requisiti e di disegno sono strettamente collegati. Il modello propone quattro sessioni di progettazione tra gli sviluppatori e gli stakeholders del software. La prima e la seconda sessione modellano le caratteristiche principali del software e le loro relazioni, identificando i requisiti di alto livello di riservatezza, integrità e disponibilità. Nella terza sessione vengono identificati rischi, vulnerabilità e minacce per il software. La quarta sessione, orientata alla progettazione, indica i requisiti di sicurezza per rimuovere le vulnerabilità identificate.

AEGIS suggerisce anche una metodologia di analisi dei rischi da utilizzare durante le sessioni 3 e 4 finalizzate alla progettazione. Questo metodo di analisi dei rischi ha le seguenti fasi principali:

- Determinazione delle vulnerabilità.
- Determinazione del costo e della probabilità di un attacco in ambiente distribuito (inclusi i ruoli delle persone coinvolte e i task che verranno eseguiti sul software).
- Selezione dei requisiti di sicurezza basate sulle indicazioni dell'esperto di sicurezza.
- Valutazione costi-benefici dei requisiti di sicurezza selezionati.
- Il confronto tra il costo di ogni attacco, commisurato con la probabilità che possa verificarsi, e il costo dei requisiti di sicurezza.
- Selezione dei requisiti di sicurezza sulla base dell'efficacia e dei costi.

8.2.4 Secure Software Development Model (SSDM)

SSDM[33] è un processo che incorpora diverse attività di sicurezza in un modello SDLC a cascata (cascade). Secondo SSDM, la modellazione delle minacce dovrebbe essere eseguita in fase di specifica dei requisiti. Il risultato di questa modellazione dovrebbe essere una check-list contenente tutte le potenziali vulnerabilità e attacchi. Tali elenchi di fatto dovrebbero essere dati in input alla fase di sviluppo.

Dopo la modellazione delle minacce, è necessario definire una policy che indichi chiaramente come saranno raggiunti gli obiettivi di sicurezza prefissati.

Tale policy, come sottolineato dal SSDM, è un insieme di decisioni di gestione di alto livello come ad esempio minimizza l'impatto degli errori in tutto il processo di sviluppo, correggendoli non appena vengono rilevati. I test di penetrazione rappresentano, nel modello SSDM, l'unica attività SSD per la fase security assurance.

8.2.5 Aprville and Pourzandi's Secure Software Development Life Cycle Process

[4]Aprville e Pourzandi[34] propongono un processo SSDLC sulla base della loro esperienza, maturata durante lo sviluppo di un software di instant messaging. Secondo il loro processo [5], il primo passo nella fase di specifica dei requisiti è quello di individuare gli obiettivi di alto livello, per quanto riguarda la sicurezza (riservatezza, integrità e disponibilità) del software in fase di sviluppo, considerando il suo ambiente di distribuzione. Per gli obiettivi di sicurezza a basso livello, la modellazione delle minacce dovrebbe essere di supporto nella costruzione di un insieme di requisiti di sicurezza. La priorità di tali requisiti può essere

---

[32] I. Flechais, C. Mascolo, and M.A. Sasse, "Integrating Security and Usability into the Requirements and Design Process," International Journal of Electronic Security and Digital Forensics, Inderscience Publishers, Geneva, Switzerland, 2007, vol. 1, no. 1, pp. 12-26.

[33] A.S. Sodiya, S.A. Onashoga, and O.B. Ajayi, "Towards Building Secure Software Systems," Issues in Informing Science and Information Technology, Informing Science Institute, California, USA, 2006, vol. 3, pp. 635-646.

[34] A. Apvrille and M. Pourzandi, "Secure Software Development by Example," IEEE Security and Privacy, IEEE CS Press, 2005, vol. 3, no. 4, pp. 10-17.