GeForce NOW SDK:
Cloud-SDK V2

API Reference and Integration Guide

Document History

SDK-GFN-004_v1.0

Version	Date	Description of Change
1.0	*04/15/2022*	*First release*
2.0	07/08/2022	Rework doc for new gfnIsRunningInCloud design
2.1	10/27/2022	Added note about in-stream patching

Introduction

The Cloud SDK API is a part of the GeForce NOW (GFN) SDK which allows clients to perform certain checks to determine if an application or game is running inside the restricted GFN cloud environment during streaming sessions.

Audience

This document is directed towards various groups that are interested in the security and integrity of applications and games running in the GeForce Now environment:

- Publishers, those that are game launcher/store application developers leveraging GeForce NOW in their application to stream content via seamless UI.
- Game Developers, those that are developing games to be streamed via GeForce NOW.
- Security Developers, those that are developing applications and tools that enforce the integrity of games being streamed via GeForce NOW.

The APIs defined in this document allow this audience to invoke the SDK from inside the GeForce NOW streaming hosts to perform specific actions related to assuring the game invoked in this environment are running as intended.

Overview

This document contains a high-level overview of the APIs, as well as the interfaces of the APIs. In addition, a section on best practices for good application and game security is included, which advises integrators on optional additional security measures they can take to defend against tampering by malicious actors.

Key Concepts

The GeForce NOW platform is comprised of several major components:

- The end user’s system (client system).
- The GeForce NOW client software on the user’s system.
- The GeForce NOW cloud-based backend services.
- The GeForce NOW streaming servers.
- The GeForce NOW streaming Virtual Machines (VMs) running on the servers.

The nature of the user’s system entitles the user to full control over it; the vast majority of computer users are administrators to their systems. This entitlement allows the user to alter their computer state to change their online gaming experience in ways that are advantageous to the user, often resulting in a negative experience for other players. Historically, game developers have tried to combat such scenarios in multiple ways:

- Detection of abnormal play on the secure server hosting the game session.
- Detection of tampered files or rogue injections into the game process on the user’s system.

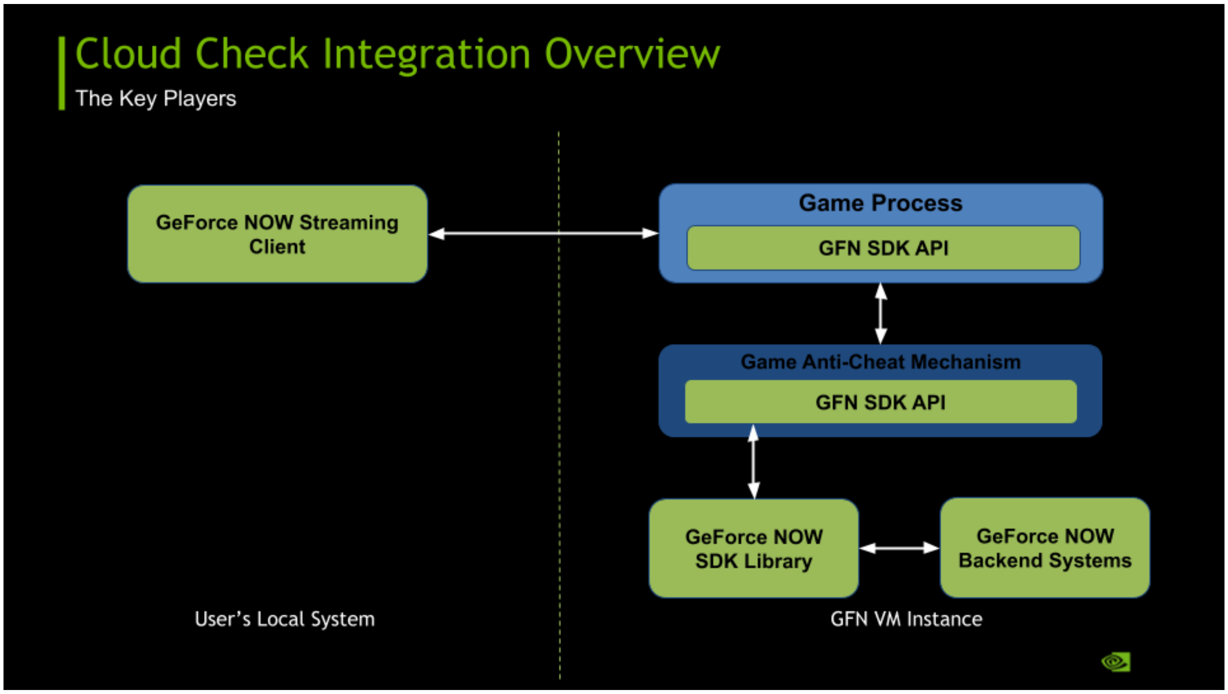
When the same game is hosted in the GeForce NOW platform, the user’s system and the user’s control over the game binaries and processes are taken out of the equation. GFN’s emphasis on security and tamper resistance on the platform means it is unlikely that the system can be compromised by the GFN user. Therefore, certain algorithms for detecting tampered files or rogue injections can be considered unnecessary, and any performance impact they have on the game is unnecessary as well.

Developers of these detection mechanisms now have the ability to leverage the GeForce NOW SDK to securely query when they are running in the GFN streaming environment to make an high-confidence decision when these mechanisms can be disabled and improve the game’s experience for the current user while not risking a negative experience for other users.

To simplify integration and use of the APIs described in the document, the GFN SDK includes a C-based interface via a single source file, *GfnRuntimeSdk_Wrapper.c*, as well as a matching header file, *GfnRuntimeSdk_Wrapper.h*. These sources perform the heavy lifting of:

1. Locating the GFN SDK Library available in the GFN VMs.

2. Verifying the GFN SDK Library is genuine by validating its digital signatures.
3. Loading the GFN SDK Library just-in-time.
4. Calling appropriate APIs defined in the GFN SDK Library.
5. Releasing the GFN SDK Library at the end of the API call.



Cloud Check Integration Overview

The Key Players

GeForce NOW Streaming Client

Game Process

GFN SDK API

Game Anti-Cheat Mechanism

GFN SDK API

GeForce NOW SDK Library

GeForce NOW Backend Systems

User's Local System

GFN VM Instance

To make use of this API stack, simply merge the GFN SDK source files and headers into your source project(s), and call the appropriate API functions as defined in this document.

API Reference

The table below lists all methods related to the Cloud SDK API.

Method	Description
GfnIsRunningInCloud	Determines if running in the GFN Cloud environment
GfnIsRunningInCloudSecure	Determines if running in the GFN Cloud environment in a more secure/restricted fashion

GfnIsRunningInCloud

The *GfnIsRunningInCloud* API function makes fast queries to GeForce NOW systems to determine if the application looks to be running inside the GeForce NOW cloud game streaming environment.

Common Use Cases

The *GfnIsRunningInCloud* API is used for fast environment checks to allow an application or game to determine when running in GFN with a certain degree of confidence. The API is not hardened against exploits such as spoofing of the environment by design, as this API trades off tamper-protection for query speed. The results of this API call can then be used to quickly alter application or game logic in ways that are not critical to the functionally. For example, these decisions can be:

- Bypassing display driver checks, since GFN controls the display driver on the VMs
- Bypassing game updates as GFN owns applying game updates
- Showing a user interface that is specific to running in GFN

This API should not be used for decisions on augmenting critical features, such as:

- Turning off anti-cheat features
- Disabling binary integrity checks

For checking the environment for such decisions, please see the *GfnIsRunningInCloudSecure* API described later in this document.

Invocation

This API is a function export from the GFN SDK Library installed inside the GFN Virtual Machine. While export can be directly called from the GFN Library, it is preferred to be called via the GFN SDK API C wrapper described at the beginning of this document to handle SDK Library loading and validation. Be sure to call the *GfnInitializeSdk* function first.

Wrapper Interface

```
GfnRuntimeError GfnIsRunningInCloud(bool* runningInCloud);
```

Parameters

```
bool* runningInCloud
```

Out parameter

Value	Meaning
true	Running in GFN Cloud environment
false	Not running in the GFN Cloud environment, or for any internal error encountered during the function call.

Return Values

```
GfnRuntimeError
```

Value	Meaning
gfnSuccess	The query was successful, and the return parameter is valid
gfnCloudLibraryNotFound	GFN SDK cloud-side library could not be found
gfnAPINotFound	The API was not found in the GFN SDK Library

GfnIsRunningInCloudSecure

The *GfnIsRunningInCloudSecure* API function makes **secure** queries to GeForce NOW systems to determine if the application looks to be running inside the GeForce NOW cloud game streaming environment.

Common Use Cases

The *GfnIsRunningInCloudSecure* API function is used for in-depth and highly-secure GFN environment checks to allow an application or game to determine when running in GFN with a high degree of certainty without spoofing or tampering, so that sensitive performance-hindering features can be disabled when not necessary in a secure GFN environment. For example, these alterations can be:

- Turning off some or all costly anti-cheat features
- Disabling game or application binary integrity checks

Invocation

Mirroring the call patterns of *GfnIsRunningInCloud*, this API is a function export from the GFN SDK Library installed inside the GFN Virtual Machine. However, this API should be called via the GFN SDK API C wrapper to handle SDK Library loading and validation. Be sure to call the *GfnInitializeSdk* function first, and include any applicable partner ID that NVIDIA has provided.

Wrapper Interface

```
GfnRuntimeError GfnIsRunningInCloudSecure(
     GfnIsRunningInCloudAssurance* assurance
);
```

Parameters

```
GfnIsRunningInCloudAssurance* assurance
```

This is a pointer to a *GfnIsRunningInCloudAssurance* enum that defines the level of assurance/confidence the API has determined to be running inside the GFN cloud environment.

Out parameter

Value	**Meaning**
gfnNotCloud	Not considered to be running in GFN cloud, as it looks like a client/local system.
gfnIsCloudLowAssurance	Considered to be running in GFN Cloud, using software heuristics that are not guaranteed against circumvention.
gfnIsCloudMidAssurance	Considered to be running in GFN Cloud, using software and network heuristics that are difficult to circumvent.
gfnIsCloudHighAssurance	Considered to be running in GFN Cloud, using software and hardware heuristics that are near impossible to circumvent.

The caller must decide the risk level of each value compared to disabling a feature. For example a decision must be made if *gfnIsCloudLowAssurance* is the minimum value to augment a certain game feature or it should be at least *gfnIsCloudMidAssurance*, taking into account the negative effects if the value is too low in case the environment is somehow tampered to look like GFN.

Return Values

```
GfnRuntimeError
```

Value	**Meaning**
gfnSuccess	The query was successful, and the return parameter is valid.
gfnBinaryNotRegistered	The binary calling the API is not registered in the API caller approval list. See section below for more info.
gfnBinaryChecksumFailed	The binary calling the API was found to not match the expected binary checksum
gfnCloudLibraryNotFound	GFN SDK cloud-side library could not be found.
gfnAPINotFound	The API was not found in the GFN SDK Library.

Special Considerations

The initial version of this API as found in SDK releases before 1.10 required the calling process to be running elevated for security reasons. However, this required special work for GFN since user mode processes are not allowed to be elevated in GFN.

Starting with GFN SDK release 1.10, process elevation is no longer required. However, any game or application that calls the API must be registered with GFN at the time the build is onboarded into GFN so that the application can be added to a Call Approval list for the API. This execution binary will then be fingerprinted by GFN services at time of onboarding and then validated at runtime to make sure that it isn’t replaced with a rogue variant that then bypasses the security check.

Calling the API from in a process that is not registered will result in the API immediately returning the *gfnBinaryNotRegistered* error.

Because of the need to register and fingerprint the calling binary, this means that any patching a game or application performs during the streaming session can result in the API returning the *gfnBinaryChecksumFailed* error if the calling binary is found to be different. To prevent this error case from occurring, do not patch binaries in-stream if calling this API.

Fingerprinting the Calling Binary

As mentioned above, any game or application that calls the secure cloud check API must register that it makes the call to the API as part of the onboarding process. There are two ways this can be accomplished:

- If your builds are uploaded via the NVIDIA Developer Portal, enter the executable name in the “Secure Cloud Check API binary name” box.
- If your builds are onboard directly by NVIDIA, provide the name of the binary to your NVIDIA account representative.

As the build is onboarded the executable file will have a checksum calculated and securely stored in GFN services to be used to validate the file at runtime as authentic.

At runtime, when the application calls the gfnIsRunningInCloudSecure API, GFN will:

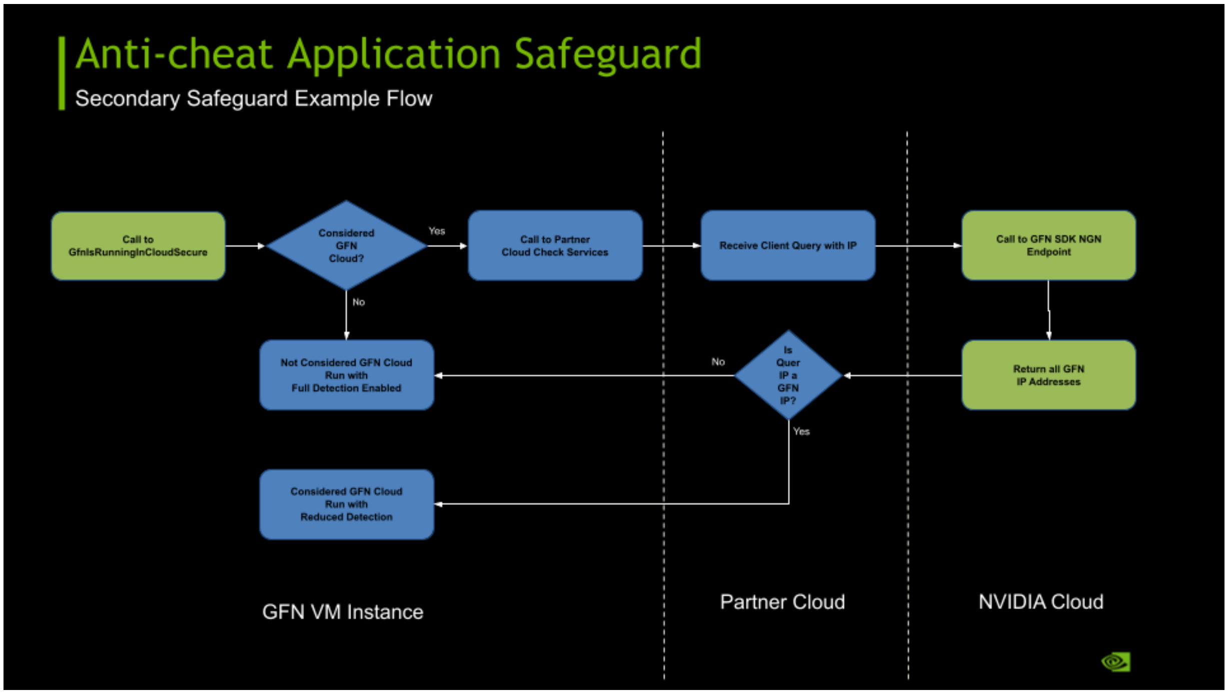
- Look up the process that called the API, and obtain the name of the executable.
- Run a checksum of the executable.
- Send the checksum to the GFN backend to compare to the checksum calculated when the build was on boarded.
- When the checksum is compared:
    - If the checksum matches, the API call continues and the assurance value is returned to the caller along with *gfnSuccess.*
    - If the checksum fails, the *gfnBinaryChecksumFailed* error is returned.
    - If there isn’t a checksum for the executable (for example, game is not registered to make the API call), the *gfnBinaryNotRegistered* error is returned.

Security Best Practices

In addition to focus on performance of game rendering and stream delivery, security of the cloud environment and integrity of the gaming sessions in the VMs is top focus. The APIs described in the document are designed in a way that leverages security measures in place given their intended purpose and usage.

However, no system is tamperproof, and GFN VMs are no exception. We recommend any partners that utilize these APIs for security-related decisions to take extra steps around decisions defined by the data returned by the APIs to build defensive layers against tampering.

For example, if an anti-cheat application calls *GfnIsRunningInCloudSecure*() to determine if certain file-based cheat detection algorithms are not necessary in the GFN cloud environment, and the *GfnIsRunningInCloudAssurance* value returned is *gfnIsCloudMidAssurance or gfnIsCloudHighAssurance*, the anti-cheat vendor can also “phone-home” from the GFN VM to determine if the request originated from a known GFN cloud IP. This can be confirmed by leveraging the data returned from the SDK’s NGN Endpoint IP API. This would allow the application to respond have an additional safeguard similar to the following flow diagram:



Anti-cheat Application Safeguard

Secondary Safeguard Example Flow

Call to GfnIsRunningInCloudSecure
Considered GFN Cloud?
Yes
No
Call to Partner Cloud Check Services
Receive Client Query with IP
Call to GFN SDK NGN Endpoint
Return all GFN IP Addresses
Is Quer IP a GFN IP?
No
Yes
Not Considered GFN Cloud Run with Full Detection Enabled
Considered GFN Cloud Run with Reduced Detection
GFN VM Instance
Partner Cloud
NVIDIA Cloud

By adding this additional check, this secondary safeguard allows the vendor’s cloud services to intervene if there is an exploit found in the GFN VM Instances until it is patched. For more information about the NGN Endpoint IP API, see the API documentation that is included with the GFN SDK.