Jaymod 2.1.1

Guide

UNDER CONSTRUCTION

Jaymod 2.1.1: Guide: UNDER CONSTRUCTION

Table of Contents

List of Tables

Foreword

First and foremost, thanks for taking the time out to give this mod a try. There are many other Wolfenstein: Enemy Territory® modifications out on the internet nowadays with the release of the SDK. This started as a project to just have a little fun and explore how the code of a great game works, but it turned into much more after my clan's game server kept experiencing instability with the popular Shrubmod.

I hope you have fun with this mod, as this is its intention.

—Jaybird

Preface

> “ The cost of adding a feature isn't just the time it takes to code it. The cost also includes the addition of an obstacle to future expansion. ... The trick is to pick the features that don't fight each other. ”
>
> —John Carmack

Audience

This guide is intended for anyone who runs or is interested in running a Jaymod 2.1.1 server.

Although ET is available on several platforms, Linux will be used as the reference platform for this documentation. Many concepts here apply equally to OSX and, to some degree, more generically to the Windows platform.

It is assumed the reader is sufficiently familiar with Linux administration to be able to perform basic operating system tasks.

Command Syntax

Linux command syntax appears in **monospace** font prefixed with a typical shell prompt denoting account/privileges for which the command should be entered with, followed by a number sign (#) or dollar sign ($) indicating root or non-root accounts, respectively. Do not enter the prompt text as part of the command.

Note

The assumed Linux shell is **bash**.

ET console command syntax appears in **monospace** font prefixed with the typical console prompt, a right-bracket (]). Do not enter the prompt text as part of the command.

Table 1. Syntax Conventions

Convention	Description
braces {}	Braces indicate required items.
brackets []	Brackets indicate optional items.
ellipses ...	Ellipses indicate an arbitrary number of similar items.
italics	Italic style indicates a variable. Substitute a real value for the variable.
vertical bar |	A vertical bar indicates a choice within braces or brackets.

Chapter 1. XP-save

Operation

Hitboxes are controlled using cvars and are effective immediately upon changing the value.

Table 1.1. XP-save Operation

CVAR	DESCRIPTION
g_xpSave	enable/disable XP-save feature
g_xpSaveTimeout	set XP-save timeout
g_xpMax	set XP-limit amount
g_xpCap	set XP-limit action

Chapter 2. Hitboxes

Hitboxes are the mechanism used to score bullet hits on players. As of Jaymod 2.1.0, an entirely new hitbox architecture was implemented. I would like to acknowledge that we continue to use contributed code from Zinx Verituse for server-side player-animation tracking.

The new architecture allows for Jaymod to support multiple hitmodes in a single release. We have taken advantage of this and implemented backwards-compatible modes which simulate boxes you are already accustomed to, while at the same time permitted us to offer, in my humble opinion, more realistic boxes and to address some known weaknesses.

During the course of development, I added some advanced visual and data debugging support to the architecture which allows us to perform testing and comparisons between different hitboxes during actual gameplay. Some of these visual debugging features are also useful for server admins and players seeking to learn and compare the new modes available.

—Mr.Mxyzptlk, January 2007.

Improvements

antilag	Antilag has been completely overhauled and integrated allowing for **all** bullet-fire weapons to benefit. Prior to this, mounted weapons such as Browning and MG42 did not benefit from antilag.
efficiency	More hitboxes and more sophistication usually results in a performance penalty. This is not the case with the new architecture. We use a hierarchal system to cull as many uneeded hit-checks as possible, while also no longer requiring temporary entities from the game engine. The net savings are very significant, resulting in more pound-for-pound CPU relief in typical gameplay, even when comparing a standing-up player with 7-hitboxes each in *real* mode, as compared to 2-hitboxes each in pre-2.1.0 mode.

Operation

Hitboxes are controlled using cvars and are effective immediately upon changing the value.

Table 2.1. Hitbox Operation

CVAR	DESCRIPTION
g_hitmode	set active hitmode
g_hitmodeAntilag	set antilag history in milliseconds
g_hitmodeAntilagLerp	enable/disable antilag lerping
g_hitmodeDebug	set bitflags for hitmode debugging
g_hitmodeFat	set increased torso-box size in inches
g_hitmodeGhosting	set lifetime of hit ghosting in milliseconds
g_hitmodeReference	set reference hitmode for comparison
g_hitmodeZone	set zone for debugging

Appendix A. Thanks and Credits

First and foremost, the most humble and sincere thanks go to everyone at **Clan FU** for their help and participation in the development and testing of this mod. Without their generous support, this mod would not be in existence.

My utmost, sincere thanks and appreciation go to **Zinx Verituse**, **Bani**, and **Forty** for their contributions.

Many, many thanks to **Mr.Mxyzptlk** for his cross-platform build/release system, C++ conversion and continued work on the mod.

Thanks to **Meyerinchains** for his contribution of the M97 model and animations.

Thanks to **Jeremy "Dr. Evil" Swigart** and the magic of the **Omni-bot** team for their continued help with the implementation and debugging support of Omni-bot in Jaymod.

Thanks at **Lazyphreak** at the Jaymod forums for taking the time to get all the original Shrub flags.

Thanks to **ETPub** for the contribution to the ET community in general. Jaymod incorporates modified bits of code from their project.

There are several server admins that frequent the Jaymod forums assisting those that require it without hesitation or commission. They simply choose to contribute, and for that I thank all of you and hope you will continue what you do.

Thanks to **SplashDamage®** for taking the time to develop and release an incredible free **id Software®** based game. They have made a good many fan because of their generosity and dedication to the gaming community. Thanks to the SplashDamage **forums community** for the help they provide to everyone wanting to mod for ET.

—Jaybird

Command Reference

Name

!status — display server resources

Synopsis

```
!status
```

Description

!status displays critical server resources which may help determine when a server is reaching its limits due to gameplay, maps, settings and other complexities.

CVAR Reference

Name

g_censor — enable/disable word-censor feature

Synopsis

```
g_censor [0|1]
```

Description

g_censor enables or disables word-censor functionality. A value of `0` will disable. If enabled, the server will censor game chat words matching those found in `censor.db`.

Name

g_hitmode — set active hitmode

Synopsis

```
g_hitmode [ 0 | 1 | 2 | 3 | 4 | 5 ]
```

Description

g_hitmode sets the active hitmode which controls how bullet-hits are calculated and scored against player models.

Table 4. g_hitmode Values

VALUE	NAME	BOXES	BRIEF
0	default	n/a	n/a
1	entity	1	torso (body)
2	etmain	3	head, torso, legs
3	basic	3	head, torso, legs
4	standard	3	head, torso, legs
5	real	7	head, lhand, rhand, torso(2), lfoot and rfoot

entity	Mimics player's word-collision with only a large torso (body) box. This is only useful for reference purposes, or servers uninterested in accuracy.
etmain	Baseline (stock) ET behavior composed of { head, torso, legs } boxes. Priority-box scoring in order of { head, legs, torso }. That is to say if a bullet can hit a player's *head*, regardless of whether it travels through another hitbox on the same player such as *torso*, the headshot will be scored. For example, shooting a player on a ladder from below, resulst in more headshots than you might expect. This behavior is to maintain backwards-compatibility.
basic	Similar but improved over *etmain* with { crouch, prone, playdead } adjustments for added realism. Uses priority-box scoring in order of { head, legs, torso }. This mode is somewhat similar to that of older ETPub versions.
standard	Similar but improved over *basic* with animation-accurate head-box. Uses priority-box scoring in order of { head, legs, torso }. This mode matches that of Jaymod prior to 2.1.0 version.
real	Realistic, tight behavior with head, left-hand, right-hand, torso(2), left-foot and right-foot boxes all tracking animation. Uses closest-box scoring, resulting in more difficult headshots when impeded by other hitboxes.

See Also

Hitboxes, g_hitmodeAntilag, g_hitmodeAntilagLerp, g_hitmodeDebug, g_hitmodeFat, g_hitmodeGhosting, g_hitmodeReference, g_hitmodeZone

Name

g_hitmodeAntilag — set antilag history in milliseconds

Synopsis

`g_hitmodeAntilag` [0..10000]

Description

g_hitmodeAntilag sets antilag history in milliseconds the server will use for backwards-reconciliation. A value of `0` will disable antilag.

See Also

Hitboxes, g_hitmode, g_hitmodeAntilagLerp, g_hitmodeDebug, g_hitmodeFat, g_hitmodeGhosting, g_hitmodeReference, g_hitmodeZone

Name

g_hitmodeAntilagLerp — enable/disable antilag lerping

Synopsis

`g_hitmodeAntilagLerp [0|1]`

Description

g_hitmodeAntilagLerp enables or disables the use of LERP calculations. A value of `0` will disable LERP.

See Also

Hitboxes, g_hitmode, g_hitmodeAntilag, g_hitmodeDebug, g_hitmodeFat, g_hitmodeGhosting, g_hitmodeReference, g_hitmodeZone

Name

g_hitmodeDebug — set bitflags for hitmode debugging

Synopsis

`g_hitmodeDebug` [flags]

Description

g_hitmodeDebug sets bitflags for hitmode debugging.

Table 5. g_hitmodeDebug Bitflags

BIT	DESCRIPTION
1	draw active hitboxes
2	draw reference hitboxes
4	draw world-collision-box
16	log lifecycle (construct/destruct/alloc/free) to client console
32	log snapshot use to client console
64	log state changes for state-driven models to client console
256	log ray text to client console
512	log volume text to client console

Caution

Hitbox visuals are not free. Each rendered box consumes an game-engine entity which is limited. If you are not careful, high numbers of players with high rates of hits and long lifetimes may lead to entity exhaustion.

Bits enabling client console logging can cause easily exceed ET limitations and cause your client to drop its connection. In other words, do not create rapid hits while this kind of debugging is active.

Tip

Use the !status command to monitor game entities.

See Also

Hitboxes, g_hitmode, g_hitmodeAntilag, g_hitmodeAntilagLerp, g_hitmodeFat, g_hitmodeGhosting, g_hitmodeReference, g_hitmodeZone

Name

g_hitmodeFat — set increased torso-box size in inches

Synopsis

`g_hitmodeFat` [-10..25]

Description

g_hitmodeFat sets increased torso-box size in inches. Negative values will attempt to reduce size.

See Also

Hitboxes, g_hitmode, g_hitmodeAntilag, g_hitmodeAntilagLerp, g_hitmodeDebug, g_hitmodeGhosting, g_hitmodeReference, g_hitmodeZone

Name

g_hitmodeGhosting — set lifetime of hit ghosting in milliseconds

Synopsis

`g_hitmodeGhosting` [0..10000]

Description

g_hitmodeGhosting sets lifetime of hit ghosting in milliseconds. A value of `0` will disable ghosting. This a visual-debugging technique which takes, upon recording a hit, the victim's hitboxes are drawn (frozen) in place. The box which has scored a hit is rendered in a different color.

Caution

Hitbox visuals are not free. Each rendered box consumes an game-engine entity which is limited. If you are not careful, high numbers of players with high rates of hits and long lifetimes may lead to entity exhaustion.

Tip

Use the !status command to monitor game entities.

See Also

Hitboxes, g_hitmode, g_hitmodeAntilag, g_hitmodeAntilagLerp, g_hitmodeDebug, g_hitmodeFat, g_hitmodeReference, g_hitmodeZone

Name

g_hitmodeReference — set reference hitmode for comparison

Synopsis

```
g_hitmodeReference [ 1 | 2 | 3 | 4 | 5 ]
```

Description

g_hitmodeReference sets reference hitmode for comparison.

Table 6. g_hitmode Values

VALUE	NAME	BOXES	BRIEF
1	entity	1	torso (body)
2	etmain	3	head, torso, legs
3	basic	3	head, torso, legs
4	standard	3	head, torso, legs
5	real	7	head, lhand, rhand, torso(2), lfoot and rfoot

See Also

Hitboxes, g_hitmode, g_hitmodeAntilag, g_hitmodeAntilagLerp, g_hitmodeDebug, g_hitmodeFat, g_hitmodeGhosting, g_hitmodeZone

Name

g_hitmodeZone — set zone for debugging

Synopsis

`g_hitmodeZone` [1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9]

Description

g_hitmodeZone set zone for debugging.

Table 7. g_hitmodeZone Values

VALUE	DESCRIPTION
1	head
2	left-hand
3	right-hand
4	torso
5	left-torso
6	right-torso
7	legs
8	left-foot
9	right-foot

See Also

Hitboxes, g_hitmode, g_hitmodeAntilag, g_hitmodeAntilagLerp, g_hitmodeDebug, g_hitmodeFat, g_hitmodeGhosting, g_hitmodeReference,

Name

g_shrubbot — enable/disable shrubbot feature

Synopsis

```
g_shrubbot [0|1]
```

Description

g_shrubbot enables or disables shrubbot (*!command*) functionality. A value of `0` will disable.

Name

g_xpCap — set XP-limit action

Synopsis

`g_xpCap` [0 | 1 | 2]

Description

g_xpCap sets the action to take when g_xpMax is exceeded.

Table 8. g_xpCap Values

VALUE	DESCRIPTION
0	Allow the player to continue gaining XP for skills which are not maxed out.
1	Disallow any further XP gain but keep it at max amount.
2	Automatically reset player's XP to 0.

See Also

XP-save, g_xpSave, g_xpSaveTimeout, g_xpMax,

Name

g_xpMax — set XP-limit amount

Synopsis

`g_xpMax` [0..999999]

Description

g_xpMax sets XP-limit amount. A value of `0` will disable. If a positive value, the server will limit the amount of XP a player can gain before a specified action is taken. The specified action is specified by g_xpCap.

See Also

XP-save, g_xpSave, g_xpSaveTimeout, g_xpCap,

Name

g_xpSave — enable/disable XP-save feature

Synopsis

`g_xpSave` [`0` | `1`]

Description

g_xpSave enables or disables XP-save functionality. A value of `0` will disable. If enabled, the server will save XP for up to the amount of time specified by g_xpSaveTimeout.

See Also

XP-save, g_xpSaveTimeout, g_xpMax, g_xpCap,

Name

g_xpSaveTimeout — set XP-save timeout

Synopsis

`g_xpSaveTimeout` [*`seconds`*]

`g_xpSaveTimeout` [[[[*`days`*] *`hours`*] *`minutes`*] *`seconds`*]

Description

g_xpSaveTimeout sets XP-save timeout. A value of 0 indicates XP-save will never timeout.

For convenience you may specify { days, hours, minutes, seconds } using their respective suffixes { d, h, m, s } as a single value. For example, to specify 10 days plus 4 hours use `10d4h0m0s` or `10d4h`. Alternatively, you can simply specify your value in seconds and do the math yourself.

See Also

XP-save, g_xpSave, g_xpMax, g_xpCap,