Make interface Ihiding: x  
Make hide class: x  
Implement hide class in the hiding skill: x  
SkillCheck if hiding is succefull:  
Disturb function works:

1. OnHiderHurt() x
2. OnConnectionLost()
3. OnHiderDeath()
4. OnCasting()
5. OnHidingAgain()

OnCasting()

Hiding should stop when you try to cast a spell.

Function word 2x opgeroepen: eerst bij de start en dan eens na de timer afloopt.  
hide.cs: HideTimer(): OnTick()

Debug list:

Start hiding

hide.TryToHide() hide.cs

HideState.TryToHide hide.cs

HideState.TryToHide m\_HiderTimer.Start() hide.cs

Hide.TryToHide() = True

“After 3 seconds”

inside HidingTimer: OnTick() startfunction: hide.cs

Start hiding

hide.TryToHide() hide.cs

Disturb new hiding request

Hide.TryToHide() = False

inside HidingTimer: m\_Hiding.m\_state == HidingState.TryingToHid startIf: hide.cs

OnHide Hiding.cs

FinishedSequence

setting m\_hider.hiding = null

inside HidingTimer: m\_Hiding.m\_state == HidingState.TryingToHid endIf: hide.cs

inside HidingTimer: OnTick() endfunction: hide.cs

**Bug:** else if (m\_Hiding.m\_state == HidingState.TryingToHide && m\_Hiding.m\_hider.UseSkill(SkillName.Hiding))

**Solution:** else if (m\_Hiding.m\_state == HidingState.TryingToHide) //&&m\_Hiding.m\_hider.UseSkill(SkillName.Hiding))