PrivX® Secrets Vault - Your stronghold for secrets that need to be managed

PrivX® Lean privileged access management

What is PrivX?

PrivX is lean, quick-to-implement and easy-to-use access management software for privileged access to on-prem and cloud environments. PrivX helps you to enable and control access to servers, network devices and other critical infrastructure according to user roles and privileges.
Secrets are easy to create but hard to keep

IT environments are full of secrets, including tokens, passwords, certificates and encryption keys. They open access to mission-critical information, which makes managing and securing these secrets a top priority. At the same time, IT environments consist of a mix of technologies and are at very different stages of maturity: some of them are much further along the way towards cloudification or even containerization whereas others are run mainly by physical servers. Most of them are somewhere in between in a hybrid stage.

In our dealings with our customers, we've learned that there's no silver bullet method for securing access secrets, but the best approach depends on the use case.

To vault or not to vault – that is the question

SSH.COM’s PrivX was designed with a clear vision in mind. The easiest way to manage secrets is when you don't have to manage them at all. That is why PrivX’s core idea was built around just-in-time access tokens - or ephemeral certificates - that are created on the fly at the time of establishing the connection and that expire automatically after authorization. What’s more, the user never sees or handles any secrets during the process, which is great for security.

The really unique part is that this approach leaves no secrets behind to vault or manage, freeing companies from traditional secrets management altogether. In fact, PrivX turns privileged access management (PAM) passwordless for many use cases, since by integrating PrivX with identity and access management systems (IAM), users just single sign-on (SSO) to their targets without having to stress about credentials. This is not only a security boost but makes day-to-run operations run smooth as silk.

Temporary access tokens and ephemeral access tokens are a great fit for many multi-cloud and hybrid environments. They allow enterprises align with Zero Trust and Zero Standing Privileges frameworks where you keep your environment free from risky credentials, reduce the complexity of managing them and adopt passwordless and credential-less paradigms.

It might seem contradictory that when we promote our vault, we recommend our customers to go vaultless and passwordless whenever possible. However, not all customer environments or use cases support this approach. That is why PrivX also offers other options, such as Vault, where access secrets can be stored, secured and retrieved when needed.

So when should our customers vault their secrets instead of using just-in-time access tokens that leave no secrets behind?
Typically, storing secrets has been associated with interactive access and privileged access management (PAM). However, automated file transfers, like application-to-application (A2A), machine-to-machine (M2M) or robotic process automation (RPA) communication, easily outnumber those made by humans.

To illustrate the point, let’s take a closer look at applications. Applications have identities but their access credentials and privileges are often all over the place – and they typically exist outside IAM or directory services like Active Directory (AD).

Moreover, automated applications might be generating thousands of requests per second as opposed to interactive sessions, where the rate is often significantly lower.

Some of the key points to consider include:

- How many applications (for example, web servers) connect to your databases?
- What level of access do those applications need?
- If it’s another type of target, like the Amazon Simple Storage Service (Amazon S3) object storage service, is it a read, use or write type of access?
- What types of secrets grant access to the targets?
- Can you track and audit their sessions if needed?

For database access, your application needs a username and a password. For S3 access, it needs an API token. If we’re dealing with server-to-server access, it might need an SSH key. That is a lot of variation already just based on the target.
Why application secrets need vaulting?

In the past, many environments were considered high-trust. Your IT team set up a security perimeter, inside which automated file transfers kept soldiering on. The required credentials might have been in plaintext format, hardcoded in the application, embedded in configuration files, part of configuration management or integrated with version control.

Even back in the day, this was a dubious approach, since valuable secrets were basically up for grabs inside the security perimeter. But you might have been able to get away with it, since only the initiated got access to your high-trust network.

Nowadays, enterprises are connected everywhere and we live in the age of Zero Trust, where no access should be trusted by default but be verified each time it’s established. The traditional security perimeter is gone, since many critical maintenance, upgrade and update task have been outsourced and applications are in contact with each other globally.

There is a genuine need for solution that manages all credentials & secrets, ensures that they are used securely, puts them under proper access controls and ensures the right level of privilege per application access – typically minimal amount of privilege to get the job done.

PrivX Secrets Vault does just that.

When to vault interactive session secrets?

Again, our recommendation is rather simple: if you can use automatically expiring and just-in-time generated ephemeral certificates. If you cannot, PrivX Secrets Vault is your secure access stronghold for managing secrets for tools that are used in software and infrastructure deployment, testing, orchestration, and configuration.

Furthermore, environments and technologies evolve, so you can start migrating from a vault-based approach to a passwordless and vault-less paradigm when you modernize your environment, when you are ready for it, and always at your own pace.

We want to offer our customers a migration path to the passwordless world.
Don’t leave secrets exposed

Many critical tasks like automated file transfers or DevOps development cycles still operate outside proper identity governance and administration (IGA), leaving secrets noncompliant or exposed for misuse. With PrivX Secrets Vault, you onboard those secrets into a vault and mitigate risks.

Protect all types of secrets

Secure, protect and control access to
- tokens
- certificates
- passwords
- encryption keys
- pieces of code

that are used through a UI, CLI or HTTP API to access sensitive or mission-critical data.

Easily manage non-interactive access with proper level of granularity

Bring automated connections, like application-to-application (A2A), machine-to-machine (M2M) Robotic Process Automation (RPA) under role-based access controls (RBAC). Make access lifecycle management fast and easy by associating non-human identities with a role. Manage their access and their level of privilege based on a dozen of roles instead of per identity – the number of which is measured in the thousands.

Keep your identities and roles in sync automatically

Integrate machine identities and human identities alike to the trusted identity provider of your choice. PrivX Secrets Vault automatically stays in sync with any changes in identities, making sure that the right identity is associated with the right role or that access rights are revoked automatically – and almost in real time – if the identity is removed from your IAM.
PrivX Secrets Vault benefits

Audit and track sessions
Stay in control of all activities in your environment by auditing and tracking each session whether they are interactive or non-interactive.

Centralize access and workflow processes
Put your access management under a single pane of glass with our multi-tenant vault. Use consistent, centralized and well-defined workflows to manage secrets and access that they enable.

Personal password manager
Allow your privileged users to store their own secrets into the vault, using it as their secure personal password manager for highly valuable secrets.

Enable collaborative secrets management
Enable collaborative secrets management based on a role for privileged users instead of them sharing secrets or using hard-coded credentials. We’ve included ready-made templates to help you get started.
Use cases

1
Interactive UI access for environments not supporting certificates
The PrivX Secrets Vault is based on role-based access controls (RBAC). The user logs in to PrivX and always sees an up-to-date list of accessible targets based on his or her role. If the target requires a password, the user retrieves the masked password from the vault and enters it to PrivX when prompted. The secrets available to the user are always limited by the role and the level of privilege associated with that role.

2
Interactive command line interface (CLI) access in DevOps
For those contexts where a command line interface is needed, PrivX Secrets Vault ensures that secrets are accessible through an easy-to-use CLI tool. For example, for DevOps teams this means that instead of the DevOps team sharing or using hard-coded secrets assigned for individual users, they can use a centralized Vault where these secrets are stored.

3
A2A, M2M and RPA access
PrivX Secrets Vault enables application-to-application (A2A), machine-to-machine (M2M) Robotic Process Automation (RPA) communication without risky, hard-coded target system credentials or credentials embedded in the code.

Instead, these credentials are stored in the Vault, which the non-human identities then access through a REST API to enable their automated connections. With the Vault, you can centrally manage, control, track and audit these processes and their secrets as needed.

4
Collaborative admin secrets management for critical IT infrastructure access
Use PrivX Secrets Vault as a centralized safe for a team of administrators to store encryption keys, access tokens or breakglass credentials. These can be used to access web applications, network devices, databases or servers. No need to share secrets, and Vault is the only location where the secret has to be updated. Vault, you can centrally manage, control, track and audit these processes and their secrets as needed.

5
Digital safe for breakglass credentials
Use a single PrivX Secrets Vault instance as a centralized safe for critical secrets to be handled with extraordinary care, like breakglass credentials. You can isolate the instance from your IAM integrations and set specific rules (like firewall or VPN configurations) on how these secrets can be accessed even if network connections are down.
## Key features

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>The secrets vault is a micro service</td>
<td>Stores the secret data in encrypted format along with the metadata in the database. Internally uses the keyvault API for performing secret encryption and decryption.</td>
</tr>
</tbody>
</table>
| Role-based access control to target hosts   | • Users can be dynamically mapped to roles  
• View hosts that are accessible by specific roles                                                                                         |
| Context-based roles                         | Restrict access based the day of the week, time of the day or IP address for specific roles.                                                                                                               |
| Restrictions based on access groups         | Restrict access based on access groups that can be, for example, departments or functions inside an organization. Even as an admin, you only get access to the area of the network your department is entitled to access. |
| Directory service integration               | Users and groups synced with Microsoft AD, Azure AD via Graph API, Google G Suite, LDAP and OpenID Connect providers (e.g. AWS Cognito, Okta, Ubisecure)                                                      |
| Sign-in and access control to PrivX         | • Single sign on (SSO) through directory services applications via Kerberos  
• Username & password for local users  
• Multi-factor authentication (MFA), time-based one-time password (TOTP) e.g. Google Authenticator, Duo, Authy |
| Authentication to target hosts               | • OpenSSH certificate  
• Virtual Smart Card for RDP  
• Stored, vaulted credentials  
• Username & password                                                                                                                  |
| Supported protocols                         | SSH (v2), RDP, HTTP(S) and SFTP                                                                                                                                                                          |
| Fast and responsive user experience         | HTML5 single page UI over REST APIs                                                                                                                                                                       |
| Complete HTTP REST API                      | Anything the UI does can be executed via the API                                                                                                                                                          |
| Support for Thales Vormetric Data Security Manager (DSM) | Integrate with a centralized security management solution                                                                                                                                              |