**Computer Architecture Midterm 2020**

A screenshot of a cell phone

Description automatically generated

#B0000008 – A0000004 = 10000004 (decimal: 268435460)

li $at,0xB0000008 #load target address  
jr $at #jump to that location

A picture containing bird

Description automatically generated

The code segment causes an infinite loop that jumps into the next label. The $at register stores the jal address 0x0c000000. The return $ra register stores the same address. They return to jal again. The code segment is store word instruction that makes the machine code relative to the address, resulting machine code will change based on where the jump is inserted. The control sequence is diverted to the virus code. When the virus code returns, control returns to the jump. So it repeatedly executes the same sequence, thus going into an infinite loop.

A screenshot of a cell phone

Description automatically generated

Lw $t0 0x0004 $sp

Binary: 10001111101010000000000000000100

Hex: 0x8FA80004

Lw rt, offset(base) [I-type]

|  |  |  |  |
| --- | --- | --- | --- |
| 100011 | 11101 | 01000 | 0000000000000100 |

lw(6) $sp(5) $t0(5) offset(16)

BEQ $s0 $s1 0x0005

Binary: 00010010000100010000000000000101

Hex: 0x12110005

BEQ rs, rt, offset(base) [I-type]

|  |  |  |  |
| --- | --- | --- | --- |
| 000100 | 10000 | 10001 | 0000000000000101 |

lw(6) $s0(5) $s1(5) offset(16)

ADD $s3 $s2 $v0

Binary: 00000010010000101001100000100000

Hex: 0x02429820

ADD rd, rs, st, [R-type]

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 000000 | 10010 | 00010 | 10011 | 00000 | 100000 |

Special(6) $s2(5) $v0(5) $s3(5) 0(5) ADD(6)

J 0x01000A9

Binary: 00001000000100000000000010101001

Hex: 0x0081000A9

J target [J-type]

|  |  |
| --- | --- |
| 000010 | 00000100000000000010101001 |

J(6) target(26)

JAL 0x000000FA

Binary: 00001100000000000000000011111010

Hex: 0x0C0000FA

J target [J-type]

|  |  |
| --- | --- |
| 000011 | 00000000000000000011111010 |

J(6) target(26)