

Eyes Only

- [Eyes Only](#)
 - [About](#)
 - [How Eyes Only works](#)
 - [Follow along](#)
 - [Eyes Only Foundation](#)
 - [Eyes Only Field](#)
 - [Eyes Only Override Role](#)
 - [Eyes Only Before Query Business Rule](#)
 - [Eyes Only Reset Business Rule](#)
 - [Outro](#)
-

About

Eyes Only is a security feature for ServiceNow records, so only those associated with the record can view. This is helpful when your organization has sensitive information that does not need to be seen by others outside a select few.

This guide is to help ServiceNow admins setup Eyes Only to limit who can see designated records. The goal for Eyes Only is to work in conjunction with the default out-of-the-box (OOTB) ServiceNow role base security model to insure future upgradeability across ServiceNow releases.

How Eyes Only works

Eyes Only is a custom true/false field on the *Task [task]* table. If the Eyes Only field is set to true, then only the following people can read the record:

- opened by
- closed by
- assigned to
- watch list
- additional assignee list
- work notes list
- resolved by
- accounts with the eyes_only_override role



Requested For & Caller ID

The *Task [task]* base table does not have either requested for or caller fields. It makes logical sense that these too should have access to the records associated with the account. This will be addressed below.

If you follow the steps below, you can add, remove, or change criteria as warranted to fit your business requirements.

Follow along

ServiceNow offers many ways to solve a problem or configure an operational business model. These instructions are a way and does not represent the way on creating a Eyes Only feature. The best way to use these instructions is to read through them and see how the feature can be adopted and modified to fit your organization requirements.



Update sets

There is an update set that implement this guide; Eyes Only. The update set can be found at <https://github.com/ChristopherCarver/EyesOnly>.



Modifying OOTB Tables

This guide focuses on creating a robust feature tied closely to the out of the box (OOTB) tables already existing in ServiceNow. There are alternative implementation solutions within ServiceNow to accomplish the same result. Your mileage may vary depending on scope defined by and practices set by your organization and/or development team. This feature is meant to be as open and flexible to meet varying modifications to suit present and future requirements.

Eyes Only Foundation

The foundation for the Eyes Only feature is based on the use of before query business rules. This allows for a tighter security control. Using before query business rules allows for a cleaner experience for accounts with record access roles, when compared to access control lists (ACLs), without sacrificing any security.

Instead of creating a single before query business rule on the base table *Task [task]* table, it is better practice to create a new before query business rule for each table that extends another as warranted; e.g., incidents, request items, sc tasks, etc. etc.. The reason for this is the *Task [task]* table does not contain other user reference fields present in derived (child) tables from the base parent table of *Task [task]*.



Coverage

This guide will setup a before query business rule on the *Incident [incident]* table. The same can easily be applied to other tables by following the example provided. There are over 40 OOTB tables that

extend (children of) the *Task [task]* table. Choose which ones are appropriate for your deployment to implement.

Eyes Only Field

The OOTB *Task [task]* table is the base table from which many records are extended from and makes it a good place to create a custom field to represent Eyes Only, as then all tables that extend the *Task [task]* table will inherit the custom field automatically. The purpose of the Eyes Only feature is to insure only those directly associated with the record can see and has access to the record.

Instructions:

1. Navigate to **System Definition > Tables**.
2. Filter the listing where **Name** is **task**.
3. Open the **Task** record.
4. Under the **Columns** tab, click **New**.
5. Under the **Dictionary Entry New record** section, fill in the following fields:
 - Type: True/False
 - Column label: Eyes only
 - Column name: u_eyes_only
6. Under the **Default Value** tab, fill in the following fields:
 - Default value: false
7. Click **Submit**.

Eyes Only Override Role

The *eyes_only_override* role allows accounts to view Eyes Only records even if they are not associated with the record.

Instructions:

1. Navigate to **User Administration > Roles**.
2. Click **New**.
3. Under the **Role New record** section, fill in the following fields:
 - Name: eyes_only_override
 - Description: Accounts can view Eyes Only records.
4. Click **Submit**.

Eyes Only Before Query Business Rule

The Eyes Only before query business rule should allow only the following accounts access to a record:

- caller

- opened by
- closed by
- assigned to
- watch list
- additional assignee list
- work notes list
- resolved by
- accounts with the eyes_only_override role



Other Tables

The following before query business rule applies the Eyes Only feature to incident records. The same before query business rule can be applied to other tables to extend Eyes Only functionality; adjust accordingly.

Instructions:

1. Navigate to **System Definition > Business Rules**.
2. Click **New**.
3. Under the **Business Rule New record** section, fill in the following fields:
 - Name: Eyes Only - Incident Query
 - Table: Incident [incident]
 - Advanced: true
4. Under the **When to run** tab, fill in the following fields:
 - When: before
 - Insert: false
 - Update: false
 - Delete: false
 - Query: true
5. Under the **Advanced** tab, fill in the following fields:
 - Script:

```
(function executeRule(current, previous /*null when async*/ ) {

    if (gs.hasRole('admin') || gs.hasRole('eyes_only_override')) {
        return;
    }

    var userReference = gs.getUser().getRecord().getValue('sys_id');

    // note the following columns are from the base task table that the record extend
    //   assigned_to, opened_by, closed_by, watch_list, additional_assignee_list, wor
    // the other fields unique to their table for the business rule

    current.addQuery('u_eyes_only', false)
        .addOrCondition('assigned_to', userReference)
        .addOrCondition('opened_by', userReference)
        .addOrCondition('closed_by', userReference)
```

```

        .addOrCondition('watch_list', 'CONTAINS', userReference)
        .addOrCondition('additional_assignee_list', 'CONTAINS', userReference)
        .addOrCondition('work_notes_list', 'CONTAINS', userReference)
        .addOrCondition('caller_id', userReference)
        .addOrCondition('resolved_by', userReference);

    })(current, previous);

```

1. Click **Submit**.

Eyes Only Reset Business Rule

There is a condition when you want to automatically reset the Eyes Only field to false and that is when the *assigned to* field is empty. A common use scenario is the re-assignment of records and when this occurs the *assigned to* field is emptied. If the *Eyes only* field is toggled true, then the members of the *Assignment group* would not see the ticket. So, if the *Assigned to* field is empty or blank, then the *Eyes only* field need to be reset to false.

Instructions:

1. Navigate to **System Definition > Business Rules**.
2. Click **New**.
3. Under the **Business Rule New record** section, fill in the following fields:
 - Name: Eyes Only - Task Reset
 - Table: Incident [incident]
 - Advanced: true
4. Under the **When to run** tab, fill in the following fields:
 - When: before
 - Insert: true
 - Update: true
 - Delete: false
 - Query: false
 - Condition:
 - Option: Assigned to
 - Operation: is empty
 - Logic condition: OR
 - Option: Assigned to
 - Operation: is empty string
5. Under the **Actions** tab, fill in the following fields:
 - Set field values:
 - Option: Eyes only
 - Operater: To
 - Option: false
6. Click **Submit**.

Outro

You have now successfully setup the Eyes Only feature. Congratulations.