GAME ENGINES: A 0-DAY'S TALE

Luigi Auriemma[1] and Donato Ferrante[2]
ReVuln
http://revuln.com
info@revuln.com
http://twitter.com/revuln
17 May 2013

Abstract *This paper details several issues affecting different game engines. All the vulnerabilities discussed in this paper are 0-days, at time of writing. This paper has been released as a companion paper along with the authors' talk Exploiting Game Engines For Fun And Profit[3] presented at the NoSuchCon conference[4].*

CONTENTS

[1] http://twitter.com/luigi_auriemma
[2] http://twitter.com/dntbug
[3] http://revuln.com/files/Ferrante_Auriemma_Exploiting_Game_Engines.pdf
[4] http://www.nosuchcon.com

1 BREACH

From Wikipedia[5]: "Breach is a team-based first-person shooter multiplayer video game developed by Atomic Games. It was announced on March 26, 2010 at PAX East 2010 for Windows PCs and the Xbox 360. Breach was distributed online for the Xbox 360 by Xbox Live Arcade, and on Windows by Steam. It features dynamic destructible environments and a cover system". Breach is based on the game engine: *Hydrogen Engine*.

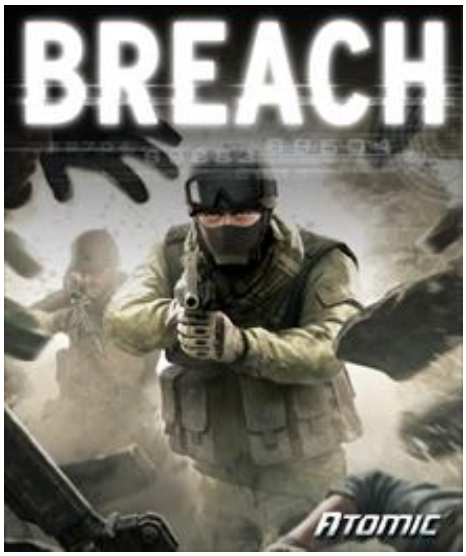

Figure 1: Breach

1.1 OVERVIEW

There are four different issues affecting the *Hydrogen Engine*, two of them are related to *Index Numbers*. Please refer to our presentation slides[6] for additional information about this strategy to optimize data-representation.

1.2 INDEX NUMBERS OFF-BY-TWO

The engine uses index numbers for storing numbers inside the packets. This solution is adopted to save space since only a part of the 64-bit number is transmitted. The functions (*BreachServer*+0x6992b0, *BreachServer*+0x698f90) that read the index numbers are affected by an *off-by-two stack based overflow* caused by the support of 80-bit numbers against a 64-bit buffer.

```
00A992B0 SUB ESP,0C
00A992B3 MOV ECX,DWORD PTR SS:[ESP+14]
[...]
00A9936F CMP ESI,50
00A99372 JB BreachServer.00A992D1
```

[5] http://en.wikipedia.org/wiki/Breach_%28video_game%29
[6] http://revuln.com/files/Ferrante_Auriemma_Exploiting_Game_Engines.pdf

1.3 32-bit Variables Used as Index Numbers

Some of the functions that call the previous functions for reading the index numbers pass 32-bit variables located on the stack and it's possible to overflow them by writing 64-bit values.

```
00A992B0 SUB ESP,0C
00A992B3 MOV ECX,DWORD PTR SS:[ESP+14]
00A992B7 PUSH EBX
00A992B8 PUSH EBP
00A992B9 PUSH ESI
00A992BA XOR ESI,ESI
00A992BC XOR EBP,EBP
00A992BE PUSH EDI
00A992BF OR EAX,FFFFFFFF
00A992C2 MOV DWORD PTR SS:[ESP+14],ESI
00A992C6 MOV DWORD PTR SS:[ESP+18],ESI
00A992CA MOV DWORD PTR DS:[ECX],ESI
00A992CC MOV DWORD PTR DS:[ECX+4],ESI
00A992CF MOV BL,1
00A992D1 TEST EAX,EAX
00A992D3 JE BreachServer.00A99378
00A992D9 MOV ECX,DWORD PTR SS:[ESP+20]
00A992DD MOV EAX,DWORD PTR DS:[ECX] ; ctrl'd via overflow
00A992DF MOV EAX,DWORD PTR DS:[EAX+54]
00A992E2 PUSH 5
00A992E4 LEA EDX,DWORD PTR SS:[ESP+17]
00A992E8 PUSH EDX
00A992E9 CALL EAX  ; read 5 bits (code execution)
[..] ; copy them in EDI
00A99363 INC ESI
00A99364 ROL AL,1
00A99366 MOV BL,AL
00A99368 MOVZX EAX,CL
00A9936B INC EBP
00A9936C AND EAX,10 ; if 5 bits & 0x10 continue
00A9936F CMP ESI,50
00A99372 JB BreachServer.00A992D1
```

1.4 Reallocation Integer Overflow

The following function is used to reallocate buffers and read packet contents. The function optimizes the reading of small blocks of data by using a stack buffer of 64 bytes and it's possible to overflow it by specifying the size 0x7fffffff to which 1 is added to bypass a signed comparison:

```
004D6D32 MOV EAX,DWORD PTR SS:[ESP+10] ; our 32-bit size
004D6D36 TEST EAX,EAX
004D6D38 JGE SHORT BreachServer.004D6D55
004D6D3A MOV DWORD PTR SS:[ESP+90],-1
```

```
004D6D45 LEA ECX,DWORD PTR SS:[ESP+14]
004D6D49 CALL BreachServer.00A98E20
004D6D4E XOR EAX,EAX
004D6D50 JMP BreachServer.004D6E01
004D6D55 INC EAX ; +1
004D6D56 PUSH EAX
004D6D57 LEA ECX,DWORD PTR SS:[ESP+18]
004D6D5B CALL BreachServer.00A98CA0 ; realloc function
[...]
00A98CA0 PUSH ESI
00A98CA1 PUSH EDI
00A98CA2 MOV EDI,DWORD PTR SS:[ESP+C]
00A98CA6 CMP EDI,40
00A98CA9 MOV ESI,ECX
00A98CAB JG SHORT BreachServer.00A98CC3 ; signed comparison
```

1.5 Double Free

There is a double-free vulnerability affecting the following code:

```
004D65D4 PUSH 8
004D65D6 LEA EAX,DWORD PTR SS:[ESP+20]
004D65DA PUSH EAX
004D65DB MOV ECX,ESI
004D65DD CALL EDX ; read 8-bit size
[...]
004D65F6 JL SHORT BreachServer.004D6631 ; read data if > 0
[...]
004D65FD CALL BreachServer.00A38470 ; allocation
[...]
004D6620 CALL EDX ; read
[...]
004D6635 CALL BreachServer.00A41530 ; strdup
[...]
004D6631 PUSH EDI
004D6632 LEA ECX,DWORD PTR DS:[EBX+44]
004D6635 CALL BreachServer.00A41530
004D663A TEST EDI,EDI
004D663C JE SHORT BreachServer.004D6647
004D663E PUSH EDI
004D663F CALL BreachServer.00A384B0 ; free
004D6644 ADD ESP,4
004D6647 MOV EDX,DWORD PTR DS:[ESI]
004D6649 MOV EDX,DWORD PTR DS:[EDX+54]
004D664C PUSH 8
004D664E LEA EAX,DWORD PTR SS:[ESP+24]
004D6652 PUSH EAX
004D6653 MOV ECX,ESI
004D6655 CALL EDX ; read 8-bit size
```

```
[...]
004D6662 JL SHORT BreachServer.004D66B2 ; read data if > 0
[...]
004D66B2 PUSH EDI ; EDI has the old value
004D66B3 LEA ECX,DWORD PTR DS:[EBX+48]
004D66B6 CALL BreachServer.00A41530
004D66BB TEST EDI,EDI
004D66BD JE SHORT BreachServer.004D66C8
004D66BF PUSH EDI
004D66C0 CALL BreachServer.00A384B0 ; double-free
```

2 Unreal Engine 3

From Wikipedia[7]: "The third and current generation of the Unreal Engine (UE3) is designed for DirectX (versions 9-11 for Windows and Xbox 360), as well as systems using OpenGL, including the PlayStation 3, OS X, iOS, Android, Stage 3D for Adobe Flash Player 11, JavaScript/WebGL, PlayStation Vita and Wii U. Its renderer supports many advanced techniques including HDRR, per-pixel lighting, and dynamic shadows. It also builds on the tools available in previous versions. In October 2011, the engine was ported to support Adobe Flash Player 11 through the Stage 3D hardware-accelerated APIs. Epic has used this version of the engine for their in-house games. Aggressive licensing of this iteration has garnered a great deal of support from many prominent licensees. Epic has announced that Unreal Engine 3 runs on both Windows 8 and Windows RT. In addition to the game industry, UE3 has also seen adoption by many non-gaming projects".

Figure 2: A game based on Unreal Engine 3

The following sections detail vulnerabilities affecting games based on the *Unreal Engine 3*.

[7] https://en.wikipedia.org/wiki/Unreal_Engine

3 Monday Night Combat

From Wikipedia[8]: "Monday Night Combat is a downloadable third-person shooter video game developed by Uber Entertainment. It was published by Microsoft Studios on the Xbox 360 and by Uber Entertainment and Valve Software for Microsoft Windows. It was released on August 11, 2010 on the Xbox 360 as part of Microsoft's 2010 Xbox Live Summer of Arcade and is distributed through Xbox Live Arcade It was released on January 24, 2011 for Windows via Steam. [...] As of year-end 2011, Monday Night Combat has sold over 307,000 copies on Xbox Live Arcade".

Figure 3: Monday Night Combat

3.1 Overview

Monday Night Combat is based on *Unreal Engine 3*, and like other games supporting the Steam platform, this game supports some custom Steam related commands via *Unreal Engine Control channel*[9].

3.2 Array Overflow

There is an array overflow, which allows attackers to corrupt the heap and allow remote code execution. This problem relies on the way the engine implements some *Steam*[10] commands.

[8] http://en.wikipedia.org/wiki/Monday_Night_Combat
[9] http://udn.epicgames.com/Three/NetworkingOverview.html
[10] http://www.steampowered.com

```
00A8B9DA MOVZX EAX,BYTE PTR SS:[ESP+5F] ; SUBBLOB 8bit
00A8B9DF MOV EDX,DWORD PTR DS:[ESI+4FD8]
00A8B9E5 LEA EAX,DWORD PTR DS:[EAX+EAX*2]
00A8B9E8 ADD EAX,EAX
00A8B9EA ADD EAX,EAX
00A8B9EC MOV ECX,DWORD PTR DS:[EDX+EAX+4] ; ar[SUBBLOB][12]
00A8B9F0 CMP ECX,EBP ; ECX must be 0 or 1
00A8B9F2 JE SHORT MNCDS.00A8B9FD
00A8B9F4 DEC ECX
00A8B9F5 CMP ECX,EBP
00A8B9F7 JNZ MNCDS.00A8B8FF
00A8B9FD LEA ECX,DWORD PTR SS:[ESP+28]
00A8BA01 PUSH ECX
00A8BA02 LEA ECX,DWORD PTR DS:[EDX+EAX] ; heap corruption
                                        ; w. AUTHBLOBSTRING
00A8BA05 CALL MNCDS.00477FD0
00A8BA0A LEA ECX,DWORD PTR SS:[ESP+28]
00A8BA0E MOV DWORD PTR SS:[ESP+8C0],-1
00A8BA19 CALL MNCDS.006C7D70
00A8BA1E JMP MNCDS.00A8C861
```

4 HOMEFRONT

From Wikipedia[11]: "Homefront is a first-person shooter video game developed by Kaos Studios and published by THQ. Players are members of a resistance movement fighting against a near-future North Korean military occupation of the United States. It was released for Microsoft Windows, PlayStation 3 and Xbox 360 on March 15, 2011 in North America, March 17, 2011 in Australia, March 18, 2011 in Europe, and April 14, 2011 in Japan".

Figure 4: Homefront

4.1 OVERVIEW

Homefront is based on a customized version of the *Unreal Engine 3*, with *RCON*[12] support.

4.2 INVALID READ ACCESS

The RCON command *CT* followed by 0x7fffffff triggers an invalid read access, while attempting to read the address 0x7fffffff.

4.3 NULL POINTER

The RCON command *CD* triggers a NULL pointer:

```
00AFBD83 MOV EAX,DWORD PTR DS:[ECX]
00AFBD85 MOV EDX,DWORD PTR DS:[EAX+1C]
00AFBD88 CALL EDX
```

[11] http://en.wikipedia.org/wiki/Homefront_%28video_game%29
[12] http://en.wikipedia.org/wiki/Remote_administration

4.4 16-bit Adjacent Memory Overwrite

The RCON command *CT* followed by a negative number, allows the setting of 16-bit adjacent memory to 0.

```
00AFBECA MOV EDI,DWORD PTR SS:[ESP+28] ; our size
00AFBECE CMP EDI,0FF ; signed comparison
00AFBED4 JLE SHORT HFDedica.00AFBEDB
00AFBED6 MOV EDI,0FF
00AFBEDB MOV EDX,DWORD PTR SS:[ESP+24]
00AFBEDF LEA ECX,DWORD PTR DS:[EDI+1]
00AFBEE2 PUSH ECX
00AFBEE3 SHL EAX,9
00AFBEE6 PUSH EDX
00AFBEE7 LEA EAX,DWORD PTR DS:[EAX+ESI+28]
00AFBEEB PUSH EAX
00AFBEEC CALL HFDedica.0045F7F0 ; wcsncpy_s
00AFBEF1 MOV ECX,DWORD PTR DS:[ESI+10228]
00AFBEF7 SHL ECX,8
00AFBEFA XOR EAX,EAX
00AFBEFC ADD ECX,EDI
00AFBEFE XOR EDX,EDX
00AFBF00 ADD ESP,0C
00AFBF03 CMP DWORD PTR SS:[ESP+2C],EAX ; dst[size*2] = 0
00AFBF07 MOV WORD PTR DS:[ESI+ECX*2+28],DX
```

4.5 Stack-based Overflow

The RCON command *CT* followed by a negative number, can be used to trigger a stack-based overflow:

```
00B03490 MOV EAX,DWORD PTR DS:[EAX+100]
00B03496 LEA EDX,DWORD PTR DS:[ESI+1]
00B03499 PUSH EDX ; our size + 1
00B0349A PUSH EAX ; our string
00B0349B LEA EAX,DWORD PTR SS:[ESP+140]
00B034A2 PUSH EAX ; stack buffer
00B034A3 MOV DWORD PTR SS:[ESP+950],2
00B034AE CALL HFDedica.0045F7F0 ; wcsncpy_s
```

5 The Haunted: Hells Reach

From the Steam store[13]: "The Haunted is a fast paced third person action horror game that focuses on delivering an intense multiplayer experience. Your goal is to liberate cursed places and survive the assault from the minions of Hell. The game features several multiplayer modes such as co-op survival, demons vs. humans and demonizer".

Figure 5: The Haunted: Hells Reach

5.1 Overview

The Haunted doesn't use the standard *Unreal Engine 3* protocol. In this protocol there are some opcodes that allow attackers to perform a *denial of service attack*[14] against the victim's systems.

5.2 OutOfMemory via Custom Opcodes

There are three opcodes, with the following formats:

- 0x16 [32-bit size]
- 0x1b [32-bit size]
- 0x1c [32-bit size]

These opcodes can be used to invoke the *VirtualAlloc*[15] function with arbitrary 32-bit values. Since *VirtualAlloc* returns *NULL* when trying to allocate a 0-byte buffer, it's not possible to use this vulnerability to as an integer overflow. However it's possible to shutdown the server immediately by requesting allocations from 0xfffff001 to 0xffffffff bytes by triggering an *Out Of Memory* condition[16].

[13] http://store.steampowered.com/app/43190/
[14] https://en.wikipedia.org/wiki/Denial-of-service_attack
[15] http://msdn.microsoft.com/en-us/library/windows/desktop/aa366887%28v=vs.85%29.aspx
[16] http://en.wikipedia.org/wiki/Out_of_memory

6 SANCTUM

From Wikipedia[17]: "Sanctum is a first-person shooter tower defense video game, developed by independent developer Coffee Stain Studios. It has been available for pre-purchase via Steam for Microsoft Windows since March 24, 2011. The successor, Sanctum 2 was announced in 2012 and is set for release summer 2013".

Figure 6: Sanctum

6.1 OVERVIEW

Sanctum doesn't use the standard *Unreal Engine 3* protocol. In this protocol there are some opcodes that allow attackers to perform attacks against the victim's systems.

- 0x13 [32-bit size]
- 0x18 [32-bit size]
- 0x19 [32-bit size]

6.2 OUTOFMEMORY VIA CUSTOM OPCODES

If *VirtualAlloc* fails, then the following code is reached:

```
01693367   53             PUSH EBX
01693368   68 5818F101    PUSH OFFSET 01F11858 ; "Core"
0169336D   8D4424 18      LEA EAX,[ESP+18]
01693371   68 DCD72902    PUSH OFFSET 0229D7DC ; "OutOfMemory"
01693376   50             PUSH EAX
01693377   895C24 1C      MOV DWORD PTR SS:[ESP+1C],EBX
0169337B   E8 1017E1FE    CALL 004A4A90
```

[17] http://en.wikipedia.org/wiki/Sanctum_%282011_video_game%29

6.3 Memset-Zero via Custom Opcodes

```
00448300    8B5424 08       MOV EDX,DWORD PTR SS:[ESP+8]
00448304    8B01            MOV EAX,DWORD PTR DS:[ECX]
00448306    8B40 08         MOV EAX,DWORD PTR DS:[EAX+8]
00448309    03D2            ADD EDX,EDX  ; value * 8
0044830B    03D2            ADD EDX,EDX
0044830D    03D2            ADD EDX,EDX
0044830F    895424 08       MOV DWORD PTR SS:[ESP+8],EDX
00448313    FFE0            JMP EAX
...
00457000    53              PUSH EBX  ; ReadBits
00457001    8B5C24 08       MOV EBX,DWORD PTR SS:[ESP+8]
00457005    56              PUSH ESI
00457006    57              PUSH EDI
00457007    8B7C24 14       MOV EDI,DWORD PTR SS:[ESP+14]
0045700B    8D47 07         LEA EAX,[EDI+7]  ; value + 7
0045700E    C1F8 03         SAR EAX,3  ; (signed)value / 8
00457011    50              PUSH EAX
00457012    6A 00           PUSH 0
00457014    53              PUSH EBX
00457015    8BF1            MOV ESI,ECX
00457017    E8 D2878001     CALL <JMP.&MSVCR100.memset>
```

For example, if *size* equals 0x11223344:

- value = 0x11223344
- value * 8 = 0x89119a20
- value + 7 = 0x89119a27
- value / 8 = 0xf1223344

7 idTech 4

From Wikipedia[18]: "id Tech 4, popularly known as the Doom 3 engine, is a game engine developed by id Software and first used in the video game Doom 3. The engine was designed by John Carmack, who also created previous engines such as those for Doom and Quake, which are also widely recognized as marking significant advances in the field. This OpenGL-based game engine has also been used in Quake 4, Prey, Enemy Territory: Quake Wars, Wolfenstein and Brink".

Figure 7: A game based on idTech 4 engine

The engine exposes a function named *idBitMsg::ReadData*, which can be used to achieve remote code execution against games using customized version of this engine. Some games, including *DOOM3* are not affected by this issue. However, others such as *Enemy Territory: Quake Wars* and *Brink*, are affected due to customizations to the original *idTech 4* engine. The *idBitMsg::ReadData* function is defined as follows (code from *DOOM3* GPL source code[19]):

```
int idBitMsg::ReadData( void *data, int length ) const {
 int cnt;
 ReadByteAlign();
 cnt = readCount;

 if ( readCount + length > curSize ) {
  if ( data ) {
   memcpy( data, readData + readCount, GetRemaingData() );
  }
  readCount = curSize;
 } else {
  if ( data ) {
```

[18] http://en.wikipedia.org/wiki/Id_Tech_4
[19] https://github.com/TTimo/doom3.gpl

```
   memcpy( data, readData + readCount, length );
  }
  readCount += length;
 }

 return ( readCount - cnt );
}
```

8 Enemy Territory: Quake Wars

From Wikipedia[20]: "Enemy Territory: Quake Wars (ET:QW) is a tactical shooter video game, and is a prequel to events in Quake II. [..] Quake Wars features similar gameplay to Wolfenstein: Enemy Territory, but with the addition of controllable vehicles and aircraft as well as multiple AI deployables, asymmetric teams, much larger maps and the option of computer-controlled bot opponents. Unlike the previous Enemy Territory games, Quake Wars is a commercial release rather than a free download. Enemy Territory: Quake Wars was developed by Splash Damage using a modified version of id Software's id Tech 4 engine with MegaTexture rendering technology. It was released for Microsoft Windows, Linux, Mac OS X, PlayStation 3 and Xbox 360. The game received mostly positive reviews upon release, although it received some criticism on consoles".

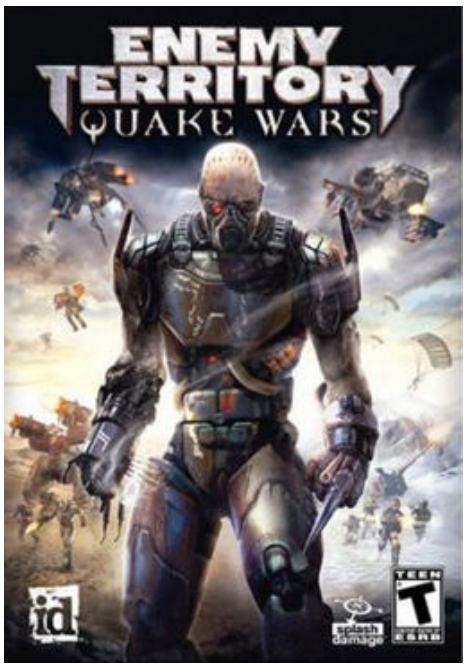

Figure 8: Enemy Territory: Quake Wars

8.1 Overview

The function *idBitMsg::ReadData* exposed by the engine is exploitable by attackers to execute code remotely on victim game clients.

[20] http://en.wikipedia.org/wiki/Enemy_Territory:_Quake_Wars

9 BRINK

From Wikipedia[21]: "Brink is a first-person shooter video game developed by Splash Damage for Microsoft Windows, PlayStation 3 and Xbox 360. It was released in North America on 10 May 2011, in Australia on 12 May, in Europe on 13 May and in Japan on 16 August. Brink has Steamworks integration, including Valve Anti-Cheat. It runs on id Tech 4 and has an updated rendering framework with improved support for multiple CPU cores. Brink is a first-person shooter with a focus on parkour-style movement. Online multiplayer servers hold up to 16 players; players can play cooperatively or competitively, or against artificially-intelligent bots. Despite the game having received negative to mixed reviews, Brink has reportedly sold 2.5 million copies to date".

Figure 9: Brink

9.1 OVERVIEW

The function *idBitMsg::ReadData* exposed by the engine is exploitable by attackers to execute code remotely on victim game servers.

9.2 ENDLESS LOOP

There is a endless loop condition, which can be triggered by exploiting the behavior of the *ReadBits* function, which is defined as:

- Function input: number of bits to read from the given stream
- Function output: a N-bit number
- Note: in case of any error the function returns -1

[21] en.wikipedia.org/wiki/Brink_%28video_game%29

With the above function logic, an attacker can quickly perform a *denial of service* attack by abusing the following code. Please note that the return value is not checked properly when the *ReadBits* function returns.

```
0070BDB0 PUSH EBX
0070BDB1 PUSH ESI
0070BDB2 PUSH EDI
0070BDB3 MOV EDI,ECX
0070BDB5 XOR ESI,ESI
0070BDB7 PUSH 10                          ; Arg1 = 10
0070BDB9 MOV DWORD PTR DS:[EDI+18],ESI
0070BDBC CALL ReadBits                    ; brink.ReadBits
0070BDC1 MOV EBX,DWORD PTR SS:[ESP+10]
0070BDC5 MOVZX EAX,AX ; assign return value to wchar_t
0070BDC8 CMP EAX,-1
0070BDCB JE SHORT 0070BDF5
0070BDCD PUSH EBP
0070BDCE MOV EBP,DWORD PTR SS:[ESP+18]
0070BDD2 TEST AX,AX
0070BDD5 JZ SHORT 0070BDF4
0070BDD7 LEA ECX,[EBP-1]
0070BDDA CMP ESI,ECX
0070BDDC JGE SHORT 0070BDE3
0070BDDE MOV WORD PTR DS:[ESI*2+EBX],AX
0070BDE2 INC ESI
0070BDE3 PUSH 10              ; Arg1 = 10
0070BDE5 MOV ECX,EDI
0070BDE7 CALL ReadBits        ; brink.ReadBits
0070BDEC MOVZX EAX,AX
0070BDEF CMP EAX,-1
0070BDF2 JNE SHORT 0070BDD2
0070BDF4 POP EBP
0070BDF5 XOR EDX,EDX
0070BDF7 POP EDI
0070BDF8 MOV WORD PTR DS:[ESI*2+EBX],DX
0070BDFC MOV EAX,ESI
0070BDFE POP ESI
0070BDFF POP EBX
0070BE00 RETN 8
```

10 QUAKE 4

From Wikipedia[22]: "Quake 4 is the fourth title in the series of Quake first-person shooter computer games. The game was developed by Raven Software and published by Activision. Raven Software collaborated with id Software, the creators and historical developers of preceding Quake games. In this case, id Software supervised the development of the game as well as providing the Doom 3 engine, now referred to as "id Tech 4" and released under the GNU General Public License on 22 November 2011, upon which it was built."

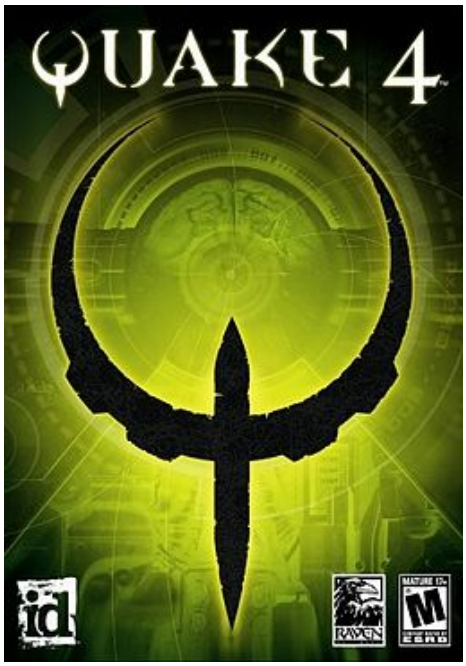

Figure 10: Quake 4

10.1 OVERVIEW

The authors found a server-side stack-based overflow due to an incorrect handling of the *GetInfo* query packet[23].

10.2 GETINFO STACK-BASED OVERFLOW

The stack-based overflow is located in the following code. As you can see there are three nested loops, where an attacker can control the values used inside these loops. In order to exploit this vulnerability an attacker needs to *spoof the IP*[24] of the packet to pretend that the packet has been sent from the *Quake 4* Master Server[25].

[22] http://en.wikipedia.org/wiki/Quake_4
[23] http://int64.org/docs/gamestat-protocols/doom3.html
[24] http://en.wikipedia.org/wiki/IP_address_spoofing
[25] q4master.idsoftware.com

```
10051B30 PUSH EBP
10051B31 MOV EBP,ESP
10051B33 AND ESP,FFFFFFF8
10051B36 PUSH -1
10051B38 PUSH 10282E07
10051B3D MOV EAX,DWORD PTR FS:[0]
10051B43 PUSH EAX
10051B44 MOV DWORD PTR FS:[0],ESP ; Installs SE handler
10051B4B SUB ESP,528
[...]
10051BB7 PUSH -10       ; Arg1 = -10
10051BB9 MOV ECX,ESI
10051BBB CALL ReadBits ; Quake4Ded.ReadBits (loop 1)
[...]
10051C06 PUSH -10       ; Arg1 = -10
10051C08 MOV ECX,ESI
10051C0A CALL ReadBits ; Quake4Ded.ReadBits (loop 2)
[...]
10051C31 PUSH -10       ; Arg1 = -10
10051C33 MOV ECX,ESI
10051C35 CALL ReadBits ; Quake4Ded.ReadBits (loop 3)
[...]
10051C50 MOV ECX,DWORD PTR SS:[EBP+8]
10051C53 PUSH 20        ; Arg1 = 20
10051C55 CALL ReadBits ; Quake4Ded.ReadBits (our value)
10051C5A MOV ECX,DWORD PTR DS:[102F8404]
10051C60 PUSH EAX
10051C61 MOV DWORD PTR DS:[EDI],EAX ; stack based
                                    ; buffer-overflow
```

11 CryEngine 3

From Wikipedia[26]: "On March 11, 2009, Crytek announced that it would introduce CryEngine 3 at the 2009 Game Developers Conference, held from March 25 to March 27. The new engine was being developed for use on Microsoft Windows, PlayStation 3, Xbox 360, and Wii U. As for the PC platform, the engine is said to support development in DirectX 9, 10, and 11. As of June 1, 2009, it was announced that Crysis 2 would be developed by Crytek on their brand new engine. CryEngine 3 was released on October 14, 2009.

On March 1, 2010, a new tech demo of the engine was released for the i3D 2010 symposium, which demonstrates 'Cascaded Light Propagation Volumes for Real Time Indirect Illumination'. On June 11, 2011, the Australian Defence Force revealed that Navy personnel would train on a virtual landing helicopter dock ship made using the CryEngine 3 software. As of July 1, 2011, the Mod SDK version of CryEngine 3 specifically to create custom maps, mods and content for Crysis 2 is available on Crytek's website. Crytek also released a free-to-use version of the CryEngine for non-commercial game development. It was released as of August 17, 2011 under the name CryEngine 3 SDK.

Crytek announced on September 9, 2011 that they would be using CryEngine 3 to bring the original Crysis to consoles. It was released for Xbox Live and PlayStation Network on October 4, 2011".

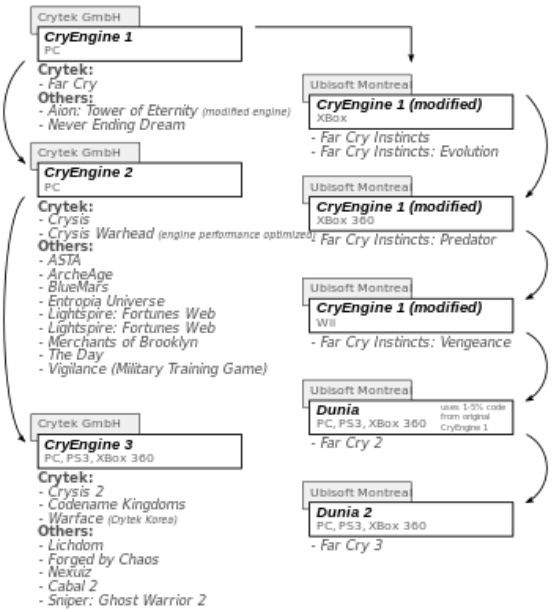

Figure 11: CryEngine's games tree

[26] http://en.wikipedia.org/wiki/CryEngine

11.1 Overview

There are two vulnerabilities in *CryEngine 3* due to improper handling of *fragmented packets* via *CryEngine*. For more information about fragmented packet issues, please refer to the authors' presentation slides[27].

11.2 Heap Overflow via Fragmented Packets

There is a heap overflow vulnerability, which can be triggered by sending a sequence of fragmented packets with opcode 0x93. By using this sequence an attacker is able to reach the following vulnerable code, and take control over the process execution:

```
39581AC6 MOV EAX,DWORD PTR DS:[EDI]
39581AC8 MOV EDX,DWORD PTR DS:[ESI]
39581ACA AND EAX,FFFFFFFC
39581ACD MOV ECX,DWORD PTR DS:[EAX]
39581ACF AND EDX,FFFFFFFC
39581AD2 MOV EDX,DWORD PTR DS:[EDX]
39581AD4 CMP ECX,EDX
39581AD6 JL 39581B94
39581ADC JNE SHORT 39581B2F
39581ADE LEA ECX,[ESP+4C]
39581AE2 PUSH ECX
39581AE3 LEA EDX,[EDI+4]
39581AE6 PUSH EDX
39581AE7 LEA ECX,[ESP+34]
39581AEB MOV DWORD PTR SS:[ESP+58],ESI
39581AEF MOV DWORD PTR SS:[ESP+34],OFFSET
39581AF7 MOV DWORD PTR SS:[ESP+38],OFFSET
39581AFF MOV DWORD PTR SS:[ESP+3C],OFFSET
39581B07 MOV EDX,DWORD PTR DS:[EAX+10]
39581B0A PUSH ECX
39581B0B CALL EDX
```

11.3 Memory Corruption via Fragmented Packets

There is a integer overflow vulnerability, which can be triggered by using a truncated fragment packet, which has a packet size lesser than 4. By sending, for instance a 2-byte packet, the following vulnerable code can be reached:

```
395818D7 MOV EDX,DWORD PTR DS:[ESI]      ; packet size
395818D9 ADD ECX,DWORD PTR DS:[EBX+44]
395818DC LEA EAX,[EDI+EAX+1E]
395818E0 MOV EAX,DWORD PTR SS:[EBP+10]
395818E3 SUB EDX,4
395818E6 PUSH EDX                        ; /Arg3
395818E7 ADD EAX,4                       ; |
395818EA PUSH EAX                        ; |Arg2
395818EB LEA ECX,[EDI+ECX+23]            ; |
```

[27] http://revuln.com/files/Ferrante_Auriemma_Exploiting_Game_Engines.pdf

```
395818EF PUSH ECX                          ; |Arg1
395818F0 CALL <JMP.&MSVCR100.memcpy>       ; \MSVCR100.memcpy
```

12 NEXUIZ

From Wikipedia[28]: "Nexuiz is a free first-person shooter video game developed and published by Alientrap. The game and its media are released under the GNU General Public License (GPL). A remake, also called Nexuiz has been released for Steam and Xbox 360 using CryEngine 3. Version 1.0 of the original game was released on May 31, 2005. The current version, 2.5.2, was released on October 1, 2009. Nexuiz classic uses DarkPlaces engine, a significantly modified Quake engine".

Figure 12: Nexuiz

12.1 CONNECTIONSETUP INTEGER OVERFLOW

This vulnerability affects only Nexuiz, and the vulnerable code is the following:

```
395E9FD4 CMP DWORD PTR SS:[ESP+390],25  ; check packet size
395E9FDC JNB SHORT CryNetwo.395E9FF4
[...]
395EA22D LEA EBP,DWORD PTR DS:[ESI+2D]  ; integer overflow
395EA230 SUB ESI,EBP
395EA232 ADD ESI,DWORD PTR SS:[ESP+390]
[...]
395EA279 PUSH ESI                       ; /n
395EA27A PUSH EBP                       ; |src
395EA27B PUSH EBX                       ; |dest
395EA27C CALL <JMP.&MSVCR100.memcpy>    ; \memcpy
```

[28] http://en.wikipedia.org/wiki/Nexuiz

13 Revision History

- 17 May 2013: Version 1.0 released.