

A decorative graphic on the right side of the page. It features three sets of concentric circles in shades of blue. The top set is the largest, the middle set is medium-sized, and the bottom set is the smallest. Thin blue lines extend from the top-left and top-right towards the circles, and a thicker line extends from the bottom-right towards the largest circle.

# Implementation of a Map Veto System

In a Team Fortress 2 Dedicated Server

This document describes the process undergone in the creation of a map veto system. The map veto system was created for the SourceOP.com TF2 dedicated servers that are on a rotation. The map veto system will allow players to choose to skip maps. The veto system has currently been implemented on the original “Valve Maps” server and is going well.

**Tony Paloma**  
**10/15/2008**

## Table of Contents

Abstract.....	2
Introduction .....	2
Implementation Details .....	2
Required SourceOP Plugin Changes.....	2
Lua Script.....	3
Known Issues.....	4
Future Improvements .....	4
Acknowledgements.....	4

## Abstract

This document describes the process undergone in the creation of a map veto system. The map veto system was created for the SourceOP.com TF2 dedicated servers that are on a rotation. The map veto system will allow players to choose to skip maps. The veto system has currently been implemented on the original “Valve Maps” server and is going well. The community is providing feedback, all of which is being considered.

## Introduction

The map veto system was created to satisfy the SourceOP.com community’s request for control over the map cycles on their Team Fortress 2 game servers. The vote itself will ask players currently playing in the game server if they would like to play the predetermined next map or if they would prefer to skip it and change to the map after. This vote occurs six minutes before the end of the map to ensure the vote is started before the map ends due to its time limit.

The map veto system will automatically adjust the percentage required for the veto to succeed. By default, a map’s veto vote will start at 50% required yes votes for a veto to succeed. If the veto succeeds, the next time the map comes up for a veto, it will require 10% more yes votes in order to succeed. Likewise, failures will result in a 10% decrease. However, the percentage is not allowed to decrease below 50% and cannot increase past 100%.

## Implementation Details

This section describes details of the implementation.

### Required SourceOP Plugin Changes

Before the map veto script could be coded in Lua, the SourceOP plugin itself had to be modified to provide required functionality.

Additions made:

Addition	Details
LevelShutdown game hook	Support for a level shutdown hook was necessary so that the required vote percentage could be saved on a per-map basis.
IntermissionStarting game hook	This was necessary due to the way that maps are skipped. The map won’t actually be skipped until intermission starts. If a map was to be skipped prior to intermission, the game would automatically go into a game-over half-intermission state leaving players frozen.
convar.SetString	This was a required Lua function to allow us to change the value of certain cvars (such as “nextmap”).
game.GetLevelFromCycle	The map veto system needed a way to get not just the next map, but also the one after. This function allows for that.

game.GetTimeLimit	This function allows the veto system to see the time limit of the map. This is required for us to start the vote six minutes before the limit.
game.GetTimeRemaining	This functions returns how long is remaining in the map. This is also used to start the veto on time.
util.KeyValuesToTable	Takes in a string and converts it to a Lua table. Necessary for loading the saved map required percentages.
util.TableToKeyValues	Takes in a table and returns a string. Necessary for saving the table that stores how many yes votes each map requires for a map to succeed.
GoToIntermission game rules hook	This function had to be hooked in order to call the new IntermissionStarting hook.

## Lua Script

The Lua script is the main implementation of the map veto system. It provides three console variables. These variables and their default values are listed below:

- DF\_mapveto\_enabled 0
- DF\_mapveto\_defaultpercent 50
- DF\_mapveto\_percentstep 10

The script creates three game hooks. One on IntermissionStarting, LevelInit, and LevelShutdown.

The LevelInit hook will initialize all internal variables and create a timer that runs every five seconds. This timer will check if it's time to show the vote. If it's time, and the map veto system is enabled, the vote will be displayed.

The vote itself builds off of previously used vote functionality. There is a simple Vote class that, when created, will display a vote to all users. After the specified time, the vote will be tallied, and the result is sent to another function. This function will then decide how to handle the results.

In the case of the map veto system, this function is mapveto\_voteover. This function will display what the raw results were and tell players whether or not the map will be skipped. If it is to be skipped, the "nextlevel" console variable is changed and a flag is set to say that the next map should be skipped. The only purpose of setting the "nextlevel" variable is to ensure that players saying "nextmap" see the correct map in the response. It is erased later before the map changes and is not used for picking the next map in our case. This function also adjusts the required veto percentage for next time as necessary.

When the map is over, SourceOP calls the IntermissionStarting hook. This function first looks to see if the "skip next map" flag mentioned in the paragraph above is set. If it is, it will reset the "nextlevel" console variable (so that the map cycle flows naturally), reset the flag (so that the system doesn't keep skipping maps when it's not supposed to), and advances the map cycle by calling `gamerules.AdvanceMapCycle()`. This function simply calls the function within TF2's `gamerules` class that changes to the next map, but blocks the actual call that changes the map. This creates the effect of "advancing the map cycle."

The table that stores the required veto percentage for each map is loaded only once when the script is loaded and saved at the end of each map.

## **Known Issues**

The veto vote will never be displayed if a round or win limit is set and this limit is hit before the six minute mark.

Player's inadvertently vote incorrectly because the 1 and 2 keys used to vote are also used to change weapon.

## **Future Improvements**

The veto vote will be changed so that '7' votes for "Yes" and '8' votes for "No." This addresses one of the issues above.

## **Acknowledgements**

I did this for the lulz.