

Die System Integrity Protection (SIP)

Um OS X auf deinem Nicht-Apple-Gerät zu installieren, muss die [SIP](#) – zumindest in Teilen – abgeschaltet werden, da z.B. unsigned Kexte für Netzwerk und Audio und auch die FakeSMC.kext geladen werden müssen.

Die [SIP](#) besteht aus einer Bitmaske mit 9 Bits, von denen wir normalerweise 7 verwenden, mit der man wie mit Schaltern einzelne Funktionen der [SIP](#) an- oder abschalten kann.

Hier die Funktionen der ersten 7 Bits:

Haben alle Bits den Wert 0 – alle Schalter auf aus – ist die [SIP](#) vollständig aktiviert. Haben alle Bits den Wert 1 – alle Schalter auf an – ist die [SIP](#) vollständig deaktiviert.

Schreiben wir die Binärzahl auf, sieht das folgendermaßen aus:

Bit 1	CSR_ALLOW_UNTRUSTED_KEXTS	Erlaubt das Laden unsigned Kexts
Bit 2	CSR_ALLOW_UNRESTRICTED_FS	Erlaubt die uneingeschränkte Nutzung des Dateisystems
Bit 3	CSR_ALLOW_TASK_FOR_PID	Erlaubt das Injecten von Code
Bit 4	CSR_ALLOW_KERNEL_DEBUGGER	Erlaubt das Ausführen von Kernel Debuggern
Bit 5	CSR_ALLOW_APPLE_INTERNAL	Eine weitere Prüfinstanz, welche unsigned Kexte blocken kann
Bit 6	CSR_ALLOW_UNRESTRICTED_DTRACE	Erlaubt das Ausführen von dtrace-basierenden Monitoring & Reporting Tools
Bit 7	CSR_ALLOW_UNRESTRICTED_NVRAM	Erlaubt das Ändern von NVRAM-Einträgen

SIP aktiviert SIP deaktiviert

%000 0000 %111 1111

Im Binärzahlensystem hat das erste Bit den Wert 1, das zweite den Wert 2, das dritte 4, das vierte 8 usw. Das erste Bit steht rechts, links davon folgen Bit 2, dann Bit 3 usw. Um eine Zahl als Binärzahl zu kennzeichnen, wird sie mit einem Präfix oder einem Suffix versehen, z.B. 1001**b** oder %1001.

Im Hexadezimalen Zahlenbereich haben wir die Ziffer 0-9 wie im dezimalen Zahlenbereich, danach folgen die Buchstaben A-F. Die Dezimalzahl 10 ist also die Hexadezimalzahl A, die 11 B, die 12 C, die 13 D, die 14 E und die 15 F. Mit diesen 16 Ziffern – also von 0-F – können wir genau vier Bits darstellen (auch Nibble genannt). Da ein Byte 8 Bits umfasst, brauchen wir zwei hexadezimale Ziffern, um ein Byte darstellen zu können (ein Byte = zwei Nibble). Um eine Zahl als Hexadezimalzahl zu kennzeichnen, wird sie mit einem Präfix oder einem Suffix versehen, z.B. 1F**h** oder 0x1F.

binär binär binär binär dezimal hexadezimal
Bit4 Bit3 Bit2 Bit1

0	0	0	0	0	0
0	0	0	1	1	1
0	0	1	0	2	2
0	0	1	1	3	3
0	1	0	0	4	4
0	1	0	1	5	5
0	1	1	0	6	6
0	1	1	1	7	7
1	0	0	0	8	8
1	0	0	1	9	9
1	0	1	0	10	A
1	0	1	1	11	B
1	1	0	0	12	C
1	1	0	1	13	D
1	1	1	0	14	E
1	1	1	1	15	F

Die Tabelle zeigt in den linken vier Spalten die möglichen Werte der einzelnen Bits eines Nibbles, daneben stehen die entsprechenden Werte dezimal und hexadezimal.

Für Clover (hexadezimal) oder Ozmosis (dezimal) müssen wir die Binärwerte umrechnen.

Dezimal:

Ist die [SIP](#) aktiviert ist das einfach: 0.

Ist die [SIP](#) komplett deaktiviert müssen wir die einzelnen Werte der 7 Bits zusammenzählen:

$$64 + 32 + 16 + 8 + 4 + 2 + 1 = 127$$

Hexadezimal:

Ist die [SIP](#) aktiviert, haben wir den Wert 0x00. Ist die [SIP](#) deaktiviert, haben den Wert 0x7F (8 + 4 + 2 + 1 = F für das erste (rechte) Nibble, 4 + 2 + 1 = 7 für das zweite Nibble). Da beim zweiten Nibble das höchste Bit fehlt (wir haben ja nur 7 Bit), wird das fehlende Bit als 0 gesetzt.

Beispiel:

Sind die Bits 1-3 und 6 und 7 gesetzt, die Bits 4 und 5 nicht, haben wir

dezimal: $64 + 32 + 0 + 0 + 4 + 2 + 1 = 103$

hexadezimal: (0 +) 4 + 2 + 0 und 0 + 4 + 2 + 1 = 0x67

Es gibt noch Bit 8 und 9 in der [SIP](#), deren Gebrauch zum Erstellen einer Hackintosh-Installation mir bisher nicht bekannt ist:

Bit 8 CSR_ALLOW_DEVICE_CONFIGURATION

Bit 9 CSR_ALLOW_ANY_RECOVERY_OS

Das Bit 8 hat den Wert dezimal 128, hexadezimal 80, das Bit 9 dezimal 256. Hexadezimal brauchen wir eine zusätzlich Stelle der Zahl, da wir mit 0xFF (255, %1111 1111) die höchste zweistellige hexadezimale Zahl erreicht haben. Das Bit 9 hat dann also den Wert 0x100.

Beispiel:

Sind alle 9 Bits gesetzt kommen wir auf

binär: %1 1111 1111

dezimal: 511 (256 + 128 + 64 + 32 + 16 + 8 + 4 + 2 + 1)

hexadezimal: 0x1FF

Und hier noch die häufigsten bekannten Werte:

CsrActiveConfig Hexadezimal	CsrActiveConfig Dezimal	Ergebnis
0x03	3	unsigned Kexte werden geladen
0x67	103	Eigentlich alles offen
0x7F	127	Wirklich alles offen