Exhibit D

I, Rudy Pedraza, declare as follows:

1. I am a witness in this case.
2. The code shown attached to this declaration was copied from a publicly available web page on October 8, 2009.
3. The url accessed to copy the listed code was: http://osxbook.com/book/bonus/chapter7/tpmdrmmyth/
4. I copied and saved the code in text format to a file on my computer.
5. On October 8, 2009, I compiled and ran the code using the steps shown in the screenshot attached to this declaration.
6. I performed all of these steps on a MacBook Air laptop computer, which I own and had purchased from Apple.
7. To compile the code as shown in the screenshot, I utilized the "gcc" compiler.
8. The "gcc" compiler was installed from the Xcode distribution that is included with OS X Leopard.
9. It is my belief that any owner of OS X Leopard and an Intel based Apple computer could perform these steps with the same results.

I declare under penalty of perjury under the laws of the United Sates that the foregoing is true and correct to the best of my knowledge and belief.

Executed on October 8, 2009

Rudy Pedraza /by

Rudy Pedraza

Attachment A to Declaration (Source Code)

```
/*
 * smc_read.c: Written for Mac OS X 10.5. Compile as follows:
 *
 * gcc -Wall -o smc_read smc_read.c -framework IOKit
 */

#include <stdio.h>
#include <IOKit/IOKitLib.h>

typedef struct {
	uint32_t key;
	uint8_t __d0[22];
	uint32_t datasize;
	uint8_t __d1[10];
	uint8_t cmd;
	uint32_t __d2;
	uint8_t data[32];
} AppleSMCBuffer_t;

int main(void)
{
	io_service_t service = IOServiceGetMatchingService(kIOMasterPortDefault,
IOServiceMatching("AppleSMC"));

	if (!service)
		return -1;

	io_connect_t port = (io_connect_t)0;
	kern_return_t kr = IOServiceOpen(service, mach_task_self(), 0, &port);
	IOObjectRelease(service);
	if (kr != kIOReturnSuccess)
		return kr;

	AppleSMCBuffer_t inputStruct = { 'OSK0', {0}, 32, {0}, 5, }, outputStruct;
	size_t outputStructCnt = sizeof(outputStruct);

	kr = IOConnectCallStructMethod((mach_port_t)port, (uint32_t)2,
		(const void*)&inputStruct, sizeof(inputStruct),
		(void*)&outputStruct, &outputStructCnt);

	if (kr != kIOReturnSuccess)
		return kr;

	int i = 0;
```

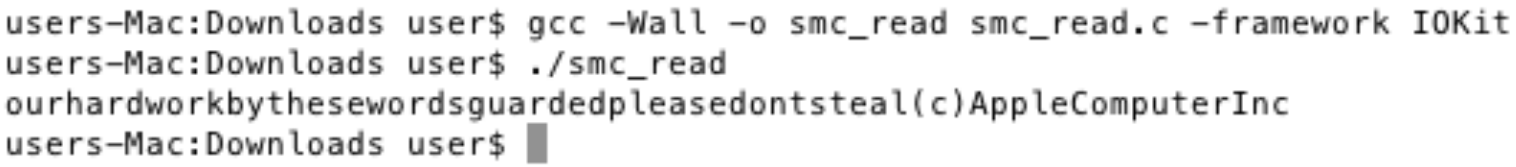
```
        for (i = 0; i < 32; i++)
                printf("%c", outputStruct.data[i]);

        inputStruct.key = 'OSK1';
        kr = IOConnectCallStructMethod((mach_port_t)port, (uint32_t)2,
                (const void*)&inputStruct, sizeof(inputStruct),
                (void*)&outputStruct, &outputStructCnt);

        if (kr == kIOReturnSuccess)
                for (i = 0; i < 32; i++)
                        printf("%c", outputStruct.data[i]); printf("\n");

        return IOServiceClose(port);
}
```

Attachment B to Declaration (Screen Shot)



Attachment C to Declaration