Access Control Security

Last Modified on 26/08/2020 10:44 am AEST

Overview

The ReadiNow Access Control system determines which users can view, create, and modify data records, reports, screens, and other services. Access Control is dynamic in that arbitrary custom, data driven, security rules can be easily created that take advantage of the data on records and the relationships between records. For example, you can define a security rule that applies to managers such that they can read orders associated with active customers if the customer is in the same region as the manager. In this example, the security rule would apply to a 'Manager' security role, and it would be configured to traverse the relationship from the manager to their region, to the customers in that region, verify that the customer is active, then traverse to the orders for that customer.

Consider the following users and data. Two users in the manager role, Alice and Bob. Alice is in the Asia region, Bob is in the Europe region. Customer A is in the Asia region. Customer B is in the Europe region. The single access rule above will allow Alice to see Customer A's orders, and Bob to see Customer B's orders. If Customer A's status changes from Active to Inactive, then Alice will no longer see Customer A.

New relationship types can be custom created in the administration user interface, as can new access rules. This example gives a flavour of the type of access control that is possible.

User Accounts, Records and Permissions

The end objective of access control is to determine if a given user account is permitted to perform a given action in relation to a given record or system entity.

User accounts are connected via a relationship to the person that owns the account. From there a system relates to many other records along various types of relationships. It is also possible for permission to be granted to some resources on some basis other than following a path to the user.

Permissions are the types of actions that can be secured. These include 'view', 'create', 'modify', and 'delete'. The create permission is special in that it relates to types of records, rather than individual records.

User Roles

User Roles allow user accounts to be grouped together to make the administration of access rules more convenient. User roles typically specify a job function.

In many other systems that offer User Roles, or Security groups, it can be necessary to maintain many security roles for different regions. That is not the case in ReadiNow, as the relationship-following nature of the access rules still act on individual users within the role. So objectives such as region, customer, or department-based access can often be achieved with a single access rule.

ReadiNow, nonetheless, offers flexible options for combining user roles. User Roles can be members of other user roles. For example, if 'User Role B' is a member of 'User Role A', then this mean that the users in role B are also implicitly in role A. Or put differently, access rules and other content made available to role A are also made available to users in role B.

In this example, both Alice and Bob can be in the same Manager role. The example Access Rule for 'Orders', previously described, would be applied to both users, but will still differentiate which particular Order records Alice and Bob can individually access based on their specific region relationship, according to the query in that

rule.

Users can be assigned to multiple roles. User roles can be members of multiple other roles. User roles can be nested multiple levels deep.

There is a built-in 'Everyone' role, which implicitly applies to all users. Access rules and content granted to the 'Everyone' role is made available to all users. Role membership is not supported for the 'Everyone' role, in order to prevent accidental exposure of other roles to all users.

Record Access Rules

Access rules are at the heart of the ReadiNow access control system. Each record access rule includes:

1. The User Role (or single user account) that it applies to
2. The type of record (the Object) that it applies to. For example 'Customer'.
3. The permissions granted by the rule. (Read, update, delete, create)
4. The query that describes what specific records can be accessed

The query can be thought of as a filtering mechanism. It is a report built using the ReadiNow reporting engine. The report engine allows record filters to be applied, such as checking certain values on the record. If a given record is acceptable to the report's filter, then the access rule grants the user or role the specified permission(s) to that record.

For example, an access rule could be defined that grants view and modify to all incidents that have their status set to urgent. This would grant view and modify permission to all urgent incidents, to any users in the role nominated by the access rule. (Note that in this example, the rule does not relate to individual users).

The report engine also allows relationships to be followed, and filters to be applied based on the values of the related record. Relationships can be followed to multiple levels, and recursively. If the relationship types lead to a user account, then it can be filtered to only match the current user.

Our example 'Orders' record access rule query describes a relationship path from the 'order' record to the current user. In this example, the report query would follow from Orders to the Customer, to the Region, to users in that region.

Calculations can also be applied in the filtering process.

Navigation Access

The Navigation Access security interface allows administrators to specifically nominate certain screens and reports as being available to certain user roles.

The record access control engine may sound complex at first. However, it allows for arbitrary security arrangements to be defined. The Navigation Access mechanism is one such example, as it is internally driven by a small number of record access rules.

Security Relationships

Security Relationships allow permissions that are granted to one record to be inferred onto a related record. The Security Relationships feature is not to be confused with the relationship aspect of record access control that we

have seen so far.

In the ReadiNow platform, records are frequently connected to each via relationships to build a rich data model. For example, a Disaster Recovery Plan may relate to Plan Steps; Policy documents; Implementation Notes, and so on. It is common to want to grant access to group of interconnected records as a whole.

Record Access Rules allow complex rules to be specified, but they do so for only one type of record at a time. In the above example, an access rule might describe who gains access to a Disaster Recovery Plan. Independently, the relationship types “plan has steps”, “plan has policy documents”, “plan has notes”, and so on can be marked as being security relationships. If a user gains read access to any plan, then they will automatically also gain read access to its steps, documents, notes, and so on. Similarly, if they have modify access to the plan, then modify access is inferred to the related records.

The security access relationship applies to the relationship type (for example “plan has steps”), and not to individual relationship instances (for example, “my plan A has step A1”). Therefore, they should only be used in circumstances where access should always be granted.

Security relationships are followed recursively, such that security can be conveyed through rich data models. The permission inferred to the related entity is always the same as the permission granted to the parent entity. In this example, if a user can read a plan, but not modify it, then the same applies to the plan steps.

Inheritance

Objects (record types) can be arranged to inherit field and relationships that are defined on other Objects. For example, a ‘Manager’ object can be defined that inherits from an ‘Employee’ object. In this case the Manager is referred to as the derived object and the Employee object is referred to as the parent object. The Manager Object inherits any fields and relationships that are defined on the Employee object. The Manager Object can then also define additional fields and relationships.

For all intent and purpose, Manager records also act as Employee records. They will appear in Employee reports; and can be used in any relationships that call for Employee.

For record access control security, any access rule that grants someone access to some parent object will, by default, also grant access to the derived object. For example, an access rule that grants permission to see all employees will also, by default, grant access to all managers. This can be changed by adjusting the record access control query to only apply to the exact type.

Combined Access

In the ReadiNow Access Control system, all security rules grant access in an additive manner.

That is to say, if any one access rule, or role, causes a user to have some permission to some record, then the user definitely has that permission on that record. There is no notion of ‘denied’ permissions.

Put differently, the total access that a user gains to the system is the sum, or union, of the access granted by each access rule that applies to the user.

This design bolsters security as it means the security engine starts by assuming the user has no access whatsoever, other than that granted by a record access control rule. It also provides the basis for substantial performance optimisations. When designing security policys, only grant what is required, rather than granting broad access to many users then seeking to deny individual content.

Security Summary Report

The record access control rules may appear complex at first. However, the system is intended to allow you to model your security policy, and then drive actual security automatically from data; rather than needing to maintain individual record access policy on large numbers of resources, security groups, or user roles.

To assist with reviewing record access rules, the ReadiNow platform provides a summary report mechanism that, for a given User Role, calculates all Objects that may be accessible; the permissions granted; as well as the specific reasons; whether it be for access rules, security relationships, or inheritance. This allows the administrator to gain a clear picture of the objects that a role can see.

Internal Security Filter

The only exception to the notion of no denials is the internal security filter. Record access control is performed in two phases. In order to gain access to a record, the user must receive permission according to the customer configurable access rule, as described above. In addition, a second system level security layer enforces access control rules to protect certain system and application records.

Summary

The platform provides a comprehensive robust Access Control model that:

- Allows security policy to be described
- Then automatically applied based on data
- Taking advantage of the rich ecosystem of related, interconnected, records
- Starting with the assumption that a user has no access

Custom implementations can be as simple as automatically granting access to records that have a certain status; reaching through to complex rules that span multiple interconnected records, applying calculations along the way.