RX26 · VOLUNTEERS & DATA INTEGRITY IN ROCK RMS

The Blueprint

How to build automated volunteer Rock access — grant it, sync it, and take it away — using data views, group sync, workflows, and jobs. Built and running at LCBC Church.

The idea in one sentence: staff check a box on a serve-group member, and Rock does the rest — collects a signed confidentiality agreement, places the volunteer in the right security roles, creates their login, and removes it all again when they stop using it.

GRANT

1 · Staff checks a box
Access attribute on the volunteer's serve-group membership

→

2 · Agreement
Job + workflow emails a confidentiality agreement to sign

→

3 · Approved = Yes
Signing flips a person attribute that gates everything

→

4 · Group sync
Data views sync the person into security roles

→

5 · Login
Base role sync creates the account and sends credentials

REMOVE

No login in 60 days
Data view finds inactive volunteers with access

→

Job → workflow
Runs every 15 minutes

→

Leave a trail
Note on the profile listing what they had

→

Clear & reset
Clear access attributes, set Approved = No

→

Sync drops them
They fall out of the data views and leave the roles

What you'll build

Piece	Purpose	Step
Data classification	Decide which data volunteers may see before building any roles	0
Security roles (additive)	One base role plus a separate role for each job a volunteer does	1
Group member attributes	The checkboxes staff use to grant access	2
Person attribute: *Approved for Volunteer Rock Access*	Yes/No gate that every sync data view requires	3
Signature document, workflow, job	Collect the confidentiality agreement and flip the gate	4
Data view chain + group sync	Turn checkboxes into security role membership and logins	5
Manual high-trust roles	Finance, communication, and group admin access go through a ticket	6
Offboarding data views, job, workflow	Remove access automatically and leave a record	7

0 Classify your data first

Before you build roles, decide what kinds of data you hold and who may see each kind. The roles then follow from that decision.

Data type	Examples	Volunteers	Staff
General	Contact info, family, serving, first-step data	Yes	All
Care	Care requests, personal struggles	Chaplains only	All
Financial 1	Check scanning, a few giving-related attributes	Yes (finance role)	All
Financial 2	Full giving history and transactions	No	Select
Key metrics	Attendance dashboards, engagement scores	No	Select
Confidential	Confidential notes, church discipline	No	Select
Security	Security flags, background check results, incident reports	No	Select

1 Build additive security roles

Don't put every volunteer in one large "volunteer" role. Give everyone the same small base role, then **add** a role for each job they do. Access grows with the volunteer and is simple to explain.

Role	Who it's for	How it's granted
Base Rock Access Volunteer	Everyone with access. People records and basic data entry.	Checkbox (automatic)
Process Volunteer (one per ministry)	Helps run that ministry's processes (connection requests, attributes)	Checkbox (automatic)
Onboarding (VOP) Volunteer	Moves people through the volunteer onboarding process; sees only that connection type	Checkbox (automatic)
Connection Volunteer	Shared connection-request access for anyone who is a Process or VOP volunteer	Derived (automatic)
Financial Volunteer	Check scanning, offering	Ticket (manual)
Group Admin Volunteer	Manages members of volunteer groups	Ticket (manual)
Communication Volunteer	Sends email communications	Ticket (manual)

> **Example**
> Janice joins a campus team to help with volunteer onboarding. She gets *Base + VOP Volunteer*. A year later kids' ministry asks her to help with more of its processes, so *Process Volunteer* is added. Nothing is removed and nothing extra is granted.

For each Process Volunteer role, sit down with the ministry and decide, for every process, whether a volunteer can **View** it, **Facilitate** it (contact the person on the other side), or **Be Assigned** the request. "Can be assigned" is about accountability: often a volunteer can run the whole process, but a staff member should still own it.

2 Add the access checkboxes to serve groups

- Create **group member attributes** (Boolean) in a category such as *Volunteer Rock Access*: `Base Rock Access`, `Process Volunteer`, `VOP Volunteer`.
- Put them where every serve group gets them, either on a shared parent group type or on each serve group type.
- Add the same attributes to the volunteer **onboarding connection opportunity** so staff can choose access while the person is still being onboarded. The automation starts only after the request is marked *Connected* and the attributes carry over to the group member.
- For a ministry that needs its own process role (LCBC: *The Advantage*), add an extra attribute only on that ministry's serve group.

Design choice that matters

Process access comes from the **ministry that owns the group type**, not from the specific group. A door greeter marked as a Process Volunteer gets all Guest Experience processes. If you want them in Student Ministry processes, they need to be in a Student Ministry serve group.

3 Create the gate attribute

Create a **person attribute** *Approved for Volunteer Rock Access* (Boolean, default **No**). Every person data view that syncs into a security role (Step 5) requires **Yes**. Nobody gets access until it's set, and setting it back to No is how access is removed.

4 Collect the confidentiality agreement automatically

1. **Signature document template**: your data confidentiality agreement, built with Rock's electronic signature feature.
2. **Data view: "Volunteers needing access without agreement"**: group members with any access attribute checked who have *no* signed agreement document *and* no active agreement workflow. The second condition prevents duplicate emails.
3. **Workflow: "Request Confidentiality Agreement"**
   - Emails the volunteer a link to sign.
   - When they sign, the document is saved to the profile's Documents tab.
   - Sets *Approved for Volunteer Rock Access* = **Yes**.
   - If an agreement is already on file, set Yes and skip the email.
4. **Job: Data View to Workflow**: points the data view at the workflow and runs on a short schedule.

5 Build the data view chain and group sync

This is the engine. Each layer filters on the one before it, so you can test each layer on its own and reuse them.

1. Group types	2. Groups	3. Group members	4. People	5. Security role
All serve group types per ministry e.g. "Kids Min Group Types"	All groups in those group types "Kids Min Volunteer Groups"	Members of those groups with the attribute checked "GM – Kids Min Process Volunteer"	People in layer 3 with Approved = Yes "Add to Security Group – Kids Min Process"	Group sync from the layer 4 data view "Kids Ministry Process Volunteer"

- Build layers 1–3 **per ministry** for Process Volunteer, plus one VOP layer across all ministries and one **Connection Volunteer** layer (anyone with Process *or* VOP checked).
- On each security role, add a **Group Sync** using the layer 4 data view. Members are added when they enter the view and removed when they leave it.
- The **Base Rock Access** person data view includes everyone with any access attribute checked and Approved = Yes.

Logins and welcome email (Base role only)

On the Base role's group sync, turn on **creating user accounts during sync** and set a **welcome communication**. New volunteers get a username and a temporary password they must change at first login. Write the template so that volunteers who already have a login get a reminder of their username and a reset link instead.

Set expectations

With data view persistence and sync schedules, it can take a couple of hours from signing to having a working login. Tell your staff so they don't open a ticket after five minutes.

6 Keep high-trust roles manual

Finance, Communication, and Group Admin roles are **not** on the checkbox. Staff ask for them with a help desk ticket, and an admin adds the person to the role directly. These roles can move money, send email to many people, or change

rosters, so a person should approve each one.

7 Automate offboarding

Rule: a volunteer with Rock access who hasn't logged in for **60 days** loses it.

1. **Data view: "Volunteers in access roles"**: members of any volunteer data-entry or finance security role.
2. **Data view: "People with Rock interactions (60 days)"**: page interactions on your internal Rock site in the last 60 days.
3. **Data view: "Finance workers actively serving (60 days)"**: finance volunteers who served recently but may not log in often. Leave them out so they keep access.
4. **Data view: "JOB – Volunteer Offboarding"**: (1) *not in* (2) *and not in* (3), and **not flagged as a system user**.
5. **Job: Data View to Workflow**, every 15 minutes, launches the offboarding workflow.

The offboarding workflow

1. **Record first:** read the person's access attributes from each serve group and add a **note** to the profile listing what was removed, so staff can restore it later.
2. **Clear** each access attribute on each serve group membership.
3. Set *Approved for Volunteer Rock Access* = **No**.
4. **Email** the volunteer to tell them access was removed and how to get it back.

You don't need to remove anyone from a role by hand. Once the attributes are cleared, the person leaves the data views and group sync removes them from the roles. Manually added roles (Step 6) aren't synced, so either remove them in the workflow or have staff restore them by ticket using the note.

> **Automation doesn't replace staff judgment**
> The 60-day rule catches people who drift away. When staff know a volunteer has stopped serving, they should uncheck the attributes right away instead of waiting 60 days.

Build checklist

- [ ] Data types classified and signed off by leadership
- [ ] Base role security set to people records and data entry only
- [ ] One Process Volunteer role per ministry, plus VOP and Connection roles
- [ ] Ministry process grid (View / Facilitate / Assign) agreed
- [ ] Access attributes on serve groups and the onboarding opportunity
- [ ] Person attribute "Approved", default No
- [ ] Signature template, agreement workflow, and job
- [ ] Data view chain built and checked layer by layer
- [ ] Group sync on every automatic role
- [ ] Base role sync creates logins and sends the welcome email
- [ ] Manual-role request path set up in your help desk
- [ ] Offboarding data views, job, and workflow
- [ ] Rule: no minors with Rock access
- [ ] Staff guide published (how to grant, what to expect, how to restore)

> **Gotchas we hit**
> - **Duplicate agreement emails**: exclude people with an active workflow, not only people without a signed document.
> - **Finance volunteers getting offboarded**: they serve every week but rarely log in. Exclude recent serving activity.
> - **System and integration accounts**: always exclude system users from offboarding.
> - **Restoring access**: if the profile note didn't exist, nobody could tell what a returning volunteer used to have. Write the note before clearing anything.

Kevin Wishard · Digital Engineering Team, LCBC Church · RX26 · Adapt names, timing, and roles to your church. Check group sync options against your Rock version.