

Toggle menu
Blue Gold Program Wiki

Navigation

- [Main page](#)
- [Recent changes](#)
- [Random page](#)
- [Help about MediaWiki](#)

Tools

- [What links here](#)
- [Related changes](#)
- [Special pages](#)
- [Permanent link](#)
- [Page information](#)

Personal tools

- [Log in](#)

personal-extra

Toggle search

Search

Random page

Views

- [View](#)
- [View source](#)
- [History](#)
- [PDF Export](#)

Actions

Module:Effective protection level

From Blue Gold Program Wiki

The printable version is no longer supported and may have rendering errors. Please update your browser bookmarks and please use the default browser print function instead.

Documentation for this module may be created at [Module:Effective protection level/doc](#)

```

local p = {}

-- Returns the permission required to perform a given action on a given
title.
-- If no title is specified, the title of the page being displayed is used.
function p._main(action, pagename)
    local title
    if type(pagename) == 'table' and pagename.prefixedText then
        title = pagename
    elseif pagename then
        title = mw.title.new(pagename)
    else
        title = mw.title.getCurrentTitle()
    end
    pagename = title.prefixedText
    if action == 'autoreview' then
        local level = mw.ext.Flagger.getStabilitySettings(title)
        level = level and level.autoreview
        if level == 'review' then
            return 'reviewer'
        elseif level ~= '' then
            return level
        else
            return nil -- not '*'. a page not being PC-protected
is distinct from it being PC-protected with anyone able to review. also not
'', as that would mean PC-protected but nobody can review
        end
    elseif action ~= 'edit' and action ~= 'move' and action ~= 'create'
and action ~= 'upload' and action ~= 'undelete' then
        error( 'First parameter must be one of edit, move, create,
upload, undelete, autoreview', 2 )
    end
    if title.namespace == 8 then -- MediaWiki namespace
        if title.text:sub(-3) == '.js' or title.text:sub(-4) ==
'.css' or title.contentModel == 'javascript' or title.contentModel == 'css'
then -- site JS or CSS page
            return 'interfaceadmin'
        else -- any non-JS/CSS MediaWiki page
            return 'sysop'
        end
    elseif title.namespace == 2 and title.isSubpage then
        if title.contentModel == 'javascript' or title.contentModel
== 'css' then -- user JS or CSS page
            return 'interfaceadmin'
        elseif title.contentModel == 'json' then -- user JSON page
            return 'sysop'
        end
    end
    if action == 'undelete' then
        return 'sysop'
    end
end

```

```

        local level = title.protectionLevels[action] and
title.protectionLevels[action][1]
        if level == 'sysop' or level == 'editprotected' then
            return 'sysop'
        elseif title.cascadingProtection.restrictions[action] and
title.cascadingProtection.restrictions[action][1] then -- used by a
cascading-protected page
            return 'sysop'
        elseif level == 'templateeditor' then
            return 'templateeditor'
        elseif action == 'move' then
            local blacklistentry = mw.ext.TitleBlacklist.test('edit',
pagename) -- Testing action edit is correct, since this is for the source
page. The target page name gets tested with action move.
            if blacklistentry and not blacklistentry.params.autoconfirmed
then
                return 'templateeditor'
            elseif title.namespace == 6 then
                return 'filemover'
            elseif level == 'extendedconfirmed' then
                return 'extendedconfirmed'
            else
                return 'autoconfirmed'
            end
        end
    end
    local blacklistentry = mw.ext.TitleBlacklist.test(action, pagename)
    if blacklistentry then
        if not blacklistentry.params.autoconfirmed then
            return 'templateeditor'
        elseif level == 'extendedconfirmed' then
            return 'extendedconfirmed'
        else
            return 'autoconfirmed'
        end
    elseif level == 'editsemiprotected' then -- create-semiprotected
pages return this for some reason
        return 'autoconfirmed'
    elseif level then
        return level
    elseif action == 'upload' then
        return 'autoconfirmed'
    elseif action == 'create' and title.namespace % 2 == 0 and
title.namespace ~= 118 then -- You need to be registered, but not
autoconfirmed, to create non-talk pages other than drafts
        return 'user'
    else
        return '*'
    end
end

setmetatable(p, { __index = function(t, k)

```

```
        return function(frame)
            return t._main(k, frame.args[1])
        end
    end })
```

```
return p
```

Retrieved from

["https://www.bluegoldwiki.com/index.php?title=Module:Effective_protection_level&oldid=3586"](https://www.bluegoldwiki.com/index.php?title=Module:Effective_protection_level&oldid=3586)

Namespaces

- [Module](#)
- [Discussion](#)

Variants

This page was last edited on 15 November 2020, at 11:23.

Blue Gold Program Wiki

The wiki version of the Lessons Learnt Report of the Blue Gold program, documents the experiences of a technical assistance (TA) team working in a development project implemented by the Bangladesh Water Development Board (BWDB) and the Department of Agricultural Extension (DAE) over an eight+ year period from March 2013 to December 2021. The wiki lessons learnt report (LLR) is intended to complement the BWDB and DAE project completion reports (PCRs), with the aim of recording lessons learnt for use in the design and implementation of future interventions in the coastal zone.

- [Privacy policy](#)
- [About Blue Gold Program Wiki](#)
- [Disclaimers](#)

Developed and maintained by Big Blue Communications for Blue Gold Program



[Blue Gold Program Wiki](#)