**Question 1: When it comes to Smash the Stack, what are the stack canaries? Give a definition and an example.**

Answer: A Stack canary is a value placed in the stack so that it will be overwritten by a stack buffer that overflows to the return address. It helps detect trouble which in this case is mainly buffer overflow. An example of a stack canary would be a terminator canary which contains NULL, CR, LF, and EOF that terminate most string operations.

**Question 2: The following code contains errors that could be exploited. How would you fix it?**

***hashOut.data = hashes + SSL\_MD5\_DIGEST\_LEN;***

***hashOut.length = SSL\_SHA1\_DIGEST\_LEN;***

***if ((err = SSLFreeBuffer(&hashCtx)) != 0)***

***goto fail;***

***if ((err = ReadyHash(&SSLHashSHA1, &hashCtx)) != 0)***

***goto fail;***

***if ((err = SSLHashSHA1.update(&hashCtx, &clientRandom)) != 0)***

***goto fail;***

***if ((err = SSLHashSHA1.update(&hashCtx, &serverRandom)) != 0)***

***goto fail;***

***if ((err = SSLHashSHA1.update(&hashCtx, &signedParams)) != 0)***

***goto fail;***

***goto fail;***

***if ((err = SSLHashSHA1.final(&hashCtx, &hashOut)) != 0)***

***goto fail;***

***err = sslRawVerify(...);***

Answer:

***hashOut.data = hashes + SSL\_MD5\_DIGEST\_LEN;***

***hashOut.length = SSL\_SHA1\_DIGEST\_LEN;***

***if ((err = SSLFreeBuffer(&hashCtx)) != 0){***

***goto fail;***

***}***

***if ((err = ReadyHash(&SSLHashSHA1, &hashCtx)) != 0){***

***goto fail;***

***}***

***if ((err = SSLHashSHA1.update(&hashCtx, &clientRandom)) != 0){***

***goto fail;***

***}***

***if ((err = SSLHashSHA1.update(&hashCtx, &serverRandom)) != 0){***

***goto fail;***

***}***

***if ((err = SSLHashSHA1.update(&hashCtx, &signedParams)) != 0){***

***goto fail;***

***}***

***if ((err = SSLHashSHA1.final(&hashCtx, &hashOut)) != 0){***

***goto fail;***

***}***

***err = sslRawVerify(...);***

**Question 3: How could you catch this error in the future?**

Answer: This error can be caught in the future through dynamic and static analysis. This includes negative testing in test cases, checking unreachable code, removing misleading indentation, and a variety of other countermeasures.

**Question 4: Heartbleed bug could be traced to a single line of code. What was that line of code?**

Answer: memcpy(bp, p1, payload);