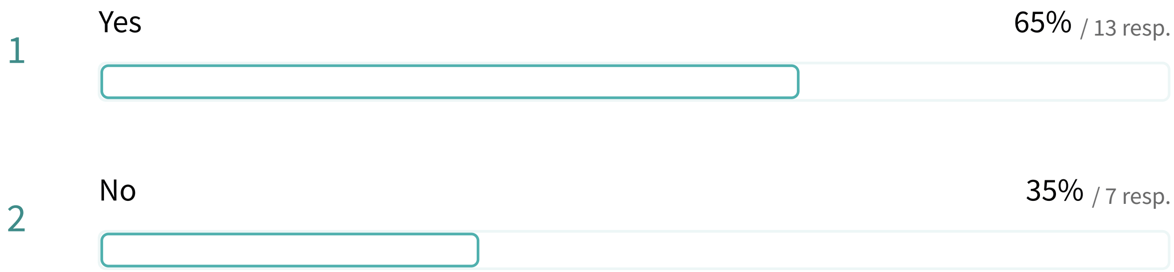
FIM Survey

20 responses

Does your organisation use FIM in OpenStack?

20 out of 20 answered



The FIM systems used in OpenStack are:

Openstack-Horizon-Shib	1
SAML/Shibboleth	7
OIDC	3
Azure/AD	3
Keycloak	2
OpenAM	1

(the total is more than 13 because some respondents gave multiple answers)

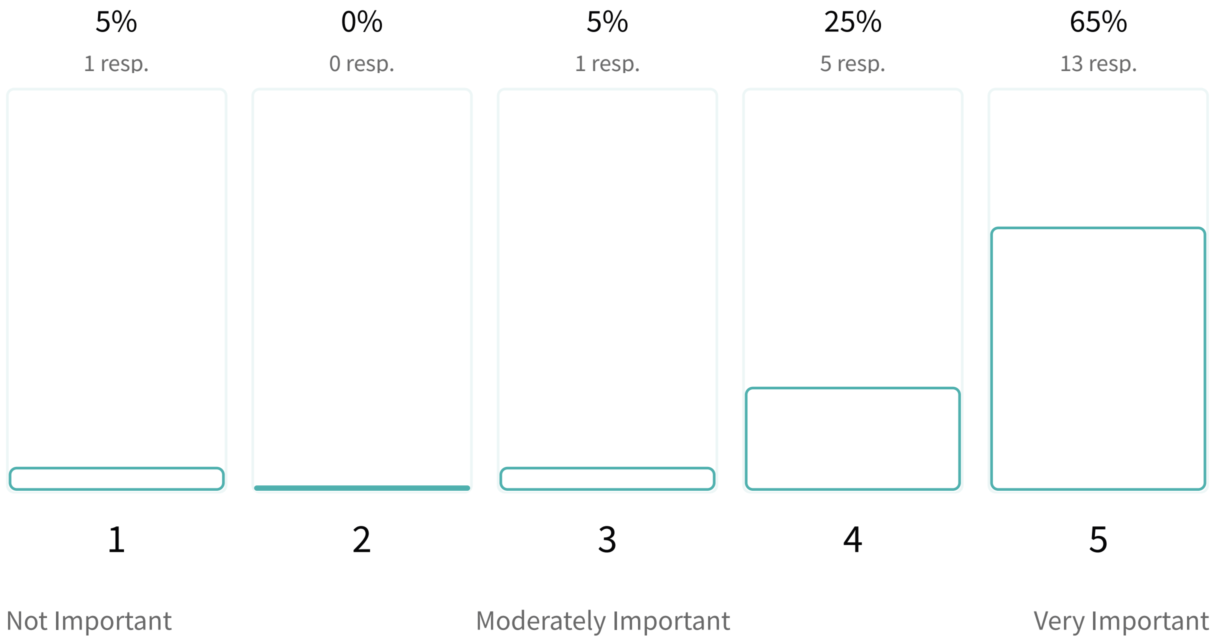
Those who do not use FIM with Openstack use:

Local authn/creds	6
Don't use Openstack	1

How important is the inclusion of FIM in OpenStack to you?

20 out of 20 answered

4.5 Average rating



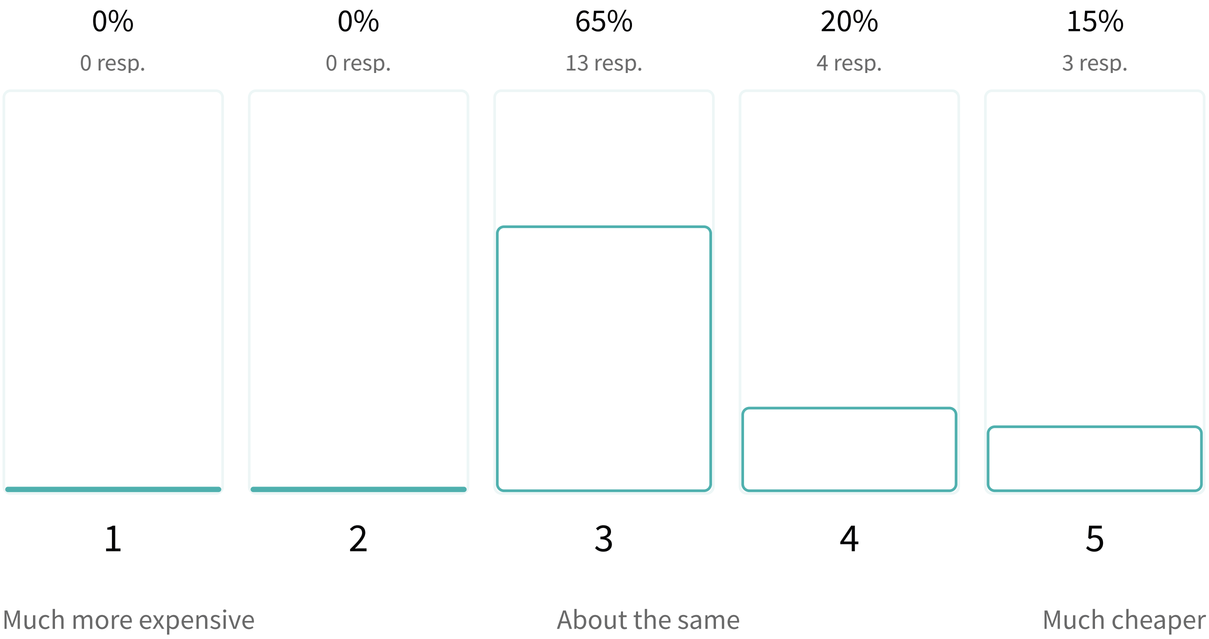
Why is FIM important to the 13 organisations that use it?

1. FIM frees us from the need to maintain users' database.
2. The service is to be used outside of the organization.
3. Validation of user accounts by home institution.
4. We need our keystone to consume identities, and we want to use current generic web authentication workflows for all our services.
5. AAI is a requirement in our user community.
6. Providing services to plenty of organisations.
7. We manage accounts via other central entities and operate multiple OpenStack clouds that use this shared identity. For simplicity of deployment we like provisioning users/projects automatically via federated users/groups.
8. Impractical to maintain separate identity (system) at this org size.
9. Allows us to verify the identity of users when they log in with their identity provider and gets us out of the business of managing accounts and passwords.
10. (We are a) large organization with a large number of independent Open Stack Installations (>50)
11. We want to have one central identity provider.
12. (It is) needed by most services.
13. We have a large community of users, it would be overkill to manage the user registration, accounting, passwords, etc. (and) we need to be sure users are coming from R&E organizations.

What are the financial implications to your organisation of adopting FIM compared to your existing (or previous) non-FIM system?

20 out of 20 answered

3.5 Average rating

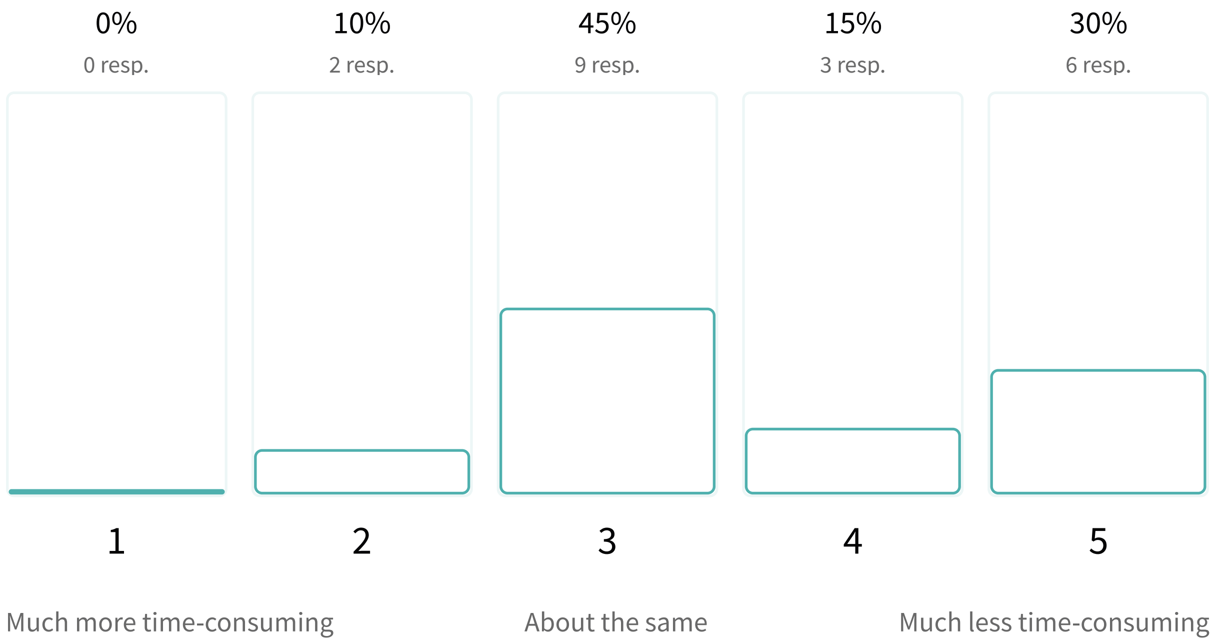


Most respondents could not put a financial figure on the cost implications of adopting FIM. Of those that said FIM was much cheaper, only one commented with "You can only scale big using FIM"

What are the time implications for your users of adopting FIM compared to your existing (or previous) non-FIM system?

20 out of 20 answered

3.6 Average rating



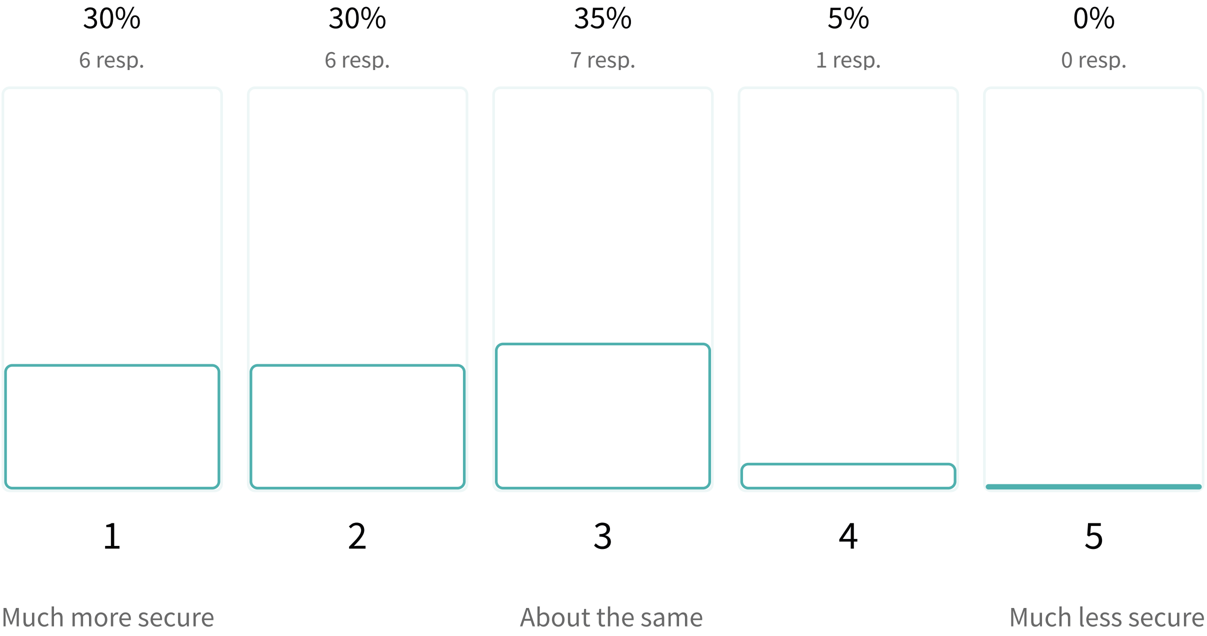
Of those that said FIM was less or much less time consuming, the following comments were received:
1. Registration time is just few clicks wrt alternative procedure for identity verification
2. Users don't need to create another account, just use the one they already use on a daily basis.
3. Earlier separate registration and approval. Now it is straightforward
4. Well, they don't have to create a new account, so there's that.
5. Multiple Operation people were spending >30% of their time just managing accounts before we implemented hooking into corporate Active Directory System.

Of those that said FIM was more time consuming, the following comments were received:
1. Forcing users to use password authn with LDAP severely limits usability.

What are the security implications of adopting FIM compared to your existing (or previous) non-FIM system?

20 out of 20 answered

2.1 Average rating



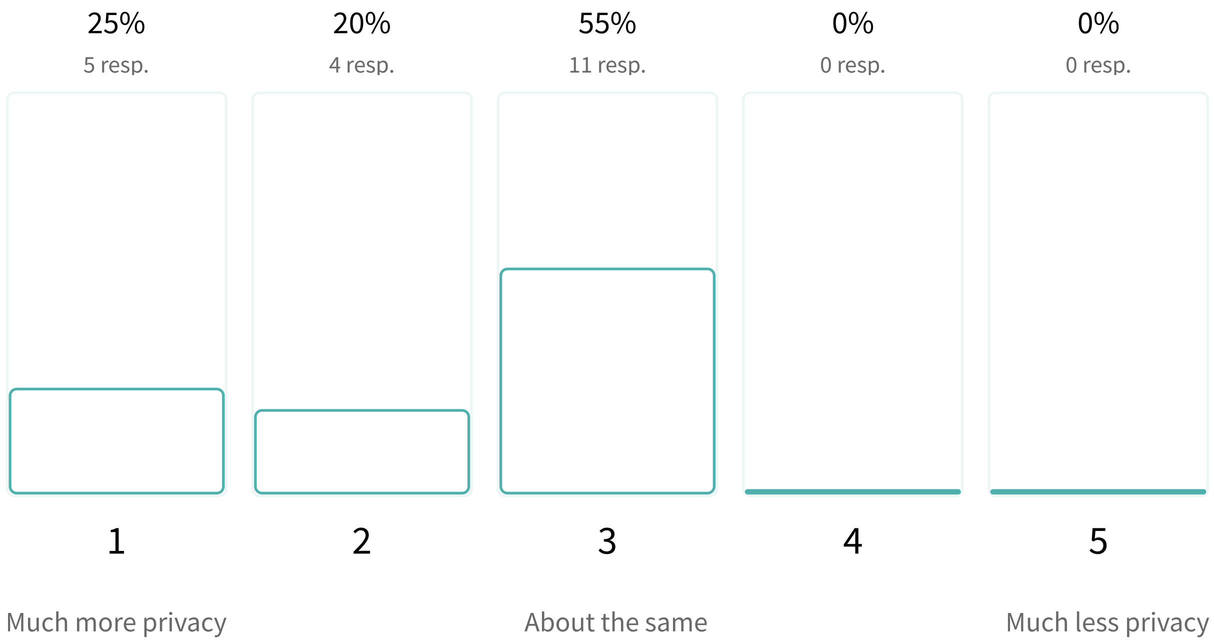
Of those that said FIM was more or much more secure, the following comments were received:
1. We can audit access controls from a central system and we can easily require MFA for user logins.
2. Don't have to manage safety of credentials, since we store none. We are not subject to brute-force, since services are accessed via identity tokens only.
3. R&S and Sirtfi helps. Also using the institutional accounts helps with traceability
4. Revocation is central. Centralized group management assignment are key as well
5. When a user leaves the home organisation, and his account is deactivated, this user cannot log in anymore in the open stack environment. Effectively, his permissions are revoked while openstack maintainers don't know (and now don't care) that he is gone.
6. It's easier to conform to standard organizational security processes.
7. Account management (expiration etc.) is handled by LDAP semi-automatically
8. We will get to know who accesses what and when
9. Managing identity universe is hard
10. Everything which gets us out of the business of [managing] passwords. Also we can offload verifying a user to the identity providers they are coming from.
11. Old account remove quicker.
12. No passwords inserted in web forms, no passwords kept in the keystone, no recovery/force-passwords-on-the-phone

Of those that said FIM was less secure, the following comments were received:
1. Less control on resources

What are the privacy implications of adopting FIM compared to your existing (or previous) non-FIM system?

20 out of 20 answered

2.3 Average rating



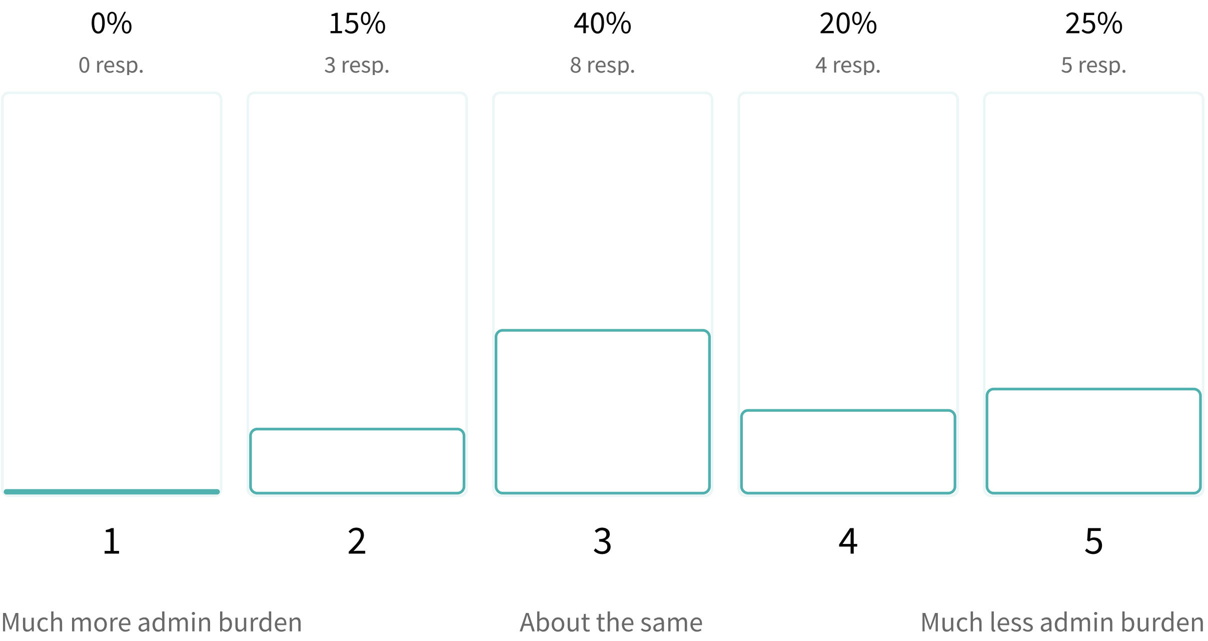
Of those that said FIM was more or much more privacy protecting, the following comments were received:

1. No private information is stored on our servers.
2. Using only one set of credentials helps
3. Single point of enforcement for policy is key
4. We do not store anything more than an r&s category. Moving to a pure Keystone FIM might allow us to store less than that, which might be nice.
5. User password is not stored in 2 different DBs
6. Do not need to collect as much information about users
7. Token is used to access. We provide access to departments.

What are the administrative implications of adopting FIM compared to your existing (or previous) non-FIM system?

20 out of 20 answered

3.5 Average rating



Of those that said FIM was more or much more of an administrative burden, the following comments were received:
1. Admins have to know how to use the FIM and Keystone federation instead of just updating Keystone state manually.
2. Cost
3. When registration to our cloud platform is performed through Federated Auth we already know if a user is eligible or not, which is a great advantage from an administrative POV.
[Ed. we suspect this respondent should have selected Less rather than More burden, and that this comment should be in the group below]

Of those that said FIM was less or much less of an administrative burden, the following comments were received:
1. Lots of administration is delegated to institutions
2. Maintaining the federated integrations is more admin work, but the manual process work is reduced in central user management.
3. Do not need to manage credentials for federated users (password resets, etc)
4. No manual approval (needed)
5. No more account management, users shall do it for us
6. A unified authentication management has allowed us to let users use the same accounts for OpenStack and OpenShift, which we are operating. So we and users don't have to deal with multiple accounts or systems of access management.

Of those that said FIM was about the same administrative burden, the following comments were received
1. We'll still be responsible for user lifecycle, but we won't need to handle user verification. We will still need to help users access the service, complicated by the IdP discovery process.
2. Sometimes need to interact with IdP managers for properly releasing needed attributes. Inclusion of new IdPs requires (simple) procedure.
3. Running of the service is not really related to FIM (administrative wise)
4. There is more gravitas to account creation than before -- a good thing, thus more thought involved on provisioning of services and their role driven nature to scale things up