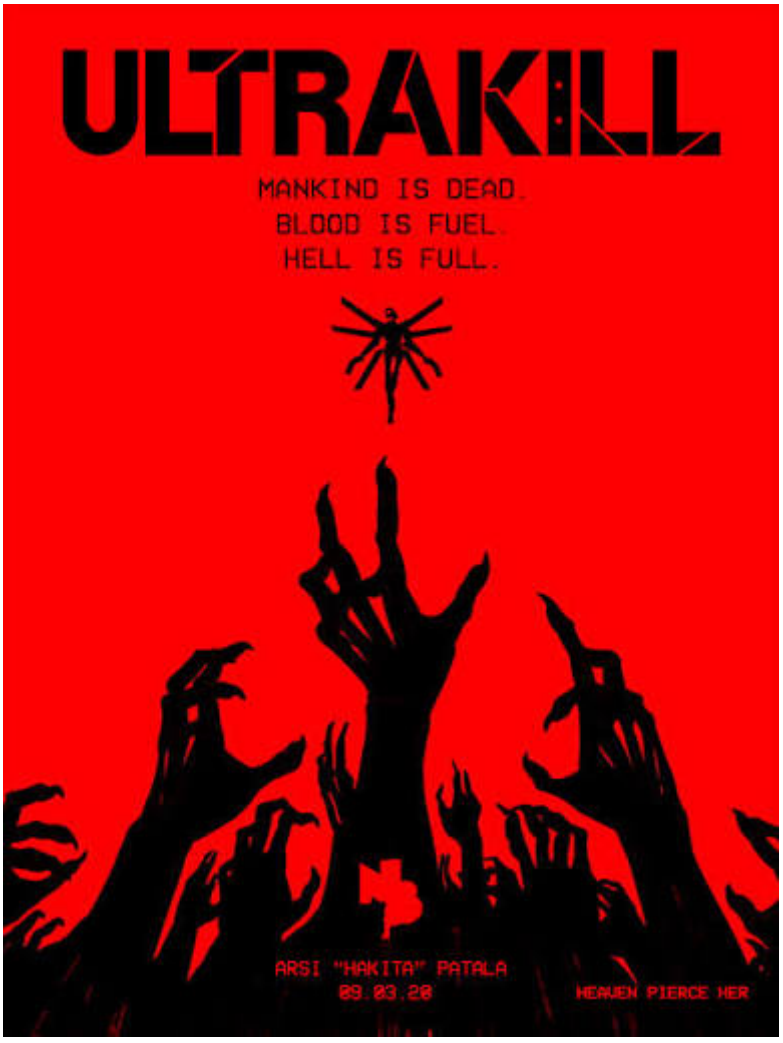
CHEAT MENU ULTRAKILL



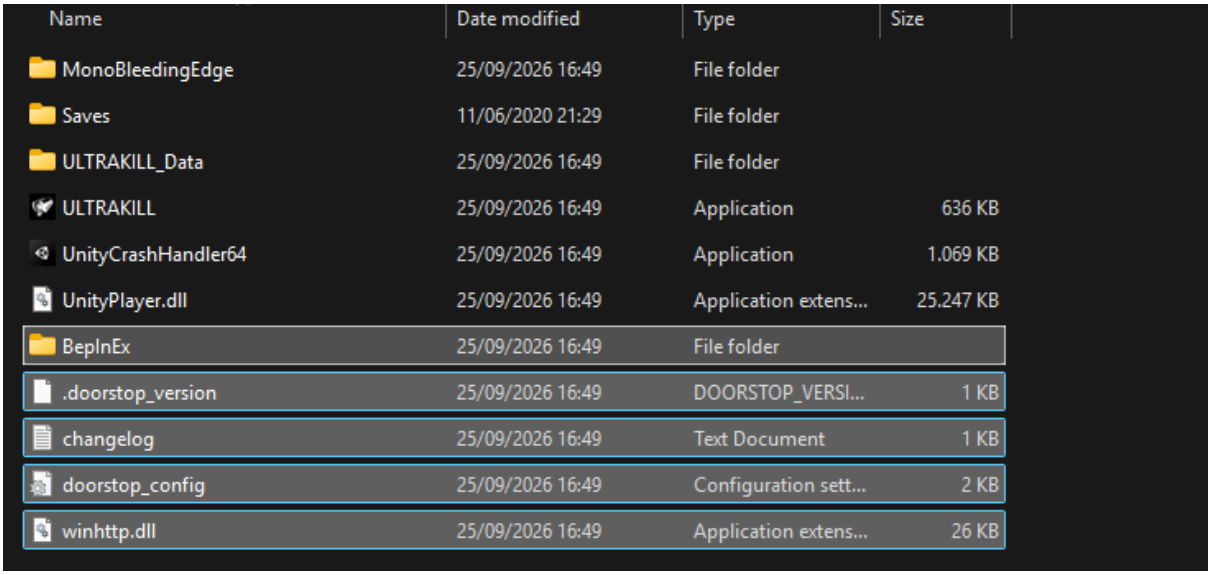
oleh:
Brilliansya Katon Rasyid - 2802505355

Informasi Project:

- Target: UltraKill (Unity Engine 2019)
- tujuan: membuat Mod Menu/Cheat menu dengan fitur *God Mode, Infinite Ammo, Infinite Stamina, One Hit Kill*, dan *Add Money*
- tools yang dibutuhkan: BepInEx (Framework Unity Mono), dnSpy (Decompiler .NET), dan Visual Studio.

TAHAPAN INSTALASI:

1. Install dan ekstrak BepInEx ke dalam folder game (setara dengan file Ultrakill.exe), seperti pada gambar dibawah

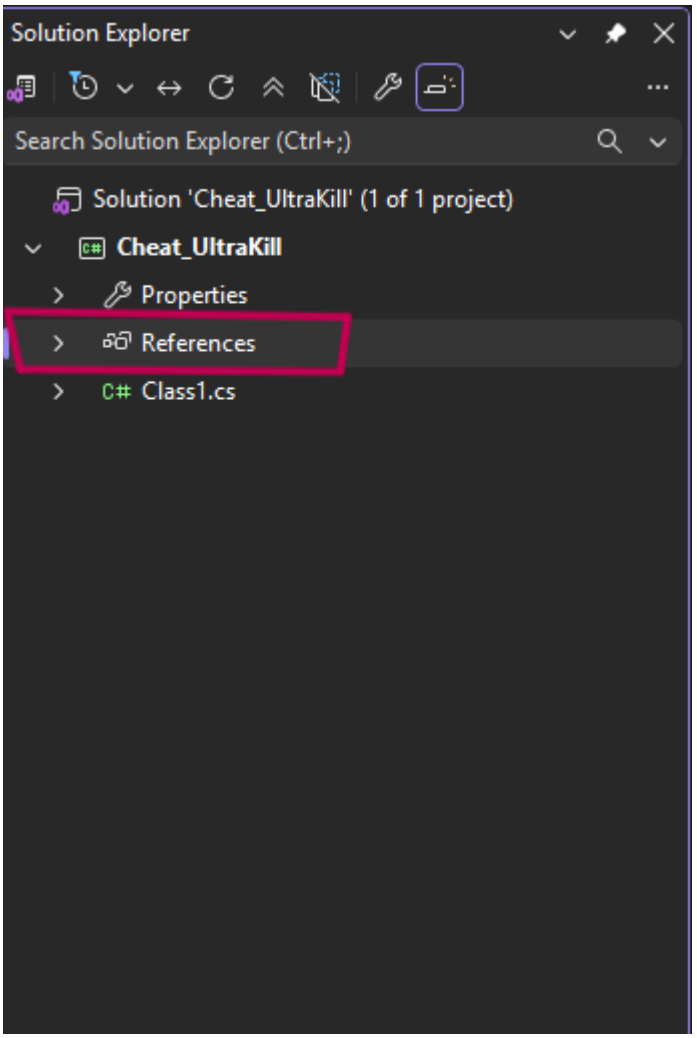


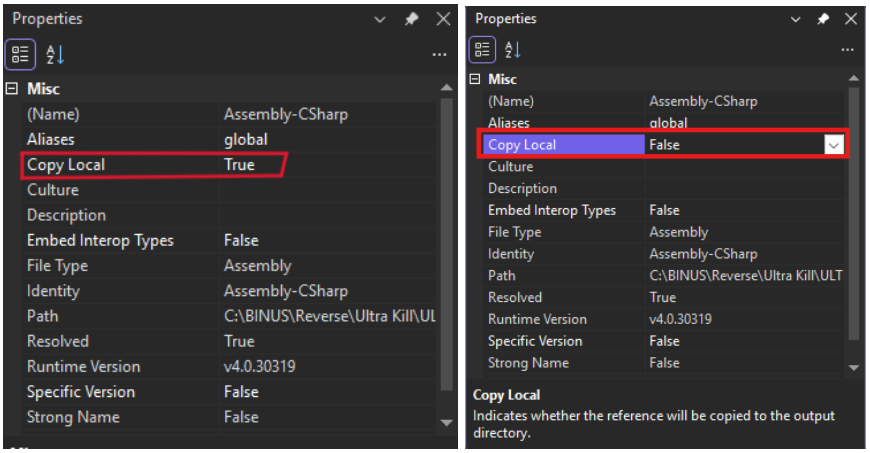
Name	Date modified	Type	Size
MonoBleedingEdge	25/09/2026 16:49	File folder	
Saves	11/06/2020 21:29	File folder	
ULTRAKILL_Data	25/09/2026 16:49	File folder	
ULTRAKILL	25/09/2026 16:49	Application	636 KB
UnityCrashHandler64	25/09/2026 16:49	Application	1.069 KB
UnityPlayer.dll	25/09/2026 16:49	Application extens...	25.247 KB
BepInEx	25/09/2026 16:49	File folder	
.doorstop_version	25/09/2026 16:49	DOORSTOP_VERSI...	1 KB
changelog	25/09/2026 16:49	Text Document	1 KB
doorstop_config	25/09/2026 16:49	Configuration sett...	2 KB
winhttp.dll	25/09/2026 16:49	Application extens...	26 KB

2. Jika BepInEx sudah terpasang dan game dijalankan maka ketika kita melihat LogOutput akan memberikan informasi seperti pada gambar dibawah, dimana <0 plingins to load> menandakan bahwa BepInEx aktif namun belum ada plugin yang terpasang, <Chainloader startup complete> menandakan proses pemuatan selesai.

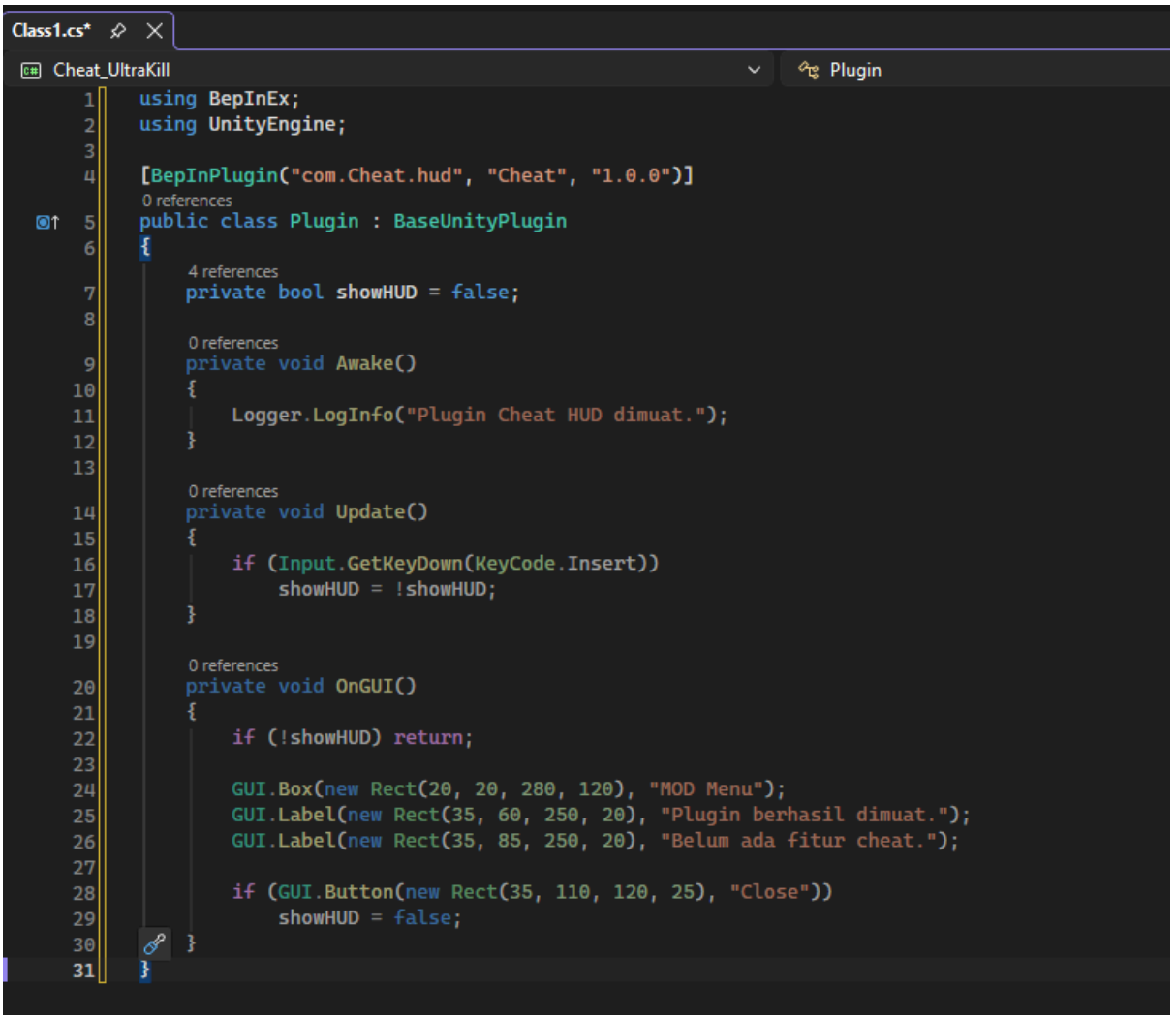
```
[Message:   BepInEx] BepInEx 5.4.23.5 - ULTRAKILL (9/25/2026 4:49:36 PM)
[Info   :   BepInEx] Running under Unity v2019.4.0.717686
[Info   :   BepInEx] CLR runtime version: 4.0.30319.17020
[Info   :   BepInEx] Supports SRE: False
[Info   :   BepInEx] System platform: Bits64, Windows
[Message:   BepInEx] Preloader started
[Info   :   BepInEx] Loaded 1 patcher method from [BepInEx.Preloader 5.4.23.5]
[Info   :   BepInEx] 1 patcher plugin loaded
[Info   :   BepInEx] Patching [UnityEngine.CoreModule] with [BepInEx.Chainloader]
[Message:   BepInEx] Preloader finished
[Message:   BepInEx] Chainloader ready
[Message:   BepInEx] Chainloader started
[Info   :   BepInEx] 0 plugins to load
[Message:   BepInEx] Chainloader startup complete
```

3. Setelah BeplnEx berhasil terpasang, buat proyek baru di Visual Studio. Tambahkan referensi file <.dll> pada Solution Explorer dengan cara klik kanan pada **References > Add Reference**. Tambahkan file berikut: BeplnEx.dll, 0Harmony.dll, Assembly-CSharp.dll, UnityEngine.dll, UnityEngine.CoreModule.dll, UnityEngine.InputLegacyModule.dll, dan UnityEngine.PhysicsModule.dll. Referensi ini berfungsi untuk memberikan akses ke dalam *class Unity, class internal game*, dan *framework modding*. Pastikan untuk mengubah pengaturan <Copy Local> menjadi <False> pada semua referensi tersebut guna mencegah duplikasi file DLL di folder output plugin.

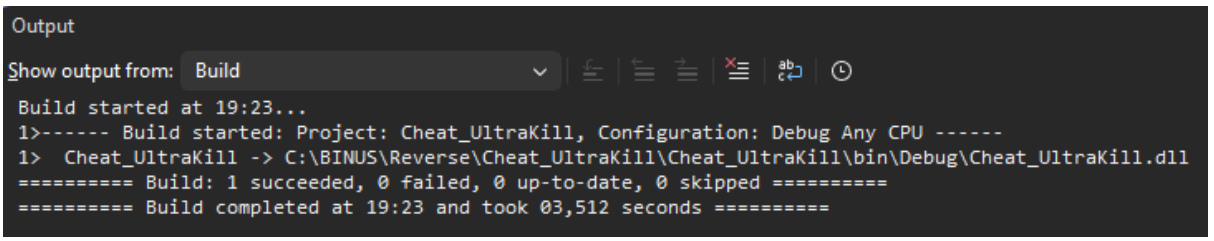




4. setelah itu kita buat script *template* buat MOD/Cheat Menu. Lalu kita build untuk memastikan *template* script tidak error, untuk buildnya tekan Ctrl + Shift + B. jika tidak error maka akan seperti gambar dibawah.

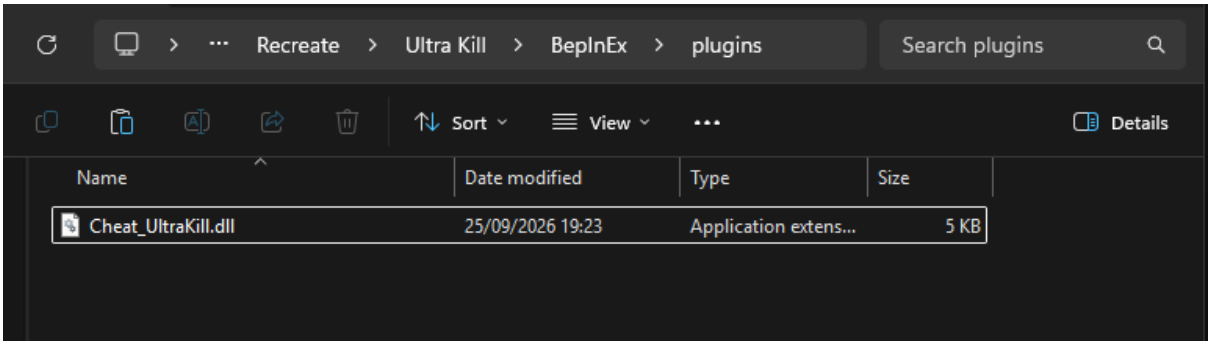


(*template codigan*)



(*Jika build berhasil*)

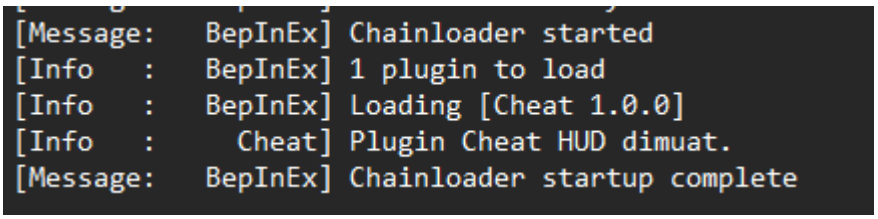
5. Ambil file <.dll> dari hasil *build* dari folder proyek (\bin\debug\file.dll), lalu copy paste ke dalam folder game (UltraKill\BepInEx\plugins). Jalankan game dan periksa LogOutput untuk memastikan pesan <[Cheat] Plugin Cheat HUD dimuat.> muncul, yang menandakan menu modnya berhasil berjalan di dalam game.



(*file MOD*)

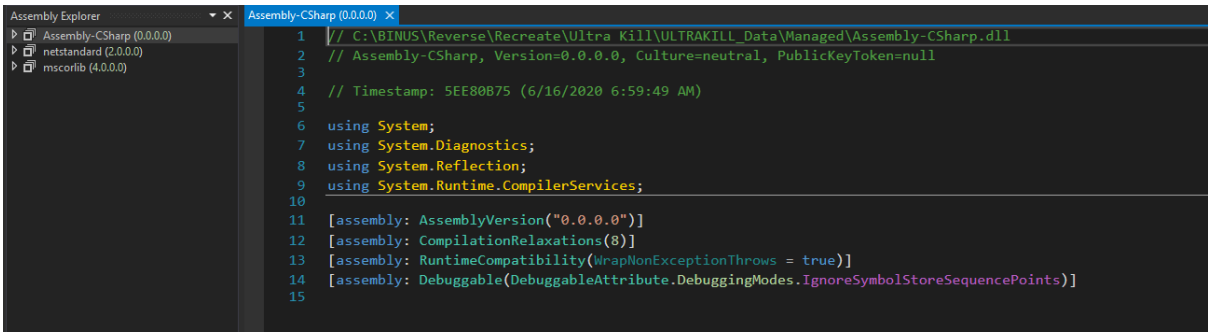


(*MOD in game*)



(*LogOutput)*

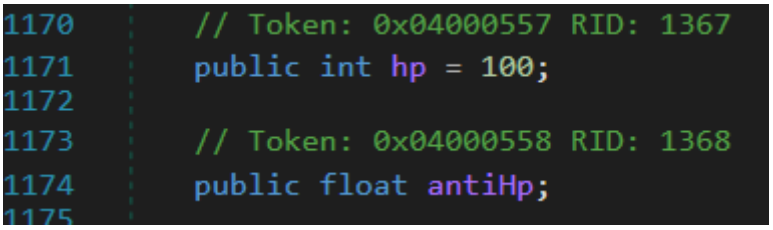
6. Buka aplikasi dnSpy dan masukan file <Assembly-CSharp.dll> dari direktori game untuk memulai mencari *field* maupun *methode* yang perlu dimodifikasi.



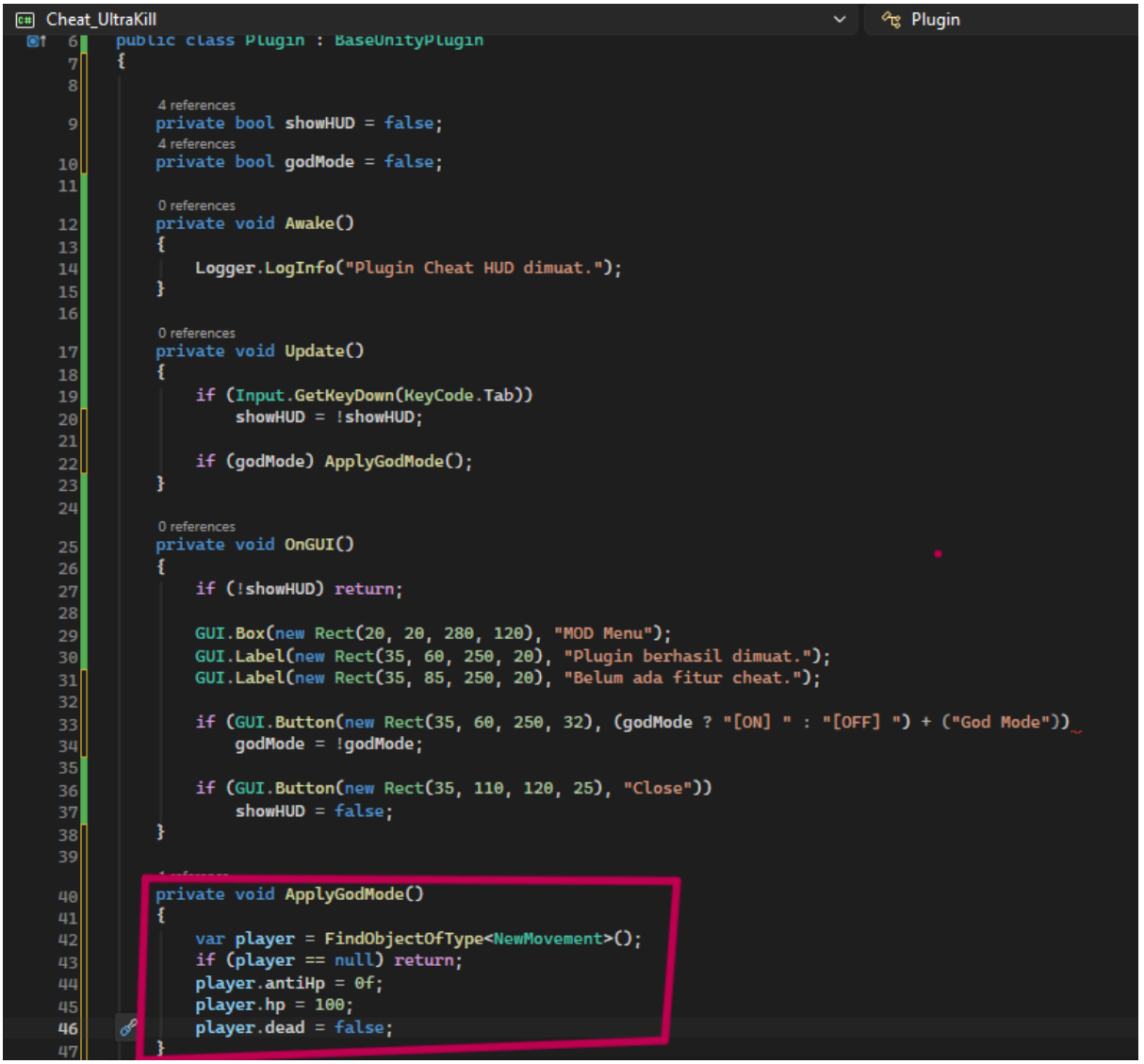
PEMBUATAN FITUR CHEAT

God Mode:

pertama kita mau membuat cheat God Mode, dimana pada Mod ini kita tidak bisa mati ketika diserang lawan. untuk membuat cheat God Mode ini kita butuh memodifikasi field <hp> dan <antihp>, untuk mencari nilai <hp> dan <antihp> kita bisa menggunakan Ctrl+Shift+K untuk search Assembly, lalu search <NewMovement>. pada game UltraKill, setiap damage yang diterima akan menambah nilai <antihp> terlebih dahulu. Jika <antihp> melewati batas tertentu, barulah nilai <hp> akan berkurang. Agar player menjadi kebal, paksa <antihp = 0>, sehingga damage tidak akan pernah nyampe ke <hp>, tetapkan <hp> di set 100, dan atur status <dead = false>.



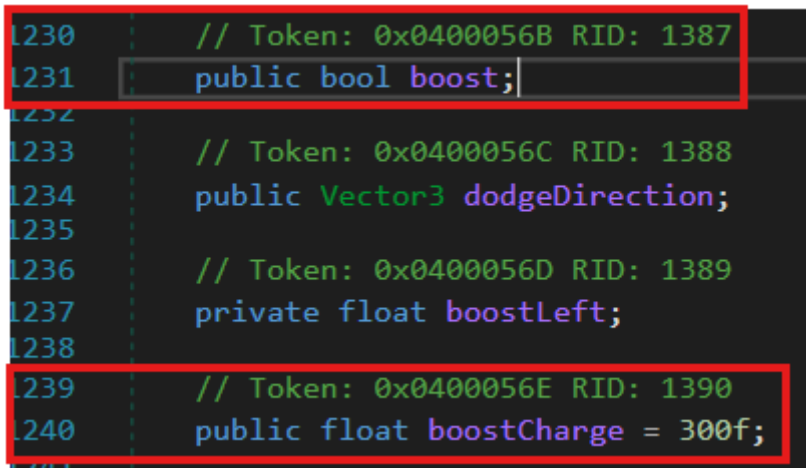
(*dnSpy*)



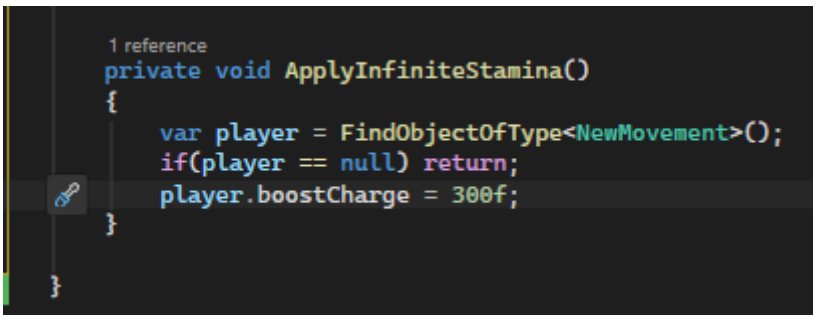
(*God Mode code*)

inilah script untuk GodMode nya. GodMode berbasis <hp>/<antiHp> tidak menangani kematian akibat jatuh ke void pit, karena sistem void pit di UltraKill memanggil fungsi kematian secara terpisah dan tidak menyentuh <hp>/<antiHp>. kemudian kita melakukan langkah yang sama untuk mod-mod lainnya.

INFINITY STAMINA:
Stamina dash pada UltraKill disimpan di field <boostCharge>. Untuk nilai maksimumnya adalah 300. Setiap pemain melakukan dash, nilai akan berkurang, dan pulih secara bertahap. Dengan memaksa <boostCharge = 300>, stamina selalu dalam kondisi penuh, sehingga dash dapat dilakukan berkali-kali.



(*dnSpy*)



(*Infinite Stamina code*)

INFINITE AMMO:
pada UltraKill, amunisi senjata diatur sebagai charge yang terletak pada class WeaponCharge, bukan sebagai jumlah peluru, setiap senjata mempunyai field charge nya sendiri, misal revolver maks 100, nailgun maks 2, railcannon maks 5, dll. class ini menyediakan <maxCharge()> dimana setiap <maxCharge()> ini dipanggil seluruh senjata akan dalam kondisi charge penuh, sehingga powershot bisa dipakai tanpa perlu dicharge.

```
74      // Token: 0x04000A2E RID: 2606
75      public float rev0charge = 100f;
76
77      // Token: 0x04000A2F RID: 2607
78      public float rev1charge = 400f;
79
80      // Token: 0x04000A30 RID: 2608
81      public float nai0charge = 2f;
82
83      // Token: 0x04000A31 RID: 2609
84      public float nai0heatUp;
85
86      // Token: 0x04000A32 RID: 2610
87      public float nai1heatUp;
88
89      // Token: 0x04000A33 RID: 2611
90      public float raicharge = 5f;
```

(*dnSpy*)

```
// Token: 0x06000565 RID: 1381 RVA: 0x0003A50C File Offset: 0x0003870C
public void MaxCharges()
{
    this.rev0charge = 100f;
    this.rev1charge = 400f;
    this.nai0charge = 2f;
    this.nai0heatUp = 0f;
    this.nai1heatUp = 0f;
    if (this.raicharge < 5f)
    {
        this.raicharge = 5f;
        this.PlayRailCharge();
    }
    if (!this.gc)
    {
        this.gc = base.GetComponent<GunControl>();
    }
    if (this.gc && this.gc.currentWeapon)
    {
        this.gc.currentWeapon.SendMessage("MaxCharge", SendMessageOptions.DontRequireReceiver);
    }
```

(*Infinite Ammo Code*)

ADD MONEY:
uang di game ini disimpan di file <Save/generalprogress.bepis>. Tidak ada field runtime yang menyimpan uang di memori, shop selalu memanggil <gameprogressSaver.GetMoney()> setiap kali shop dibuka. untuk menambah uang, plungin memanggil <GameProgressSaver.AddMoney(int)> yang secara internal membaca file save, menambah nilai <money> pada <GameProgressMoneyAndGear>, <AddMoney> tidak bisa dipanggil setiap frame karena bisa merusak file save.

```
// Token: 0x060001A5 RID: 421 RVA: 0x000129A8 File Offset: 0x00010BA8
public static void AddMoney(int money)
{
    BinaryFormatter binaryFormatter = new BinaryFormatter();
    string path = Directory.GetParent(Application.dataPath) + "/Saves/generalprogress.bepis";
    try
    {
        FileStream fileStream;
        GameProgressMoneyAndGear gameProgressMoneyAndGear;
        if (File.Exists(path))
        {
            fileStream = new FileStream(path, FileMode.Open);
            gameProgressMoneyAndGear = (binaryFormatter.Deserialize(fileStream) as GameProgressMoneyAndGear);
            fileStream.Close();
        }
        else
        {
            gameProgressMoneyAndGear = new GameProgressMoneyAndGear();
            gameProgressMoneyAndGear.secretMissions = new int[10];
        }
        gameProgressMoneyAndGear.money += money;
        fileStream = new FileStream(path, FileMode.Create);
        binaryFormatter.Serialize(fileStream, gameProgressMoneyAndGear);
        fileStream.Close();
    }
    catch
    {
        Debug.Log("Save Failed");
        GameObject.FindWithTag("OptionsManager").GetComponent<ProgressChecker>().SaveLoadError();
    }
}
```

```
// Token: 0x060001A6 RID: 422 RVA: 0x00012A60 File Offset: 0x00010C60
public static int GetMoney()
{
    BinaryFormatter binaryFormatter = new BinaryFormatter();
    string path = Directory.GetParent(Application.dataPath) + "/Saves/generalprogress.bepis";
    int result;
    try
    {
        if (File.Exists(path))
        {
            FileStream fileStream = new FileStream(path, FileMode.Open);
            GameProgressMoneyAndGear gameProgressMoneyAndGear = binaryFormatter.Deserialize(fileStream) as GameProgressMoneyAndGear;
            fileStream.Close();
            result = gameProgressMoneyAndGear.money;
        }
        else
        {
            result = 0;
        }
    }
    catch
    {
        Debug.Log("Load Failed");
        GameObject.FindWithTag("OptionsManager").GetComponent<ProgressChecker>().SaveLoadError();
        result = 0;
    }
    return result;
}
```

(*dnSpy*)

```
GUI.Label(new Rect(35, 232, 250, 20), "Tambah uang (P):");
moneyInput = GUI.TextField(new Rect(35, 254, 160, 28), moneyInput, 9);

if (GUI.Button(new Rect(200, 254, 85, 28), "Add Money"))
{
    if (int.TryParse(moneyInput, out int amount))
    {
        if (amount > 0 && amount <= 999999999)
        {
            GameProgressSaver.AddMoney(amount);
            Logger.LogInfo($"AddMoney: +{amount}, total: {GameProgressSaver.GetMoney()}");
        }
    }
}
```

(*Add Money code*)

ONE HIT KILL:
setiap membunuh musuh di game ini akan melewati method <DeliverDamage> di class <EnemyIdentifier>. Method ini meneruskan damage ke method <GetHurt> sesuai tipe musuh dan di dalam <GetHurt> nilai health musuh dikurangin dengan <health - = 1f * multiplier>. Parameter <multiplier> adalah nilai yang dikirim dari senjata pemain. Dengan menggunakan Harmony Prefix untuk memaksa <multiplier = 9999f> setiap kali <DeliverDamage> dipanggil, semua musuh akan mati dalam satu tembaka. <OneHitKill> akan dideklarasikan static agar bisa diakses dari kelas patch Harmony.

```
31
32      // Token: 0x06000116 RID: 278 RVA: 0x0000C680 File Offset: 0x0000A880
33      public void DeliverDamage(GameObject target, Vector3 force, Vector3 hitPoint, float multiplier, bool tryForExplode, float critMultiplier)
34      {
35          if (this.weaknesses.Length != 0)
36          {
37              foreach (string b in this.weaknesses)
38              {
39                  if (this.hitter == b)
40                  {
41                      multiplier *= this.weaknessMultiplier;
42                  }
43              }
44          }
45          if (this.nails.Count > 10)
46          {
47              for (int j = 0; j < this.nails.Count - 10; j++)
48              {
49                  if (this.nails[j] != null)
50                  {
51                      Object.Destroy(this.nails[j].gameObject);
52                  }
53                  this.nails.RemoveAt(j);
54              }
55          }
56          if (!this.beingZapped && this.hitter == "railcannon" && this.nailsAmount > 0)
57          {
58              this.beingZapped = true;
59              foreach (Nail nail in this.nails)
60              {
61                  nail.Zap();
62              }
63              base.Invoke("AfterShock", 1f);
64          }
65          if (this.beingZapped)
66          {
67              Debug.Log("Damage: " + multiplier);
68          }
69          if (this.magneted && this.hitter != "deathzone")
70          {
71              this.magneted = false;
72          }
73          switch (this.type)
74          {
75          case EnemyType.Zombie:
76              if (this.zombie == null)
77              {
78                  this.zombie = base.GetComponent<Zombie>();
```

```
239
240     // Token: 0x06000119 RID: 281 RVA: 0x0000CC98 File Offset: 0x0000AE98
241     public void InstaKill()
242     {
243         Debug.Log("Instakill Phase 2");
244         if (!this.dead)
245         {
246             if (this.magneted)
247             {
248                 Object.FindObjectOfType<StyleHUD>().AddPoints(240, "<color=cyan>CATAPULTED</color>");
249             }
250             this.dead = true;
251             switch (this.type)
252             {
253             case EnemyType.Zombie:
254                 if (this.zombie == null)
255                 {
256                     this.zombie = base.GetComponent<Zombie>();
257                 }
258                 this.zombie.GoLimp();
259                 break;
260             case EnemyType.Spider:
261                 if (this.spider == null)
262                 {
263                     this.spider = base.GetComponent<SpiderBody>();
264                 }
265                 if (!this.spider.dead)
266                 {
267                     this.spider.Die();
268                 }
269                 break;
270             case EnemyType.Machine:
271                 if (this.machine == null)
272                 {
273                     this.machine = base.GetComponent<Machine>();
274                 }
275                 this.machine.GoLimp();
276                 break;
277             case EnemyType.Statue:
278                 if (this.statue == null)
279                 {
280                     this.statue = base.GetComponent<Statue>();
281                 }
282                 this.statue.GoLimp();
283                 break;
284             case EnemyType.Drone:
```

(*dnSpy*)

untuk mod OneHitKill butuh membuat satu file patch tambahan yaitu <OneHitKillPatch.cs>. class patch ini menargetkan method <DeliverDamage> di class <EnemyIdentifier> melalui atribut <HarmonyPatch>, dan method prefix didalamnya dijalankan sebelum method asli. Karena parameter <multiplier> dikirim dengan <ref>, nilai damage dapat diperbesar menjadi 999f secara langsung, membuat semua musuh mati dalam sekali tembak.

OneHitKillPatch.cs | KeyCode [decompiled] | Class1.cs

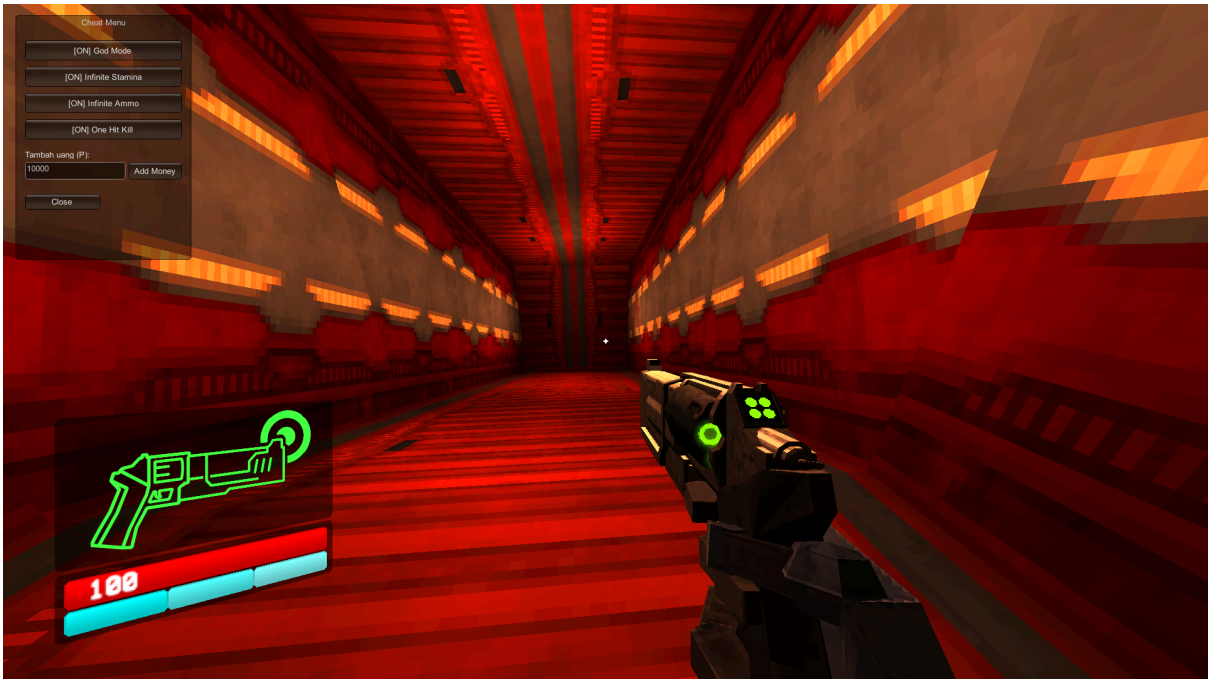
Miscellaneous Files | OneHitKillPatch | Prefix(ref float multiplier)

```
1   using HarmonyLib;
2
3   [HarmonyPatch(typeof(EnemyIdentifier))]
4   [HarmonyPatch("DeliverDamage")]
5   public static class OneHitKillPatch
6   {
7       [HarmonyPrefix]
8       public static void Prefix(ref float multiplier)
9       {
10          if (Plugin.oneHitKill)
11              multiplier = 9999f;
12      }
13  }
```

(*OnehitKill Patch code*)

setelah itu dibuild, lalu copy filenya modnya dan pindahkan ke folder UltraKill - BeplnEx - plugins - paste. Lalu uji gamenya.

HASIL CHEAT:




```
[Message:   BepInEx] BepInEx 5.4.23.5 - ULTRAKILL (9/25/2026 4:49:36 PM)
[Info   :   BepInEx] Running under Unity v2019.4.0.717686
[Info   :   BepInEx] CLR runtime version: 4.0.30319.17020
[Info   :   BepInEx] Supports SRE: False
[Info   :   BepInEx] System platform: Bits64, Windows
[Message:   BepInEx] Preloader started
[Info   :   BepInEx] Loaded 1 patcher method from [BepInEx.Preloader 5.4.23.5]
[Info   :   BepInEx] 1 patcher plugin loaded
[Info   :   BepInEx] Patching [UnityEngine.CoreModule] with [BepInEx.Chainloader]
[Message:   BepInEx] Preloader finished
[Message:   BepInEx] Chainloader ready
[Message:   BepInEx] Chainloader started
[Info   :   BepInEx] 1 plugin to load
[Info   :   BepInEx] Loading [Cheat 1.0.0]
[Info   :     Cheat] Harmony patches applied.
[Message:   BepInEx] Chainloader startup complete
[Info   :     Cheat] AddMoney: +10000, total: 16566
[Info   :     Cheat] AddMoney: +10000, total: 26566
```