Steal Application Access Token, Technique T1528 - Enterprise

Archived: 2026-04-02 12:28:42 UTC

Adversaries can steal application access tokens as a means of acquiring credentials to access remote systems and resources.

Application access tokens are used to make authorized API requests on behalf of a user or service and are commonly used as a way to access resources in cloud and container-based applications and software-as-a-service (SaaS).[1] Adversaries who steal account API tokens in cloud and containerized environments may be able to access data and perform actions with the permissions of these accounts, which can lead to privilege escalation and further compromise of the environment.

For example, in Kubernetes environments, processes running inside a container may communicate with the Kubernetes API server using service account tokens. If a container is compromised, an adversary may be able to steal the container’s token and thereby gain access to Kubernetes API commands.[2]

Similarly, instances within continuous-development / continuous-integration (CI/CD) pipelines will often use API tokens to authenticate to other services for testing and deployment.[3] If these pipelines are compromised, adversaries may be able to steal these tokens and leverage their privileges.

In Azure, an adversary who compromises a resource with an attached Managed Identity, such as an Azure VM, can request short-lived tokens through the Azure Instance Metadata Service (IMDS). These tokens can then facilitate unauthorized actions or further access to other Azure services, bypassing typical credential-based authentication.[4][5]

Token theft can also occur through social engineering, in which case user action may be required to grant access. OAuth is one commonly implemented framework that issues tokens to users for access to systems. An application desiring access to cloud-based services or protected APIs can gain entry using OAuth 2.0 through a variety of authorization protocols. An example commonly-used sequence is Microsoft's Authorization Code Grant flow.[6][7] An OAuth access token enables a third-party application to interact with resources containing user data in the ways requested by the application without obtaining user credentials.

Adversaries can leverage OAuth authorization by constructing a malicious application designed to be granted access to resources with the target user's OAuth token.[8][9] The adversary will need to complete registration of their application with the authorization server, for example Microsoft Identity Platform using Azure Portal, the Visual Studio IDE, the command-line interface, PowerShell, or REST API calls.[10] Then, they can send a Spearphishing Link to the target user to entice them to grant access to the application. Once the OAuth access token is granted, the application can gain potentially long-term access to features of the user account through Application Access Token.[11]

Application access tokens may function within a limited lifetime, limiting how long an adversary can utilize the stolen token. However, in some cases, adversaries can also steal application refresh tokens[12], allowing them to obtain new access tokens without prompting the user.
