

Anatomy of Rogue Encounter Editor

Next to the Complex Encounter this is the single most confusing and Difficult thing to understand about making a Realmz scenario. I don't want to sugar coat it for you.

OK. Lets dive in and get the dirty work done.

Figure 11.0

As in many other screens, you can edit strings and play sounds from this screen as well. But lets try to explain how the Rogue encounter works and how to go about filling in all the information you need to make it work correctly.

As with both Complex and Simple encounters you can also set a Prompt String. However, Rogue Encounters also have the option of a Prompt Sound that plays.

The first thing you have to do is decide if this is a trap or you want to test the characters acrobatic skill or have them attempt to force a lock or door open.

Lets do the easy things first. If you want to test them against an Acrobatic Act or Forcing a Lock or Door open, just check the box needed.

If you want the actions to be harder to do then place a -% modifier. If you want it to be easier then place a +% modifier.

In the example above, the lock is not particularly hard nor well hidden so there is a +20% chance to detect the trap, plus 10% chance to disable the trap and a +10 chance to pick the lock.

The same holds true for Acrobatic Act and Force Lock. You can put a +/- there to show that it may be tougher or easier than a standard actions of this type. These modifiers are added to the characters base chance of success at the particular task at hand.

In the above example it's a locked and trapped object. If you place a check next to Detect Trap and Pick Lock, that will give those options to the player when they first try this encounter.

NOTE: DO NOT put a check next to disable trap even though it is trapped. That option will be set automatically once they have been successful at detecting a trap if it exists.

You should always check DETECT TRAP if it is some type of lock. Even if it is NOT trapped. The player should not know it is not trapped and should always have the option to try and detect one whether it is actually there or not.

If the object IS trapped as in the above example, you must also decide on whether the trap (if sprung) will affect only the Rogue trying to open the lock or if it will affect the entire party. i.e. A poison needle will only affect the character trying to open the lock. However, if the chest explodes into a fireball, it's safe to say that everybody in the party should be affected.

Building a better mouse trap: OK. Lets get into the nuts and bolts of building the trap. Most of the work consists of telling Realmz what kind of trap it is and what it does to the poor slob who sets it off.

Here is a list of fields that define the trap and what they do.

Number of Lock Tumblers: When you attempt to pick a lock you get a number of colored bars. The object is to get all the bars showing at least yellow before time runs out. This field sets the number of bars (called tumblers) to use. You can have as few as one and as many as 6.

Trap Damage: If the trap just does physical damage, say a pit or hammer that hits the Rogue then you can set the amount of damage it will do. In this case it will do from 4 to 12 points of damage.

Trap Sound: The Sound ID to be played, if any, when the trap is triggered.

Power Level: If the trap triggers a spell then the power level of the spell is placed here. If you have the trap trigger a fireball spell, it can be power level 1 (ouch) or power level 7 (OOOOOOOOohhhh OUCH!)

The damage done by a spell is figured for the spell and you do not need to place any damage in the TRAP DAMAGE field if you cast a spell.

% Chance / Level to Knock: Realmz sports some spells that can use magic to open a lock. If you want the lock to be immune to magical means of opening it, set this to 0. If you want a spell such as Open Lock to have some effect, then put a % chance/power level of the spell. In this example, the lock is immune to magical means and can only be opened the hard way.

% Chance / Level to Disarm: Realmz sports some spells that can use magic to disarm a trap. If you want the trap to be immune to magical means of disarming, set this to 0. If you want a spell such as Disarm Trap to have some effect, then put a % chance/power level of the spell. In this example there is a 5% chance for each power level of the spell cast. So if the character casts "Disarm Trap" from the Complex Encounter that calls this Rogue Encounter, and casts it at Power Level 6. There is a 30% chance that the trap will be disarmed.

Let's turn our heads to the top of the screen where we see all kinds of headings.

Figure 11.1

The % Modify we explained earlier.

Result Codes: If you want any particular action to return to the Complex Encounter and start to execute a specific Result Code, then you can place a number here. In this example it will not return any results to the Complex Encounter except if there is a success of Picking the Lock. If they pick the lock, it returns to the Complex Encounter that called it and begins to

execute Result Code #2

Text Codes: If you want specific strings to be displayed on success/fail of any action, you can place the String ID here. This is not required.

Sound Codes: If you want specific sounds to be played on success/fail of any action, you can place the Sound ID here. This is not required.

If a particular action is taken and no Result Code is specified, (as is the case in this example and for the majority of the actions) then the game will not do anything on success/fail. i.e. If they perform the detect trap, the game will set the Disable Trap flag but will not return to the Complex Encounter. If they attempt to pick a lock and succeed, it will return with Result Code #2, otherwise it will trigger the trap and just sit there.

Some trial and error is a necessary evil of learning how to script these babies.