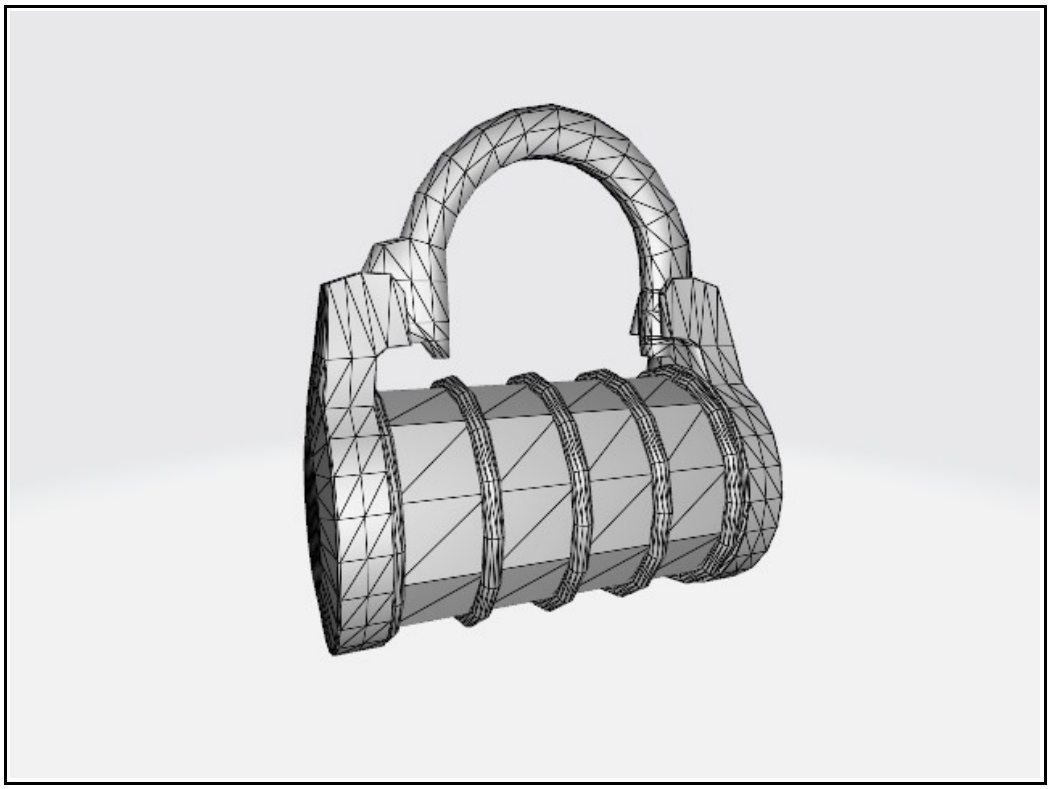
Models

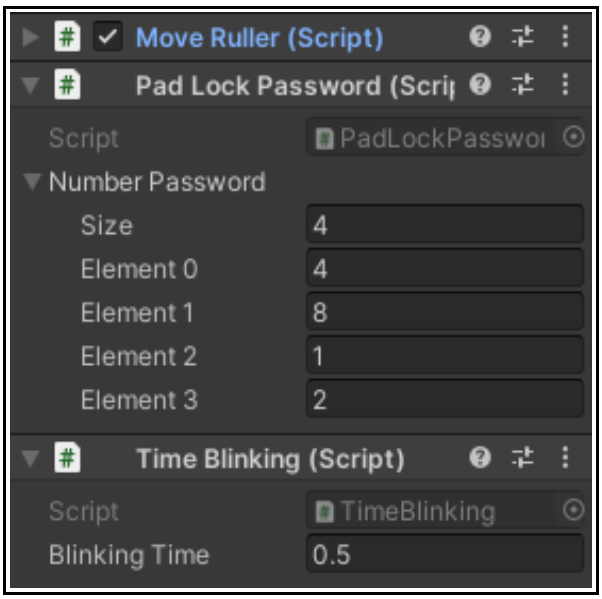
This little asset contains one models of PadLock high poly with 3328 faces. The model has one set of textures (Albedo, Metallic and Normal map) that can be used in ***HD Render Pipeline*** *and* ***Universal Render Pipeline*** . The textures size are 2048x2048.



(PadLock model)

Scripts

To make the model inside Unity more interesting, there are some scripts that are used to choose the rullers and rotate them. In addition we can choose a combination to open the lock



PadLock Password

Number password:
In this area we can choose the 4 numbers to enter the password. When we rotate the Rullers and the numbers combine with the password an event dictated by the user happens

Time Blinking: Here you can change the time for the blinking of the ruller when it's selected.

Note: it is possible to change the keys for rotate and change ruller through direct access to the scripts in use. Open in editor the script "Move Ruller" and in `MoveRulles() and RotateRullers();` void change the KeyCode as follows: `if (Input.GetKeyDown(KeyCode.D))` where the "D" (in this case) is the letter to change for the key of our choice.
Then you can change all four letters to make the rotate and change of the rullers custom. The letters entered by default are: "W" and "S" for rotate ruller and "D" and "A" to change left or right the rullers.

Note 2: As mentioned it's left to the user to create the event once the right code has been entered. In script "PadLockPassword" below the void `Password()` the event is inserted. By default, the ability to turn off roller blinking once the password has been reached has been added:

```
public void Password()
    {
        if (_moveRull._numberArray.SequenceEqual(_numberPassword))
        {
            // Here enter the event for the correct combination
            Debug.Log("Password correct");

            // Es. Below the for loop to disable Blinking Material after the correct password
            for (int i = 0; i < _moveRull._rullers.Count; i++)
            {
                _moveRull._rullers[i].GetComponent<PadLockEmissionColor>()._isSelect = false;
                _moveRull._rullers[i].GetComponent<PadLockEmissionColor>().BlinkingMaterial();
            }

        }
    }
```