

Project Title: Prevent User Deletion if Assigned to an Incident

Platform: ServiceNow

Team ID : LTVIP2025TMID30459

Team Size : 4

Team Leader : Nulu Mohan

Team member : Gnanchand Isukapati

Team member : Chitturi Kalyani

Team member : Dunnala Hemanth Bhagavan

INTRODUCTION

- **Project Overview**

This project implements a safeguard in ServiceNow to prevent deletion of users assigned to open incidents.

- **Purpose**

To ensure data integrity and avoid accidental deletions that disrupt workflows.

IDEATION PHASE

- **Problem Statement**

Deleting users assigned to active incidents can break workflows and lead to data loss.

- **Empathy Map**

- User: IT Admin
- Pain: Loss of incident tracking
- Need: A safeguard against deletion
- Gain: Workflow safety

- **Brainstorming**

- Pop-up alert – ✗ not enough
- Restrict role – ✗ too broad
- ✓ Business Rule to check before deletion – Selected

REQUIREMENT ANALYSIS

- **Customer Journey Map**

Admin → Deletes user → System checks incident → Block if assigned → Allow if not

- **Solution Requirement**

- Functional: Prevent deletion if incidents assigned

- Non-functional: Fast, reliable, light-weight

- **Data Flow Diagram**

Delete User → Trigger Business Rule → Check Incident Table → Block/Allow

- **Technology Stack**

- ServiceNow

- JavaScript

PROJECT DESIGN

- **Problem-Solution Fit**

Solves workflow disruption from user deletions

- **Proposed Solution**

Business Rule on sys_user table, triggered before delete

- **Solution Architecture**

Admin Deletes → Rule Checks → Incident Found?

→ Yes → Show Error → Abort

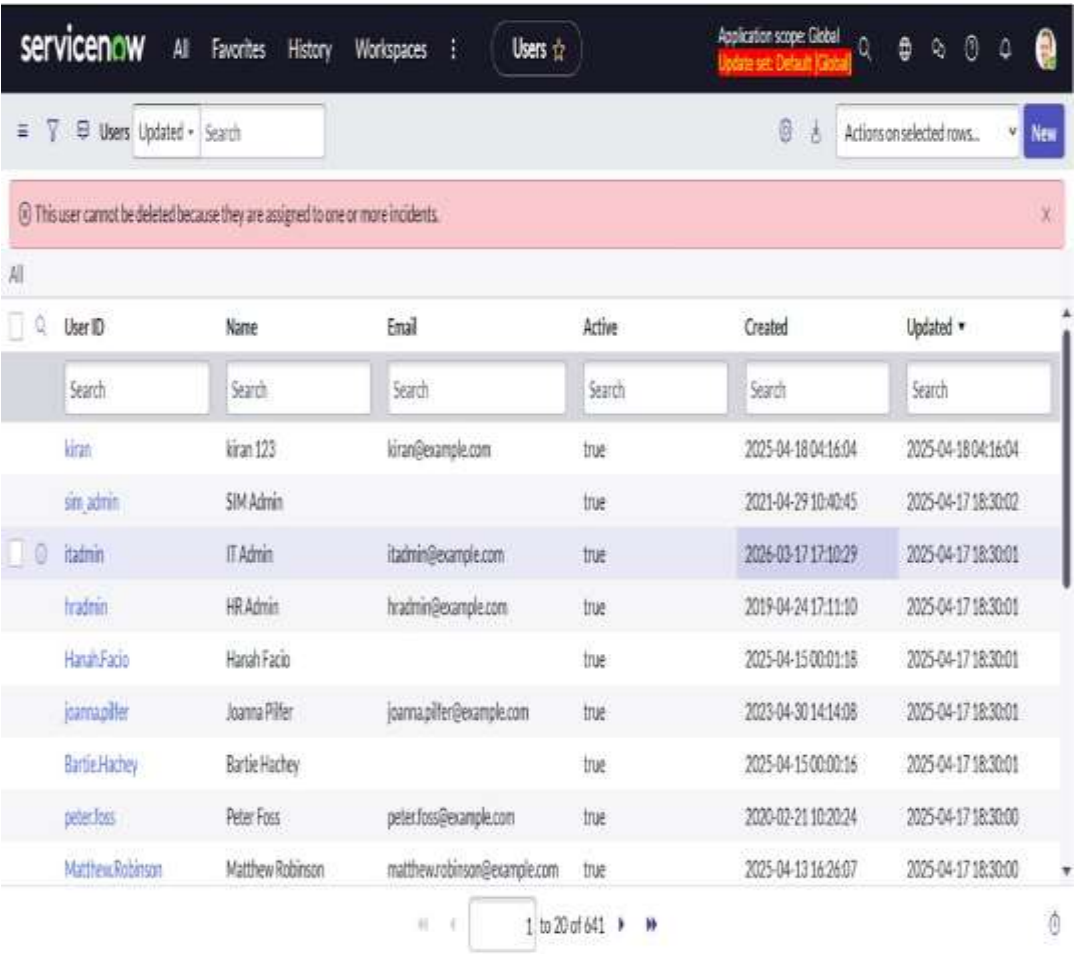
→ No → Allow Deletion

PROJECT PLANNING

Task	Time
Create Users	1 hr
Assign Incidents	1 hr
Business Rule Setup	1 hr
Testing + Screenshots	1 hr
Report & Video	1 hr

FUNCTIONAL AND PERFORMANCE TESTING

- **Performance Testing**
- Assigned user: Deletion blocked
- Unassigned user: Deletion succeeds
- Rule works under 0.2s
- No lag or performance issues



User ID	Name	Email	Active	Created	Updated
Search	Search	Search	Search	Search	Search
kiran	kiran 123	kiran@example.com	true	2025-04-18 04:16:04	2025-04-18 04:16:04
sim_admin	SIM Admin		true	2021-04-29 10:40:45	2025-04-17 18:30:02
itadmin	IT Admin	itadmin@example.com	true	2026-03-17 17:10:29	2025-04-17 18:30:01
hradmin	HR Admin	hradmin@example.com	true	2019-04-24 17:11:10	2025-04-17 18:30:01
Hannah.Facio	Hannah Facio		true	2025-04-15 00:01:18	2025-04-17 18:30:01
joanna.piller	Joanna Piller	joanna.piller@example.com	true	2023-04-30 14:14:08	2025-04-17 18:30:01
Bartie.Hachey	Bartie Hachey		true	2025-04-15 00:00:16	2025-04-17 18:30:01
peter.foss	Peter Foss	peter.foss@example.com	true	2020-02-21 10:20:24	2025-04-17 18:30:00
Matthew.Robinson	Matthew Robinson	matthew.robinson@example.com	true	2025-04-13 16:26:07	2025-04-17 18:30:00

RESULTS

- **Output**
- Business Rule created
- User deletion blocked if assigned
- Successful deletion for others
- Screenshots captured

servicenow Application scope: Global [Update per Default \(Joblog\)](#)

Users Updated Search Actions on selected rows... New

All

User ID	Name	Email	Active	Created	Updated
Aje	Aje Kumar	aje@example.com	true	2025-04-18 06:59:51	2025-04-18 06:59:51
Kiran	Kiran 123	kiran@example.com	true	2025-04-18 04:16:04	2025-04-18 04:16:04
sim_admin	SIM Admin		true	2021-04-29 10:40:45	2025-04-17 18:30:02
iadmin	IT Admin	iadmin@example.com	true	2026-03-17 17:30:29	2025-04-17 18:30:03
hradmin	HR Admin	hradmin@example.com	true	2019-04-24 17:11:10	2025-04-17 18:30:01
Harsh Facko	Harsh Facko		true	2025-04-15 00:01:18	2025-04-17 18:30:01
joanna.piller	Joanna Piller	joanna.piller@example.com	true	2023-04-30 14:34:08	2025-04-17 18:30:01
Bartle Hachey	Bartle Hachey		true	2025-04-15 00:00:16	2025-04-17 18:30:01
peter.foss	Peter Foss	peter.foss@example.com	true	2020-02-21 10:30:24	2025-04-17 18:30:00
Matthew Robinson	Matthew Robinson	matthew.robinson@example.com	true	2025-04-13 16:38:07	2025-04-17 18:30:00

1 to 20 of 642

servicenow Incident - INC0010001 Application scope: Global [Update per Default \(Joblog\)](#)

Incident: INC0010001 Discuss Follow Update Resolve Delete

Number: Channel:

* Caller: State:

Category: Impact:

Subcategory: Urgency:

Service: Priority:

Service offering: Assignment group:

Configuration item: Assigned to:

* Short description:

Description:

Related Search Results

servicenow Application scope: Global [Update per Default \(Joblog\)](#)

Users Updated Search Actions on selected rows... New

All

User ID	Name	Email	Active	Created	Updated
Kiran	Kiran 123	kiran@example.com	true	2025-04-18 04:16:04	2025-04-18 04:16:04
sim_admin	SIM Admin		true	2021-04-29 10:40:45	2025-04-17 18:30:02
<input checked="" type="checkbox"/> iadmin	IT Admin	iadmin@example.com	true	2026-03-17 17:30:29	2025-04-17 18:30:03
hradmin	HR Admin	hradmin@example.com	true	2019-04-24 17:11:10	2025-04-17 18:30:01
Harsh Facko	Harsh Facko		true	2025-04-15 00:01:18	2025-04-17 18:30:01
joanna.piller	Joanna Piller	joanna.piller@example.com	true	2023-04-30 14:34:08	2025-04-17 18:30:01
Bartle Hachey	Bartle Hachey		true	2025-04-15 00:00:16	2025-04-17 18:30:01
peter.foss	Peter Foss	peter.foss@example.com	true	2020-02-21 10:30:24	2025-04-17 18:30:00
Matthew Robinson	Matthew Robinson	matthew.robinson@example.com	true	2025-04-13 16:38:07	2025-04-17 18:30:00
Athena Cantu	Athena Cantu		true	2025-04-14 23:53:36	2025-04-17 18:30:00

1 to 20 of 641

ADVANTAGES & DISADVANTAGES

- **Advantages**

- Protects data and workflow
- Simple to implement
- Real-time prevention

- **Disadvantages**

- Doesn't check related records
- Needs manual extension for other modules

CONCLUSION

- The Business Rule safeguards the ServiceNow platform by stopping deletions of users assigned to incidents. It improves stability and responsibility in IT operations.

FUTURE SCOPE

- Extend for closed incidents
- Add logging/audits
- Apply to Problems/Changes

APPENDIX

- Business Rule Code:

Business Rule Script:

```
(function executeRule(current, previous) {  
    var incGr = new GlideRecord('incident');  
    incGr.addQuery('assigned_to', current.sys_id);  
    incGr.setLimit(1);  
    incGr.query();  
    if (incGr.next()) {  
        gs.addErrorMessage('This user cannot be deleted because they are assigned to one or more i');  
        current.setAbortAction(true);  
    }  
})(current, previous);
```

THANK YOU