Written by: guyguy April 2, 2012

Contents

1 Introduction

This document shows how to use the survival records script for setting up automatically updated survival records that are displayed on Google Spreadsheets.

Small note I am going to try to update this document with a lot of screenshots to make it an easy step-by-step tutorial. However, the spreasheet is constantly being changed so the actual document may end up looking different.

1.1 Purpose

Why should we bother with keeping track of survival records anyway? This project is not as serious as it seems and mostly serves as a sideproject of mine. It serves as a way for me to learn Python and to practice a little bit of programming. Other than that, the records are not meant to be taken very seriously. However, it is good to keep track of and recognize achievements made by our community of survival players.

There are many people playing survival mode in Left 4 Dead 1 and Left 4 Dead 2 from all over the world, and it is a good opportunity to make new online friends and collaborate with them towards a common goal. The Left 4 Dead series are known to be high quality cooperative games that stress teamwork, and the addition of goals and records makes the game refreshing and endlessly replayable for many survival enthusiasts.

1.2 How it works

After each game of survival, the game displays the statistics of the round that you just played. This includes the duration of the round and the number of common infected, special infected and tanks killed. The numbers will typed by someone (maybe you!) onto a Google Spreadsheet for the purpose of keeping track of records. Once the numbers are input onto the spreadsheet, the spreadsheet maintainer runs the provided Python script in order to update the spreadsheet. The best time, kills, and other statistics are calculated and displayed on the updated spreadsheet. In the next sections, the software behind the recording system will be explained.

1.2.1 Google Spreadsheets

Google spreadsheets is part of the Google Docs office suite, which is an online service designed to allow users to collaborate on office documents [1], [2]. The spreadsheets are a collection of tables, and consist of multiple pages called worksheets. Figure 1 shows what a Google Spreadsheet looks like in a web browser.

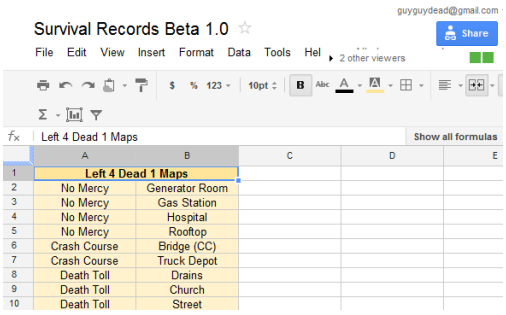


Figure 1 — Google Spreadsheet Opened in Web Browser

One of the interesting features of Google Docs is that any changes made to the web documents are instantly updated and seen by people on the Internet. In this way, we can imagine having a document that outputs real-time statistics, in our case, of the game Left 4 Dead.

1.2.2 Google Data API

As well as being able to edit spreadsheets from a web browser, Google has provided a way to communicate to their servers and automatically update Google Spreadsheets: the Google Data API (Application Programming Interface) [3]. What this means is that anyone can download the provided programming library and use it to read from and write to a Google Spreadsheet by writing a simple program.

1.2.3 Python

Python is a relatively new programming language suitable for scripting [4]. I chose Python because it is an interesting language that I wanted to learn through this project. Figure 2 shows the output of a sample run of the Python script to update the Google Spreadsheet.

```
adding new players
  Added 0 players
adding l4d1 records
  Added 0 records
adding l4d2 records
  Added 0 records
updating maps
   batch update 0 cells...
   single update 0 cells...
updating players
   batch update 1 cells...
```

Figure 2 — Python Script Updating the Spreadsheet

2 Spreadsheet Overview

This section will explain how the Google Spreadsheet works without going into the details of the underlying script. This section is for users that want to look at the survival records through the web browser and are not necessarily concerned with how it is updated. The spreadsheet is divided up into a number of different worksheets. Each worksheet will be explained in the following sections. The steps to adding records to the spreadsheets will be explained in Section 3.

2.1 Maps

The worksheet titled maps displays the list of official survival maps in Left 4 Dead 1 and 2 (see Figure 3.1). The first column indicates the campaign and the second column indicates the map. The names used here should be used in the rest of the document. For example, refer to the map as “Street” as shown and not other variations such as “The Street”.

2.2 Players

This worksheet displays a list of survival players (see Figure 3.2). Note that firstly, this is definitely not a complete list of all the players, only the players that are registered in the system. Secondly, the names of some players have been modified for readability. Their full name might appear under the aliases column. When referring to a player, you can use either the player’s name or one of his aliases.

Some extra information displayed are the number of records with that player in the system and the country. The country where the player is located is only a guess based on looking at public steam profiles, and might not actually be accurate. In the cases where I didn't know where the player was located, his country is listed as "Unknown".

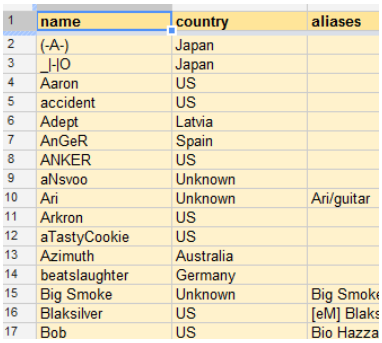
2.3 Records

This worksheet (Figure 3.3) displays the lists of all records in the system. These are the master lists of all records, not just best time, best kills, and also there are no exceptions or special groupings considered.

Dates Since the date is not an important factor in the survival statistics, not all the dates may be accurate. Firstly, the dates recorded consider the North American central timezone. Secondly, when the actual date that the survival game took place could not be found due to lack of screenshots, I made up an approximate date.

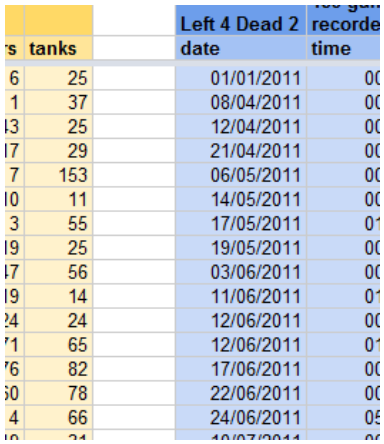
	A	B
10	Death Toll	Street
11	Death Toll	Boathouse
12	Dead Air	Crane
13	Dead Air	Construction Site
14	Dead Air	Terminal
15	Dead Air	Runway
16	Blood Harvest	Warehouse
17	Blood Harvest	Bridge (BH)
18	Blood Harvest	Farmhouse
19	Last Stand	Lighthouse
20	Sacrifice	Traincar
21	Sacrifice	Port
22		
23	**Left 4 Dead 2 Maps**	
24	Dead Center	Mall Atrium
25	The Passing	Riverbank
26	The Passing	Underground
27	The Passing	Port (P)
28	Dark Carnival	Motel
29	Dark Carnival	Stadium Gates

3.1: Maps Worksheet



	name	country	aliases
2	(-A-)	Japan	
3	_|-|O	Japan	
4	Aaron	US	
5	accident	US	
6	Adept	Latvia	
7	AnGeR	Spain	
8	ANKER	US	
9	aNsvoo	Unknown	
10	Ari	Unknown	Ari/guitar
11	Arkron	US	
12	aTastyCookie	US	
13	Azimuth	Australia	
14	beatslaughter	Germany	
15	Big Smoke	Unknown	Big Smok
16	Blaksilver	US	[eM] Blak
17	Bob	US	Bio Hazza

3.2: Players Worksheet



	s	tanks	Left 4 Dead 2 date	recorde time
	6	25	01/01/2011	0(
	1	37	08/04/2011	0(
	13	25	12/04/2011	0(
	17	29	21/04/2011	0(
	7	153	06/05/2011	0(
	10	11	14/05/2011	0(
	3	55	17/05/2011	0(
	19	25	19/05/2011	0(
	17	56	03/06/2011	0(
	19	14	11/06/2011	0(
	24	24	12/06/2011	0(
	71	65	12/06/2011	0(
	76	82	17/06/2011	0(
	50	78	22/06/2011	0(
	4	66	24/06/2011	0(

3.3: Records Worksheet

Figure 3 — Examples of the Worksheets

2.4 Statistics

This worksheet is where all the major statistics are calculated. It consists of a number of tables for Left 4 Dead 1 and 2, which are mostly self-explanatory. The tables show the best record in the system for all of the official Left 4 Dead 1 and 2 maps. I will now explain all the possible statistics that are tracked.

2.4.1 Time

This statistic measures the time the round has lasted. It is traditionally how all survival games have been rated.

2.4.2 Trash Factor

This statistic was named the trash factor or trash4cash factor after the player who first started mentioning this statistic to be used for measuring the difficulty of survival games. It is the ratio of total SI kills to minutes

played. Specifically,

$$TF_1 = \frac{smokers + boomers + hunters}{minutes} \tag{1}$$

$$TF_2 = \frac{smokers + boomers + hunters + chargers + spitters + jockeys}{minutes}. \tag{2}$$

This factor is meant to be interpreted as a difficulty rating of 1-10, where 10 is a most difficult round due to high number of special infected. In actuality, the trash factor may also exceed 10, but this seems to happen rarely, during survival rounds that should be considered difficult.

2.4.3 Kill Factor

This statistic takes the average kills of each kind of special infected including tanks. The reasoning for this is to penalize rounds that have a small number of a specific kind of special infected due to them getting stuck. For example, if all jockeys are stuck at some point, the number of special infected may still be significant, but the difficulty in dealing with jockeys are removed and the survival round was overall less challenging. Kill factor is calculated as follows:

$$KF_1 = \frac{(smokers + boomers + hunters + tanks)/4}{minutes} \tag{3}$$

$$KF_2 = \frac{(smokers + boomers + hunters + chargers + spitters + jockeys + tanks)/7}{minutes}. \tag{4}$$

This factor should yield values around 1.0. Factors well below 1.0 should be considered easy rounds and factors well above 1.0 should be considered difficult rounds.

2.4.4 Other Factors

Other factors include tank factor, gore factor, and common factor. These are pretty self-explanatory; tank factor is the total tank kills, common factor is the total common kills, and gore factors is the total amount of SI kills.

2.5 Top 10

This shows the same statistics as before, except the best 10 times are recorded for each map.

2.6 NA Statistics

These statistics are the same as the previous section's statistics, except that it only includes records with over 50% of the players residing in North America. The players with "Unknown" country are not counted as players from North America.

2.7 Groups

The groups worksheet lists three kinds of groups: game modes, strategies, and player groups. Another worksheet, the group statistics worksheet, shows the statistics for records restricted to various groupings. The groups are described in the following sections.

2.7.1 Playergroups

Player groups are groups of players. Some statistics might only count records with members belonging to the player group, e.g., North American statistics.

2.7.2 Strategies

The strategies are used to record the various strategies used in survival. These are specific to a map. For example, the `circuit` strategy for Mall Atrium involves running from tanks in a circuit, which is unique to that map. Most of the strategies are explained by the description. Also note that many of the strategies listed are alternative strategies that differ considerably from the main (typically most effective) strategies.

2.7.3 Game Modes

The game modes are modes of play that can be applied to any map. However, some game modes only apply to either Left 4 Dead 1 or Left 4 Dead 2. Most of the game modes are explained by the description.

`hardmode` Game Mode A common problem in both Left 4 Dead 1 and 2 is the fact that the special infected get eventually get stuck in almost all maps, which makes survival rounds actually easier as time progresses. `hardmode` is a server plugin (yet to be released, currently only for Left 4 Dead 2...) that removes special infected that are frozen in position and not attacking, allowing new special infected to attack. This makes the game harder than the normal (buggy) survival game settings.

`nomv` Game Mode In Left 4 Dead 2, it is considered acceptable to move throwables, health items, and weapons to any holdout spots. The `nomv` mode means to play Left 4 Dead 2 survival without moving any of the items except for the gascans and propane tanks. Note that moving items is generally not accepted in Left 4 Dead 1.

`smashtv` Game Mode This mode involves players that are not really in fixed positions and not camping in one particular spot, but hovering around an area, killing zombies in all directions.

3 Adding to the Spreadsheet

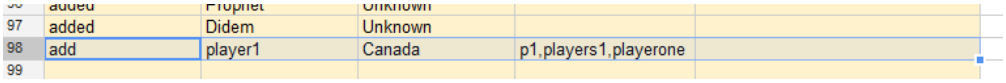
This section explains the steps to adding new records through the Google Spreadsheet.

Important Note Remember that in order for any new records or players to be added to the system, the Python script must first be ran by the spreadsheet maintainer. This means that any additions that you make onto the spreadsheets will not show up automatically.

3.1 Adding Players

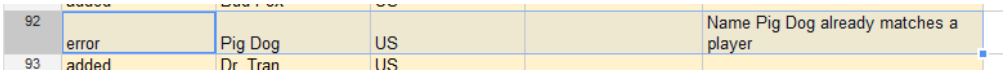
Before the players can be considered for records, they must be added to the system. This is performed inside the `Add Player` worksheet. The `Add Player` has four columns that must be filled out: status, name, country, and aliases. To add a player, first select an empty row. Then fill out the desired name for the player (preferably in short form using normal ASCII characters), the country to list the player under, and a list of aliases, separated by commas (no spaces). Then under the status, type `add`. The errors column should be left blank. Figure 4.1 shows an example of adding a player.

If the status remains on add, then the player has not yet been added but will be added once the spreadsheet has been updated via the script. Once the player has actually been added to the system, the status will change from `add` to `added`. If there was some kind of error during adding, such as a duplicate player, then the status will be changed to `error` as seen in Figure 4.2.



97	added	Didem	Unknown		
98	add	player1	Canada	p1,players1,playerone	
99					

4.1: Adding a New Player



92	error	Pig Dog	US		Name Pig Dog already matches a player
93	added	Dr. Tran	US		

4.2: Add Player Error

Figure 4 — Adding a Player

3.2 Adding Records

The most involved process is adding a record. Adding a record is available through the `Add Record1` worksheet for Left 4 Dead 1 records and the `Add Record 2` worksheet for Left 4 Dead 2 records. When adding records, the names used must already exist in the system. Otherwise, errors will appear during the script update process. The following fields must be filled out to add a record:

status Set status to `add`.

date Put the date that the game took place in the format year-month-day, e.g., `2012-03-21` for March 21, 2012.

time Put the time that the game lasted here. There are two possible formats.

- minutes:seconds.hundredths of a second, e.g., `38:29.73` for 38 minutes, 29 seconds, and 73 hundredths of a second or 730 milliseconds. The number of minutes can exceed 60.
- hours:minutes:seconds.hundredths of a second, e.g., `01:28:29.23` means 1 hour, 28 minutes, 29 seconds, 230 milliseconds.

map The name of the map played. This name must be the same name listed in the `Maps` worksheet.

players The name of all the players that played the game separated by commas (no spaces). Each of the names written here must match either a name or alias in the `Players` worksheet.

common Number of common infected killed

hunters Number of hunters killed

smokers Number of smokers killed

boomers Number of boomers killed

chargers Number of chargers killed (Left 4 Dead 2 only)

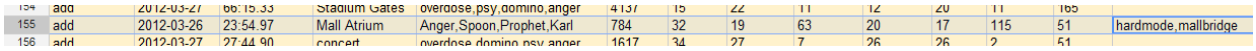
spitters Number of spitters killed (Left 4 Dead 2 only)

jockeys Number of jockeys killed (Left 4 Dead 2 only)

tanks Number of tanks

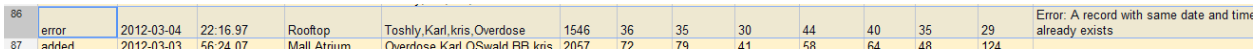
groups Add either a game mode or a strategy here (explained in Section 2.7) that applied to this survival game. Do not add playergroups. If there are multiple groups applicable, separate them with commas (no spaces).

Once the record has been added to the list, the status will say `added` and the record will be considered for all of the statistics that apply including normal statistics, top 10 statistics, and group statistics. If there was an error due to incorrect values, the status will say `error` and a corresponding error message will show up in the `errors` column. An example of adding a record is shown in Figure 5.1 and an example error message is shown in Figure 5.2.



154	add	2012-03-27	66:15.33	Stadium Gates	overdose,psy,domino,anger	4137	15	22	11	12	20	11	165	
155	add	2012-03-26	23:54.97	Mall Atrium	Anger,Spoon,Prophet,Karl	784	32	19	63	20	17	115	51	hardmode,mallbridge
156	add	2012-03-27	27:44.90	concert	overdose,domino,psy,anger	1617	34	27	7	26	26	2	51	

5.1: Adding a New Record



86	error	2012-03-04	22:16.97	Rooftop	Toshly,Karl,kris,Overdose	1546	36	35	30	44	40	35	29	Error: A record with same date and time already exists
87	added	2012-03-03	56:24.07	Mall Atrium	Overdose,Karl,OSwald,BB,kris	2057	72	79	41	58	64	48	124	

5.2: Add Record Error

Figure 5 — Adding a Record

3.3 Editing Records

A number of commands allow you to edit existing records through the `Add Record1` and `Add Record 2` worksheets.

3.3.1 Adding a Group

This allows you to add additional groups to an existing record. The required entries are:

- The date and time are required.
- The additional groups to be added should be entered either in the `groups` field or under the `players` field.
- The status should be set to `add-group`.

3.3.2 Editing the Date

This allows you to edit the date of an existing record.

- The correct time is required.
- The correct map is required.
- The correct list of players is required (Player names separated by commas, no spaces).
- Fill out the new date under the `date` field.
- The status should be set to `edit-date`.

3.3.3 Editing the Players

This allows you to edit the players in an existing record.

- The time is required.
- The date is required.
- The new list of players is required (Player names separated by commas, no spaces).
- The status should be set to `edit-players`.

3.3.4 Editing the Counts

This allows you to edit the common infected, special infected, and tank counts in an existing record.

- The time is required.
- The date is required.
- All counts of common infected, special infected, and tank counts will be changed simultaneously and should be filled in.
- The status should be set to `edit-counts`.

3.3.5 Editing the Map

This allows you to edit the map in an existing record.

- The time is required.
- The date is required.
- The new map is required.
- The status should be set to `edit-map`.

3.3.6 Editing Players, Maps, and Counts

This allows you to edit the map, players, and counts simultaneously.

- The time is required.
- The date is required.
- The new map is required.
- All counts of common infected, special infected, and tank counts will be changed simultaneously and should be filled in.
- The new list of players is required (Player names separated by commas, no spaces).
- The status should be set to `edit`.

4 Setting up the Script

In this section, I go over how to setup the Python script on your system such that it can automatically update the google spreadsheet.

Note It is very possible that this section gets obsolete quickly.

4.1 Prerequisites

Install the following prerequisites:

git `http://git-scm.com/download` This step is optional and can be avoided for Windows users.

Python 2.7 Use the Python 2.x version, not Python 3 `http://www.python.org/getit/`

Google Data Python Library See this getting started guide: `http://code.google.com/apis/gdata/articles/python_client_lib.html`. The steps are basically to download the library and run `python ./setup.py install`.

Google Account Register for a free account for Google Docs: `https://docs.google.com`. A gmail account will suffice.

4.1.1 Setting up the Google Spreadsheet

First, log in to Google Docs and create a spreadsheet as seen in Figure 6.

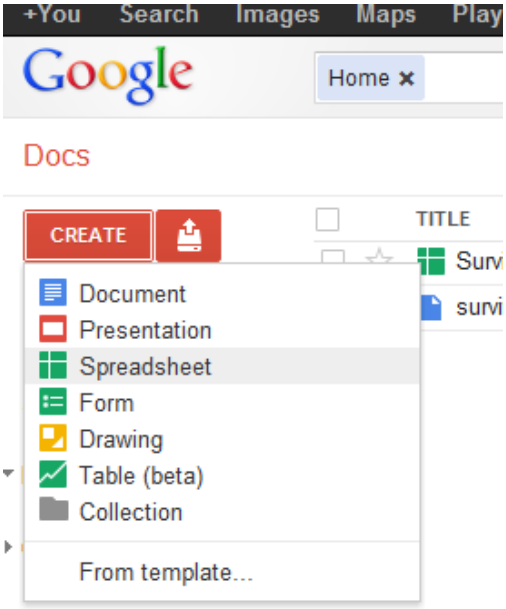


Figure 6 — Creating a Spreadsheet through the Google Docs Web Interface

Note Do not use an existing spreadsheet because all of the data will be erased. This cannot be undone!

Next, create twelve worksheets within the newly created google spreadsheet. By default, they should be named and ordered as follows:

1. Maps
2. Players
3. All Records
4. Statistics

5. Statistics Top 10
6. NA Statistics
7. NA Top 10
8. Group Stats
9. Add Player
10. Add Record1
11. Add Record2
12. Groups

The worksheets that are not involved with adding can be locked from public users editing it. This is because the script will overwrite any changes made to the spreadsheet anyway.

4.1.2 Default Settings

Currently the script will use the default settings and assume that the worksheets are ordered as mentioned in the previous section. Also, the script assumes that the google spreadsheet is actually the first document on your google account. If this is not true, you need to change the script from the default settings. Open up the script `survival_records.py` and near the bottom (line 2357) find the line that says:

```
use_defaults = True
```

Change the `True` to `False` and you will no longer be using the default settings. You will be able to specify which spreadsheet and which worksheets the script will write to.

4.2 Windows Instructions

In order to download the code from the git repository, you can visit the github webpage: `https://github.com/guyguydead/left4dead`. Now download the repository as a zip file as shown in Figure 7. Extract the zip file and rename the folder `left4dead-script` (your choice of name).

Figure 7 — Downloading the Repository from the github Webpage

Open up a command prompt using `start...run...cmd`. Ensure that your Python program can run by typing `python`. The command should be found and you get the prompt of >>> as seen in Figure 8. press CTRL-D to exit the python shell. If it's not working see instructions for finding python.exe at `http://www.imladris.com/Scripts/PythonForWindows.html`.

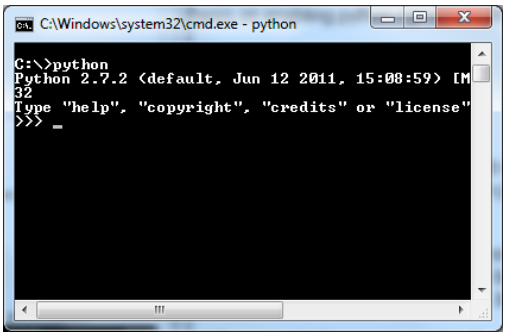


Figure 8 — Ensuring that the Python Program Runs Under Windows

Now that your command prompt is working, go to the left4dead script folder using the cd (change directory) command:

```
c:
cd \left4dead-scripts
```

run the script, supplying the username and password of your google account:

```
python ./survival_records.py --user=you_username --pw=your_password
```

Now the script will run and update the Google Spreadsheet. Please be patient and wait for the script to complete.

4.3 Linux Instructions

Clone a copy of the repository to get the source code and also enter the directory:

```
git clone git://github.com/guyguydead/left4dead.git
cd left4dead
```

run the script, supplying the username and password of your google account:

```
python ./survival_records.py --user=guyguydead --pw=your_password
```

Now the script will run and update the Google Spreadsheet. Please be patient and wait for the script to complete.

5 Future Work

A major future work addition will be to migrate the records into a proper database such as MySQL or SQLite. This was not done in the first place because of my lack of knowledge of databases. However, I might be willing to learn more about them and how to interface with them through Python as a hobby. I might also consider creating a graphical user interface for the script in the future. Other future work:

- Basic error checking has been added but more can be made for invalid inputs
- Some basic functionality has been added for groups but creating and editing groups should be added.
- More work on the documentation, specifically for spreadsheet maintainers and script developers.
- The script is very slow and feels hacked together. Some major refactoring would be in order but I may never get around to doing this.

References

[1] (2012) Google spreadsheets. [Online]. Available: http://www.google.com/google-d-s/spreadsheets/

[2] (2012) Google docs. [Online]. Available: https://docs.google.com

[3] (2012) Google data api. [Online]. Available: http://code.google.com/p/gdata-python-client/

[4] (2012) Python programming language. [Online]. Available: http://www.python.org/