



PWNING GAME SERVERS

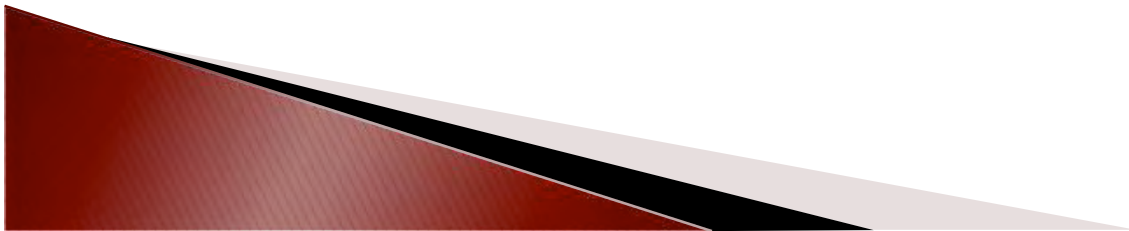
WELL, REALLY, JUST SRCDS, BUT WHO REALLY CARES?

BRUCE POTTER, LOGAN LODGE

GDEAD@SHMOO.COM, LOLO@SHMOO.COM

GAMING IS BIG

- ▶ GAMING IS A USD47 BILLION GLOBAL MARKET.
 - CONSOLE GAMING ALONE IS ESTIMATED TO BE USD27 BILLION IN SIZE.
 - PC ONLINE GAMES IS A USD6.5 BILLION INDUSTRY. PROJECTED TO BE USD13 BILLION BY 2012.
 - ONLINE MMOGS IS A USD3.5 BILLION INDUSTRY.
 - CASUAL GAMING IS A USD1.5 BILLION INDUSTRY.
 - READ MORE:
[HTTP://WWW.TECHVIBES.COM/BLOG/GAMEON-FINANCE-2.0-KEY-GAMING-INDUSTRY-TRENDS-AND-MARKET-OVERVIEW#IXZZOCH84UUJO&B](http://www.techvibes.com/blog/gameon-finance-2.0-key-gaming-industry-trends-and-market-overview#IXZZOCH84UUJO&B)

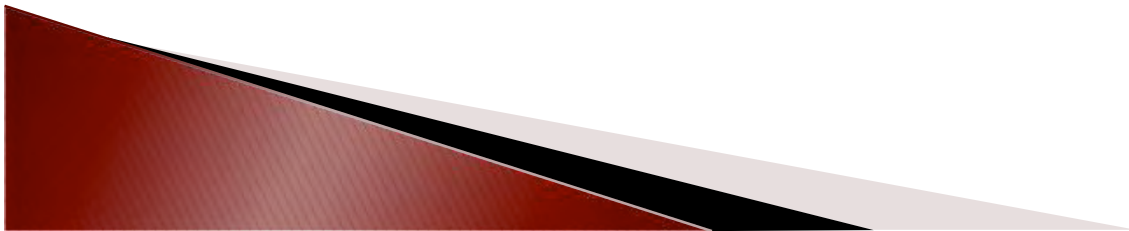


GAMING STATS



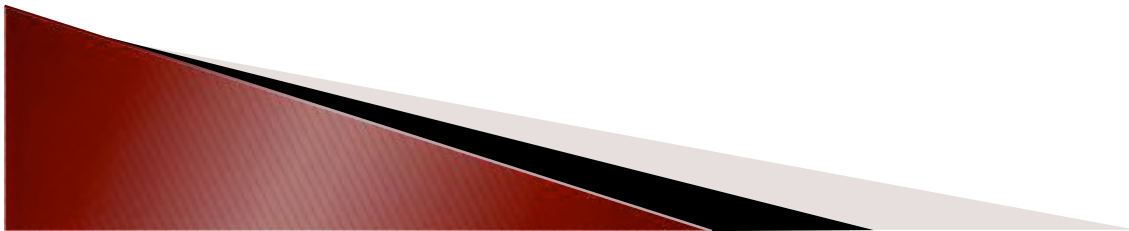
TALKING ABOUT GAMING ISN'T

- ▶ AT LEAST NOT TALKING ABOUT THEM IN A COHERENT FASHION
 - THERE ARE A LOT OF REVIEWS THAT FOCUS ON GAMEPLAY
 - THERE ARE INDUSTRY ANALYSIS SITES
 - THERE ARE MANY REVIEWS OF GAMING HARDWARE
 - FEW PUBLIC DISCUSSIONS OF SECURITY
 - FEW PUBLIC DISCUSSIONS REGARDING THE MERITS AND IMPACT OF UNDERLYING TECHNOLOGY
 - BEGINNING TO SEE CULTURAL/ANTHROPOLOGICAL DISCUSSIONS



TODAY, LET'S TALK ABOUT GAME SERVERS

- ▶ PUBLISHER PROVIDED GAME SERVERS
 - PRETTY MUCH ALL CONSOLE GAMING
 - SOME PC GAMES SUCH AS WOW
- ▶ COMMUNITY DRIVEN GAME SERVERS
- ▶ WHAT DRIVES PEOPLE TO RUN GAME SERVERS?
 - IT'S A LOT LIKE OSS
 - CONVENIENCE, FAME, MONEY, FUN



GAMES PEOPLE ARE PLAYING

THERE ARE OTHERS.. QUAKE, UT, ETC..

CALL OF DUTY

- COD2
- COD4
- COD WW



HALF-LIFE

- HALF LIFE
- TFC
- COUNTER STRIKE



SOURCE ENGINE

- HL2DM
- TF2
- LEFT4DEAD
- CS:SOURCE



BATTLEFIELD

- BF
- BF2



FOR THE REST OF THIS TALK, WE'LL FOCUS ON SOURCE DEDICATED SERVER (SRCDS)

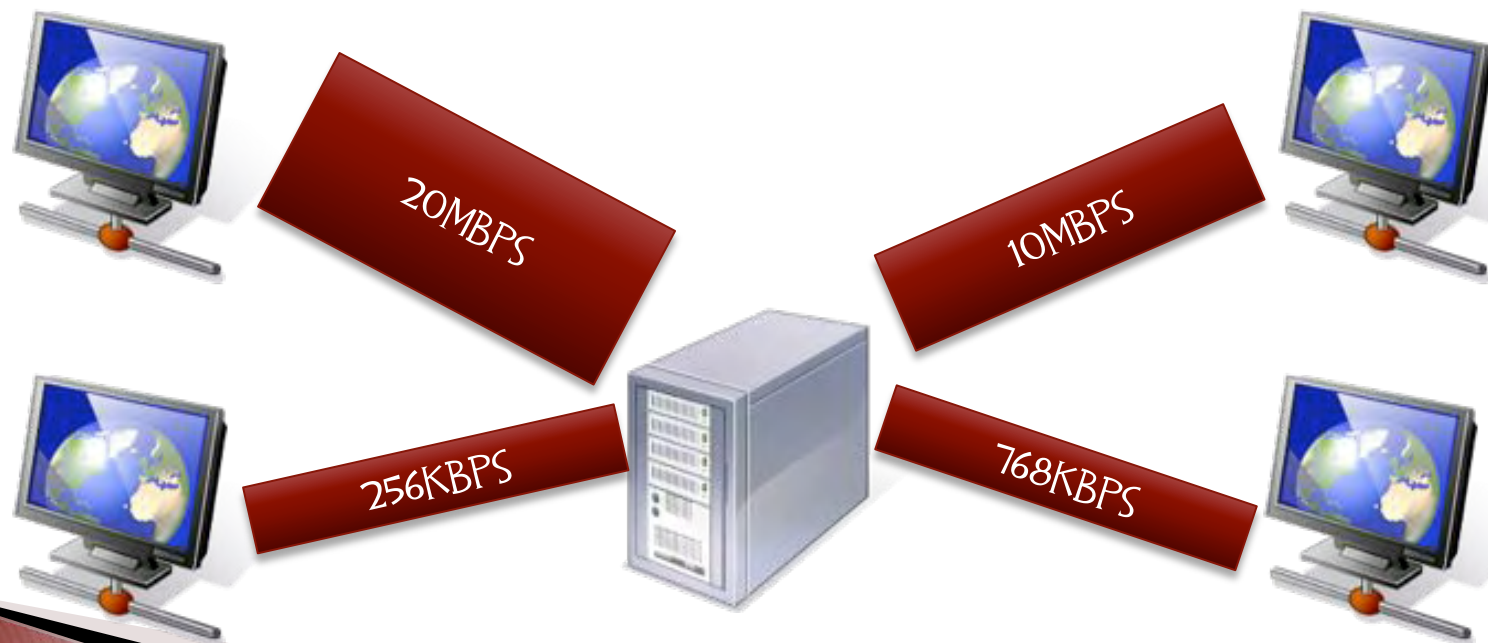
▶ WHY?

- VALVE HAS CREATED A PLATFORM THAT NOT ONLY SUPPORTS THEIR NEEDS, BUT ALLOWS FOR MASSIVE AMOUNTS OF CUSTOMIZATION
- HUGE NUMBER OF SERVERS DEPLOYED
 - GAMETRACKER.COM HAS CS:S AS #2 (10700 SERVERS, 25000 PLAYERS), L4D AS #6 (3000 SERVERS, 3000 USERS), TF2 AS #9 (2300 SERVERS, 8500 USERS) AT 8PM ON A MONDAY
 - STEAM HAS ~1.5MILLION ACTIVE USERS EACH DAY... MANY OF THEM PLAYING SOURCE-BASED GAMES
- THE REAL REASON: I PLAY A LOT OF TF2
 - 150 HOURS ON DEMOMAN.. YEAH, IT'S A PROBLEM



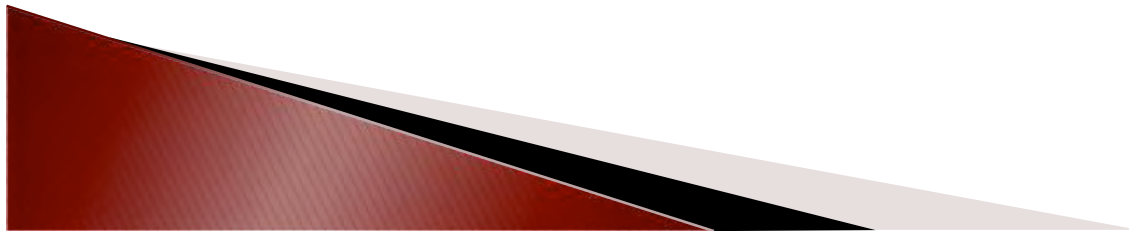
HOW SOURCE SERVER WORKS

- ▶ THE BIG CHALLENGE: PROVIDE A GOOD GAMING EXPERIENCE FOR THE PERSON WITH A 1.6GHZ PIV ON A 256K DSL LINE AND A PERSON WITH AN I7 ON THE ASS END OF A 25MBPS FIOS CONNECTION



SERVER INTERNALS

- ▶ SRDS IS A COMPLEX PIECE OF SOFTWARE
 - TRIES TO PROVIDE REAL TIME SERVICE ON OS'S THAT AREN'T RTOS
 - ENFORCES COMPLEX MATHEMATICAL MODELS OF WHERE PLAYERS ARE, WHERE THEY ARE GOING, AND WHAT THEY'RE DOING
 - DISTRIBUTES CONTENT TO CLIENTS THAT NEED IT
 - ATTEMPTS TO CONTROL CHEATING
 - ALLOWS SPECTATING OF THE MATCHES
 - SUPPORTS REMOTE ADMINISTRATION
 - IS HIGHLY EXTENSIBLE
- ▶ PRETTY IMPRESSIVE FOR FREE
 - BUT THEN AGAIN, THE BETTER IT IS, THE MORE PEOPLE WILL STAND UP SERVERS, AND THE MORE PEOPLE WILL BUY THE CLIENT AND PLAY



WHAT MAKES TF2 SO POPULAR?

▶ SIMPLE PREMISE

- SIMPLE PREMISE IS KEY
 - PONG HAD 8 WORDS ON FRONT... "AVOID MISSING BALL FOR HIGH SCORE" AND "INSERT COIN"
- RED TEAM? KILL BLUE.
- BLUE TEAM? KILL RED.
- SOMETIMES THERE'S A CART, OR FLAG, OR SOMETHING.. BUT MOSTLY IT'S ABOUT DESTROYING THE OTHER SIDE

▶ ATTENTION TO DETAIL ON ART DIRECTION AND SUPPORTING TECHNOLOGY

- SERIOUSLY, TAKE A LOOK AT THIS
- [HTTP://WWW.VALVESOFTWARE.COM/PUBLICATIONS/2007/NPAR07_ILLUSTRATIVERENDERINGINTEAMFORTRESS2_SLIDES.PDF](http://www.valvesoftware.com/publications/2007/NPAR07_ILLUSTRATIVERENDERINGINTEAMFORTRESS2_SLIDES.PDF)




REMOTE ADMINISTRATION


- ▶ YOU CAN'T ALWAYS BE SITTING IN FRONT OF YOUR SERVER TO CHANGE THE SETTINGS
- ▶ RCON IS THE SRCDS MECHANISM FOR SENDING GAME COMMANDS TO THE SERVER
 - CHANGE NUMBER OF ROUNDS, RATES, LEVEL, BAN, KICK, ETC..
 - CAN BE SENT THROUGH THE GAME VIA CONSOLE
 - ALSO THIRD PARTY SCRIPTS LIKE SRCDS.PY
- ▶ DANGER: RCON ACCESS IS FUNCTIONALLY EQUIVALENT TO SHELL ACCESS
 - CAN EXECUTE PROGRAMS AND SAVE FILES WITH THE PRIVILEGE OF THE USER RUNNING SRCDS
 - DON'T RUN AS ROOT!



PLUGINS

- ▶ SRCDS HAS A ROBUST SET OF THIRD PARTY PLUG-INS
 - CUSTOM SOUNDS
 - GAMEPLAY MODIFICATIONS
 - PROTECTION MECHANISMS
 - SERVER ADMINISTRATION
 - KICKS/BANS
 - ▶ METAMOD, AS AN EXAMPLE, PROVIDES A CLEAN INTERFACE FOR PLUGIN WRITERS TO THE SRCDS ENGINE
 - SOURCEMOD IS A POPULAR ADMIN AND GAMEPLAY MOD ENGINE THAT USES METAMOD
 - ▶ RATHER THAN GIVING OUT RCON PASSWORDS, USE SOMETHING LIKE SOURCEMOD
- 

PATCH MANAGEMENT

- ▶ VALVE RELEASES PATCHES THAT CAN BE APPLIED AUTOMATICALLY VIA THEIR UPDATE TOOL
 - ▶ VALVE RELEASES PATCHES... UH.. “WHENEVER” CAN BE DISRUPTIVE TO SERVER ADMINS
 - ▶ SRCDS IS HIGHLY OPTIMIZED FOR DIFFERENT PLATFORMS
 - PATCHES CAN CAUSE ISSUES ON AMD BUT NOT INTEL, FOR EXAMPLE
 - ▶ DIFFERENT GAMES CAN BE BROKEN BY DIFFERENT PATCHES
 - ▶ OVER TIME, THE GAMES BLOAT.. COUNT ON IT
- 

CHEATING

- ▶ CHEATING COMES IN MANY SHAPES AND SIZES
- ▶ WITH SRCDS, THERE ARE MANY CHEATING MECHANISMS “BUILT IN”
 - MATERIALS, SOUNDS, ETC CAN ALL BE CUSTOMIZED ON BOTH THE SERVER *AND* THE CLIENT SIDE
 - OBVIOUSLY, CAN BE USED TO MAKE THE GAME MORE UNIQUE AND FUN
 - IT CAN ALSO BE USED TO GIVE YOURSELF AN ADVANTAGE



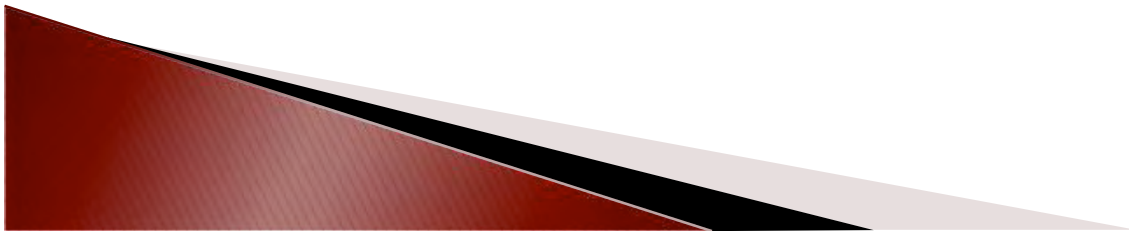
WHAT A DIFFERENCE A SOUND CAN MAKE

- ▶ It's a movie... doesn't work in a PDF. Download this preso from www.nomoose.org to watch



REPLACE THE UNCLOAK SOUND...

- ▶ It's a movie... doesn't work in a PDF. Download this preso from www.nomoose.org to watch



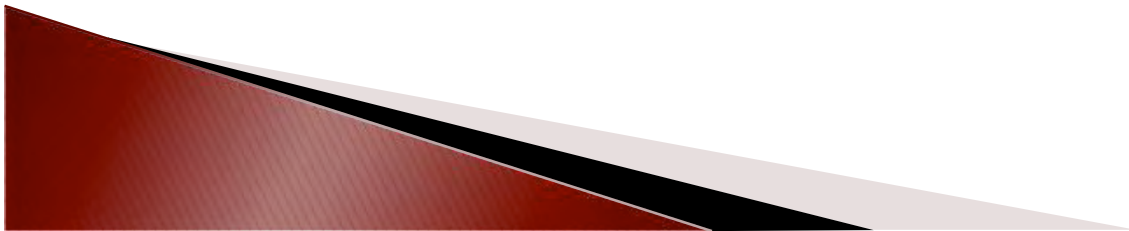
LET'S CHANGE SOME MATERIALS...

- ▶ It's a movie... doesn't work in a PDF. Download this preso from www.nomoose.org to watch



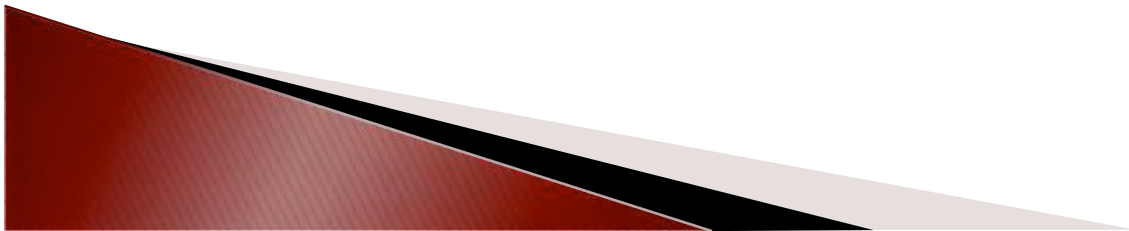
LITTLE MORE ADVANCED...

- ▶ It's a movie... doesn't work in a PDF. Download this preso from www.nomoose.org to watch



AND THEN JUST CARNAGE...

- ▶ It's a movie... doesn't work in a PDF. Download this preso from www.nomoose.org to watch

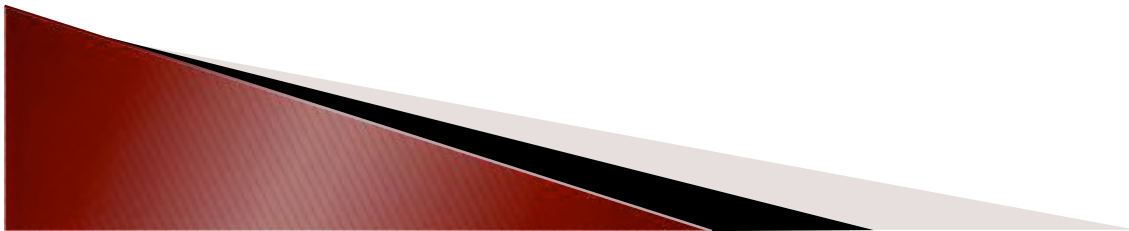


STOPPING CLIENT SIDE CHANGES

- ▶ VALVE IMPLEMENTED A GAME VARIABLE, SV_PURE, TO TRY AND CONTROL THIS
 - SV_PURE=0 IS THE DEFAULT. NO ENFORCEMENT
 - SV_PURE=1 CAUSES THE CLIENT TO SCAN THE MATERIALS, SOUNDS, AND MODELS TO VERIFY THEY'RE THE SAME AS THE ORIGINAL VALVE CONTENT
 - SOME CUSTOM CONTENT IS ALLOWED (SPRAYS AND SUCH)
 - CUSTOM MATERIALS CAN BE WHITELISTED SERVER SIDE
 - SV_PURE=2 RESULTS IN NO CUSTOM CONTENT
- ▶ SV_PURE INCREASES LOAD TIME
- ▶ SV_PURE USES CRC32
 - FINDING A COLLISION IN CRC32 IS A BIT EASIER THAN MD5 ;)

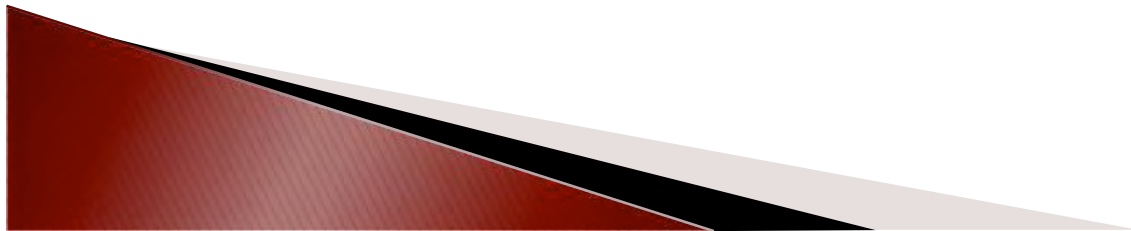
DARKSTORM V.2.6

- ▶ Darkstorm is a publicly available cheat written by Calvin (<http://www.projectvdc.com/wordpress/>) and other members from the Game-Deception web forum Credits to: (Patrick, wav, tabris, Lawgiver, aVitamin, gir489, CampStaff, and s0beit)
- ▶ Open source code
- ▶ Written in C++
- ▶ Lots of cheats available



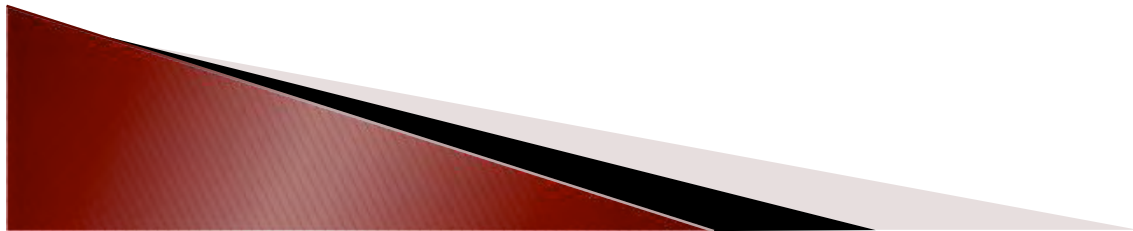
STEP 1 – LOAD IT

- ▶ LOAD OUR DLL
- ▶ STANDARD DLL INJECTION TECHNIQUES APPLY
 - Get Process ID (via: name of Window or Process name)
 - Allocate space in process' virtual address space
 - Create a remote thread in the target's process space and have it kick off your DLL
- ▶ Darkstorm injects into the hl2.exe process



HIDING FROM VAC2

- ▶ REMOVE THE PE HEADER
- ▶ UNLINK OUR MODULE FROM THE PEB LINKED LIST
- ▶ DETOURS TO HOOK VARIOUS API CALLS
- ▶ PE RANDOMIZER TO MAKE SIGNATURE BASED DETECTION MORE CHALLENGING (CREDIT: CHT1)
- ▶ THESE METHODS SOUND FAMILIAR...WHERE HAVE WE SEEN THEM BEFORE? VIRUS? USERLAND ROOTKIT?



DARKSTORM VAC HIDING

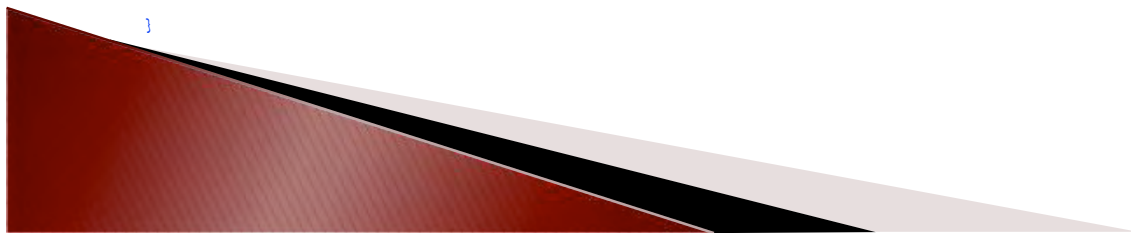
- ▶ DARKSTORM ONLY USES ONE OF THESE METHODS (CONTAINS CODE FOR PEB UNLINKING, BUT NOT IN USE)

```
VOID CMEMORYTOOLS::REMOVEPEHEADER( DWORD DWMODULEBASE )
{
    PIMAGE_DOS_HEADER PDOSHEADER = (PIMAGE_DOS_HEADER)DWMODULEBASE;
    PIMAGE_NT_HEADERS PNTHEADER = (PIMAGE_NT_HEADERS)( DWMODULEBASE + (DWORD)PDOSHEADER->E_LFANEW );

    IF(PDOSHEADER->E_MAGIC != IMAGE_DOS_SIGNATURE) //VALID PE HEADER?
        RETURN;

    IF(PNTHEADER->SIGNATURE != IMAGE_NT_SIGNATURE) //VALID PE HEADER?
        RETURN;

    IF(PNTHEADER->FILEHEADER.SIZEOFOPTIONALHEADER)
    {
        DWORD DWPROTECT;
        WORD SIZE = PNTHEADER->FILEHEADER.SIZEOFOPTIONALHEADER; //POINTER TO THE OPTIONAL HEADER PORTION OF THE PE
        VIRTUALPROTECT( (PVOID)DWMODULEBASE, SIZE, PAGE_EXECUTE_READWRITE, &DWPROTECT ); //ALLOW US TO WRITE
        RTLZEROMEMORY( (PVOID)DWMODULEBASE, SIZE ); //ZERO IT OUT
        VIRTUALPROTECT( (PVOID)DWMODULEBASE, SIZE, DWPROTECT, &DWPROTECT ); //RESET THE PERMISSIONS
    }
}
```



UNDOING SOME PROTECTION

```
VOID UNPROTECTCVARS( VOID )
{
    CONCOMMAND *PVAR = (CONCOMMAND*)G_PCVAR->GETCOMMANDS( );           //POINTER TO LIST OF COMMANDS

    CONVAR *PCONSISTENCY = G_PCVAR->FINDVAR( 'SV_CONSISTENCY' );         //POINTER TO SV_CONSISTENCY
    CONVAR *PCHEATS = G_PCVAR->FINDVAR( 'SV_CHEATS' );                   //POINTER TO SV_CHEATS

    WHILE( PVAR )                                                       //CYCLE THROUGH COMMANDS
    {
        IF( PVAR->ISFLAGSET( FCVAR_CHEAT ) )
            PVAR->M_NFLAGS &- FCVAR_CHEAT;                             //FLIP THE BIT FOR FCVAR_CHEAT
        IF( PVAR->ISFLAGSET( FCVAR_REPLICATED ) )
            PVAR->M_NFLAGS &- FCVAR_REPLICATED;                       //FLIP THE BIT FOR FCVAR_REPLICATED
        IF( PVAR->ISFLAGSET( FCVAR_PROTECTED ) )
            PVAR->M_NFLAGS &- FCVAR_PROTECTED;                       //FLIP THE BIT FOR FCVAR_PROTECTED
        IF( PVAR->ISFLAGSET( FCVAR_SPONLY ) )
            PVAR->M_NFLAGS &- FCVAR_SPONLY;                           //FLIP THE BIT FOR FCVAR_SPONLY
        PVAR = (CONCOMMAND*)PVAR->GETNEXT( );
    }

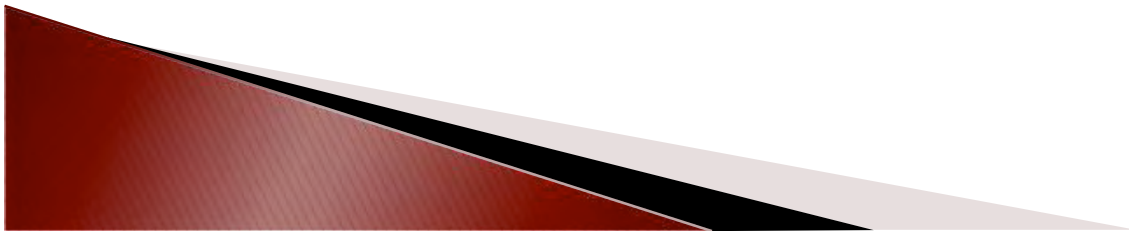
    PCONSISTENCY->SETVALUE( 0 );
    PCHEATS->SETVALUE( 1 );                                           //ALLOW 'CHEAT' COMMANDS TO BE RUN SERVER SIDE
}
}
```



CREATEMOVE

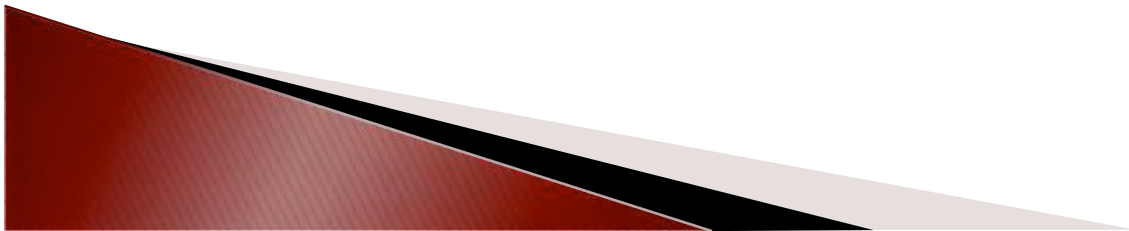
```
void __stdcall Hooked_CreateMove( int sequence_number, float input_sample_frametime, bool active )
```

- ▶ Called on every tick of the game
- ▶ Essentially the entry point for our code to run
- ▶ From here it cycles through the enabled cheats and executes the appropriate routines



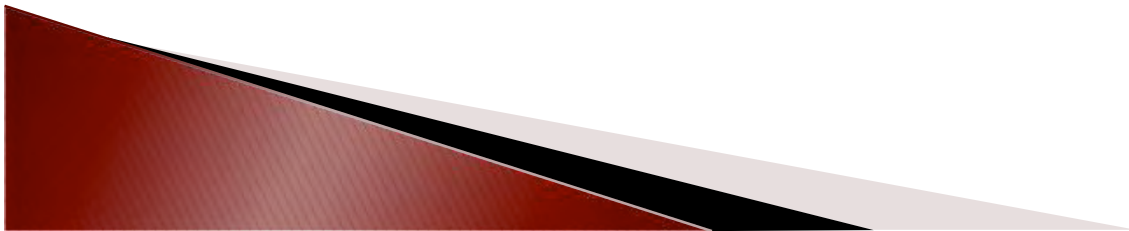
AIMBOT

- ▶ This is the first stop for this cheat code
- ▶ (Un)fortunately the most recent update to TF2 broke this cheat
- ▶ There was a class method (`EyeAngles()`) that was made private in the latest update
- ▶ The AIMBOT relied on this method to 'Aim'



AIMBOT – CONT.

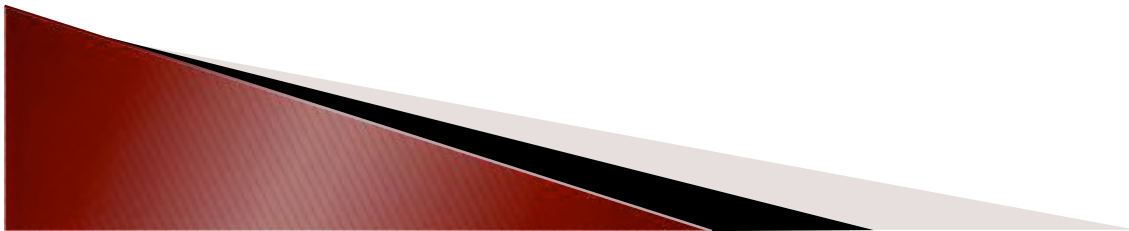
- ▶ Source code for this is too long, so a quick overview:
 - Get list of entities (players & objects)
 - Found a player? not me? Solid? other team?
 - Get my loc and fov, get vector from me to target. Is target in my fov?
 - Change my Eye position to face target's hitbox
 - There are 5 to choose from. But really, who's going to choose one other than the head...
 - Fire!



SPEEDHACK

```
if( gCvars.misc_speed_on && g_pCvar ) //is the speed hack enabled and do we have an interface to console commands
{
    ConVar* pSpeed = g_pCvar->FindVar( "host_timescale" ); //pointer to CVAR host_timescale

    if( pSpeed )
    {
        if( gCvars.misc_speed > 1 && blsSpeedKey( gCvars.misc_speed_key ) ) //hack enabled? button pushed?
        {
            pSpeed->SetValue( gCvars.misc_speed ); //set the CVAR to the specified value
        }
        else
        {
            pSpeed->SetValue( 1.0f ); //set CVAR to 1
        }
    }
}
```



RAPID FIRE

```
if( gCvars.misc_autopistol && pCommand->buttons & IN_ATTACK && //cheat enabled, do we have access to the required interface
    ( iGetWeaponID( pBaseWeapon ) == WEAPONLIST_SCOUTPISTOL || //are we holding the right weapon?
      iGetWeaponID( pBaseWeapon ) == WEAPONLIST_ENGINEERPISTOL ||
      iGetWeaponID( pBaseWeapon ) == WEAPONLIST_SPYPISTOL ) )
{
    static bool bInAttack = false; //stores whether we're attacking or not, set externally
    if ( bInAttack ) //if we're attacking (i.e. pushing the left mouse button)
        pCommand->buttons |= IN_ATTACK; //flip the bit that says you're firing (rapid fire)
    else
        pCommand->buttons &= ~IN_ATTACK; //else, flip it back

    bInAttack = !bInAttack; //reset our state
}
```




CONSTANT CRITS

```
__asm
{
    mov ecx, pBaseWeapon;    //structure containing weapon info (crit chance, etc.)
    mov eax, [ecx+0x16B4];    //grab persistent seed
    push eax;                //save it
    mov eax, [ecx];
    mov eax, [eax+0x528];    //IsShotCritical
    call eax;
    mov iResult, eax;        //Save value at AL - 1 for crit, 0 for sad panda
    pop eax;
    mov ecx, pBaseWeapon;
    mov [ecx+0x16B4], eax;    //restore persistent seed
}

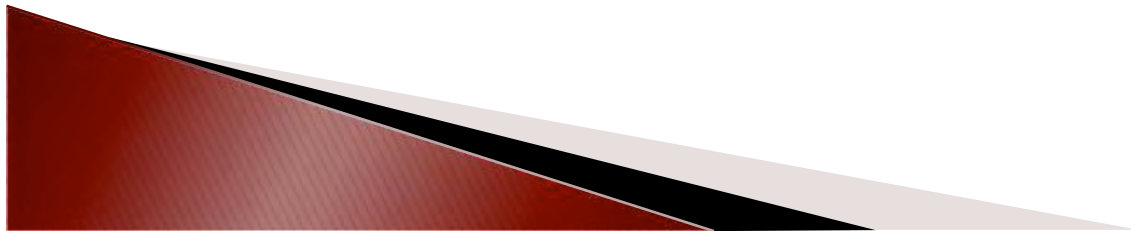
if( pCommand->buttons & IN_ATTACK )
{
    pCommand->buttons &= ~IN_ATTACK;    //not crit time, cry some twinkletoes
    bWaitFire = true;
}

if( bWaitFire && (BYTE)iResult )
{
    pCommand->buttons |= IN_ATTACK;    //crit time, attack!
    bWaitFire = false;
}
```



PROTOCOL BREAKDOWN

- ▶ DEMO OF WIRESHARK DISSECTOR FOR SRCDS TRAFFIC
- ▶ [HTTP://WWW.SHMOO.COM/SRCDS/](http://www.shmoo.com/srcds/)



QUESTIONS?

- ▶ THERE'S A LOT MORE HERE.. BUT IT'S A START
- ▶ INTERESTED? CAPTURE WHAT YOU LEARN AND SHARE IT B/C THERE ARE OTHERS WHO RE INVENTING THE WHEEL EVERY DAY
- ▶ WWW.NOMOOSE.ORG
- ▶ GDEAD@SHMOO.COM, LOLO@SHMOO.COM

