Checklist Secure SDLC

A. *Checklist* Persyaratan Keamanan

No.	Persyaratan Keamanan	Detail Persyaratan Keamanan	Keterangan	*Checklist*
1.1	**Sumber dari Persyaratan Keamanan**			
1.1.1	**Identifikasi Persyaratan *Core Security***			
	a. Persyaratan kerahasiaan (*confidentiality*)	1) Apakah diterapkan kriptografi, dengan: a) Mekanisme terang-terangan (*overt*). Misalnya: enkripsi, hash? b) Mekanisme terselubung (*covert*). Misalnya: steganografi, *digital watermarking*? 2) Apakah diterapkan penyamaran (*masking*)? 3) Apakah diterapkan kerahasiaan pada siklus hidup informasi: a) *Data in transit*? b) *Data in process*? c) *Data in storage*?		
	b. Persyaratan integritas (*integrity*)	Apakah terdapat diterapkan pelindungan untuk memastikan: 1) Integritas sistem? 2) Integritas data?		
	c. Persyaratan ketersediaan (*availability*)	Apakah diterapkan pelindungan dari: 1) Upaya penghancuran perangkat lunak/data? 2) Serangan *Denial of Service* (DoS)?		
	d. Persyaratan autentikasi	Apakah diterapkan autentikasi untuk memastikan keabsahan dan memvalidasi identitas pada: 1) Pengguna? 2) Administrator? 3) *Super user* (jika ada)?		
	e. Persyaratan otorisasi	Apakah diterapkan kontrol akses untuk memastikan otoritas entitas yang diautentikasi pada sumber daya?		
	f. Persyaratan akuntabilitas	Apakah diterapkan fungsi untuk pemeliharaan catatan riwayat tindakan pengguna (*audit trail*)?		
1.1.2	**Mengidentifikasi Persyaratan Umum**			

No.	Persyaratan Keamanan	Detail Persyaratan Keamanan	Keterangan	*Checklist*
	a. Persyaratan manajemen sesi	Apakah diterapkan *session identifier* untuk melacak perilaku pengguna dan mempertahankan status terautentikasi?		
	b. Persyaratan manajemen *error* dan *exception*	Apakah diterapkan pelindungan terhadap informasi *error* dan *exception*?		
	c. Persyaratan Manajemen Parameter Konfigurasi	Apakah diterapkan pelindungan terhadap informasi parameter dan kode konfigurasi perangkat lunak yang menyusun perangkat lunak?		
1.1.3	**Mengidentifikasi Persyaratan Operasional**			
	a. Persyaratan Lingkungan Penerapan	Apakah lingkungan dimana perangkat lunak akan digunakan telah diidentifikasi dan dianalisis?		
	b. Persyaratan Pengarsipan	Apakah terdapat kebutuhan pengarsipan untuk mendukung kelangsungan bisnis atau untuk memenuhi persyaratan peraturan?		
	c. Persyaratan Anti Pembajakan (*Anti-Piracy*)	Apakah diterapkan pelindungan anti-pembajakan pada perangkat lunak yang akan dikomersialkan?		
1.1.3	**Mengidentifikasi Persyaratan Lain**			
	a. Persyaratan Urutan dan Waktu	Apakah terdapat pelindungan terhadap kondisi *race condition* atau serangan *Time of Check/Time of Use* (TOC/TOU)?		
	b. Persyaratan Internasional	Apakah terdapat persyaratan hukum dan teknologi secara internasional yang harus dipatuhi?		
	c. Persyaratan Pengadaan	Apakah terdapat persyaratan keamanan perangkat lunak sudah dipenuhi pada perangkat lunak yang akan dibeli?		
1.2	**Klasifikasi Data**			
1.2.1	**Tipe Data**			
	a. Data Terstruktur	Apakah terdapat data terstruktur (*database*) pada perangkat lunak?		
	b. Data Tidak Terstruktur	Apakah terdapat data tidak terstruktur (gambar, video, *email*, dokumen, dan teks) pada perangkat lunak?		
1.2.2	Memberi Label pada Data	Apakah terdapat pelabelan pada data sesuai aspek C-I-A?		
1.2.3	Kepemilikan dan Peran pada Data	1) Apakah pemilik data sudah terdefinisi? 2) Apakah pemilik data sudah menjalankan perannya dalam mengelola data?		
1.2.4	*Data Lifecycle Management* (DLM)	Apakah terdapat prosedur dan praktik *Data Lifecycle Management* (DLM)?		

No.	Persyaratan Keamanan	Detail Persyaratan Keamanan	Keterangan	*Checklist*
1.2.5	Persyaratan Privasi	1) Apakah terdapat pedoman yang mengatur implementasi privasi pada data? 2) Apakah kontrol privasi sudah diterapkan sesuai persyaratan?		
1.3	**Pemodelan *Use Case* dan *Misuse Case***			
1.3.1	Menganalisis Skenario *Use Case*	1) Apakah sudah dibuat diagram *use case*? 2) Apakah sudah analisis terhadap skenario *use case*?		
1.3.2	Menganalisis Skenario *Misuse Case*	1) Apakah sudah dibuat diagram *misuse case*? 2) Apakah sudah analisis terhadap skenario *misuse case*?		
1.3.3	Membuat Model Serangan	1) Apakah sudah diidentifikasi serangan yang relevan dengan perangkat lunak? 2) Apakah sudah dianalisis siapa saja yang dapat menjadi sumber ancaman?		
1.3.4	Memilih Kontrol Mitigasi	Apakah sudah dianalisis kontrol keamanan untuk memitigasi serangan pada perangkat lunak?		
1.4	**Manajemen Risiko**			
1.4.1	Penilaian Risiko	Apakah sudah dilakukan aktivitas penilaian risiko berikut: 1) Karakterisasi sistem? 2) Identifikasi ancaman? 3) Identifikasi kerentanan? 4) Analisis kontrol? 5) Penentuan kemungkinan? 6) Analisis dampak? 7) Penentuan risiko? 8) Rekomendasi kontrol? 9) Dokumentasi hasil?		
1.4.2	Mitigasi Risiko	Apakah sudah dilakukan aktivitas berikut: 1) Penentuan opsi mitigasi risiko? 2) Penentuan strategi mitigasi risiko? 3) Menentukan pendekatan pada implementasi kontrol? 4) Mengkategorikan kontrol? 5) Analisis biaya dan manfaat? 6) Menentukan risiko residual?		
1.4.3	Evaluasi dan Penilaian Risiko	Apakah proses evaluasi dan penilaian sudah dilakukan secara berkala?		

B. *Checklist* Desain Keamanan

No.	Persyaratan Keamanan	Detail Persyaratan Keamanan	Keterangan	*Checklist*
2.1	**Pertimbangan Desain *Core Security***			
2.1.1	Desain Kerahasiaan	Apakah sudah terdapat desain aspek kerahasiaan menggunakan: 1) Teknik kriptografi, dengan mempertimbangkan: a) Algoritma enkripsi? b) Ukuran kunci? c) Manajemen kunci? 2) Penyamaran (*masking*)?		
2.1.2	Desain Integritas	Apakah sudah terdapat desain aspek integritas menggunakan: 1) *Hashing* (fungsi *hash*)? 2) Integritas referensi? 3) Penguncian sumber daya?		
2.1.3	Desain Ketersediaan	Apakah sudah terdapat desain aspek ketersediaan menggunakan: 1) Pengkodean perangkat lunak? 2) Replikasi? 3) *Failover*? 4) Skalabilitas?		
2.1.4	Desain Autentikasi	Apakah sudah terdapat desain untuk autentikasi menggunakan: 1) Autentikasi 2FA atau MFA? 2) *Single Sign-On* (SSO)?		
2.1.5	Desain Otorisasi	Apakah sudah terdapat desain untuk otorisasi menggunakan mekanisme kontrol akses: 1) Direktori? 2) *Access Control List* (ACL)? 3) Matriks kontrol akses? 4) Kapabilitas menggunakan token yang tidak dapat dipalsukan? 5) Kontrol akses berorientasi prosedur?		
2.1.6	Desain Akuntabilitas	Apakah sudah terdapat desain untuk mendukung *audit trail*?		
2.2	**Pertimbangan Desain Tambahan**			
2.2.1	Bahasa Pemrograman	Apakah bahasa pemrograman yang akan digunakan sudah ditentukan?		
2.2.2.	Jenis, Format, Jangkauan, dan Panjang Data	Apakah sudah ditentukan: 1) Tipe data primitif atau *built-in*?		

No.	Persyaratan Keamanan	Detail Persyaratan Keamanan	Keterangan	*Checklist*
		2) Tipe data yang ditentukan oleh pemrogram? 3) Nilai dan operasi yang diizinkan? 4) Ketidakcocokan dan kesalahan konversi?		
2.2.3	Keamanan *Database*	Apakah sudah terdapat desain keamanan *database* dengan mempertimbangkan: 1) *Polyinstantiation*? 2) Enkripsi *database*? 3) Normalisasi? 4) *Trigger* dan *view*?		
2.2.4	Desain Antarmuka	Apakah sudah terdapat desain antarmuka pada: 1) Antarmuka pengguna (*user interface/*UI)? 2) Antarmuka pemrograman aplikasi (*application programming interface/*API)? 3) Antarmuka manajemen keamanan (*security management interface/*SMI)? 4) Antarmuka *out-of-band*? 5) Antarmuka *log*?		
2.2.5	Interkonektivitas	Apakah sudah terdapat desain interkonektivitas pada perangkat lunak hulu dan hilir?		
2.3	**Pemodelan Ancaman**			
2.3.1	Dekomposisi Perangkat Lunak	Apakah sudah dilakukan dekomposisi perangkat lunak dengan mempertimbangkan: 1) Ketergantungan eksternal? 2) Titik masuk (vektor serangan)? 3) Aset? 4) *Attack surface*? 5) Tingkat kepercayaan? 6) Analisis aliran data? 7) Analisis transaksi ? 8) Diagram *data flow*?		
2.3.2	Menentukan dan Mengurutkan Ancaman	1) Apakah sudah mengidentifikasi ancaman dengan metode pengkategorian ancaman (misalnya STRIDE)? 2) Apakah sudah mengurutkan ancaman menggunakan model peringkat ancaman-risiko (misalnya DREAD)?		

No.	Persyaratan Keamanan	Detail Persyaratan Keamanan	Keterangan	*Checklist*
2.3.3	Menentukan Penanggulangan dan Mitigasi	1) Apakah sudah mengurutkan ancaman berdasarkan peringkat risiko? 2) Apakah sudah mengidentifikasi penanggulangan berdasarkan pemetaan ancaman-penanggulangan?		

C. *Checklist* Pengembangan Keamanan

No.	Persyaratan Keamanan	Detail Persyaratan Keamanan	Keterangan	*Checklist*
3.1	**Kerentanan dan Kontrol Umum pada Perangkat Lunak**			
3.1.1	*Database* Kerentanan	1) Apakah organisasi mengacu ke daftar kerentanan perangkat lunak (OWASP Top 10 Project atau CWE/25)? 2) Apakah terdapat *database* kerentanan yang ditemukan pada perangkat lunak yang diimplementasikan?		
3.2.3	Praktek Pengkodean Defensif	Apakah teknik pengkodean defensif untuk mengurangi *attack surface* sudah dipraktekan?		
3.2	**Proses Perangkat Lunak yang Aman**			
3.2.1	Versi pada Kode Sumber	Apakah versi pada kode sumber sudah diterapkan?		
3.2.2	Analisis Kode	Apakah analisis kode diterapkan menggunakan: 1) Analisis kode statis? 2) Analisis kode dinamis?		
3.2.3	Reviu Kode	Apakah dilakukan evaluasi kode sumber secara sistematis dan manual?		
3.2.4	Pengujian Pengembang	Apakah pengembang sudah melakukan pengujian menggunakan alat dan teknik pengujian yang sistematis yang meliputi: 1) Pengujian unit? 2) Pengujian integrasi? 3) Pengujian regresi? 4) Pengujian perangkat lunak?		
3.3	**Mengamankan Lingkungan Pengembangan**			
3.3.1	Mengamankan Akses Fisik ke Perangkat Lunak yang Membangun Kode	Apakah akses fisik ke perangkat lunak yang membangun kode sudah diamankan?		

No.	Persyaratan Keamanan	Detail Persyaratan Keamanan	Keterangan	*Checklist*
3.3.2	Menggunakan *Access Control List* (ACL)	Apakah digunakan *Access Control List* (ACL) untuk mencegah akses pengguna yang tidak sah?		
3.3.3	Menggunakan Perangkat Lunak Kontrol Versi	Apakah perangkat lunak kontrol versi sudah digunakan?		
3.3.4	*Build Automation*	Apakah *build automation* sudah digunakan?		
3.3.5	Penandatanganan Kode (*Code Signing*)	Apakah *code signing* sudah digunakan?		

D. *Checklist* Pengujian Keamanan

No.	Persyaratan Keamanan	Detail Persyaratan Keamanan	Keterangan	*Checklist*
4.1	**Validasi Permukaan Serangan (*Attack Surface*)**			
4.1.1	Pengujian Pasca Pengembangan	Apakah sudah dilakukan *dynamic code analysis*, yaitu pemeriksaan kode pada saat program dijalankan?		
4.1.2	Pengujian Keamanan Menggunakan Metode Pengujian Keamanan	Apakah dilakukan pengujian keamanan menggunakan: 1) Pengujian *white box*? 2) Pengujian *black box*? 3) Pengujian validasi kriptografi?		
4.1.3	Melakukan Pengujian Keamanan Perangkat Lunak untuk Jaminan Kualitas	Apakah dilakukan pengujian keamanan perangkat lunak untuk menjamin kualitas yang meliputi: 1) Pengujian validasi *input*; 2) Pengujian untuk kontrol cacat injeksi (*injection flaw*); 3) Pengujian untuk kontrol serangan *scripting* (*scripting attack*); 4) Pengujian untuk kontrol nir-penyangkalan (*non-repudiation*); 5) Pengujian untuk kontrol *spoofing*; 6) Pengujian untuk kontrol *error* dan *exception handling* (*failure testing*); 7) Pengujian untuk kontrol eskalasi hak istimewa (*privilege escalation*); 8) Pengujian untuk pelindungan anti-pembalikan (*anti-reversing*); 9) Pengujian ketahanan (*stress testing*).		
4.2	**Manajemen Data Pengujian**			

No.	Persyaratan Keamanan	Detail Persyaratan Keamanan	Keterangan	*Checklist*
4.2.1	Mengidentifikasi *Output* Data Pengujian untuk Memastikan Persyaratan Perangkat Lunak	1) Apakah *output* dari data pengujian sudah sesuai dengan harapan atau telah memenuhi persyaratan? 2) Apakah menggunakan data tiruan di lingkungan pengujian atau simulasi?		
4.2.1	Melakukan Pengujian dengan Transaksi Sintetis	Apakah transaksi pada pengujian dilakukan menggunakan data *dummy* yang tidak terkait dengan proses bisnis sebenarnya?		
4.2.3	Solusi pada Manajemen Data Pengujian	Apakah digunakan solusi (alat atau layanan) pada manajemen data pengujian?		
4.2.4	Pelaporan dan Pelacakan Cacat	Apakah terdapat mekanisme untuk melaporkan cacat (*defect/flaw*) dan kemudian melacak *bug* pengkodean, cacat desain (*design flaw*), anomali perilaku (*logic flaw*), *error*, *fault*, dan kerentanan pada perangkat lunak?		

E. *Checklist* Penerapan Keamanan

No.	Persyaratan Keamanan	Detail Persyaratan Keamanan	Keterangan	*Checklist*
5.1	**Pertimbangan Penerimaan Perangkat Lunak**			
5.1.1	Kriteria Penyelesaian	Apakah terdapat kriteria penyelesaian untuk fungsionalitas dan keamanan perangkat lunak dengan *milestone* yang jelas?		
5.1.2	Manajemen Perubahan	Apakah terdapat mekanisme untuk menangani permintaan perubahan pada penerapan perangkat lunak?		
5.1.3	Persetujuan untuk Menerapkan atau Merilis	Apakah terdapat mekanisme persetujuan/penolakan untuk menerapkan/merilis perangkat lunak?		
5.1.4	Kebijakan Penerimaan dan Pengecualian Risiko	Apakah terdapat kebijakan untuk penerimaan dan pengecualian risiko?		
5.1.5	Dokumentasi Perangkat Lunak	Apakah terdapat dokumentasi perangkat lunak yang memadai, yang meliputi: 1) Perancangan? 2) Penginstalan? 3) Pengaturan konfigurasi? 4) Penggunaan? 5) Pengaturan?		
5.2	**Verifikasi dan Validasi (V&V)**			

No.	Persyaratan Keamanan	Detail Persyaratan Keamanan	Keterangan	*Checklist*
5.2.1	Reviu	Apakah dilakukan reviu pada akhir setiap tahap untuk memastikan bahwa perangkat lunak berfungsi seperti yang diharapkan dan memenuhi spesifikasi bisnis yang diharapkan?		
5.2.2	Pengujian	Apakah sudah dilakukan pengujian untuk memastikan persyaratan sudah dipenuhi dan menentukan penyimpangan yang diharapkan dengan: 1) Pengujian deteksi kesalahan? 2) Pengujian penerimaan? 3) Pengujian pihak independen?		
5.3	**Sertifikasi dan Akreditasi (C&A)**			
5.3.1	Mendapatkan Sertifikasi	Apakah dilakukan sertifikasi perangkat lunak yang mencakup evaluasi penjaminan: 1) Hak pengguna, hak istimewa, dan manajemen profil? 2) Sensitivitas data dan aplikasi serta pengendalian yang sesuai? 3) Konfigurasi perangkat lunak, fasilitas, dan lokasi? 4) Interkonektivitas dan ketergantungan? 5) Mode keamanan operasional?		
5.3.2	Mendapatkan Akreditasi	Apakah terdapat mendapatkan akreditasi atas penerimaan formal pihak manajemen pada penggunaan perangkat lunak?		
5.4	**Instalasi**			
5.4.1	Penguatan (*Hardening*)	1) Apakah dilakukan penguatan (*hardening*) sistem operasi *host* dengan menggunakan *baseline*, *update*, dan *patch*? 2) Apakah dilakukan penguatan (*hardening*) perangkat lunak melibatkan pengaturan konfigurasi dan perancangan perangkat lunak agar aman secara *default*?		
5.4.2	Konfigurasi Lingkungan	1) Apakah sudah dipastikan bahwa lingkungan pengembangan dan pengujian cocok dengan lingkungan produksi? 2) Apakah terdapat *checklist* pra-instalasi untuk menentukan parameter yang diperlukan dalam menjalankan perangkat lunak dengan konfigurasi yang tepat?		
5.4.3	Manajemen Rilis	Apakah terdapat mekanisme untuk memastikan rilis perangkat lunak dengan benar ke dalam lingkungan komputasi operasi?		
5.4.4	*Bootstrap* dan *Startup* yang Aman	Apakah *Power-On Self-Test* (POST) sudah dijalankan setelah instalasi perangkat lunak?		

F. *Checklist* Pemeliharaan Keamanan

No.	Persyaratan Keamanan	Detail Persyaratan Keamanan	Keterangan	*Checklist*
6.1	**Operasi, Pemantauan dan Pemeliharaan**			
6.1.1	Melaksanakan Pengamanan Operasi	Apakah sudah diterapkan kontrol keamanan operasi yang meliputi: 1) Kontrol detektif? 2) Kontrol preventif? 3) Kontrol pencegahan? 4) Kontrol korektif? 5) Kontrol kompensasi?		
6.1.2	Pemantauan Berkelanjutan	1) Apakah terdapat pemantauan berkelanjutan terhadap sistem, perangkat lunak, atau proses? 2) Apakah terdapat metrik terkait yang mengukur kinerja aktual dan operasi pemantauan?		
6.1.3	Audit untuk Pemantauan	Apakah audit terhadap catatan dan aktivitas perangkat lunak sudah dilakukan secara berkala?		
6.2	**Manajemen Insiden**			
6.2.1	Menentukan Peristiwa, Peringatan, dan Insiden	Apakah terdapat kebijakan atau prosedur yang memuat definisi tentang peristiwa, peringatan, dan insiden?		
6.2.2	Identifikasi Jenis Insiden	Apakah terdapat kebijakan atau prosedur yang memuat tentang jenis insiden yang ditangani di organisasi?		
6.2.3	Proses Tanggap Insiden	Apakah terdapat kebijakan atau prosedur yang memuat proses tanggap insiden?		
6.3	**Manajemen Permasalahan**			
6.3.1	Pemberitahuan Insiden	Apakah terdapat prosedur yang mengatur tentang pemberitahuan insiden?		
6.3.2	Analisis Akar Penyebab	Apakah terdapat prosedur yang mengatur tentang analisis akar penyebab dari permasalahan?		
6.3.3	Penentuan Solusi	Apakah terdapat prosedur yang mengatur tentang penentuan solusi dari permasalahan?		
6.3.4	Permintaan Perubahan	Apakah terdapat prosedur yang mengatur tentang permintaan perubahan terkait permasalahan?		
6.3.5	Menerapkan Solusi	Apakah terdapat prosedur yang mengatur tentang penerapan solusi dari permasalahan?		

No.	Persyaratan Keamanan	Detail Persyaratan Keamanan	Keterangan	*Checklist*
6.3.6	Memantau dan Melaporkan	Apakah terdapat prosedur yang mengatur tentang pelaporan permasalahan yang bertujuan memantau penerapan solusi.		
6.4	**Manajemen Perubahan**			
6.4.1	Manajemen *Patch* dan Kerentanan	Apakah terdapat prosedur yang mengatur tentang manajemen *patch* dan kerentanan?		
6.4.2	Pencadangan, Pemulihan, dan Pengarsipan	Apakah terdapat prosedur yang mengatur tentang pencadangan, pemulihan, dan pengarsipan?		
6.5	**Pembuangan**			
6.5.1	Kebijakan Akhir Masa Pakai (*End-of-Life*)	Apakah terdapat kebijakan yang mengatur tentang *End-Of-Life* (EOL) perangkat lunak?		
6.5.2	Kriteria Pembuangan (*Sunset Criteria*)	Apakah terdapat prosedur yang mengatur tentang kriteria pembuangan perangkat lunak (*sunsiet criteria*)?		
6.5.3	Proses Pembuangan (*Sunset Process*)	Apakah terdapat prosedur yang mengatur tentang proses pembuangan perangkat lunak (*sunset process*)?		
6.5.4	Pembuangan Informasi dan Sanitasi Media	Apakah terdapat prosedur yang mengatur tentang pembuangan informasi dan sanitasi media?		