

Максим Кустов

# Вирус Фейка

```
(SPR_PUNG,2,4,(NULL),S_PUNCH5,0,0); // S_PUNCH4
(SPR_PUNG,1,5,(A_ReFire),S_PUNCH,0,0); // S_PUNCH5 // do things to change the game state
(SPR_PISG,0,1,(A_WeaponReady),S_PISTOL,0,0); // S_PISTOL while (gameaction != ga_nothing)
(SPR_PISG,0,1,(A_Lower),S_PISTOLDOWN,0,0); // S_PISTOLDOWN
(SPR_PISG,0,1,(A_Raise),S_PISTOLUP,0,0); // S_PISTOLUP
(SPR_PISG,0,4,(NULL),S_PISTOL2,0,0); // S_PISTOL1
(SPR_PISG,1,6,(A_FirePistol),S_PISTOL3,0,0); // S_PISTOL2
(SPR_PISG,2,4,(NULL),S_PISTOL4,0,0); // S_PISTOL3
(SPR_PISG,1,5,(A_ReFire),S_PISTOL,0,0); // S_PISTOL4
(SPR_PISF,32769,7,(A_Light1),S_LIGHTDONE,0,0); // S_PISTOLFLASH
(SPR_SHTG,0,1,(A_WeaponReady),S_SGUN,0,0); // S_SGUN
(SPR_SHTG,0,1,(A_Lower),S_SGUNDOWN,0,0); // S_SGUNDOWN
(SPR_SHTG,0,1,(A_Raise),S_SGUNUP,0,0); // S_SGUNUP
(SPR_SHTG,0,3,(NULL),S_SGUN2,0,0); // S_SGUN1
(SPR_SHTG,0,7,(A_FireShotgun),S_SGUN3,0,0); // S_SGUN2
(SPR_SHTG,1,5,(NULL),S_SGUN4,0,0); // S_SGUN3
(SPR_SHTG,2,5,(NULL),S_SGUN5,0,0); // S_SGUN4
(SPR_SHTG,3,4,(NULL),S_SGUN6,0,0); // S_SGUN5
(SPR_SHTG,2,5,(NULL),S_SGUN7,0,0); // S_SGUN6
(SPR_SHTG,1,5,(NULL),S_SGUN8,0,0); // S_SGUN7
```

switch (gameaction)

```
case ga_loadlevel:
    G_DeLoadLevel ();
    break;
case ga_newgame:
    G_DeNewGame ();
    break;
case ga_loadgame:
    G_DeLoadGame ();
    break;
case ga_savegame:
    G_DeSaveGame ();
    break;
case ga_playdemo:
    G_DePlayDemo ();
```

```
// ENEMY THINKING
// Enemies are always spawned
// with targetplayer = -1, then
// Most numbers are spawned
// but some can be made pre
//
// Called by P_NoiseAlert.
// Recursively traverses adjacent
// sound blocking lines out of
//
mob_t* soundtarget;
void
P_RecursiveSound
( sector_t* sec,
  int soundblock
)
{
    int i;
    line_t* check;
    sector_t* other;
    // wake up all monsters in
    if (sec->validcount == validcount
        && sec->soundtraverse
    )
    {
        return; //
    }
    sec->validcount = validcount;
    sec->soundtraverse = sec;
    sec->soundtarget = soundtarget;
    for (i=0; i<sec->linecount; i++)
    {
        check = sec->lines[i];
        if (! (check->flags & SFL_BLOCK_SOUND))
            continue;
        P_LineOpening (check);
        if (openrange <= 0)
            continue; //
        cmd = &players[i].cmd;
        if (! (sideset check->sidel
            other = sideset check->other;
            else
            other = sideset check->other;
        if (check->flags & MFL_BLOCK_SOUND)
        {
            if (soundblock)
                P_RecursiveSound (sec, soundblock);
            else
                P_RecursiveSound (sec, soundblock);
        }
        static char turbomessage[90];
        extern char *player_names[4];
        sprintf (turbomessage, "%s is turbo!", player_names[i]);
        players[consoleplayer].message = turbomessage;
        //
        // P_NoiseAlert,
        // If a monster yells at a player
        // it will alert other monsters
        if (netgame && !netdemo && !(gametic%ticdup) )
        {
            if (gametic > BACKUPTICS
                && consistency[fillbuf] != cmd->consistency)
            {
                void
                P_NoiseAlert
                ( mob_t* target,
                  mob_t* emitter );
                soundtarget = target;
                validcount++;
                P_RecursiveSound (sec, soundblock);
            }
        }
        if (netgame)
        {
            consistency[fillbuf] = players[i].mo->x;
            else
                consistency[fillbuf] = rndindex;
        }
        // check for special buttons
        for (i=0; i<MAXPLAYERS; i++)
        {
            if (players[i].gametic)
            {
                if (players[i].cmd.buttons & BT_SPECIAL)
                {
                    if (i == consoleplayer)
                    {
                        fixed_s;
                        dist;
                    }
                }
            }
        }
    }
}
```

```
(SPR_CHGF,32768,5,(A_Light1),S_LIGHTDONE,0,0); // S_CHAINFLASH1
(SPR_CHGF,32769,5,(A_Light2),S_LIGHTDONE,0,0); // S_CHAINFLASH2
(SPR_MISC,0,1,(A_WeaponReady),S_MISSILE,0,0); // S_MISSILE
(SPR_MISC,0,1,(A_Lower),S_MISSILEDOWN,0,0); // S_MISSILEDOWN
(SPR_MISC,0,1,(A_Raise),S_MISSILEUP,0,0); // S_MISSILEUP
(SPR_MISC,1,0,(A_GunFlash),S_MISSILE2,0,0); // S_MISSILE1
(SPR_MISC,1,12,(A_FireMissile),S_MISSILE3,0,0); // S_MISSILE2
(SPR_MISC,1,0,(A_ReFire),S_MISSILE,0,0); // S_MISSILE3
(SPR_MISF,32768,3,(A_Light1),S_MISSILEFLASH2,0,0); // S_MISSILEFLASH1
(SPR_MISF,32769,4,(NULL),S_MISSILEFLASH3,0,0); // S_MISSILEFLASH2
(SPR_MISF,32770,4,(A_Light2),S_MISSILEFLASH4,0,0); // S_MISSILEFLASH3
(SPR_MISF,32771,4,(A_Light2),S_LIGHTDONE,0,0); // S_MISSILEFLASH4
(SPR_SAWG,2,4,(A_WeaponReady),S_SAWB,0,0); // S_SAW
(SPR_SAWG,3,4,(A_WeaponReady),S_SAWD,0,0); // S_SAWB
(SPR_SAWG,2,1,(A_Lower),S_SAWDOWN,0,0); // S_SAWDOWN
(SPR_SAWG,2,1,(A_Raise),S_SAWUP,0,0); // S_SAWUP
(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_PLSG,0,1,(A_WeaponReady),S_PLASMA,0,0); // S_PLASMA
(SPR_PLSG,0,1,(A_Lower),S_PLASMDOWN,0,0); // S_PLASMDOWN
(SPR_PLSG,0,1,(A_Raise),S_PLASMAUP,0,0); // S_PLASMAUP
(SPR_PLSG,0,3,(A_FirePlasma),S_PLASMA2,0,0); // S_PLASMA1
(SPR_PLSG,1,20,(A_ReFire),S_PLASMA,0,0); // S_PLASMA2
(SPR_PLSF,32768,4,(A_Light1),S_LIGHTDONE,0,0); // S_PLASMAFLASH1
(SPR_PLSF,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,0,1,(A_Raise),S_BFGUP,0,0); // S_BFGUP
(SPR_BFGG,0,20,(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
(SPR_BFGF,32768,11,(A_Light1),S_BFGFLASH2,0,0); // S_BFGFLASH1
(SPR_BFGF,32769,6,(A_Light2),S_LIGHTDONE,0,0); // S_BFGFLASH2
(SPR_BLOOD,2,9,(NULL),S_BLOOD2,0,0); // S_BLOOD1
(SPR_BLOOD,1,9,(NULL),S_BLOOD3,0,0); // S_BLOOD2
(SPR_BLOOD,0,9,(NULL),S_NULL,0,0); // S_BLOOD3
(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
```

```
cmd = &players[i].cmd;
memcpy (cmd, &netcmds[fillbuf], sizeof(ticcmd_t));
if (demoplayback)
    G_ReadDemoTiccmd (cmd);
if (demorecording)
    G_WriteDemoTiccmd (cmd);
// check for turbo cheats
if (cmd->forwardmove > TURBOTHRESHOLD
    && !(gametic&31) && (gametic>>5)&3) == i)
{
    static char turbomessage[90];
    extern char *player_names[4];
    sprintf (turbomessage, "%s is turbo!", player_names[i]);
    players[consoleplayer].message = turbomessage;
    //
    // P_NoiseAlert,
    // If a monster yells at a player
    // it will alert other monsters
    if (netgame && !netdemo && !(gametic%ticdup) )
    {
        if (gametic > BACKUPTICS
            && consistency[fillbuf] != cmd->consistency)
        {
            void
            P_NoiseAlert
            ( mob_t* target,
              mob_t* emitter );
            soundtarget = target;
            validcount++;
            P_RecursiveSound (sec, soundblock);
        }
    }
    if (players[i].mo)
        consistency[fillbuf] = players[i].mo->x;
    else
        consistency[fillbuf] = rndindex;
}
// check for special buttons
for (i=0; i<MAXPLAYERS; i++)
{
    if (players[i].gametic)
    {
        if (players[i].cmd.buttons & BT_SPECIAL)
        {
            if (i == consoleplayer)
            {
                fixed_s;
                dist;
            }
        }
    }
}
}
}
```

Максим Кустов

**Вирус Фейка**

«ЛитРес: Самиздат»

2019

## **Кустов М. В.**

Вирус Фейка / М. В. Кустов — «ЛитРес: Самиздат», 2019

Вредоносная информация, как и вредоносные микроорганизмы, может крайне пагубно влиять на человека. Если внимательно проследить за основными признаками болезни - пути заражения, протекание заболевания, методы лечения и другие, то можно пронаблюдать удивительную схожесть заболевания человека при проникновении информационного и биологического вируса. А учитывая набирающий популярность термин "информационная война", интересно проследить за аналогией двух видов оружия массового поражения - информационного и биологического.

# Содержание

Вирус Fake	5
Конец ознакомительного фрагмента.	6

## Вирус Fake

Мы все много раз слышали заезженные фразы, вроде того, что мы живем в век информации, и что главной ценностью является информация. Но до конца тяжело разобраться, что это означает. В действительности каждый под этим термином понимает свой смысл.

Кто-то считает, что владение нужной информацией крайне важно в бизнесе. И они правы. Например, для биржевых брокеров узнать на несколько секунд раньше нужную информацию означает миллионные прибыли. Но биржи были и ранее, а конечной целью брокеров всё равно является заработок денег, в таком случае информация является лишь средством для получения материальной выгоды.

Другие под ценностью информации понимают выход в лидирующие позиции в промышленном производстве средств обмена, обработки и хранения информационных ресурсов. И они тоже правы. Одной из отраслей, которая обладает наибольшей добавочной стоимостью, является производство различных «гаджетов», таких как компьютеры и телефоны. А если к этому добавить еще и огромный перечень профессиональной техники – фото-, видеокамеры, рации, коммутаторы, спутниковая связь, серверная и офисная техника, то есть всё то что общается волшебным языком «01101», то мы поймём, что просто окружены средствами информационных технологий. Но здесь опять мы сталкиваемся с тем, что доминирующую роль играет человек, и пока роботы не захватили власть на планете, как предрекают некоторые фантасты, все эти технологии находятся на службе у человека.

Ещё одна точка зрения о смысле определения – «информационный век», говорит о том, что существует какая-то критически важная информация, обладание которой обеспечивает безопасность целых государств или их развитие. Обладание знаниями обеспечивает экономический рост, а наличие высокотехнологичных научных разработок, позволяет создать высокоэффективное оружие, что обеспечивает безопасность государства. Ярким примером здесь является наличие у некоторых стран технологии создания атомного оружия. Особенно явно это было на начальном этапе гонки вооружений двух сверхдержав – США и СССР. Тогда отставание или доминирование одной из стран означало вопрос жизни или смерти для другой. Перечень такой крайне важной информации можно продолжать ещё долго – это и точные сроки начала войны, и количество воинских подразделений, и материальные ресурсы. Такая информация действительно крайне важна для судьбы многих тысяч людей. Но эти проблемы существовали и много лет назад, для этого и создавались разведки, шпионы, учёные, и так далее. Поэтому такое объяснение также не совсем подходит к современному веку информации.

Есть ещё множество других объяснений и толкований этого понятия. Я уверен, что многие читатели имеют свое объяснение понятия «века информационных технологий». Я предлагаю рассмотреть подробнее ещё одну точку зрения на то, почему именно сейчас наступил век информационных технологий и как существующие новшества влияют на жизнь.

## **Конец ознакомительного фрагмента.**

Текст предоставлен ООО «ЛитРес».

Прочитайте эту книгу целиком, [купив полную легальную версию](#) на ЛитРес.

Безопасно оплатить книгу можно банковской картой Visa, MasterCard, Maestro, со счета мобильного телефона, с платежного терминала, в салоне МТС или Связной, через PayPal, WebMoney, Яндекс.Деньги, QIWI Кошелек, бонусными картами или другим удобным Вам способом.