The Feed Is Ours: A Case for Custom Clients

AUTHOR: tgr <tgr@tgrcode.com>

Editor's note:
This article has shortened to fit the print format.The full version will be available online at https://phrack.org

Table of Contents

0 - Introduction
1 - A Worrying Trend
2 - Super Mario Maker 2
  2.0 - Removed Features
  2.1 - Prior Research
  2.2 - NEX
  2.3 - Opening A Public API
  2.4 - The Scene Adapts
  2.5 - The Scrape
  2.6 - Opening A Public Server
  2.7 - A New Era in SMM2
  2.8 - A Bitter Reminder
3 - A Fragile State of Affairs
4 - References

0 Introduction

In 1995, when the World Wide Web was less than 2 years old and SSL 2.0 had only been released earlier that month, Neal Stephenson published a book hypothesizing a greatly advanced version of his society.

Drawing from scientific, not fictional, ideas growing in the late 20th century from books like "Engines of Creation (1986)" and "Nanosystems (1992)", this book proposed a striking kind of post-scarcity that remains to be seen even in the bounty of today.

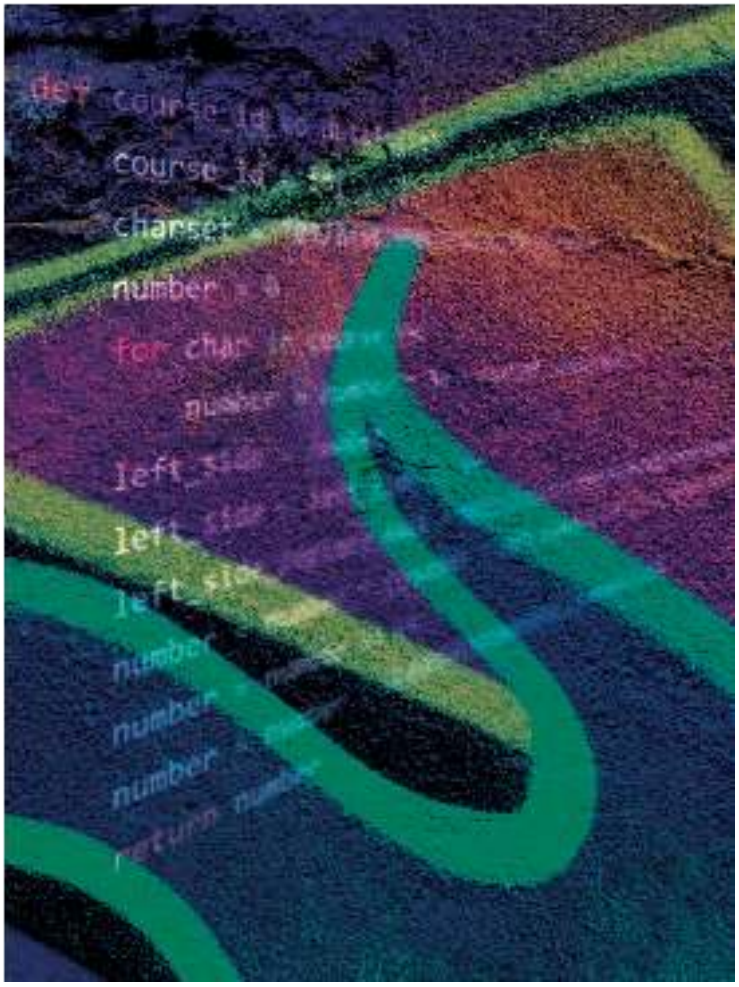
"The Diamond Age" proposed a type of nanomaterial distribution network capable of providing dependable streams of basic atoms like carbon, sulfer, oxygen and hydrogen. The "Feed", as it was called, was capable of providing "boxes of water and nutri-broth, envelopes of sushi made from nanosurimi and rice, candy bars" [0] from free matter compilers dotted throughout the urban landscape. Paid MCs are capable of creating much more complex structures, like the Primer, of which much has been said in regards to the development of Large Language Models.

The Feed is not purely altruistic, however. It is capable of reporting what is created and by whom to its operators. And unfortunately neo-Victorians do not have the best interests of other phyles at heart. In order to instill subversiveness in his daughter Fiona the secondary protagonist John Percival Hackworth secretly commissions the creation of a second Primer, which had only been intended for his employer. The engineer behind this second Primer, who operated his own private, untraceable Feed, called himself a"Reverse Engineer" [1].

[...]

The World Wide Web, TLS, large search engines - all of which started for the purposes of ensuring security and the continued proliferation of information freely, now serve to pull the internet back into its centralized origins.

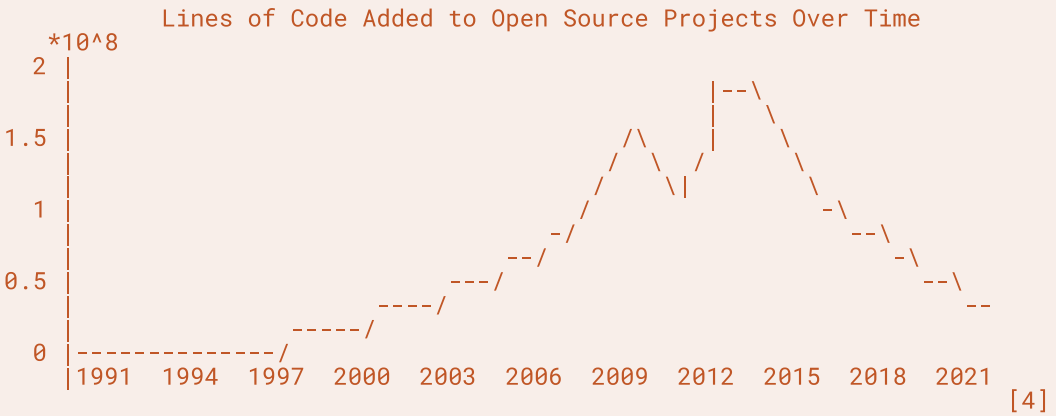
The world needs new hackers, like the reverse engineer Dr. X. and his custom Feed, to open the internet back up and free information once again.

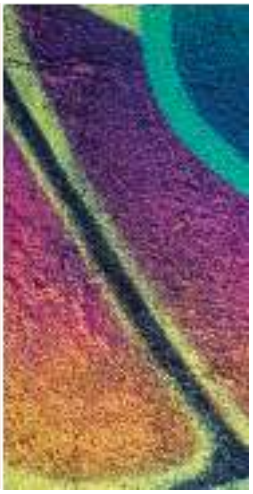


1 A Worrying Trend

Lets return to the present day. Independent blogs and webservers hosted out of commodity hardware flourished, giving rise to famed stores of culture that would become obsolete a few short years later when a new flashier competitor appeared. Forum messages, Freeware, PDFs - the internet was constantly sharing information in a freely accessible manner.

But monetization now flourishes in spaces it used to have no grasp.





One such kind of place hard hit in the last few years is the new age forum: social media.

On April 18, 2023 Reddit announced it would charge for its API [5]. Reddit had enjoyed a thriving scene of custom clients adding quality of life and accessibility features. RIF and Apollo were both apps forced to shut down.

The latter discovered they would have to pay 2 million dollars per month to Reddit in order to operate a custom client which was simply passing through requests [6]. From June 12 through to June 14 over 7000 subreddits blocked access to their content entirely for everyone, known as a blackout. Some subreddits continued beyond that timeframe. But unfortunately Reddit startedforcibly removing moderators from subreddits that stayed closed [7]. As of today business is as usual and the API pricing has not changed.

[...]

2 Super Mario Maker 2

My first custom client was for a game I saw as a perfect candidate for open data. Take the best selling video game franchise of all time [9] and allow for user created content, perhaps one of the largest such games! The roots of custom content, and especially courses, in the Super Mario series come from romhacks: injections of new machine code and assets into existing console-based Mario games. With a culture as rich as the demoscene-adjacent romhacking scene the idea had potential.

0. Removed Features

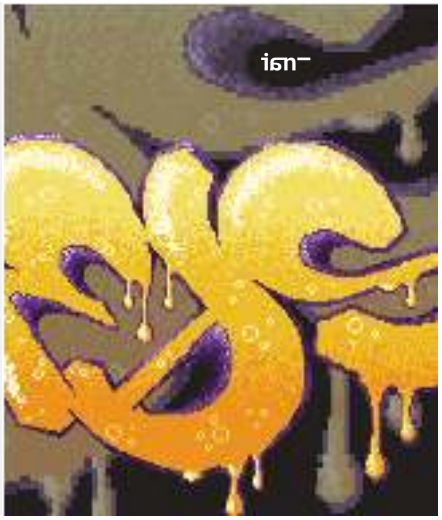
Super Mario Maker, the prequel, had a number of features that its sequel lacked. Importantly, these features were largely problems with the interface and not with the data accessible in game. One such feature is the ability to search courses using more complex queries, available in Super Mario Maker via an external site [10]. Another is the inability to view the entirety of downloaded courses via panning, accomplishable in Super Mario Maker by editing the downloaded course.

Some new features also lack the kind of searchability available in the prequel. For example, "Ninji" speedruns have no browser leaderboard. Various user leaderboards also lack a website.

1. Prior Research

The work started with Kinnay's NintendoClients [11], which implemented DAuth, AAuth, BAAS and the start of a custom client for Nintendo Online enabled games.

```
def course_id_to_dataid(id):
  course_id = id[::-1]
  charset = "0123456789BCDFGHJKLMNPQRSTVWXY"
  number = 0
  for char in course_id:
    number = number * 30 + charset.index(char)
  left_side = number
  left_side = left_side << 34
  left_side_replace_mask = 0b1111111111110000000000000000000000000000000000
  number = number ^ ((number ^ left_side) & left_side_replace_mask)
  number = number >> 14
  number = number ^ 0b00010110100000001110000001111100
  return number
```



When converted into a course ID or a maker ID the XOR serves to hide the fact that the ID is incrementing, likely an attempt to prevent a sweep through all courses or makers for nefarious purposes. We know the algorithm, however, so this reversible algorithm just means there are two ways of representing a course or maker.

Since it is incrementing where do we start? Not 0. The region of Data IDs below 3 million in SMM2 cannot be queried. With manual testing the first course found in the game is 4RF-XV8-WCG, data ID 3000004 uploaded on 6/27/19 02:05, a day before the game officially released. Using the method SearchCoursesLatest (method 73), which is used in game to return a random list of recent courses, I received a data ID close to the most recently allocated, at the time around 40000000. Using this approach one can query for 500 courses' info at once using GetCourses (method 70). Courses that have since been deleted, with their ID not being reallocated, return empty course info that could be ignored. This is the primary approach for courses.

[...]

The reason we cannot query players easily is because their "data ID" equivalent is not the internal ID used by the game. Wheras a course is associated with its data ID by every part of the game a maker ID's corresponding data ID is only used to generate that maker ID by Nintendo, summarily being ignored. The game actually uses PIDs, or Principal IDs, to refer to players, and these are randomized unsigned 64 bit integers on the switch. PID to maker ID is not reversible. While one can be used to query the other the direction we care about, maker ID to PID, can only be performed by GetUserOrCourse (method 131), and it only supports one maker ID at a time. Used in-game for a search bar it is not optimized for speed. The next best thing is just to collate all PIDs collected from other methods, assuming that one of the following applies to every player in the game:

- Created a course
- First cleared a course
- Has an active WR on a course
- Was one of the last 1000 players to play a course, whether they beat it or not
- Has played a Ninji event course while it was active
- Is within the top 1000 players on any in-game leaderboard (number of maker points, score in endless mode, etc)

[...]

6. Opening A Public Server

After the completion of the scrape it made sense to replace the feed entirely. By this I mean the classic final frontier in game modding related to protocol reimplementation: custom servers. That way no technical limitation, or action on the part of Nintendo, has an impact.

Custom clients, the topic of this paper and my main work, are exceptional starts and the only way to have live data from the feed, as well as being more directly usable by an audience. Custom servers, however, are the best way to follow up a custom client. Assuming a modded official client or another custom client it's possible to hook into a new feed entirely. The company behind the feed may not have any interest in archival, or may not send out timely

updates or may shut down the service with no recourse. With SMM2 having been around 5 years old by that time it was not, and still is not, an impossibility that Nintendo was considering shutting down the servers in a few years

This final step was possible thanks to the help of a number of developers who had begun building tools around the API following the technical discoveries made: Kramer, Wizulus, jneen and Shadow. Kramer had begun developing a Golang server implementing the NEX protocol, using NintendoClients as a base. When he reached out to the rest of us we began contributing.

[...]

Custom servers also have to convince the official client it is legitimate. All GetUser requests that refer to the current user have to send the same PID as the user's device ID. But we've already established Ryujinx sends the same constant for everyone. So to prevent a crash that PID has to be swapped with `0xcafe`. Another example is GetEndlessModePlayInfo (method 115). This method is constantly called while in a playthrough of the endless gamemode, and it is expected to return up to date info about all active endless runs. Included is all the courses that have been cleared.

So calls to StartEndlessModeCourse, DominateEndlessModeCourse (completing), PassEndlessModeCourse (skipping), SuspendEndlessMode (exiting) and FinishEndlessMode (game over) need to record the new status of the endless run or the client will behave strangely.

Number of coins, remaining lives and other variables must also be kept up-to-date.

[...]

Once we own the server we can also begin to see what information is sent by the official client but locked away forever on the official servers with no way to query. We knew about the Ninji replay because the official client queries it to represent it in-game. We also know how to parse it [37]. This replay stores the position of the player during the run every 4 frames and in what animation state they were in, so it only exists to play that run back. There had always been theories of another replay: input replays.

[...]

7. A New Era in SMM2

The public API, and the technical research following it, has changed how players engage with this game left behind by Nintendo. Streamers use course viewers, primarily one developed by Wizulus [40], to vet what users send and to skip tiring "little timmy" levels in endless mode. A search engine for courses, one of those removed features from SMM1, has been created by regularly topping off a local archive of courses collected from SearchCoursesLatest and GetCourses, enabling players to find whatever the in-game filter makes needlessly tedious to find [41]. Teams of players who had labored by hand searching for particular kinds of levels no longer need to do so, like the 0% team [42], who used the scrape to get an up to date list of uncleared levels and boost the team forwards.

[...]

8. A Bitter Reminder

So what is there to do when the hardware operating the public API, by which everything else mentioned operates, gets banned? Then our reliance on this fragile feed, and my loyalty to this kind of work, gets tested.

I know what it feels like because it's happened twice. The first time was immediately after the scrape in 2022, likely due to test requests I made to implement new methods. The second time was 2025, as the result of large scale DAuth changes from system version 20.0.0. Both times I had to buy new hardware, with the implicit reminder that it was another potential sacrifice to keep this experience going for all of the players of my favorite game.

[...]

3 A Fragile State of Affairs

The adventure continues on the Nintendo Switch 2, should we find an exploit that lets us MITM traffic for research, but it's the early days for that. We have a whole scene of experts who made this possible, and we will need their help or find new blood. Every source of user created content deserves a scene as rich as SMM2 now is.

My work on other custom clients continues. Prior to the shutdown of the Nintendo Network in 2024, for the WiiU and 3DS, I endeavored to create a scrape for every game

on both platforms. It required me to use my NEX custom client knowledge to create another: one that could request from every possible game that supported NEX [44].

[…]

New updates to network protocols, some whose outdated features had been depended on, will change the feasibility of custom clients for many domains, especially ones that do not want to make money. TLS 1.3, ECH and dynamic certificate pinning will make it much harder to research custom clients and implement them. Updates to old servers, requiring corresponding updates to the client, will remove the old exploits that made the custom clients possible from the picture.

We should ensure our favorite online services have a custom client. Those custom clients bring ownership of the feed back to the ones that made it possible. Whether it be social media or video games we should be allowed to do what we want with it. As long as the official client continues to slide towards corporate profit and the neo-Victorians operate the feed with their own agenda the path of the custom client remains the only way to preserve our liberties in this technological world. […]