

# Exporting and Importing Realms

Elestio enables seamless migration of Keycloak realms by supporting realm exports and imports. This capability is vital for backing up configurations, replicating environments, or transitioning between staging and production systems. The process ensures consistency across deployments while preserving all realm-level resources such as users, roles, groups, clients, and identity providers.

## Key Steps for Exporting and Importing

### Pre-Migration Preparation

Before initiating realm export or import, it's essential to prepare both the source and target environments to ensure compatibility and prevent data loss:

- **Create an Elestio Account and Deploy Keycloak**

Sign up at [elest.io](https://elest.io) and deploy a Keycloak instance. Ensure the Keycloak version in the target environment matches the source to avoid compatibility issues during import.

- **Backup Existing Configuration**

Always create a snapshot or export of the existing realm configuration before starting. This ensures a rollback path in case of issues during import.

- **Verify Resource Limits**

Confirm the Elestio service has adequate CPU, RAM, and storage to accommodate the imported realm data, especially when dealing with large user bases or multiple clients.

### Exporting a Realm

Keycloak provides CLI-based tools and startup parameters to export realm configurations. Elestio supports these via custom startup commands.

- **Export Using kcadm.sh (CLI)**

```
/opt/keycloak/bin/kcadm.sh config credentials --server http://localhost:8080 --realm master --  
user admin --password <your-password>  
  
/opt/keycloak/bin/kcadm.sh get realms/<realm-name> > myrealm-export.json
```

This method exports the realm configuration to a JSON file.

- **Export Using Environment Variable Method (Preferred on Elestio):** You can configure the container to perform a full export on startup:

```
KEYCLOAK_IMPORT=/opt/keycloak/data/import/myrealm-export.json
```

And use the following command:

```
/opt/keycloak/bin/kc.sh export --dir /opt/keycloak/data/import --realm <realm-name> --users  
realm_file
```

This will export the full realm configuration including users, clients, and roles into the myrealm-export.json file.

- **Download the Export File**

After the export completes, use the Elestio dashboard or scp/rsync to download the exported JSON file from the container.

## Importing a Realm into Elestio-Hosted Keycloak

Once the realm has been exported and downloaded, follow these steps to import it into your Elestio-hosted Keycloak instance:

- **Upload Exported JSON File:** Place the exported file in a volume accessible to the Elestio container (e.g., under `/opt/keycloak/data/import/`).
- **Configure Import Environment Variable:** In the Elestio dashboard, go to your Keycloak service → **Settings** → **Environment Variables**, and add:

```
KEYCLOAK_IMPORT=/opt/keycloak/data/import/myrealm-export.json
```

- **Trigger Import at Startup:** Elestio will automatically import the realm during the next container restart. To do this:
  - Click **Restart Service** from the Elestio dashboard.
  - Monitor logs in real-time to ensure the import process completes successfully.

## Post-Import Validation and Optimization

After importing the realm into your Elestio-hosted Keycloak instance, perform the following steps

- **Validate Realm Components:** Confirm all users, roles, groups, clients, and identity providers have been imported. Use the Keycloak Admin UI or kcadm.sh CLI to inspect the imported realm.
- **Test Application Authentication Flows:** Update client application configurations if needed. Confirm login, token exchange, and logout flows work as expected using the new

realm setup.

- **Review Access Tokens and Certificates:** Ensure keys and token lifespans are properly configured. Replace any expired or incompatible certificates.
- **Enable Monitoring and Backup:** Use Elestio's built-in monitoring tools to observe performance and usage. Schedule regular backups from the dashboard to ensure data protection.
- **Apply Security Best Practices:** Rotate admin credentials. Set up IP whitelisting and firewalls via Elestio. Review and assign minimal privileges to users and service accounts.

# Benefits of Using Elestio for Realm Management

- **Simplified Automation:** Elestio automates backup, monitoring, and scaling, removing manual overhead from managing Keycloak instances.
- **Secure by Default:** Instances are provisioned with firewalls, encryption, and unique passwords. Elestio keeps Keycloak up to date with critical security patches.
- **Scalable and Portable:** Realms can be exported and imported across environments with ease, enabling multi-region replication, staging-to-prod transitions, and more.
- **Performance Optimized:** Instances are pre-tuned for performance. Elestio supports scaling CPU, RAM, and volume size based on identity workload.

---

Revision #2

Created 16 June 2025 06:15:59 by kaiwalya

Updated 16 June 2025 06:25:05 by kaiwalya