Office of the Chief Information Security Officer
Risk Advisory: Managing Secrets

This document is intended to provide general guidance to UW organizations in understanding secrets and appropriate strategies for keeping them secure.

First, what exactly are *secrets* and what is typically done with them?

A secret is any information to which you want to control access. These can be (but are not limited to):

- Passwords
- Tokens
- Certificates
- API keys

Typically, you'll want to be able to *securely* do the following with your secrets:

- Store them
- Access them
- Share with others
- Be able to change them

Controlling secrets

Properly managing your secrets is a key element to the overall security of your organization. To help assess how best to do this within the context of your organization, consider both your people and your technologies.

Consider people

- Who knows or has access to your secrets?
- What if one of those people leaves the organization?
- What if one of those people has their credentials compromised?
- Consider the risks of shared accounts

Consider technologies

- Are you storing (hard-coding) secrets directly in your app's source code? (Please don't.)
- If a file is storing secrets, are its (and its parent directories') permissions set correctly?
- Did you use email to send secrets or files containing secrets? (Again, please don't.)
- Are you sending files with secrets to an insecure location (maybe a source code repository or other cloud storage location)?
- Are files with secrets being stored as part of a virtual machine image?
- Are you authenticating over an unencrypted protocol?

Tools

Fortunately, many tools have emerged to assist people and organizations in the secure management of secrets.

Password managers

Password managers are essentially digital vaults which are protected by a single password or passphrase (as well as two-factor authentication) and they offer many advantages.

- Will generate long, unique, random passwords
- Eliminate the need for password reuse
- Can typically store notes, attachments, and other sensitive info in addition to credentials
- Often provide the ability to share secrets securely
- Some examples include: LastPass (LastPass Enterprise is available to all UW employees), 1Password, Dashlane, Keeper, etc.

Managing secrets at scale

Using a 3rd party solution for larger-scale secrets management may be advisable given your organization's needs.

- Provide a way to securely provision apps with programmatic access to necessary secrets
- Make secrets management (such as rotating/changing) considerably easier (and possibly less error-prone)
- Some examples include: AWS Secrets Manager, Microsoft Azure Key Vault, and HashiCorp Vault, etc.

Code scanners

There are now tools/built-in functionality that can be deployed to provide automatic scanning of code and code repositories for accidental inclusion of secrets. A few examples include:

- GitHub secret scanning
- GitGuardian, for scanning Python
- Deadshot, a pull request secret scanner

Additional resources

- Best practices for storing API keys
- Storing secrets in a code repo with Stack Exchange's Black Box