



# SECURE CODING

TEAM 12 - PHASE 2

MEMBERS

Alexander Lill    Lorenzo Donini

---

Florian Mauracher    Mahmoud Naser

# MOST IMPORTANT VULNERABILITIES

```

if ((gamenode == commercial)
    || (gamenode == pack_trc))
    || (gamenode == pack_plt))
{
    skytexture = R_TextureNumForName ("SKY3");
    if (gamenode < 12)
        skytexture = R_TextureNumForName ("SKY1");
    else
        if (gamenode < 21)
            skytexture = R_TextureNumForName ("SKY2");
}
levelstartic = gametic; // for time calculation
if (wipegamestate == GS_LEVEL)
    wipegamestate = -1; // force a wipe
gamerate = GS_LEVEL;
for (i=0 ; i<MAXPLAYERS ; i++)
{
    if (playeringamei && playersi.playerstate == PST_DEAD)
        playersi.playerstate = PST_REBORN;
    memset (playersi frags, 0, sizeof(playersi frags));
}
P_SetupLevel (gameepisode, gamemap, 0, gameskill);
displayplayer = consoleplayer;
starttime = L_GetTime ();
gameraction = ga_nothing;
Z_CheckHeap ();
// clear cmd building stuff
memset (gamekeydown, 0, sizeof(gamekeydown));
joymove = joymove = 0;
mousex = mousey = 0;
sendpulse = sendpulse = paused = false;
memset (mousebuttons, 0, sizeof(mousebuttons));
memset (joybuttons, 0, sizeof(joybuttons));
void G_PlayerReborn (int player)
{
    player_t* p;
    int i;
    fragsMAXPLAYERS;
    killcount;
    itemcount;
    secretcount;
    wminfo_t* wminfo;
    memory (frags,players[player].frags, sizeof (frags));
    killcount = players[player].killcount;
    itemcount = players[player].itemcount;
    secretcount = players[player].secretcount;
    p = &players[player];
    memset (p, 0, sizeof (*p));
    memcpy (frags,frags, sizeof (players[player].frags));
    players[player].killcount = killcount;
    players[player].itemcount = itemcount;
    players[player].secretcount = secretcount;
    p->usedown = p->attackdown = true;
    p->playerstate = PST_LIVE;
    p->health = MAXHEALTH;
    p->readyweapon = p->pendingweapon = wp_pistol;
    p->weaponowned[wp_pistol] = true;
    p->weaponowned[wp_pistol] = true;
    p->ammo/am_clip = 50;
    p->POL4_0_1=NULL;
    p->POL3_0_1=NULL;
    p->POL2_0_1=NULL;
    p->POL1_0_1=NULL;
    p->POL0_0_1=NULL;
    p->POL1_0_1=NULL;
    p->POL2_0_1=NULL;
    p->POL3_0_1=NULL;
    p->POL4_0_1=NULL;
    p->POL5_0_1=NULL;
    p->POL6_0_1=NULL;
    p->POL7_0_1=NULL;
    p->POL8_0_1=NULL;
    p->POL9_0_1=NULL;
    p->POL10_0_1=NULL;
    p->POL11_0_1=NULL;
    p->POL12_0_1=NULL;
    p->POL13_0_1=NULL;
    p->POL14_0_1=NULL;
    p->POL15_0_1=NULL;
    p->POL16_0_1=NULL;
    p->POL17_0_1=NULL;
    p->POL18_0_1=NULL;
    p->POL19_0_1=NULL;
    p->POL20_0_1=NULL;
    p->POL21_0_1=NULL;
    p->POL22_0_1=NULL;
    p->POL23_0_1=NULL;
    p->POL24_0_1=NULL;
    p->POL25_0_1=NULL;
    p->POL26_0_1=NULL;
    p->POL27_0_1=NULL;
    p->POL28_0_1=NULL;
    p->POL29_0_1=NULL;
    p->POL30_0_1=NULL;
    p->POL31_0_1=NULL;
    p->POL32_0_1=NULL;
    p->POL33_0_1=NULL;
    p->POL34_0_1=NULL;
    p->POL35_0_1=NULL;
    p->POL36_0_1=NULL;
    p->POL37_0_1=NULL;
    p->POL38_0_1=NULL;
    p->POL39_0_1=NULL;
    p->POL40_0_1=NULL;
    p->POL41_0_1=NULL;
    p->POL42_0_1=NULL;
    p->POL43_0_1=NULL;
    p->POL44_0_1=NULL;
    p->POL45_0_1=NULL;
    p->POL46_0_1=NULL;
    p->POL47_0_1=NULL;
    p->POL48_0_1=NULL;
    p->POL49_0_1=NULL;
    p->POL50_0_1=NULL;
    p->POL51_0_1=NULL;
    p->POL52_0_1=NULL;
    p->POL53_0_1=NULL;
    p->POL54_0_1=NULL;
    p->POL55_0_1=NULL;
    p->POL56_0_1=NULL;
    p->POL57_0_1=NULL;
    p->POL58_0_1=NULL;
    p->POL59_0_1=NULL;
    p->POL60_0_1=NULL;
    p->POL61_0_1=NULL;
    p->POL62_0_1=NULL;
    p->POL63_0_1=NULL;
    p->POL64_0_1=NULL;
    p->POL65_0_1=NULL;
    p->POL66_0_1=NULL;
    p->POL67_0_1=NULL;
    p->POL68_0_1=NULL;
    p->POL69_0_1=NULL;
    p->POL70_0_1=NULL;
    p->POL71_0_1=NULL;
    p->POL72_0_1=NULL;
    p->POL73_0_1=NULL;
    p->POL74_0_1=NULL;
    p->POL75_0_1=NULL;
    p->POL76_0_1=NULL;
    p->POL77_0_1=NULL;
    p->POL78_0_1=NULL;
    p->POL79_0_1=NULL;
    p->POL80_0_1=NULL;
    p->POL81_0_1=NULL;
    p->POL82_0_1=NULL;
    p->POL83_0_1=NULL;
    p->POL84_0_1=NULL;
    p->POL85_0_1=NULL;
    p->POL86_0_1=NULL;
    p->POL87_0_1=NULL;
    p->POL88_0_1=NULL;
    p->POL89_0_1=NULL;
    p->POL90_0_1=NULL;
    p->POL91_0_1=NULL;
    p->POL92_0_1=NULL;
    p->POL93_0_1=NULL;
    p->POL94_0_1=NULL;
    p->POL95_0_1=NULL;
    p->POL96_0_1=NULL;
    p->POL97_0_1=NULL;
    p->POL98_0_1=NULL;
    p->POL99_0_1=NULL;
    p->POL100_0_1=NULL;
    p->POL101_0_1=NULL;
    p->POL102_0_1=NULL;
    p->POL103_0_1=NULL;
    p->POL104_0_1=NULL;
    p->POL105_0_1=NULL;
    p->POL106_0_1=NULL;
    p->POL107_0_1=NULL;
    p->POL108_0_1=NULL;
    p->POL109_0_1=NULL;
    p->POL110_0_1=NULL;
    p->POL111_0_1=NULL;
    p->POL112_0_1=NULL;
    p->POL113_0_1=NULL;
    p->POL114_0_1=NULL;
    p->POL115_0_1=NULL;
    p->POL116_0_1=NULL;
    p->POL117_0_1=NULL;
    p->POL118_0_1=NULL;
    p->POL119_0_1=NULL;
    p->POL120_0_1=NULL;
    p->POL121_0_1=NULL;
    p->POL122_0_1=NULL;
    p->POL123_0_1=NULL;
    p->POL124_0_1=NULL;
    p->POL125_0_1=NULL;
    p->POL126_0_1=NULL;
    p->POL127_0_1=NULL;
    p->POL128_0_1=NULL;
    p->POL129_0_1=NULL;
    p->POL130_0_1=NULL;
    p->POL131_0_1=NULL;
    p->POL132_0_1=NULL;
    p->POL133_0_1=NULL;
    p->POL134_0_1=NULL;
    p->POL135_0_1=NULL;
    p->POL136_0_1=NULL;
    p->POL137_0_1=NULL;
    p->POL138_0_1=NULL;
    p->POL139_0_1=NULL;
    p->POL140_0_1=NULL;
    p->POL141_0_1=NULL;
    p->POL142_0_1=NULL;
    p->POL143_0_1=NULL;
    p->POL144_0_1=NULL;
    p->POL145_0_1=NULL;
    p->POL146_0_1=NULL;
    p->POL147_0_1=NULL;
    p->POL148_0_1=NULL;
    p->POL149_0_1=NULL;
    p->POL150_0_1=NULL;
    p->POL151_0_1=NULL;
    p->POL152_0_1=NULL;
    p->POL153_0_1=NULL;
    p->POL154_0_1=NULL;
    p->POL155_0_1=NULL;
    p->POL156_0_1=NULL;
    p->POL157_0_1=NULL;
    p->POL158_0_1=NULL;
    p->POL159_0_1=NULL;
    p->POL160_0_1=NULL;
    p->POL161_0_1=NULL;
    p->POL162_0_1=NULL;
    p->POL163_0_1=NULL;
    p->POL164_0_1=NULL;
    p->POL165_0_1=NULL;
    p->POL166_0_1=NULL;
    p->POL167_0_1=NULL;
    p->POL168_0_1=NULL;
    p->POL169_0_1=NULL;
    p->POL170_0_1=NULL;
    p->POL171_0_1=NULL;
    p->POL172_0_1=NULL;
    p->POL173_0_1=NULL;
    p->POL174_0_1=NULL;
    p->POL175_0_1=NULL;
    p->POL176_0_1=NULL;
    p->POL177_0_1=NULL;
    p->POL178_0_1=NULL;
    p->POL179_0_1=NULL;
    p->POL180_0_1=NULL;
    p->POL181_0_1=NULL;
    p->POL182_0_1=NULL;
    p->POL183_0_1=NULL;
    p->POL184_0_1=NULL;
    p->POL185_0_1=NULL;
    p->POL186_0_1=NULL;
    p->POL187_0_1=NULL;
    p->POL188_0_1=NULL;
    p->POL189_0_1=NULL;
    p->POL190_0_1=NULL;
    p->POL191_0_1=NULL;
    p->POL192_0_1=NULL;
    p->POL193_0_1=NULL;
    p->POL194_0_1=NULL;
    p->POL195_0_1=NULL;
    p->POL196_0_1=NULL;
    p->POL197_0_1=NULL;
    p->POL198_0_1=NULL;
    p->POL199_0_1=NULL;
    p->POL200_0_1=NULL;
    p->POL201_0_1=NULL;
    p->POL202_0_1=NULL;
    p->POL203_0_1=NULL;
    p->POL204_0_1=NULL;
    p->POL205_0_1=NULL;
    p->POL206_0_1=NULL;
    p->POL207_0_1=NULL;
    p->POL208_0_1=NULL;
    p->POL209_0_1=NULL;
    p->POL210_0_1=NULL;
    p->POL211_0_1=NULL;
    p->POL212_0_1=NULL;
    p->POL213_0_1=NULL;
    p->POL214_0_1=NULL;
    p->POL215_0_1=NULL;
    p->POL216_0_1=NULL;
    p->POL217_0_1=NULL;
    p->POL218_0_1=NULL;
    p->POL219_0_1=NULL;
    p->POL220_0_1=NULL;
    p->POL221_0_1=NULL;
    p->POL222_0_1=NULL;
    p->POL223_0_1=NULL;
    p->POL224_0_1=NULL;
    p->POL225_0_1=NULL;
    p->POL226_0_1=NULL;
    p->POL227_0_1=NULL;
    p->POL228_0_1=NULL;
    p->POL229_0_1=NULL;
    p->POL230_0_1=NULL;
    p->POL231_0_1=NULL;
    p->POL232_0_1=NULL;
    p->POL233_0_1=NULL;
    p->POL234_0_1=NULL;
    p->POL235_0_1=NULL;
    p->POL236_0_1=NULL;
    p->POL237_0_1=NULL;
    p->POL238_0_1=NULL;
    p->POL239_0_1=NULL;
    p->POL240_0_1=NULL;
    p->POL241_0_1=NULL;
    p->POL242_0_1=NULL;
    p->POL243_0_1=NULL;
    p->POL244_0_1=NULL;
    p->POL245_0_1=NULL;
    p->POL246_0_1=NULL;
    p->POL247_0_1=NULL;
    p->POL248_0_1=NULL;
    p->POL249_0_1=NULL;
    p->POL250_0_1=NULL;
    p->POL251_0_1=NULL;
    p->POL252_0_1=NULL;
    p->POL253_0_1=NULL;
    p->POL254_0_1=NULL;
    p->POL255_0_1=NULL;
    p->POL256_0_1=NULL;
    p->POL257_0_1=NULL;
    p->POL258_0_1=NULL;
    p->POL259_0_1=NULL;
    p->POL260_0_1=NULL;
    p->POL261_0_1=NULL;
    p->POL262_0_1=NULL;
    p->POL263_0_1=NULL;
    p->POL264_0_1=NULL;
    p->POL265_0_1=NULL;
    p->POL266_0_1=NULL;
    p->POL267_0_1=NULL;
    p->POL268_0_1=NULL;
    p->POL269_0_1=NULL;
    p->POL270_0_1=NULL;
    p->POL271_0_1=NULL;
    p->POL272_0_1=NULL;
    p->POL273_0_1=NULL;
    p->POL274_0_1=NULL;
    p->POL275_0_1=NULL;
    p->POL276_0_1=NULL;
    p->POL277_0_1=NULL;
    p->POL278_0_1=NULL;
    p->POL279_0_1=NULL;
    p->POL280_0_1=NULL;
    p->POL281_0_1=NULL;
    p->POL282_0_1=NULL;
    p->POL283_0_1=NULL;
    p->POL284_0_1=NULL;
    p->POL285_0_1=NULL;
    p->POL286_0_1=NULL;
    p->POL287_0_1=NULL;
    p->POL288_0_1=NULL;
    p->POL289_0_1=NULL;
    p->POL290_0_1=NULL;
    p->POL291_0_1=NULL;
    p->POL292_0_1=NULL;
    p->POL293_0_1=NULL;
    p->POL294_0_1=NULL;
    p->POL295_0_1=NULL;
    p->POL296_0_1=NULL;
    p->POL297_0_1=NULL;
    p->POL298_0_1=NULL;
    p->POL299_0_1=NULL;
    p->POL300_0_1=NULL;
    p->POL301_0_1=NULL;
    p->POL302_0_1=NULL;
    p->POL303_0_1=NULL;
    p->POL304_0_1=NULL;
    p->POL305_0_1=NULL;
    p->POL306_0_1=NULL;
    p->POL307_0_1=NULL;
    p->POL308_0_1=NULL;
    p->POL309_0_1=NULL;
    p->POL310_0_1=NULL;
    p->POL311_0_1=NULL;
    p->POL312_0_1=NULL;
    p->POL313_0_1=NULL;
    p->POL314_0_1=NULL;
    p->POL315_0_1=NULL;
    p->POL316_0_1=NULL;
    p->POL317_0_1=NULL;
    p->POL318_0_1=NULL;
    p->POL319_0_1=NULL;
    p->POL320_0_1=NULL;
    p->POL321_0_1=NULL;
    p->POL322_0_1=NULL;
    p->POL323_0_1=NULL;
    p->POL324_0_1=NULL;
    p->POL325_0_1=NULL;
    p->POL326_0_1=NULL;
    p->POL327_0_1=NULL;
    p->POL328_0_1=NULL;
    p->POL329_0_1=NULL;
    p->POL330_0_1=NULL;
    p->POL331_0_1=NULL;
    p->POL332_0_1=NULL;
    p->POL333_0_1=NULL;
    p->POL334_0_1=NULL;
    p->POL335_0_1=NULL;
    p->POL336_0_1=NULL;
    p->POL337_0_1=NULL;
    p->POL338_0_1=NULL;
    p->POL339_0_1=NULL;
    p->POL340_0_1=NULL;
    p->POL341_0_1=NULL;
    p->POL342_0_1=NULL;
    p->POL343_0_1=NULL;
    p->POL344_0_1=NULL;
    p->POL345_0_1=NULL;
    p->POL346_0_1=NULL;
    p->POL347_0_1=NULL;
    p->POL348_0_1=NULL;
    p->POL349_0_1=NULL;
    p->POL350_0_1=NULL;
    p->POL351_0_1=NULL;
    p->POL352_0_1=NULL;
    p->POL353_0_1=NULL;
    p->POL354_0_1=NULL;
    p->POL355_0_1=NULL;
    p->POL356_0_1=NULL;
    p->POL357_0_1=NULL;
    p->POL358_0_1=NULL;
    p->POL359_0_1=NULL;
    p->POL360_0_1=NULL;
    p->POL361_0_1=NULL;
    p->POL362_0_1=NULL;
    p->POL363_0_1=NULL;
    p->POL364_0_1=NULL;
    p->POL365_0_1=NULL;
    p->POL366_0_1=NULL;
    p->POL367_0_1=NULL;
    p->POL368_0_1=NULL;
    p->POL369_0_1=NULL;
    p->POL370_0_1=NULL;
    p->POL371_0_1=NULL;
    p->POL372_0_1=NULL;
    p->POL373_0_1=NULL;
    p->POL374_0_1=NULL;
    p->POL375_0_1=NULL;
    p->POL376_0_1=NULL;
    p->POL377_0_1=NULL;
    p->POL378_0_1=NULL;
    p->POL379_0_1=NULL;
    p->POL380_0_1=NULL;
    p->POL381_0_1=NULL;
    p->POL382_0_1=NULL;
    p->POL383_0_1=NULL;
    p->POL384_0_1=NULL;
    p->POL385_0_1=NULL;
    p->POL386_0_1=NULL;
    p->POL387_0_1=NULL;
    p->POL388_0_1=NULL;
    p->POL389_0_1=NULL;
    p->POL390_0_1=NULL;
    p->POL391_0_1=NULL;
    p->POL392_0_1=NULL;
    p->POL393_0_1=NULL;
    p->POL394_0_1=NULL;
    p->POL395_0_1=NULL;
    p->POL396_0_1=NULL;
    p->POL397_0_1=NULL;
    p->POL398_0_1=NULL;
    p->POL399_0_1=NULL;
    p->POL400_0_1=NULL;
    p->POL401_0_1=NULL;
    p->POL402_0_1=NULL;
    p->POL403_0_1=NULL;
    p->POL404_0_1=NULL;
    p->POL405_0_1=NULL;
    p->POL406_0_1=NULL;
    p->POL407_0_1=NULL;
    p->POL408_0_1=NULL;
    p->POL409_0_1=NULL;
    p->POL410_0_1=NULL;
    p->POL411_0_1=NULL;
    p->POL412_0_1=NULL;
    p->POL413_0_1=NULL;
    p->POL414_0_1=NULL;
    p->POL415_0_1=NULL;
    p->POL416_0_1=NULL;
    p->POL417_0_1=NULL;
    p->POL418_0_1=NULL;
    p->POL419_0_1=NULL;
    p->POL420_0_1=NULL;
    p->POL421_0_1=NULL;
    p->POL422_0_1=NULL;
    p->POL423_0_1=NULL;
    p->POL424_0_1=NULL;
    p->POL425_0_1=NULL;
    p->POL426_0_1=NULL;
    p->POL427_0_1=NULL;
    p->POL428_0_1=NULL;
    p->POL429_0_1=NULL;
    p->POL430_0_1=NULL;
    p->POL431_0_1=NULL;
    p->POL432_0_1=NULL;
    p->POL433_0_1=NULL;
    p->POL434_0_1=NULL;
    p->POL435_0_1=NULL;
    p->POL436_0_1=NULL;
    p->POL437_0_1=NULL;
    p->POL438_0_1=NULL;
    p->POL439_0_1=NULL;
    p->POL440_0_1=NULL;
    p->POL441_0_1=NULL;
    p->POL442_0_1=NULL;
    p->POL443_0_1=NULL;
    p->POL444_0_1=NULL;
    p->POL445_0_1=NULL;
    p->POL446_0_1=NULL;
    p->POL447_0_1=NULL;
    p->POL448_0_1=NULL;
    p->POL449_0_1=NULL;
    p->POL450_0_1=NULL;
    p->POL451_0_1=NULL;
    p->POL452_0_1=NULL;
    p->POL453_0_1=NULL;
    p->POL454_0_1=NULL;
    p->POL455_0_1=NULL;
    p->POL456_0_1=NULL;
    p->POL457_0_1=NULL;
    p->POL458_0_1=NULL;
    p->POL459_0_1=NULL;
    p->POL460_0_1=NULL;
    p->POL461_0_1=NULL;
    p->POL462_0_1=NULL;
    p->POL463_0_1=NULL;
    p->POL464_0_1=NULL;
    p->POL465_0_1=NULL;
    p->POL466_0_1=NULL;
    p->POL467_0_1=NULL;
    p->POL468_0_1=NULL;
    p->POL469_0_1=NULL;
    p->POL470_0_1=NULL;
    p->POL471_0_1=NULL;
    p->POL472_0_1=NULL;
    p->POL473_0_1=NULL;
    p->POL474_0_1=NULL;
    p->POL475_0_1=NULL;
    p->POL476_0_1=NULL;
    p->POL477_0_1=NULL;

```

# BYPASSING AUTHORIZATIONS

# PRIVILEGE ESCALATION

# STORED XSS IN ALL FORMS

# SQL INJECTION IN ALL FORMS

# TEST UPLOAD OF MALICIOUS FILES

# UPLOAD OF EXPLOITABLE FILES

# TESTING FOR EXPLOITS

# EXPLOITING THE GAME

# BYPASSING AUTHENTICATION

{SPR\_SHT2,1,7,(NULL)}

```
if (*save_ip != 0x1d)
    Error ("Bad save address");
// done
Z_Free (savebuffer);

if (setsizeneeded)
    B_ExecuteSetViewSize ();
```

```
(SPR_SHT2,0,5,{A_Refire}),S_DSGUN,0,0), // S_DSGUN10
(SPR_SHT2,1,7,{NULL}),S_DSNR2,0,0), // S_DSNR1
(SPR_SHT2,0,3,{NULL}),S_DSGUNDOWN,0,0), // S_DSNR2
(SPR_SHT2,32776,5,{A_Light1}),S_DSGUNFLASH2,0,0), // S_DSGUN/get commands, check consistency
(SPR_SHT2,32777,4,{A_Light2}),S_LIGHTDONE,0,0), // S_DSGUN/and build new consistancy
(SPR_CHGG,0,1,{A_WeaponReady}),S_CHAIN,0,0), // S_CHAIN      buf = (gametic/ticdup)%BACKUP
(SPR_CHGG,0,1,{A_Lower}),S_CHAINDOWN,0,0), // S_CHAINDOWN
(SPR_CHGG,0,1,{A_Raise}),S_CHAINUP,0,0), // S_CHAINUP    for (i=0 ; i<MAXPLAYERS ; i++)
(SPR_CHGG,0,1,{A_Center}),S_CHAINC,0,0), // S_CHAINC
```

```
    // draw the pattern into the back buffer
    R_FillBackScreen();
}

void G_OutSaveGame(void)
{
    char name[100];
    char name2[VERSIONSIZE];
    char* description;
}
```

```
ed None ISPR_CHGG,0.4,(A_FireCGun),S_CHAIN2,0,0); // S_CHAIN1 if (playeringame(i)) { ISPR_CHGG,1.1,(A_FireCGun),S_CHAIN3,0,0); // S_CHAIN2 ISPR_CHGG,1.5,(A_Refire),S_CHAIN,0,0); // S_CHAIN3 ISPR_CHLF,32,68,5,(A_Light1),S_LIGHTDONE,0,0); // S_CHAINFLASH1 cmd = &players[i].cr ISPR_CHGF,32769,5,(A_Light2),S_LIGHTDONE,0,0); // S_CHAINFLASH2 ISPR_MISC,0,1,(A_WeaponReady),S_MISSILE,0,0); // S_MISSILE memcpy(cmd, &net ISPR_MISC,0,1,(A_Lower),S_MISSILEDOWN,0,0); // S_MISSILEDOWN if (demoplayback) ISPR_MISC,0,1,(A_Raise),S_MISSILEUP,0,0); // S_MISSILEUP G_ReadDemo() ISPR_MISC,1.8,(A_GunFlash),S_MISSILE2,0,0); // S_MISSILE1 if (demorecording) ISPR_MISC,1.12,(A_FireMissile),S_MISSILE3,0,0); // S_MISSILE2 G_WriteDemo() ISPR_MISC,1.2,(A_Refire),S_MISSILE,0,0); // S_MISSILE3
```

```
int length;
int i;

if (M_CheckParm("-edrom"))
    sprintf(name,"c:\doomdata\SAVEGAMENAME%d.dsg",savegame);
else
    sprintf(name,SAVEGAMENAME "%d.dsg", savegame);
descritor = fopen(name,"w");
```

Unchanged

```
    save_p = savebuffer = screens(1)+0x4000;  
  
    memcpy (save_p, description, SAVESTRINGSIZE);  
    save_p += SAVESTRINGSIZE;  
    memcpy (name2, 0, sizeof(name2));  
    sprintf (name2, "vers %s", EPROM(N));  
    memcpy (save_p, name2, VERSIONSIZE);
```

1; i<-S\_SAWG\_PAINT ; i++)  
1;  
RSHOT1.speed = 15\*FRACUNIT;  
HOT1.speed = 10\*FRACUNIT;  
SHOT1.speed = 10\*FRACUNIT  
**H**igh  
(SPR\_SAWG,0,4,(A\_Saw),S\_SAW2,0,0), // S\_SAW1  
(SPR\_SAWG,1,4,(A\_Saw),S\_SAW3,0,0), // S\_SAW2  
(SPR\_SAWG,1,0,(A\_ReFire),S\_SAW,0,0), // S\_SAW3  
(SPR\_PLASG,0,1,(A\_WeaponReady),S\_PLASMA,0,0), // S\_PLASMA  
(SPR\_PLASG,0,1,(A\_Lower),S\_PLASMADOWN,0,0), // S\_PLASMADOWN  
(SPR\_PLASG,0,1,(A\_Raise),S\_PLASMAUP,0,0), // S\_PLASMAUP  
(SPR\_PLASG,0,3,(A\_FirePlasma),S\_PLASMA2,0,0), // S\_PLASMA1  
(SPR\_PLASG,0,4,(A\_FirePlasma),S\_PLASMA3,0,0), // S\_PLASMA2  
players|conso )  
if (netgame && !neto {  
if (gematic > 1 && consist {  
}

```
; break;
emap;
LES
LES2
to secret level
save_p += VERSIONSIZE;


---


    *save_p++ = gameskill;
    *save_p++ = gameepisode;
    *save_p++ = gamemode;
for (i=1; i<MAX_LVLEPIS; i++)
    *save_p++ = playtime;
*save_p++ = leveltime>>16;
*save_p++ = leveltime>>8;
```

```
    {SPR_PLASMA,1,20,(A_Refire),S_PLASMA,0,0}, // S_PLASMA2
    {SPR_PLASMAFLASH,32768,4,(A_Light1),S_LIGHTDONE,0,0}, // S_PLASMAFLASH1
    {SPR_PLASMAFLASH,32769,4,(A_Light1),S_LIGHTDONE,0,0}, // S_PLASMAFLASH2
    {SPR_BFGG,0,1,(A_WeaponReady),S_BFG,0,0}, // S_BFG
    {SPR_BFGG,0,1,(A_Lower),S_BFGDOWN,0,0}, // S_BFGDOWN
    {SPR_BFGG,1,1,(A_Pause),S_BFGUP,0,0}, // S_BFGUP
    {SPR_BFGG,1,20,(A_BFGsound),S_BFG2,0,0}, // S_BFG1
    {SPR_BFGG,1,10,(A_GunFlash),S_BFG3,0,0}, // S_BFG2
    {SPR_BFGG,1,10,(A_FireBFG),S_BFG4,0,0}, // S_BFG3
    {SPR_BFGG,1,20,(A_Refire),S_BFG,0,0}, // S_BFG4
```

# Availability

```

    {SPA_BFGF,32768,11,(A_Light1),S_BFGFLASH2,0,0}, // S_BFGFLASH1
    {SPA_BFGF,32769,6,(A_Light2),S_LIGHTDONE,0,0}, // S_BFGFLASH2
    {SPA_PLUG,2,8,(NULL),S_BL0001,0,0}, // check for special buttons
    {SPA_PLUG,1,8,(NULL),S_BL0002,0,0}, // S_BL0001
    {SPA_PLUG,0,8,(NULL),S_BL0003,0,0}, // S_BL0002
    {SPA_PUFF,32768,4,(NULL),S_PUFF1,0,0}, // S_PUFF1
    {SPA_PUFF,1,4,(NULL),S_PUFF2,0,0}, // if (playeringame(i))
    {SPA_PUFF,2,4,(NULL),S_PUFF3,0,0}, // if (players[i].cmd.bu

```

```
length = save_p - savebuffer  
if (length > SAVEGAMESIZE)  
    L_Error ("Savegame buffer overflow");  
    WriteFile (name, savebuffer, length);  
    gameaction = ga_nothing;  
    savefilesize[0] = 0;
```

```

    (SPR_PUFF1,2,4,(NULL),S_PUFF4,0,0), // S_PUFF3
    (SPR_PUFF3,4,(NULL),S_NULL,0,0), // S_PUFF4
    (SPR_BAL1,32768,4,(NULL),S_TBALL2,0,0), // S_TBALL1
    (SPR_BAL1,32769,4,(NULL),S_TBALL1,0,0), // S_TBALL2
    (SPR_BAL1,32770,6,(NULL),S_TBALLX2,0,0), // S_TBALLX1
    (SPR_BAL1,32771,6,(NULL),S_TBALLX3,0,0), // S_TBALLX2
    (SPR_BAL1,32772,6,(NULL),S_NULL,0,0), // S_TBALLX3
    (SPR_BAL1,32773,6,(NULL),S_TBALLX4,0,0) // S_TBALLX4
};

switch (player)
{
    case BTS_PLAYER1:
        paused ^= 1;
        if (paused)
            S_Pause();
        else
            S_Resume();
        break;
}
```

# ESCALATION

8.8

Network

Low

red Low

ers = false;  
-- sk\_nightmare && gamekill != sk\_nightmare) {  
 if (SPR\_MISSL\_18(A\_GunFlash), S\_MISSILE)  
 if (SPR\_MISSL\_19(A\_FireMissile), S\_MISSILE)  
 if (SPR\_MISSL\_20(A\_Refire), S\_MISSILE, 0)  
 if (SPR\_MISSL\_32768, 3, A\_Light1), S\_MISSILE

**Unchange**

**High**

// will be set false if a demo  
erstate = PST\_REBORN;

ode; High

ISPA\_PUFF,32/68,4,(NULL),S\_PUFF2,0,  
ISPA\_PUFF,1,4,(NULL),S\_PUFF3,0,0},  
ISPA\_PUFF,2,4,(NULL),S\_PUFF4,0,0},  
ISPA\_PUFF,3,4,(NULL),S\_NULL,0,0},  
ISPP\_PUFF,22320,4,(NULL),S\_PUFF2,0,

```
    S_TextureNumForName ("SKY3");  
    : 12)  
= R_TextureNumForName ("SKY1");  
  
    (SPR_BAL1,32768,4,(NULL),S_TBALL2,  
    (SPR_BAL1,32769,4,(NULL),S_TBALL1,  
    (SPR_BAL1,32770,6,(NULL),S_TBALLX2,  
    (SPR_BAL1,32771,6,(NULL),S_TBALLX3,  
    (SPR_BAL1,32772,6,(NULL),S_NULL,0,(
```



# SQL INJECTION IN ALL FORMS

// deal with the modification  
P.UpdateLayer();

```
if (*save_p != 0x1d)
    // Error ("Bad savegame").
// done.
7. Free (savebuffer);
```

y

```
R_ExecuteSetViewSize D;  
// draw the participant into the black screen  
R_FillBlackScreen B;
```

# Privileges

commercial)

```
char name[100];  
char name[VERSIONSIZE];  
char *exception;
```

: false;

```
if (M_CheckParm("-cdrom"))  
    sprintf(name,"c:\\doomdata\\\"SAVEGAME.NLM\"%d.lsg",s  
else  
    sprintf(name,"SAVEGAME.NLM\"%d.lsg",s  
description = "SAVING GAME";
```

ERSHOT1.speed = 20\*FRACUNIT; {SPR\_MISF32771.4,{A\_Light2},  
SHOT1.speed = 20\*FRACUNIT; {SPR\_SAW1.2.4,{A\_WeaponRea  
PSHOT1.speed = 20\*FRACUNIT {SPR\_SAW1.2.4,{A\_WeaponRea  
ce && gamerskill == sk\_nightmare) {SPR\_SAWG2.1,{A\_Raiser},{S\_SAWG2.1,{A\_Raiser}}}

```
        memcpy (save_p, description, SAVESTRINGSIZE);
        save_p += SAVESTRINGSIZE;
        memset (name2, 0, sizeof(name2));
        sprintf (name2, "Version %d", ERANUM);
        memcpy (save_p, name2, VERSIONSIZE);
```

**High**

```
// will be set false if a c
```

1

episode  
special

# TEST UPLOAD OF MALICIOUS FILES

RS ; i++)  
set = true;  
  
// determine all the modifications  
P.UnionSet(avenue);

ISPR SHT2,1  
ISPR SHT2,2  
ISPR SHT2,3  
ISPR SHT2,4  
ISPR SHT2,5  
ISPR SHT2,6

# Access Comp

1 / **LOW** {  
    (SPR\_SHT2,0,5,{A\_ReFire}),S\_DSGUN,0,0}, // S\_DSGUN10  
    (SPR\_SHT2,1,7,{NULL}),S\_DSNR1, // S\_DSNR1  
    (SPR\_SHT2,0,3,{NULL}),S\_DSGUNDOWN,0,0}, // S\_DSNR2  
    (SPR\_SHT2,32776,5,{A\_Light1}),S\_DSGUNFLASH2,0,0}, // S\_DSGUN#get commands, check consistency  
    (SPR\_SHT2,32777,4,{A\_Light2}),S\_LIGHTDONE,0,0}, // S\_DSGUN#and build new consistency  
}

R\_ExecuteSetViewSize D:  
    // draw the pattern into the source area  
    R\_FillBackScreen

char name[100];  
char game[2\*VERSIONSIZE];  
char\* description;

# User Interface

more & nominate

gamemap  
else  
sprintf(name, "c:\\doomdata\\SAVEGAMEN  
description = "Save game  
Scope C

Unchanging

```
    memmove (save_p, description, SAVESTRINGSIZE);
    save_p += SAVESTRINGSIZE;
    memmove (name2, str2, nameRead);
    sprintf (name2, "%s.%s.%s", EP, str1, str2);
    memmove (save_p, name2, VERSIONSIZEx2);
    save_p += VERSIONSIZEx2;
```

if (netgame && !netcon) {  
 if (gameric > 1 && consistent) {  
 if (High) {  
 if (SG1.speed = 15\*FRACUNIT;  
 SPR\_PLSG(0,1,(A\_WeaponReady),S\_PLASMA,0,0); // S\_PLASMA  
 SG1.speed = 10\*FRACUNIT;  
 SPR\_PLSG(0,1,(A\_Lower),S\_PLASMADOWN,0,0); // S\_PLASMADOWN  
 SG1.speed = 10\*FRACUNIT;  
 SPR\_PLSG(0,1,(A\_Raise),S\_PLASMAUP,0,0); // S\_PLASMAUP  
 SG1.PLUG(1,3,(A\_FirePlasma),S\_PLASMA1,0,0); // S\_PLASMA1  
 SG1.PLUG(1,20,(A\_Refire),S\_PLASMA2,0,0); // S\_PLASMA2  
 }  
 }  
}

```
*save_p++ = gameepisode;
*save_p++ = gametime;
for (i=1; i<MAX_LVLEPIS; i++)
    save_p += playtime[i];
*save_p++ = leveltime>>16;
```

PST\_REBORN;  
High  
will be set false if a demo

**Availability**: P\_ArchivePlayers();  
P\_ArchiveWorld();  
P\_ArchiveParties();

```
High // S_BFGFBASH1  
 // S_BFGFLASH2  
 // check for special buttons  
 for (i=0 ; i<MAXPLAYERS ; i++)  
 {
```

```

    *save_p++ = 0x1d; // constancy n

    length = save_p - savebuffer;
    if (length > SAVEGAMESIZE)
        { // Error ('Savegame buffer overrun')
        M_WriteFile (name, savebuffer, length);
        gameaction := ga_nothing;
    }
}

```

```

    (SPR_PUFF,32768,4,(NULL),S_PUFF2,0,0), // S_PUFF1
    (SPR_PUFF,1,4,(NULL),S_PUFF3,0,0), // S_PUFF2
    (SPR_PUFF,2,4,(NULL),S_PUFF4,0,0), // S_PUFF3
    (SPR_PUFF,3,4,(NULL),S_NULL,0,0), // S_PUFF4
    (SPR_BAL1,32768,4,(NULL),S_TBALL2,0,0), // S_TBALL1
    (SPR_BAL1,32769,4,(NULL),S_TBALL1,0,0), // S_TBALL2
    (SPR_BAL1,32770,6,(NULL),S_TBALLX2,0,0), // S_TBALLX1
    (SPR_BAL1,32771,6,(NULL),S_TBALLX3,0,0), // S_TBALLX2
    (SPR_BAL1,32772,6,(NULL),S_NULL,0,0) // S_TBALLX3
}
{
    if (players[li].cmd.button)
    {
        switch (player)
        {
            case BTS_PLAYER:
                paused ^= 1;
                if (paused)

```

# LIVE-DEMO!



# ONE MORE THING ...



... IF SOME OF YOU GUYS  
WERE LOOKING FOR A WAY  
TO EASILY INCLUDE THE  
CVSS SCORES IN LATEX



**... IF SOME OF YOU GUYS  
WERE LOOKING FOR A WAY  
TO EASILY INCLUDE THE  
CVSS SCORES IN LATEX  
  
BUT YOU DIDN'T FIND ONE ...**



THERE IS A TOOL.



THERE IS A TOOL. NOW

[github.com/AlexanderLill/cvss3tex](https://github.com/AlexanderLill/cvss3tex)



**THANKS FOR YOUR  
ATTENTION!**

