

# Mobile Data Collection Toolkit - SurveyCTO

## Scenario on case management and user- based access control

This tutorial is made available under the terms of the  
[Creative Commons Attribution-Share Alike 4.0  
International License](https://creativecommons.org/licenses/by-sa/4.0/).



# Why might this scenario be useful to you?



This scenario will give you an example of :

- **Limiting access to a case in the cases dataset**, based on the **user** name (*users* column).

→ This means that **only users whose name is mentioned in the *users* column will be able to see the case on the SurveyCTO Collect application.** Other users will not see this case appear in the list of cases on the application.



## Scenario (1/3):

The international solidarity organization *Terre des animaux* deployed SurveyCTO in the field. A data collection system with animal tracking (case management) has been set up. The follow-up is done in the following way:

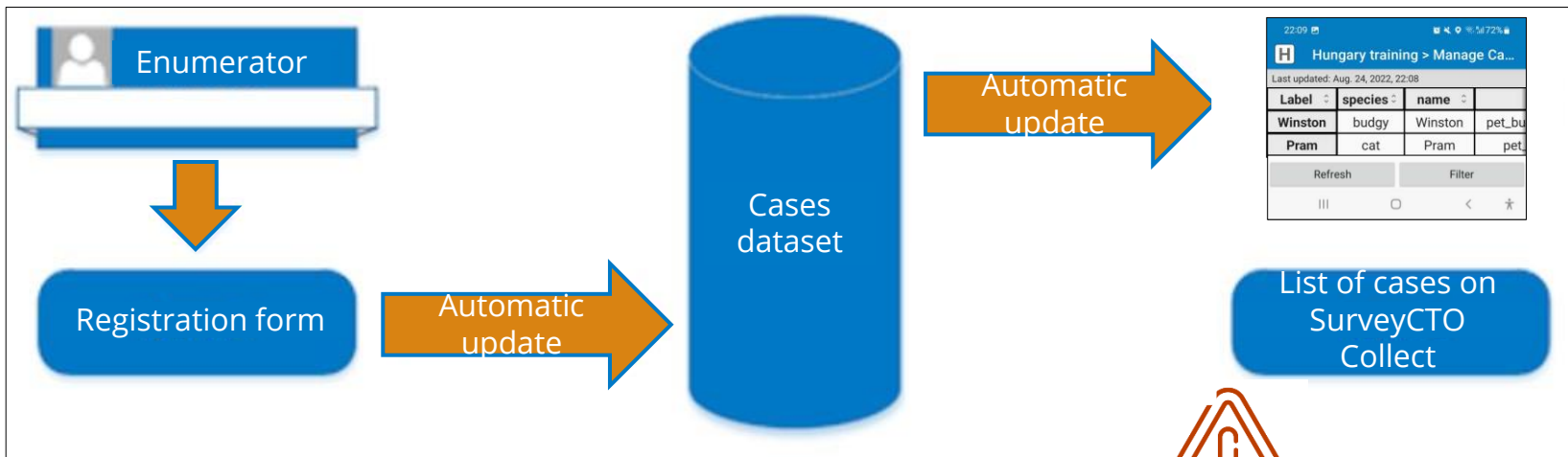
- 1) When a new animal is identified, an educator (enumerator) completes the *Registration Form*.
- 2) This animal is recorded in the cases dataset.



From that point on, the information for that animal should only be visible (on the *SurveyCTO Collect* application) to the enumerator and his or her supervisor. Other enumerators and supervisors should not see this animal in the list of cases on the application.

# Scenario (2/3):

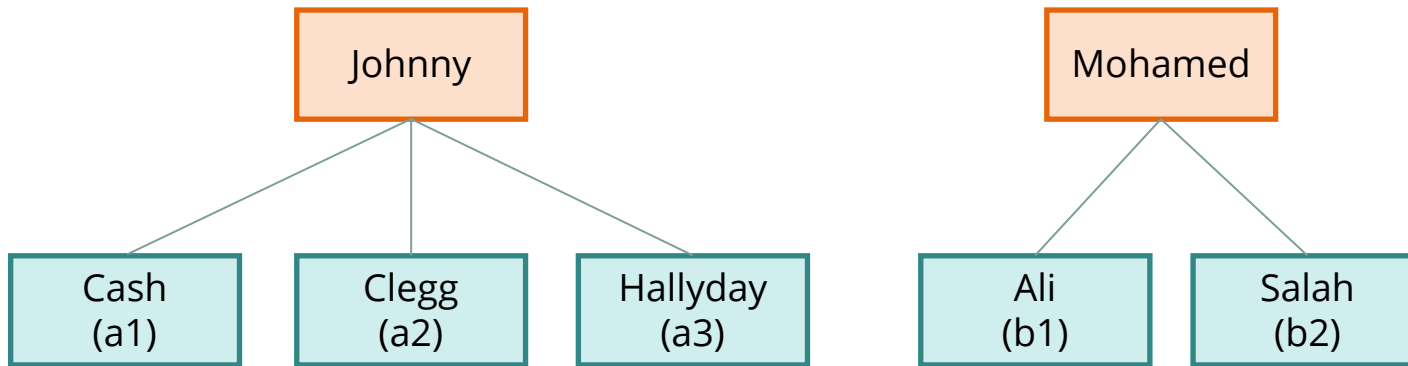
Schematically, this is what it would look like:



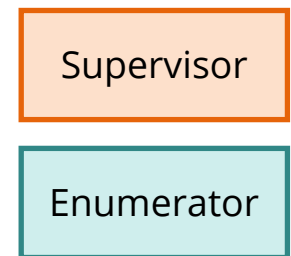
The list of cases varies for each enumerator and supervisor, depending on which cases they are allowed to see and which they are not.

# Scenario (3/3):

The organization of the enumerators and their respective supervisors is as follows:



*Caption:*



# Proposed solution on SurveyCTO:



The registration form includes an **automatically calculated field** that will populate the **users** column in the cases dataset.

→ **Only the enumerator and supervisor listed in the users column** can see this animal in the case list.

# Registration form - XLSForm

# Registration form - XLSForm :

The form coded on XLSforms includes 2 questions that will limit access to the enumerator and his or her supervisor, only:

1) The enumerator must choose his/her name from the list of enumerators:

	A	B	C
1	type	name	label
2	select_one enumerator	enumerator	Select your name (enumerator) from the list.

survey | choices | settings | (+)

	A	B	C
1	list name	name	label
6	enumerator	a1	Enumerator Cash
7	enumerator	a2	Enumerator Clegg
8	enumerator	a3	Enumerator Hallyday
9	enumerator	b1	Enumerator Ali
10	enumerator	b2	Enumerator Salah

survey | **choices** | settings | (+)



# Registration form - XLSForm :

2) An automatic calculation field will add the user name of the enumerator and that of his/her supervisor, in the *users* column of the cases dataset:

	A	B	C	D	E	F	G
1	type	name	label	hint	constraint	constraint_message	calculation
14	calculate	users					<code>concat(\${enumerator}, ',', pulldata('list_enumerator_supervisor','supervisor','enumerator','\${enumerator}'))</code>

→ Using the *pulldata* function, the name of the supervisor is automatically retrieved thanks to a list (in csv format) associating each enumerator with his/her supervisor.

	A	B	C	D
1	enumerator, supervisor			
2	a1, johnny			
3	a2, johnny			
4	a3, johnny			
5	b1, mohamed			
6	b2, mohamed			

list\_enumerator\_supervisor

# Registration form - XLSForm :

→ The registration form is as follows:

	A	B	C	D	E	F	G	H	I
1	type	name	label	hint	constraint	constraint_message	calculation	relevant	required
2	select_one enumerator	enumerator	Select your name (enumerator) from the list.						yes
3	date	date_day	Date of the day		.<=date(today())	The date cannot be in the future!			yes
4	text	name	Name of the animal.						yes
5	integer	age	How old was Steve when he got this pet?		. >= 0 and . <=100	The value must be greater than 0 and smaller than 100.			yes
6	calculate	age_group					if({age} < 5, '0 to 4', if({age} >= 5 and {age} < 12, '5 to 12', if({age} >= 12 and {age} < 19, '12 to 18', if({age} >=19 and {age} < 25, '19 to 24', if({age} >=25 and {age} < 50, '25 to 50', 'over 50 '))))))		
7	note	age_note	Please confirm that Steve was \${age} years old when he got \${name}. This means that Steve was in the following age group at the time: \${age_group}.						
8	select_one especie	species_select	What species of animal is \${name}?						{age} > 0
9	text	species_other	If other, please specify.						{species_select} = 'other'
10	calculate	species					if({species_select} = 'other', \${species_other}, \${species_select})		
11	calculate	label					\${name}		
12	calculate	formids					concat('steve_example_2', ',', 'steve_example_3')		
13	calculate	id					concat('pet', ',', \${species}, ',', \${name})		
14	calculate	users					concat(\${enumerator}, ',', pulldata('list_enumerator_supervisor','supervisor','enumeratortt',\${enumerator}))		
15									
16									

# Registration form - XLSForm :

Once the form is deployed on the server, it must be linked to a cases dataset\*. The field mapping is as follows:

The screenshot shows the XLSForm configuration interface for a dataset named 'cases'. The top navigation bar includes icons for Attach, Publish into, Edit, Upload, Download, Purge data, Delete, and Settings. Below the navigation bar, the dataset information is displayed: 'Dataset ID: cases, Version: 27, Records: 5'. The main content area is titled 'Forms that publish to dataset with ID 'cases''. A specific form, 'Steve\_TEST\_registration' (ID: Steve\_example\_1), is selected. The configuration for this form includes a section for 'Treatment of fields inside repeat groups' with two radio button options: 'Publish to wide format (one row per submission)' (selected) and 'Publish to long format (one row per repeat instance)'. Below this is the 'Field mapping' section, which contains a table with columns for 'Form Field', 'Action', and 'Dataset Field'. Each row represents a field mapping, and each row has a 'Delete' button. The 'users' field mapping is highlighted with an orange L-shaped cursor.

Form Field	Action	Dataset Field	Delete
id	Replace	id	Delete
label	Replace	label	Delete
formids	Replace	formids	Delete
nom	Replace	nom	Delete
species	Replace	species	Delete
age	Replace	age	Delete
users	Replace	users	Delete

# Conclusion scenario - Cases dataset

# Results in the cases dataset

In the *users* column of the cases dataset are the names of the enumerators and his or her supervisor who will be able to see this survey in the case list.

	A	B	C	D	E	F	G	H	I	J	
1	id	label	formids	users	roles	sortby	enumerators	species	name	age	m
2	pet_cat_Test enumerator a3	Test enumerator a3	steve_example_2,steve_example_3	a3, johnny				cat	Test enumerator a3	20	
3	pet_other_Test enumerator a1	Test enumerator a1	steve_example_2,steve_example_3	a1, johnny				other	Test enumerator a1	31	
4	pet_dog_Enumerator a2	Enumerator a2	steve_example_2,steve_example_3	a2, johnny				dog	Enumerator a2	15	
5	pet_cat_Enumerator b1	Enumerator b1	steve_example_2,steve_example_3	b1, mohamed				cat	Enumerator b1	1	
6	pet_dog_Test enumerator b2	Test enumerator b2	steve_example_2,steve_example_3	b2, mohamed				dog	Test enumerator b2	2	
7											

cases dataset

→ For example, for the survey **Test enumerator a3** :

- The **a3** enumerator and his supervisor **johnny** will be able to see this case on the SurveyCTO Collect application.
- The other enumerators (**a1** and **a2**), as well as the supervisor **mohamed** will not see this case on the SurveyCTO Collect application.

# Other SurveyCTO tutorials available

In this toolbox, you will also find the following tutorials:

- **Groups, subgroups** and workspaces
- Centralized management of **roles, users** and **teams**  
→ **Scenario** on the use of teams, groups and roles
- **Encryption of** data collected on SurveyCTO
- **Security settings for** SurveyCTO Collect
- **Submission types** on SurveyCTO
- **Case management** on SurveyCTO
- **Importing** data into SurveyCTO
- **Workflows** of a SurveyCTO form
- **Automated quality checks** on SurveyCTO
- Data **filtering** before downloading on the SurveyCTO server
- **Bulk data deletion** on the SurveyCTO server

# Acknowledgements

This publication is supported by Terre des hommes (Tdh) and the French Development Agency (AFD). Nevertheless, the ideas and opinions presented in this toolkit do not necessarily represent those of Tdh and AFD.

This presentation was designed using resources from [Flaticon](#).

This tutorial is made available under the terms of the [Creative Commons Attribution-Share Alike 4.0 International License](#).

