access management and sophisticated resource protection policies to ensure appropriate access to each and every active user of the system. These rigorous policies will rely on defining authorization data that can be utilized for policy enforcement.

To address the problem of authorization over such a complex data structure, Nulli and EnerNOC collaborated on the design of a graph-based access control solution. EnerNOC designed the data model and implemented a custom-written API, named Atlas API, that stores and retrieves graph data as well as encapsulates and orchestrates data interactions on behalf of applications. Nulli led the implementation of ForgeRock Identity Platform™ and its integration with the Neo4j® Graph Database for authentication and authorization purposes. The key advantage of this approach is to support coexistence of customer and authorization data, such as users, organizations, resources, and policy rules; hence, to allow for high-performance access control decision making through simultaneous traversing of customer and authorization data nodes over the graph.

Introduction

The enormous number of relationships between nodes, humans and processes in an IoT platform implies modelling such an interconnected environment is going very complex. Identity and Access Management (IAM) plays a key role in IoT in the modeling process as signified by the recent growing trend of managing identities when implementing IoT infrastructures. Securing access through flexible, strong authorization models is key to managing the security paradigm for IoT.

In the IoT world, every entity has an identity and the entity's access to resources is authorized based on how its identity is related to other identities. Given the complexity of describing an IoT model, implementing IAM solutions becomes even more challenging compared to other Web services platforms. The fact that identities are related to one another makes a graph data structure the best fit for modelling human and non-human identities and their relationships. We leverage graphs that are expressive data structures to make IoT models scalable, feasible and performant.

As an IoT solution provider in the energy sector, EnerNOC is facing increasing challenges and difficulties in providing a fine-grained IAM solution for applications leveraging the EnerNOC IoT infrastructure. Nulli - a pioneer in implementing state-of-the-art IAM architectures - started a collaboration with EnerNOC - an industry leading provider of energy intelligence software - to address these challenges. This paper summarizes the design and development of a fine-grained relationship-based access control model using graphs to provide solid access control.

Who is EnerNOC?

EnerNOC is a leading provider of Energy Intelligence solutions including Demand-Response and, Energy Intelligence software to provide energy efficiency, utility bill management, and energy procurement. EnerNOC was recently acquired by Enel, a multinational power company and an integrated operator in the electricity and gas sectors, with a special focus on Europe, United States, and Latin America.

Who is Nulli?

Nulli, since 1998, has crafted identity management solutions that allow organizations to manage the end-to-end lifecycle of user identities throughout the enterprise and beyond. Nulli delivers solutions in support of business requirements through proven techniques, proprietary tools, a dedicated team of experts, and the right mix of identity management products. Nulli is an early developer/adopter of relationship-based IAM using graphs by Neo4j.

Problem Definition

In order for EnerNOC to provide energy solutions, it is essential to create an internal model of the real-world customers. For example, a customer may have multiple sites (where factories or buildings exist) and for each site there are energy meters and control devices to manipulate equipment settings. With the ever-increasing number of IoT devices, this model is becoming increasingly complex. Access to this model, via software, to manage energy usage and the physical equipment requires the ability to provide flexible access control capability.

EnerNOC offers a set of application portfolios that provide solutions to energy intelligence. The solutions are sold as packages of capabilities, e.g. silver or gold, where silver might provide basic capabilities and gold a superset of capabilities for a premium price. The application accesses the model to collect measured energy data, aggregating and analyzing energy data produced by different types of devices. In some cases the software also controls the devices to turn on or off equipment, reduce the temperature, etc. The applications are made accessible to EnerNOC clients through product agreements. A product agreement describes what package of capabilities (applications or sub-applications or features) can be accessed by a client in which context. EnerNOC can define arbitrary contexts based on each client's unique requirements. A typical example of a context is geolocation. For example, let's assume there is a building management company that manages buildings in multiple cities across the US. Every building is equipped with meters to measure energy consumption in the building. This client purchased several product agreements from EnerNOC to analyze the data collected by its devices. A product agreement shows what applications will be accessible in each building. The building management company might need different packages of applications in every city or building and that is why a product agreement needs to include that information. Figure 1. shows a sample geolocation context hierarchy.

Once a product agreement is purchased by a client, the client needs to allow its employees to access the EnerNOC applications. An employee's role determines his/her level of access. Moreover, a role is given to an employee in a certain context. For example, an employee could be a Facility Manager only in a specific power plant.

Now given that client employees are assigned roles in certain contexts and product agreements are defined on applications and contexts, we aim to provide a platform to protect EnerNOC applications and secure access to clients.

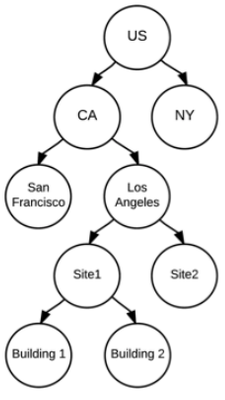

Figure 1. Sample context hierarchy

Identity and Access Management Solution

Platform Design

Externalization of IAM was a key design goal. In other words, security decisions are removed from the application developers. Authentication and authorization is performed by the IAM platform on behalf of the developer. Figure 2. shows the components required in the IAM platform. The Policy Enforcement Point (PEP) component protects EnerNOC resources by intercepting all client requests to access those resources. PEP consults the authentication server to verify the identity of the user/service who is attempting to access a protected resource. Once the user is authenticated, PEP will use the authorization server (Policy Decision Point - PDP) to evaluate whether the user is authorized to access the requested resource. PDP uses a graph database as its policy store. The graph database stores the relationship-based access control model. It describes and implements the access policies by modelling users, roles, contexts, resources (applications) and the relationships between them. All IAM related policies that are stored in the graph database are private to the IAM platform and hidden from the developers. PEP queries (traverses) the policy graph to evaluate whether the access request should be granted or denied. The identity management component provides the ability to manage user provisioning/deprovisioning.

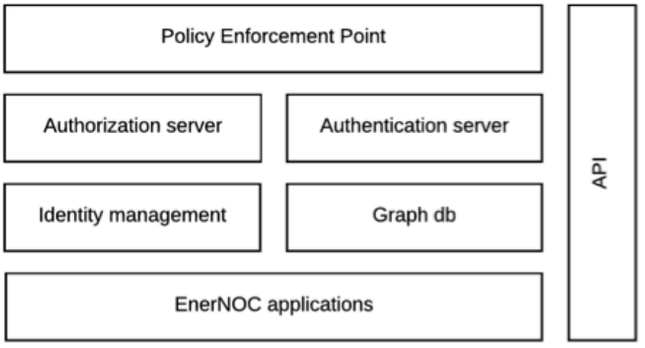

Figure 2. IAM platform components

The graph database also stores the business model entities, relationships, and attributes. EnerNOC has developed a custom-written API, named Atlas API, that makes this part of the graph model visible to other services and applications. The Atlas API encapsulates and orchestrates graph data interactions on behalf of the developers so applications can gain access to permitted resources. In conjunction with ForgeRock Identity Gateway, the API also serves as a “private” abstraction layer between the IAM components and the clients with authentication and user provisioning needs. The following sections provide further details about these components.

Access Control Model

The main difference between access control models is in the information they use for articulating an access policy. For instance, a role-based access control model uses the notion of roles in its access policies. In order to provide fine-grained access control and flexibility, a complex definition for an access policy is required. The relationship between users, roles, applications, product agreements and contexts is leveraged to provide this flexibility. The model is extensible. New entities and relationships can be added to the model to form the basis for new policies.

Roles form a hierarchy with an arbitrary depth. Applications are part of a larger graph structure which builds packages, from applications, sub applications, and features. Large context hierarchies are connected to application packages to define product agreements. Users are assigned roles in certain contexts. Last but not least, all these relationships and connections have access management significance. These requirements led us to adopt relationship-based access control model. Furthermore, in order to model the entities and their relationships, we decided to use the graph data structure and utilize its expressive power.

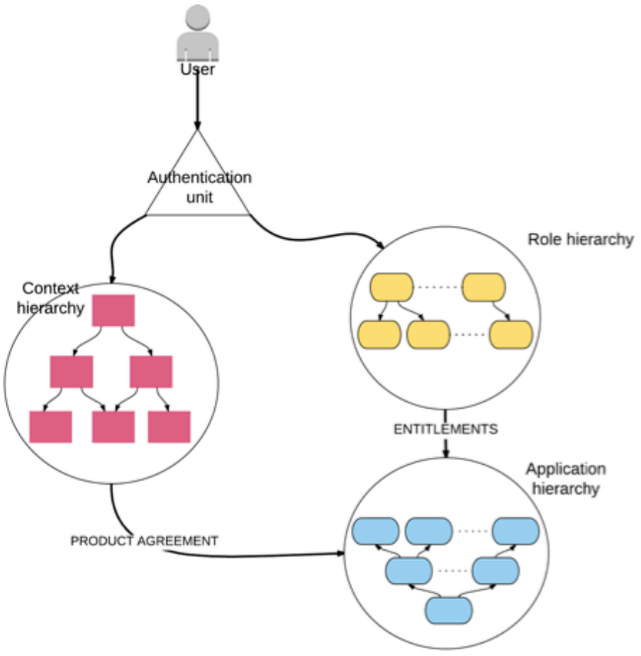

Figure 3. Sample authorization graph

Figure 3. illustrates a high level view of the graph structure designed to model identities and their relationships. Note that the notion of authorization unit was introduced to the model to assign to a user, a role in a certain context. Now an access policy can be simply defined as a query against such a graph.

Here is a sample access policy:

A user U can access an application APP in a context C if there is an authentication unit AU connected to U where,

A. there is a path from AU to APP through the roles hierarchy, and
B. there is a path from AU to context C, and
C. there exists an application package connected to APP, which context C or one of its ancestors in the context hierarchy is connected to through a product agreement edge.

Neo4j - graph database service

In order to store a graph of all identities and their relationships, a database is required. In a prior implementation, the business model was stored in a relational database. Traversal and use of the model requires database joins. Using a native graph database brings many advantages. Relationships are first class objects and do not have to be “compiled”. For this reason, EnerNOC selected Neo4j as the graph database. Neo4j is the most popular graph database in the market. The graph model is general and can be supported by any graph database.

ForgeRock® - authentication & authorization service

As mentioned earlier, a PEP component is required to enforce authentication and authorization before access is allowed to protect a resource. EnerNOC chose ForgeRock Identity Gateway as the PEP. ForgeRock Identity Gateway is a lightweight gateway which is scriptable to support a wide variety of functionalities including being a PEP.

Once ForgeRock Identity Gateway intercepts a request to access an application, it needs to identify the user/service who is making the request. To do so, an authentication server is employed. EnerNOC adopted ForgeRock Access Management as the authentication server. ForgeRock Access Management supports various IAM services including authentication modules, session management, federation, OAuth 2.0, etc.

Before making a request to access an EnerNOC application, a user needs to authenticate with ForgeRock Access Mangement and then include the obtained session token in its request. ForgeRock identity Gateway then intercepts the call and validates the token with ForgeRock Access Mangement. The next step is to determine if all applicable access policies are satisfied. This requires running all the access policy queries against the graph stored in Neo4j. To do so, ForgeRock Identity Gateway can directly make a call to Neo4j REST API. However, we decided to use ForgeRock Access Mangement as the policy engine as it provides many other features for policy management and evaluation.

Since ForgeRock Access Mangement does not support relationship-based policies out of the box, we developed a Neo4j plugin to do this. The plugin enables us to a) store access policies in the form of queries, and b) run queries against Neo4j. As a result, when ForgeRock Identity Gateway catches a request with a valid token, it then sends a request to ForgeRock Access Mangement to evaluate its access policies and decide if the request is allowed or not. If ForgeRock Access Mangement approves the request, ForgeRock identity Gateway will forward it to the target application.

Conclusion

A variety of real-life attacks against IoT platforms, which can cause serious damage to organizations and governments, show the risks and challenges in providing secure IoT solutions. IoT solutions introduce a massive number of connected devices and complex relationships between devices themselves, users, owners, and services. With a rich IoT ecosystem, new flexible and complex fine-grained access control requirements need to be provided. Standard access control solutions fall short in providing the necessary capabilities. The core to defining security models and policies is “identity and access management (IAM)”, which assigns trackable identity to each and every digitally connected thing and defines an access control model together with policies and cryptographic tools to enforce it. Providing an appropriate access control model for an IoT problem is a challenge due to the variety of services and object models. Nulli working with EnerNOC has shown that the relationship-based access control model leveraging a graph data structure is the best fit for addressing the challenge of designing an IoT IAM architecture. A combination of capable graph database engines such as Neo4j with solid and feature-rich IAM tools such as the ForgeRock stack can result in a secure relationship-based IAM platform for IoT.

nulli
Identity Management