TLP:CLEAR

AMAN BERDASARKAN DESAIN

MENGUBAH KESEIMBANGAN RISIKO KEAMANAN SIBER:

PRINSIP DAN PENDEKATAN UNTUK PERANGKAT LUNAK AMAN BERDASARKAN DESAIN

CISA | NSA | FBI | ACSC | CCCS | CERT NZ | NCSC-NZ | NCSC-UK | BSI | NCSC-NL
NCSC-NO | NÚKIB | INCD | KISA | NISC-JP | IPCERT/CC | CSA | CSIRTAMERICAS

TLP:CLEAR

Communications Security Establishment

Canadian Centre for Cyber Security

Centre de la sécurité des télécommunications

Centre canadien pour la cybersécurité

Konten

IKHTISAR:
RENTAN BERDASARKAN DESAIN

Teknologi terintegrasikan ke dalam hampir setiap aspek kehidupan sehari-hari, seperti sistem-sistem yang menggunakan internet terus menghubungkan kita ke sistem-sistem kritis yang berpengaruh secara langsung terhadap kemakmuran ekonomi, mata pencaharian, dan bahkan kesehatan, mulai dari manajemen identitas pribadi hingga perawatan medis. Salah satu contoh kelemahan seperti ini adalah penyusupan siber global yang menyebabkan pembatalan operasi bedah di rumah sakit dan mengubah perawatan pasien. Teknologi yang tidak aman dan kerentanan dalam sistem-sistem kritis dapat mengundang penyusupan siber yang berbahaya, sehingga menimbulkan potensi yang membahayakan keamanan[1].

Akibatnya, penting bagi produsen perangkat lunak, menjadikan aman berdasarkan desain dan aman berdasarkan bawaan sebagai titik fokus perancangan dan proses pengembangan produk. Beberapa vendor telah membuat kemajuan besar mendorong industri ini dalam hal jaminan perangkat lunak, sementara yang lain masih tertinggal. Organisasi-organisasi penggagas sangat mendorong setiap produsen teknologi untuk membangun produk-produk mereka berdasarkan pengurangan beban keamanan siber pada pelanggan, termasuk mencegah mereka dari keharusan terus-menerus melakukan pemantauan, pembaruan dan pengendalian kerusakan pada sistem mereka guna memitigasi penyusupan siber. Kami juga mendesak produsen perangkat lunak untuk membangun produk mereka dengan cara yang memfasilitasi automasi konfigurasi, pengawasan dan pembaruan rutin. Para produsen didorong untuk bertanggung jawab dalam peningkatan keamanan bagi pelanggan mereka. Secara historis, produsen perangkat lunak telah mengandalkan perbaikan kerentanan yang ditemukan setelah pelanggan memakai produk, mengharuskan pelanggan untuk menerapkan tambalan tersebut dengan biaya mereka sendiri. Hanya dengan menggabungkan praktik-praktik aman berdasarkan desain kita akan memutus lingkaran setan membuat dan menerapkan perbaikan secara terus-menerus. **Catatan:** Istilah “aman berdasarkan desain” mencakup aman berdasarkan desain dan aman berdasarkan bawaan.

Untuk mencapai standar keamanan perangkat lunak yang tinggi ini, organisasi penggagas mendorong para produsen untuk memprioritaskan integrasi keamanan produk sebagai prasyarat kritis bagi fitur dan kecepatan ke pasar. Seiring waktu, tim teknik akan mampu mewujudkan ritme kondisi stabil baru, keamanan benar-benar terancang dan memerlukan sedikit usaha untuk pemeliharaannya.

Mencerminkan perspektif ini, Uni Eropa memperkuat pentingnya keamanan produk dalam Undang-Undang Ketahanan Siber, yang menekankan bahwa para produsen harus mengimplementasikan keamanan selama masa pakai suatu produk agar menghindarkan produsen tersebut memperkenalkan produk-produk rentan ke pasar.

[1] Organisasi penggagas mengakui bahwa istilah "safety" (keamanan) memiliki makna ganda tergantung dari konteks yang digunakan. Untuk tujuan panduan ini, "safety" mengacu kepada meningkatkan standar keamanan teknologi untuk melindungi pelanggan dari aktivitas siber yang berbahaya.

Untuk menciptakan masa depan dengan teknologi dan produk-produk terkait lebih aman bagi para pelanggan, organisasi penggagas mendesak para produsen untuk merombak program desain dan pengembangan mereka sehingga hanya mengirimkan produk yang aman berdasarkan desain dan aman berdasarkan bawaan. Jauh sebelum pengembangan, produk-produk aman berdasarkan desain dikonseptualisasikan bersama dengan keamanan pelanggan sebagai tujuan utama bisnis, bukan hanya fitur teknis saja. Produk aman berdasarkan desain dimulai dengan tujuan tersebut sebelum pengembangan. Produk-produk yang sudah ada dapat berkembang menjadi aman berdasarkan desain setelah beberapa iterasi. Produk aman berdasarkan bawaan adalah produk yang aman digunakan secara langsung setelah "dikeluarkan dari kotak" dengan sedikit atau tanpa perubahan konfigurasi dan fitur-fitur keamanan tersedia tanpa biaya tambahan. Bersama-sama, kedua filosofi ini memindahkan sebagian besar beban untuk tetap aman kepada produsen dan mengurangi kemungkinan pelanggan menjadi korban insiden keamanan yang disebabkan oleh salah konfigurasi, tambalan pelanggan kurang cepat, atau banyak masalah umum lainnya.

Badan Keamanan Siber dan Keamanan Infrastruktur (CISA), Badan Keamanan Nasional (NSA), Biro Investigasi Federal (FBI) dan mitra-mitra internasional berikut ini[2] memberikan rekomendasi dalam panduan ini sebagai peta jalan bagi produsen perangkat lunak guna memastikan keamanan pada produk mereka:

- Pusat Keamanan Siber Australia (ACSC)
- Pusat Keamanan Siber Kanada (CCCS)
- Pusat Keamanan Siber Nasional Kerajaan Inggris Raya (NCSC-UK)
- Kantor Federal untuk Keamanan Informasi Jerman (BSI)
- Pusat Keamanan Siber Nasional Belanda (NCSC-NL)
- Pusat Keamanan Siber Nasional Norwegia (NCSC-NO)
- Tim Respons Darurat Komputer Selandia Baru (CERT NZ) dan Pusat Keamanan Siber Nasional Selandia Baru (NCSC-NZ).
- Badan Internet & Keamanan Korea (KISA)
- Direktorat Siber Nasional Israel (INCD)
- Pusat Nasional Kesiapan Insiden dan Strategi untuk Keamanan Siber Jepang (NISC) dan Pusat Koordinasi Tim Respons Darurat Komputer Jepang (JPCERT/CC)
- OAS/CICTE Jaringan Tim Respons Insiden Siber Pemerintah (CSIRT) Amerika
- Badan Keamanan Siber Singapura (CSA)
- Badan Siber Nasional dan Keamanan Informasi Republik Ceko (NÚKIB)

Organisasi penggagas mengakui kontribusi oleh banyak mitra sektor swasta dalam mempercanggih keamanan berdasarkan desain dan keamanan berdasarkan bawaan. Produk ini ditujukan untuk melanjutkan pembahasan internasional mengenai prioritas kunci, investasi, dan keputusan yang diperlukan untuk mencapai masa depan di mana teknologi yang aman dan terjamin, serta tangguh berdasarkan desain dan bawaan. Untuk tujuan tersebut, organisasi-penggagas mencari umpan balik tentang produk ini dari pihak-pihak yang berkepentingan dan bermaksud mengadakan serangkaian sesi rapat dengar untuk lebih lanjut menyempurnakan, menetapkan, dan meningkatkan panduan kami untuk mencapai tujuan-tujuan bersama.

Untuk informasi lebih lanjut tentang pentingnya keamanan produk, bacalah artikel CISA, Akibat Teknologi Tidak Aman dan Hal Yang Bisa Dilakukan Untuk Mengatasinya.

[2] Selanjutnya disebut sebagai "organisasi-organisasi penggagas".

YANG BARU

Publikasi awal laporan ini menimbulkan banyak pembicaraan dalam industri perangkat lunak. Berita harian tentang organisasi dan perorangan yang tersusupi menyoroti perlunya lebih banyak pembahasan mengenai cara menangani masalah kronis dan sistemik pada produk-produk perangkat lunak.

Setelah dirilis pada bulan April 2023, organisasi-organisasi penggagas (selanjutnya disebut sebagai "kami") telah menerima umpan balik yang bijaksana dari ratusan orang, perusahaan, dan asosiasi perdagangan. Permintaan yang paling banyak diminta dalam umpan balik adalah untuk memberikan perincian lebih lanjut tentang tiga prinsip yang berlaku bagi produsen perangkat lunak maupun para pelanggan mereka. Dalam dokumen ini, kami memperluas laporan asal dan menyentuh tema-tema lain seperti besarnya produsen dan pelanggan, kedewasaan pelanggan, dan ruang lingkup prinsip-prinsipnya.

Perangkat lunak ada di mana-mana dan tidak ada laporan satupun yang mampu mencakup seluruh sistem perangkat lunak, pengembangan produk perangkat lunak, pemakaian dan pemeliharaan oleh pelanggan, serta integrasi dengan sistem lainnya secara memadai. Untuk panduan berikut ini yang tidak memetakan dengan jelas lingkungan tertentu, kami menantikan kabar dari komunitas bagaimana praktik-praktik yang dijelaskan dalam makalah ini menuntun pada peningkatan keamanan tertentu.

Laporan ini juga berlaku bagi produsen sistem dan model perangkat lunak kecerdasan buatan (AI). Meskipun mungkin berbeda dari bentuk perangkat lunak tradisional, praktik-praktik keamanan fundamental masih berlaku pada sistem dan model AI. Beberapa praktik aman berdasarkan desain mungkin memerlukan modifikasi untuk menjelaskan pertimbangan khusus AI, namun ketiga prinsip aman berdasarkan desain yang menyeluruh berlaku bagi semua sistem AI.

Kami mengakui bahwa mentransformasikan siklus hidup pengembangan perangkat lunak (SDLC) agar sesuai dengan prinsip aman berdasarkan desain ini bukanlah tugas yang sederhana dan dapat memakan waktu. Lagipula, produsen perangkat lunak yang lebih kecil mungkin mengalami kesulitan dalam menerapkan sebagian besar saran ini. Kami yakin bahwa industri perangkat lunak perlu menyediakan secara luas alat-alat dan prosedur yang membuat produknya lebih aman. Karena banyak orang dan organisasi-organisasi memusatkan perhatian mereka ke peningkatan keamanan perangkat lunak, kami yakin ada ruang bagi inovasi yang akan mempersempit kesenjangan antara produsen perangkat lunak besar dan kecil demi keuntungan semua pelanggan.

Pembaruan pada laporan asli aman berdasarkan desain ini merupakan bagian dari komitmen kami dalam membangun kemitraan dengan komunitas pemangku kepentingan yang saling berhubungan dan mendukung ekosistem teknologi kami. Ini adalah hasil umpan balik dari berbagai bagian ekosistem tersebut, dan kami akan terus mendengarkan dan belajar dari berbagai perspektif. Meskipun terdapat banyak tantangan di masa depan, kami sangat optimis ketika kami mengetahui lebih banyak orang-orang dan organisasi-organisasi yang telah mengadopsi filosofi aman berdasarkan desain, yang seringkali berhasil.

CARA MENGGUNAKAN DOKUMEN INI

Kami mendesak produsen perangkat lunak untuk mematuhi prinsip-prinsip dalam dokumen ini. Produsen perangkat lunak dapat mendemonstrasikan komitmen mereka dengan mendokumentasikan tindakan mereka secara publik, sejalan dengan langkah-langkah berikut ini: Kami mendorong para produsen perangkat lunak untuk mencari taktik yang sesuai dengan semangat prinsip ini dan membuat artefak yang akan memberi bukti yang meyakinkan bagi pelanggan saat ini maupun calon pelanggan yang masih ragu-ragu bahwa mereka mewujudkan filosofi aman berdasarkan desain.

Selain tindakan yang harus diambil oleh produsen perangkat lunak, para pelanggan juga dapat memanfaatkan dokumen ini. Perusahaan yang membeli perangkat lunak harus mengajukan pertanyaan yang sulit kepada vendor mereka, dengan menggambarkan inspirasi dari contoh yang mematuhi prinsip-prinsip yang tercantum dalam dokumen ini. Dengan demikian, pelanggan dapat membantu mengalihkan pasar ke produk yang lebih aman berdasarkan desain. Salah satu contoh pertanyaan yang dapat diajukan kepada vendor diberikan dalam Panduan CISA untuk Akuisisi Teknologi K-12.

Kami mendorong pelanggan perusahaan untuk menerapkan praktik-praktik ini ke dalam proses pengadaan, penilaian uji tuntas vendor, keputusan penerimaan risiko perusahaan, dan langkah-langkah lain yang diambil ketika mengevaluasi vendor. Para pelanggan juga harus mendorong vendor mereka untuk mendokumentasikan secara publik tindakan aman berdasarkan desain yang dilakukan oleh setiap vendor. Secara kolektif, ini dapat membuat sinyal permintaan yang kuat akan keamanan, yang dapat mendorong dan memungkinkan produsen perangkat lunak untuk mengambil langkah menuju keamanan yang lebih baik. Dengan kata lain, sebagaimana kami berusaha menciptakan filosofi aman berdasarkan desain dalam produsen perangkat lunak, kami juga perlu menciptakan budaya "aman berdasarkan permintaan" dengan pelanggan mereka.

Aman berdasarkan Desain

"Secure by design" (Aman berdasarkan desain) berarti bahwa produk-produk teknologi dibangun dengan cara yang memberikan cukup perlindungan ketika pelaku kejahatan siber berhasil mendapatkan akses ke perangkat, data, dan infrastruktur yang terhubung. Produsen perangkat lunak harus melakukan penilaian risiko untuk mengidentifikasi dan memperhitungkan ancaman siber yang meluas terhadap sistem kritis, dan kemudian mencakup proteksi dalam cetak biru produk yang memperhitungkan lanskap ancaman siber yang terus berkembang.

Praktik pengembangan teknologi informasi (TI) yang aman dan pertahanan berlapis — yang dikenal sebagai pertahanan mendalam atau defense-in-depth—juga direkomendasikan untuk mencegah pelaku kejahatan menyusupi sistem atau mendapatkan akses secara tidak sah ke data yang sensitif. Organisasi-organisasi penggagas lebih jauh merekomendasikan para produsen untuk menggunakan model ancaman yang disesuaikan selama tahap pengembangan produk untuk mengantisipasi semua potensi ancaman terhadap sistem dan memperhitungkan setiap proses penerapan sistem.

Organisasi penggagas mendorong produsen untuk melakukan pendekatan keamanan holistik bagi produk dan platform mereka. Pengembangan aman berdasarkan desain memerlukan investasi strategis sumber daya khusus dari produsen-produsen perangkat lunak pada setiap lapisan proses desain dan pengembangan produk yang tidak dapat "ditambahkan" di kemudian hari. Ini memerlukan kepemimpinan yang kuat dari para eksekutif bisnis tertinggi produsen tersebut untuk menjadikan keamanan sebagai prioritas bisnis, bukan hanya fitur teknis. Kolaborasi antara para pemimpin bisnis dengan tim-tim teknis ini berlangsung sejak tahap awal desain dan pengembangan, hingga penerapan dan pemeliharaan oleh pelanggan. Produsen didorong untuk melakukan pengorbanan dan investasi yang sulit, termasuk hal-hal yang "tidak terlihat" oleh para pelanggan (misalnya, bermigrasi ke bahasa pemrograman yang menghilangkan kerentanan yang meluas). Mereka harus memprioritaskan fitur, mekanisme, dan implementasi alat yang melindungi pelanggan daripada fitur-fitur produk yang tampak menarik namun memperbesar permukaan serangan.

Tidak ada solusi tunggal untuk mengakhiri ancaman terus-menerus dari pelaku kejahatan siber yang mengeksploitasi kerentanan teknologi, dan produk-produk yang "aman-berdasarkan-desain" akan terus mengalami kerentanan; akan namun, sejumlah besar kerentanan disebabkan oleh sejumlah kecil akar penyebab. Produsen harus mengembangkan peta jalan tertulis agar selaras dengan portofolio produk mereka yang sudah ada dengan lebih banyak praktik aman berdasarkan desain, guna memastikan hanya akan menyimpang dalam situasi luar biasa saja.

Organisasi penggagas mengakui bahwa bertanggung jawab atas keamanan bagi pelanggan dan memastikan tingkat keamanan bagi pelanggan ini dapat meningkatkan biaya pengembangan. Namun, berinvestasi dalam praktik aman berdasarkan desain saat mengembangkan produk teknologi inovatif dan memelihara yang sudah ada dapat meningkatkan postur keamanan pelanggan secara substansial dan mengurangi kemungkinan penyusupan. Prinsip-prinsip aman berdasarkan desain tidak hanya menguatkan postur keamanan bagi pelanggan dan reputasi merek bagi pengembang namun praktik ini juga menurunkan pemeliharaan dan biaya patching (tambalan) bagi produsen dalam jangka waktu panjang.

Bagian Rekomendasi bagi Produsen Perangkat Lunak yang terdaftar di bawah ini memberikan daftar praktik-praktik pengembangan produk dan kebijakan untuk dipertimbangkan oleh produsen.

Aman berdasarkan Bawaan

"Secure by default" (aman berdasarkan bawaan) artinya produk-produk tahan terhadap teknik-teknik eksploitasi umum tanpa biaya tambahan. Produk-produk ini melindungi dari ancaman dan kerentanan paling umum tanpa pengguna akhir harus mengambil langkah tambahan untuk mengamankannya. Produk-produk aman berdasarkan bawaan dirancang untuk membuat pelanggan mengetahui bahwa ketika mereka menyimpang dari bawaan yang aman, mereka meningkatkan kemungkinan penyusupan kecuali jika mereka mengimplementasikan kontrol kompensasi tambahan. Aman berdasarkan bawaan adalah salah satu bentuk aman berdasarkan desain.

- Konfigurasi yang aman harus menjadi garis dasar bawaan. Produk-produk aman berdasarkan bawaan secara otomatis mengaktifkan kontrol keamanan paling penting yang dibutuhkan untuk melindungi perusahaan dari pelaku kejahatan siber, serta memasok kemampuan untuk menggunakan dan mengonfigurasikan kontrol keamanan lebih jauh tanpa biaya tambahan.
- Kekompleksan konfigurasi keamanan tidak boleh menjadi masalah pelanggan. Staf TI organisasi sering kali dibebani dengan terlalu banyak tanggung jawab keamanan dan operasional, sehingga menyebabkan terbatasnya waktu untuk memahami dan mengimplementasikan implikasi keamanan dan mitigasi yang diperlukan untuk postur keamanan siber yang kokoh. Para produsen dapat membantu pelanggan mereka melalui optimasi konfigurasi produk yang aman—mengamankan "default path" (jalur bawaan)— memastikan produk mereka diproduksi, didistribusikan, dan digunakan dengan aman sesuai dengan standar "aman berdasarkan bawaan".

Produsen produk yang "aman berdasarkan bawaan" tidak mengenakan biaya ekstra untuk mengimplementasikan konfigurasi keamanan tambahan. Sebaliknya, itu sudah termasuk dalam produk dasar seperti sabuk pengaman yang disertakan dalam semua mobil baru.

Keamanan tidak perlu menjadi opsi mewah, namun harus dianggap sebagai hak pelanggan tanpa negosiasi atau pun membayar lebih.

REKOMENDASI BAGI PRODUSEN PERANGKAT LUNAK

Panduan gabungan ini memberikan rekomendasi kepada produsen agar mengembangkan sebuah peta jalan tertulis untuk mengimplementasikan dan menjamin keamanan TI. Organisasi penggagas menyarankan produsen perangkat lunak untuk mengimplementasikan strategi yang diuraikan di bagian bawah ini untuk bertanggung jawab atas keamanan bagi pelanggan melalui prinsip-prinsip aman berdasarkan desain dan aman berdasarkan bawaan.

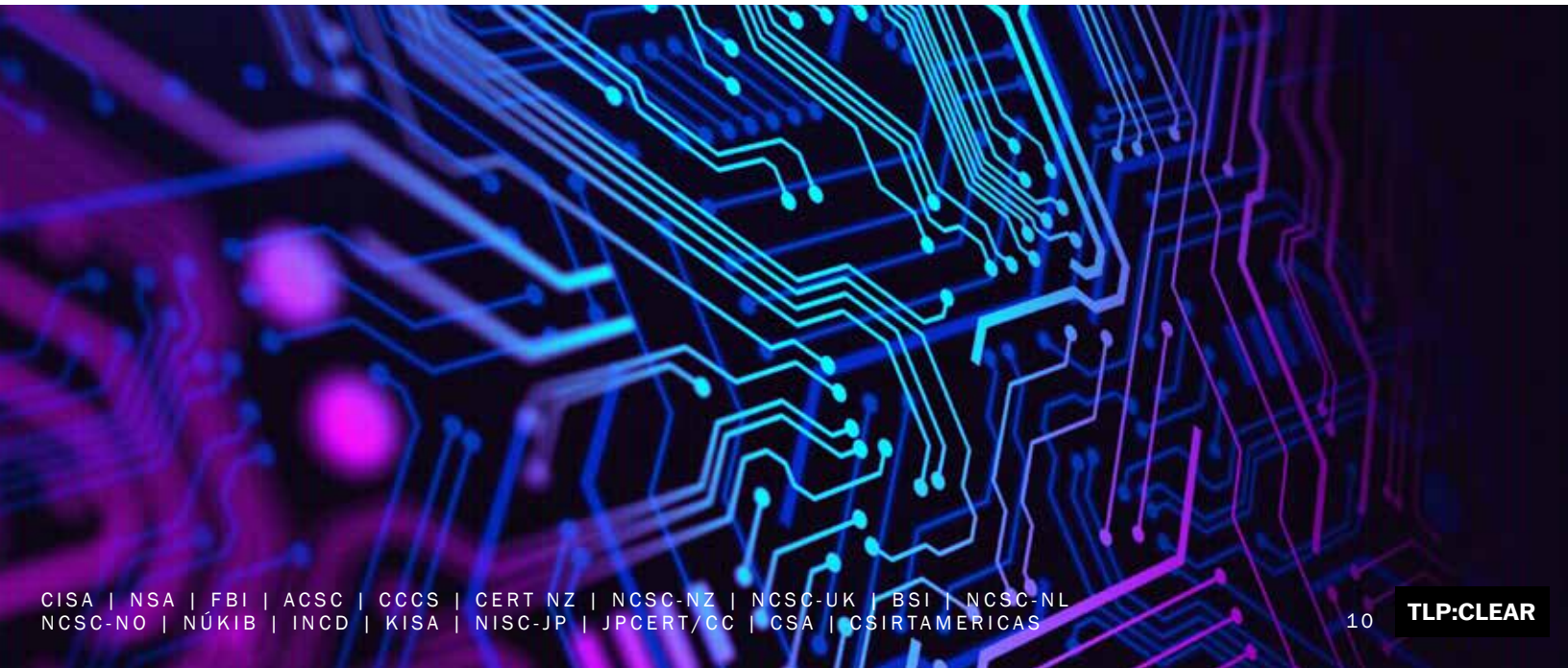
PRINSIP-PRINSIP KEAMANAN PRODUK PERANGKAT LUNAK

Produsen perangkat lunak didorong untuk mengadopsi sebuah fokus strategis yang memprioritaskan keamanan perangkat lunak. Organisasi penggagas mengembangkan tiga prinsip inti berikut untuk memandu produsen perangkat lunak dalam membangun keamanan perangkat lunak ke dalam proses desain mereka sebelum pengembangan, konfigurasi dan pengiriman produk mereka.

1. **Bertanggung jawab atas hasil keamanan pelanggan** dan mengembangkan produk berdasarkan hal tersebut. Beban keamanan tidak seharusnya ditanggung oleh pelanggan.

2. **Menerapkan transparansi dan akuntabilitas yang radikal.**
Produsen perangkat lunak harus bangga dengan diri mereka dalam memberikan produk aman dan terjamin, serta membedakan dirinya dari komunitas produsen lain berdasarkan kemampuannya. Ini termasuk berbagi informasi yang mereka dapatkan dari penerapan pelanggan, seperti penyerapan mekanisme autentikasi yang kuat secara bawaan. Ini juga bisa mencakup komitmen kuat untuk menjamin laporan kerentanan serta kerentanan dan paparan umum terkait (CVE) lengkap dan akurat. Namun, waspadalah terhadap godaan untuk menghitung CVE sebagai metrik negatif, karena angka tersebut juga merupakan tanda analisis kode dan komunitas pengujian yang sehat.

3. **Membangun struktur organisasi dan kepemimpinan untuk mencapai tujuan-tujuan ini.**
Sementara keahlian teknis kritis bagi keamanan produk, eksekutif senior merupakan pengambil keputusan utama bagi implementasi perubahan dalam suatu organisasi. Para eksekutif perlu memprioritaskan keamanan sebagai elemen kritis dalam pengembangan produk di seluruh perusahaan, dan dalam kemitraan dengan para pelanggan.



Agar dapat melaksanakan ketiga prinsip ini, produsen harus mempertimbangkan beberapa taktik operasional untuk memajukan proses pengembangan mereka.

Mengadakan rapat rutin dengan pemimpin eksekutif perusahaan untuk mendorong pentingnya aman berdasarkan desain dan aman berdasarkan bawaan dalam organisasi. Kebijakan dan prosedur harus ditetapkan untuk menghargai tim produksi yang mengembangkan produk menurut prinsip-prinsip ini, yang dapat berupa penghargaan atas mengimplementasikan praktik keamanan perangkat lunak yang luar biasa atau insentif untuk kriteria jenjang pekerjaan dan promosi.

Beroperasi berdasarkan pentingnya keamanan perangkat lunak bagi kesuksesan bisnis. Sebagai contoh, pertimbangkan menugaskan seorang "pemimpin keamanan perangkat lunak" atau "tim keamanan perangkat lunak" yang menguatkan praktik bisnis dan TI yang mengaitkan secara langsung standar keamanan perangkat lunak dan akuntabilitas produsen. Produsen harus menjamin mereka memiliki program penilaian keamanan produk yang kokoh dan independen untuk produk mereka.

Gunakan model ancaman yang disesuaikan selama alokasi dan pengembangan sumber daya untuk memprioritaskan fitur yang paling kritis dan berdampak tinggi. Model ancaman mempertimbangkan kasus penggunaan tertentu suatu produk dan memungkinkan tim pengembangan untuk melindungi produk. Yang terakhir, kepemimpinan senior harus meminta pertanggungjawaban tim dalam menghasilkan produk yang aman sebagai elemen kunci keunggulan dan kualitas produk.

Sebagai bagian dari pembaruan panduan ini pada bulan Oktober 2023, tiga prinsip ini diperluas melalui penjelasan, demonstrasi dan bukti berikut ini.

PRINSIP 1: Bertanggung Jawab Atas Keamanan Pelanggan

PENJELASAN

Praktik terbaik modern mengharuskan produsen perangkat lunak berinvestasi dalam upaya keamanan produk yang mencakup **pengerasan aplikasi, fitur aplikasi,** dan aplikasi **penataan bawaan.**

Produsen perangkat lunak perlu mengimplementasikan **pengerasan aplikasi** dengan menggunakan proses dan teknologi yang menaikkan biaya bagi pelaku kejahatan yang ingin menyusupi aplikasi. Protokol dan prosedur pengerasan aplikasi membantu produk melawan serangan oleh pelaku kejahatan yang cerdas. Istilah-istilah seperti pengerasan, keamanan produk, dan ketahanan semuanya saling berkaitan dengan kualitas produk. Gagasannya adalah keamanan harus "diintegrasikan", bukan "ditambahkan". [1] Dengan mengintegrasikan keamanan, produsen perangkat lunak bukan hanya dapat menaikkan keamanan pelanggan mereka, namun juga meningkatkan kualitas produk mereka. Contoh dari taktik mencakup memastikan masukan pengguna divalidasi dan disanitasi, dan tidak dimasukkan secara langsung ke dalam kode (yaitu, dengan menggunakan kueri berparameter sebagai gantinya), menggunakan bahasa pemrograman yang aman untuk memori, manajemen ketat siklus hidup pengembangan perangkat keras (SDLC), dan menggunakan manajemen kunci kriptografi yang didukung perangkat keras.

Aplikasi membutuhkan dukungan **fitur aplikasi** yang terhubung ke keamanan siber. Terkadang disebut "kapabilitas", fitur-fitur ini memperluas fungsionalitas produk atau layanan dengan cara yang membantu mempertahankan atau meningkatkan postur keamanan

pelanggan. Contoh fitur yang berhubungan dengan keamanan mencakup dukungan bagi keamanan lapisan transportasi (TLS) bagi semua koneksi jaringan, dukungan sistem masuk tunggal (SSO), dukungan autentikasi multi faktor (MFA), pencatatan audit peristiwa keamanan, kontrol akses berbasis peran (RBAC), dan kontrol akses berbasis atribut (ABAC).

Beberapa fitur produk ini dapat dikonfigurasi sehingga memungkinkan pelanggan dapat dengan lebih mudah mengintegrasikan produknya ke dalam lingkungan dan alur kerja mereka. Konfigurasi tersebut berarti aplikasi harus memiliki **penataan bawaan** hingga pelanggan mengonfigurasikannya. Penataan bawaan tersebut perlu diatur secara aman "out of the box" sehingga pelanggan menggunakan lebih sedikit sumber daya untuk membuat produk-produk teknologi mereka lebih aman.

Setiap elemen-elemen ini – penguat aplikasi, fitur keamanan aplikasi, dan penataan bawaan aplikasi – memegang peranan dalam keamanan aplikasi, dan postur keamanan yang dihasilkan pelanggan. Produsen perangkat lunak harus memikirkan masing-masing elemen- ini dan bagaimana mereka berhubungan Produsen sebaiknya harus memikirkan lebih dari sekadar investasi untuk memasukkan elemen-elemen ini ke dalam produk mereka. Produsen harus melangkah lebih jauh dan memperhitungkan bagaimana elemen-elemen tersebut mengubah postur keamanan dunia nyata pelanggan mereka, menjadi lebih baik atau lebih buruk.

Produsen sebaiknya bertanggung jawab atas hasil keamanan pelanggan mereka daripada mengukur diri mereka hanya berdasarkan usaha dan investasi mereka. Tanggung jawab ini harus ditempatkan di hulu, di mana produsen memiliki peluang terbesar untuk mengurangi kemungkinan gangguan.

Sayangnya, hal tersebut tidak terjadi saat ini. Terlalu banyak produsen menempatkan beban keamanan pada pelanggan daripada berinvestasi dalam **pengerasan aplikasi yang komprehensif.** Misalnya, ketika produsen menambal satu kerentanan, kami sering melihat kerentanan serupa terekspos karena mereka mengatasi gejalanya daripada akar penyebab kerusakan tersebut. Produk mungkin mengimplementasikan mitigasi yang berbeda di berbagai bagian basis kode untuk kelas kerentanan yang sama. Sebagai contoh, setelah produsen memperbaiki satu kerentanan sanitasi masukan, para peneliti atau penyerang menemukan jalur kode yang tidak mendapatkan manfaat dari peningkatan sanitasi masukan Produsen telah menerapkan perbaikan satu per satu daripada menyatukan basis kode untuk mengeliminasi kelas kerentanan tersebut di seluruh aplikasi.

Fitur aplikasi dapat menciptakan manfaat sekaligus risiko bagi pelanggan. Fitur-fitur yang memungkinkan poin integrasi dengan banyak sistem dan versi eksternal dapat sangat meningkatkan nilai suatu produk. Namun, fitur-fitur yang mendukung tanpa rencana berhenti, seperti protokol jaringan, dapat menyebabkan pelanggan rentan jika mereka kurang memahami implikasi penggunaan fitur tersebut secara terus menerus. Sebagai contoh, beberapa produk terus menggunakan protokol jaringan yang berasal dari tahun 1990-an atau 2000-an yang sekarang diketahui tidak aman. Ada banyak faktor yang dapat memperlambat pelanggan dalam meningkatkan dan menerapkan langkah-langkah keamanan modern. Mereka mungkin menggunakan produk yang terintegrasi dengan seluruh jaringan organisasi, namun tidak memiliki langkah-langkah keamanan modern, sehingga mencegah tim TI untuk memodernkannya. Namun, produsen perangkat lunak dapat memasukkan pola-pola ini ke dalam proses perencanaan mereka untuk mendorong pelanggan tetap terkini.

Penataan bawaan aplikasi adalah area tambahan yang berpotensi menimbulkan risiko bagi pelanggan. Produsen sering memilih penataan bawaan tertentu, sehingga memudahkan pelanggan menggunakan fitur aplikasi yang mereka inginkan. Sisi negatif praktik ini adalah meningkatkan permukaan serangan bagi pelanggan yang mungkin tidak membutuhkan fitur dan protokol tertentu yang diaktifkan secara bawaan. Selain itu, banyak kontrol keamanan dinonaktifkan secara bawaan atau mengharuskan pelanggan meluangkan waktu untuk mengonfigurasi penataan mereka untuk meningkatkan keamanan. Pemodelan ancaman eksplisit merupakan taktik yang dapat membantu menginformasikan keputusan fitur mana yang harus diaktifkan secara bawaan atau penataan mana yang dibutuhkan agar aman berdasarkan bawaan. Taktik lain adalah menginvestigasi cara untuk membuat fitur lebih mudah ditemukan oleh administrator.

Beberapa produsen mengirimkan produk dengan bawaan yang dapat menimbulkan risiko bagi sebagian atau seluruh pelanggan mereka. Daripada mengatur bawaan yang lebih aman, mereka

lebih sering memilih untuk membuat **panduan pengerasan** yang harus dilakukan oleh pelanggan dengan biaya sendiri. Panduan pengerasan mengalami beberapa masalah umum. Sebagian panduan pengerasan sulit didapatkan dan tidak didukung dengan baik. Sementara yang lainnya terlalu kompleks untuk diimplementasikan, seringkali mengharuskan pengembangan perangkat lunak menulis modul ekstensi. Namun, pihak lain menganggap pembaca memiliki pengalaman keamanan siber yang ekstensif untuk memahami beberapa cara penataan mengubah permukaan serangan. Para praktisi yang memiliki pemahaman kurang lengkap tentang cara kerja penyerang mungkin gagal mengimplementasikan instruksi panduan pengerasan dengan tepat, terutama jika instruksi tidak menjelaskan akibatnya dengan jelas. Selain itu, tidak semua panduan pengerasan ditulis oleh insinyur yang sangat akrab dengan taktik dan ekonomi penyerang, sehingga menyebabkan mereka membuat panduan pengerasan yang tidak efektif walaupun diimplementasikan secara tepat. Jutaan pelanggan mengambil tanggung jawab untuk mengeraskan beberapa contoh perangkat lunak atau sistem, seringkali dalam lingkungan dengan sumber daya yang terbatas. Mengandalkan panduan pengerasan saja tidak akan berhasil.

Penataan aplikasi harus terus dievaluasi apakah pengaturan itu merupakan penataan bawaan atau diatur oleh pelanggan, berdasarkan pemahaman produsen saat ini tentang lanskap ancaman. Aplikasinya harus dibuat dengan indikator yang jelas tentang potensi risiko yang dapat disebabkan oleh penataan tersebut dan indikator tersebut harus diketahui. Sama seperti mobil modern yang memiliki indikator tentang sabuk pengaman dan mengungkapkan indikator itu dengan suara peringatan jika Anda mencoba mengemudi tanpa memakainya, perangkat lunak harus mengungkapkan indikator tentang keadaan keamanan sistem. Jika satu aplikasi dikonfigurasikan tidak memerlukan MFA bagi akun administrator, sebaiknya menyadarkan administrator bahwa mereka dan seluruh organisasi mereka berada dalam bahaya jika tidak mengonfigurasikan MFA. Selain itu, jika satu aplikasi dikonfigurasi untuk mendukung protokol lawas yang sekarang diketahui mengimplementasikan kriptografi yang lemah, sebaiknya secara berkala administrator diberi kejelasan bahwa organisasi sedang dalam bahaya dan menyediakan sumber daya untuk mengatasi situasi tersebut. Kami mendesak produsen untuk mengimplementasikan dorongan rutin yang tertanam dalam produk daripada mengandalkan administrator untuk memiliki waktu, keahlian dan kesadaran untuk menafsirkan panduan pengerasan. Jelas ada peluang bagi inovasi untuk menyeimbangkan keamanan dan pertimbangan kebergunaan.

Setiap elemen di atas menciptakan situasi yang tidak dapat dipertahankan di mana pelanggan perlu meneliti, mendanai, membeli, staf, menerapkan, , dan memantau **produk keamanan** tambahan guna mengurangi peluang penyusupan. Organisasi ukuran kecil dan menengah (SMO) biasanya tidak mampu memfasilitasi opsi ini. Mereka menghadapi kekurangan keahlian, dana, dan waktu yang membebani lebar pita dan fungsi, sehingga memaksa keamanan menjadi prioritas rendah, dan secara agregat memperburuk risiko kolektif. Sebaliknya, investasi keamanan oleh beberapa produsen akan meningkat. Frasa umum yang merangkum masalah tersebut adalah industri perangkat lunak membutuhkan produk yang lebih aman, bukan lebih banyak produk keamanan. Produsen perangkat lunak harus memimpin transformasi itu.

> ***Industri perangkat lunak membutuhkan produk lebih aman, bukan lebih banyak produk keamanan. Produsen perangkat lunak harus memimpin transformasi itu.***

Saat ini, kami kadang-kadang membaca komentar dari produsen yang menjelaskan bahwa pelanggan telah tersusupi karena tidak mengaktifkan fitur keamanan tertentu atau mengikuti panduan pengerasan khusus. Alih-alih, setelah penyusupan, produsen harus menjelaskan apakah fitur keamanan tertentu atau panduan pengerasan khusus akan dapat mencegah penyusupan tersebut dan mempertimbangkan untuk membuatnya sebagai bawaan tanpa biaya. Dalam kasus di mana produk itu sendiri tidak cukup kuat dalam fase desain maupun implementasi, maka produsen harus menjelaskan bagaimana mereka bekerja untuk mengeliminasi kelas kerentanan tersebut dari lini produk mereka.

Produsen perangkat lunak bertanggung jawab untuk menjamin produk-produk mereka dirancang dan dikembangkan dengan keamanan sebagai prioritas utama. Untuk tujuan itu, mereka seharusnya **secara objektif mengukur hasil** dari upaya mereka di lapangan. Kami mengimbau produsen agar tidak hanya fokus pada upaya internal mereka, namun juga mengukur secara objektif dan secara teratur melaporkan hasil dan keefektifan upaya dan konfigurasi keamanan produk, serta membangun umpan balik yang menciptakan perubahan dalam SDLC yang mengarah pada peningkatan dalam keamanan pelanggan yang dapat diukur dan produk yang lebih aman. Pelaporan harus mencakup data anonim yang dapat digunakan oleh komunitas akademi dan peneliti keamanan untuk melacak tren tingkat tinggi dan mengukur kemajuan ekosistem secara luas.

MENDEMONSTRASIKAN PRINSIP INI

Produsen perangkat lunak dan layanan online harus mencari jalan untuk mendemonstrasikan keberhasilan dalam mengimplementasikan prinsip ini. Mereka harus berusaha memberikan bukti dalam bentuk artefak untuk diperiksa oleh pihak luar. Tidak ada satupun artefak yang dapat membuktikan bahwa produsen mengimplementasikan program aman berdasarkan desain yang kokoh, namun dengan menyediakan beragam artefak mereka akan membuktikan komitmen produsen dalam mengembangkan produk yang aman. Pendekatan ini memiliki semangat "menunjukkan, bukan memberi tahu".

Untuk mendemonstrasikan prinsip ini, produsen perangkat lunak harus mempertimbangkan langkah-langkah yang ada dalam daftar berikut. Organisasi penggagas mengakui bahwa produsen perangkat lunak akan dapat segera mengimplementasikan praktik-praktik ini dan memproduksi artefak yang sesuai pada awal perjalanan aman berdasarkan desain mereka. Selanjutnya, produsen perangkat lunak perlu memprioritaskan daftar ini tergantung pada bagaimana pelanggan menerapkan produk di lapangan untuk mencapai manfaat keamanan terbesar.

PRAKTIK-PRAKTIK AMAN BERDASARKAN BAWAAN

1. **Eliminasi kata sandi bawaan:** Kata sandi bawaan terus diimplikasikan sebagai penyebab banyak serangan setiap tahunnya. Berkomitmen untuk mengeliminasi masalah kronis ini akan menolak akses yang mudah bagi penyerang. Demikian pula, produsen harus memperhitungkan praktik kata sandi apa harus diimplementasikan, seperti minimum panjang kata sandi dan melarang penggunaan kata sandi yang pernah diterobos.
2. **Lakukan uji lapangan.** Seiring dengan teknologi yang terus berkembang dan menjadi lebih kompleks, maka semakin penting bagi produsen perangkat lunak untuk melakukan pengujian pengguna yang berfokus pada keamanan untuk memahami postur keamanan produk mereka di lapangan. Serupa dengan cara penelitian pengguna menginformasikan persyaratan pengembangan perangkat lunak, produsen perangkat lunak juga harus melakukan penelitian pengguna yang berfokus pada keamanan untuk memahami di mana letak kelemahan pengalaman pengguna keamanan (UX). Dengan mengamati cara perangkat lunak pelanggan menerapkan dan menggunakan produk mereka di lingkungan dunia nyata, produsen perangkat lunak dapat memperoleh wawasan yang berharga mengenai kegunaan dan keefektifan fitur dan kontrol keamanan mereka. Wawasan ini dapat membantu mengidentifikasi area-area yang perlu ditingkatkan dan memperbaiki produk mereka agar dapat lebih memenuhi kebutuhan keamanan pelanggan. Sebagai contoh, uji lapangan menyarankan perubahan dalam aliran, bawaan, peringatan dan pemantauan UX. Uji lapangan juga dapat menunjukkan apabila peningkatan terdahulu dalam desain produk mengurangi kecepatan tambalan keamanan, mengurangi kesalahan konfigurasi, dan meminimalkan permukaan serangan.

Produsen sebaiknya mempertimbangkan hal berikut:

- Apakah pelanggan mengimplementasikan panduan pengerasan dengan benar?
- Apakah fitur keamanan produk yang ada berfungsi sesuai harapan di lapangan?
- Apakah fitur-fitur tersebut benar-benar melawan serangan dunia nyata?
- Fitur-fitur manakah yang akan mengurangi kemungkinan penyusupan dengan lebih baik?

Catatan: Untuk mendapatkan wawasan yang lebih dalam mengenai elemen-elemen ini, produsen perangkat lunak mungkin ingin bermitra dengan pelanggan untuk melakukan latihan tim merah guna melihat bagaimana cara produk melawan serangan. Uji lapangan ini dapat dilaksanakan di lokasi fisik pelanggan, secara virtual, atau via telemetri dari aplikasi itu dengan tetap mempertahankan privasi.

3. **Perkecil ukuran "panduan pengerasan" (hardening guide).** Produsen dapat meningkatkan postur keamanan pelanggan melalui perampingan atau bahkan menghilangkan panduan pengerasan produk dan berfokus pada tindakan keamanan yang paling kritis yang harus diprioritaskan oleh pelanggan ketika menerapkan produk mereka. Daripada membanjiri pelanggan dengan serangkaian tindakan keamanan, lebih baik produsen mengidentifikasi risiko keamanan utama yang paling rentan bagi produk mereka dan memberikan panduan yang jelas dan ringkas mengenai cara memitigasi risiko-risiko ini. Selain itu, produsen harus menyediakan alat dan automasi kepada pelanggan yang menyederhanakan proses penerapan kontrol keamanan, seperti skrip yang dapat dengan mudah diterapkan di dalam lingkungan mereka. Alat-alat ini juga harus dapat memverifikasi dan dengan jelas menunjukkan perubahan yang dibuat dari garis dasar aslinya. Dengan merampingkan panduan pengerasan dan menyediakan alat dan

automasi yang mudah digunakan, produsen dapat mengurangi beban pelanggan dan membantu memastikan produk mereka diterapkan dengan cara yang aman. Salah satu taktiknya adalah untuk mempertimbangkan implementasi prinsip Pareto untuk mengurangi jumlah langkah untuk kasus pengguna yang umum (80%), dan kemudian memberikan panduan kontekstual serta alat untuk skenario yang kurang umum (20%). Dengan cara ini, produsen perangkat lunak akan membuat hal-hal sederhana tetap sederhana, dan hal-hal sulit menjadi mungkin. Uji lapangan akan menjadi alat yang kuat dalam mengukur berapa lama pelanggan dapat menemukan, memahami dan mengimplementasikan panduan pengerasan. Produsen harus mempertimbangkan bagaimana produk dapat mendorong administrator agar mengambil tindakan dalam produk itu sendiri daripada mengandalkan mereka untuk mengimplementasikan tugas dari panduan pengerasan.

4. **Secara aktif mencegah penggunaan fitur-fitur versi lama yang tidak aman.** Prioritaskan keamanan melalui alur peningkatan yang jelas daripada dengan kompatibilitas mundur. Terbitkan tulisan blog yang menunjukkan penggunaan fitur dan protokol yang aman, serta berhenti menggunakan fitur tidak aman melalui pengumuman, mungkin dalam produk itu sendiri. Sejumlah besar pelanggan telah mendemonstrasikan bahwa mereka tidak akan terus memperbarui sistem mereka dengan jaringan, identitas, dan fitur keamanan kritis lainnya yang modern. Dalam beberapa kasus, pelanggan khawatir bila fungsionalitas yang ada akan terhenti karena peningkatan. Dengan membuat pembaruan tanpa hambatan sebisa mungkin, pelanggan kemungkinan besar akan meningkatkan dan mendapatkan perbaikan keamanan lebih sering dan lebih cepat. Produsen perangkat lunak sebaiknya mendorong pelanggan secara agresif bersama alur peningkatan yang mengurangi risiko pelanggan.
5. **Mengimplementasikan peringatan yang menarik perhatian.** Mirip dengan suara sabuk pengaman pada mobil yang berbunyi terus-menerus jika sabuk pengaman tidak dipasang, produsen harus mengimplementasikan peringatan yang tepat waktu dan berulang ketika pengguna atau admin berada dalam keadaan yang tidak aman, untuk memperingatkan administrator bahwa mereka menggunakan protokol yang usang dalam lingkungan mereka dan menyarankan alur peningkatan. Mengimplementasikan peringatan secara tepat waktu dan berulang ketika pengguna atau admin, atau konfigurasi aplikasi, berada dalam keadaan tidak aman. Beritahukan mode tidak aman kepada administrator secara teratur. Fitur tambahan dapat mengharuskan super administrator untuk mengakui kurangnya MFA pada akun mereka setiap kali login, atau bahkan menonaktifkan fitur utama tertentu hingga mereka mengaktifkan MFA. Ada ruang untuk menginovasi guna mencapai tujuan ini tanpa menimbulkan peringatan kelelahan.
6. **Buat templat konfigurasi yang aman.** Templat ini dapat mengatur mengatur konfigurasi tertentu ke penataan aman berdasarkan selera risiko organisasi. Meskipun mungkin terlalu sederhana untuk memiliki templat keamanan yang rendah/menengah/tinggi, contoh tersebut menggambarkan seberapa banyak penataan dapat diperbarui untuk mengelola risiko bagi organisasi. Templat dapat didukung dengan panduan pengerasan tentang risiko yang telah diidentifikasi oleh produsen.

PRAKTIK-PRAKTIK PENGEMBANGAN PRODUK YANG AMAN

1. **Dokumentasikan kesesuaian dengan kerangka kerja SDLC yang aman.** Kerangka kerja SDLC yang aman memberikan sasaran dan contoh pada orang-orang, proses dan teknologi. Pertimbangkan untuk menerbitkan keterangan terperinci tentang kontrol kerangka kerja SDLC aman mana yang telah diimplementasikan dan jelaskan kontrol alternatif lain yang telah digunakan. Di negara AS, pertimbangan untuk menggunakan Kerangka Kerja Pengembangan Perangkat Lunak Yang Aman NIST (SSDF) Meskipun bukan sebuah daftar periksa, SSDF "menjelaskan serangkaian praktik fundamental, dan bagus untuk pengembangan perangkat lunak yang aman."
2. **Dokumentasikan Tujuan Kinerja Keamanan Siber (CPG) atau konformitas yang setara.** Ketika satu organisasi menyatakan bahwa mereka mematuhi standar SSDF NIST, mereka menegaskan bahwa SDLC mereka didasari oleh praktik terbaik yang dipahami dengan baik. Namun, bagi mereka memiliki SDLC yang tangguh masih kurang mencukupi. Mereka juga harus melindungi perusahaan mereka dan lingkungan pengembangan mereka dari para pelaku kejahatan yang akan berusaha memanipulasi properti keamanan produk saat masih dalam pengembangan. Ini bukan kelas serangan teoretis, melainkan serangan yang dilaksanakan dengan efek buruk kepada pelanggan, dan secara luas kepada keamanan nasional. Organisasi harus mempertimbangkan untuk menerbitkan detail mengenai kepatuhan organisasi terhadap CISA CPG, Kerangka Kerja Keamanan Siber (CSF) NIST, atau kerangka kerja program keamanan siber lainnya.
3. **Manajemen kerentanan.** Beberapa produsen memiliki program manajemen kerentanan yang berfokus kepada penambalan kerentanan yang ditemukan secara internal atau eksternal, dan banyak lagi. Lebih banyak program-program yang lebih matang memasukkan analisis kerentanan berbasis data dan akar penyebabnya, mengambil langkah-langkah untuk mengeliminasi seluruh kelas kerentanan secara sistemik[3]. Mereka mengimplementasikan program-program formal seputar penataan perencanaan kualitas, pengendalian kualitas, peningkatan kualitas, dan pengukuran kualitas. Mereka memandang manajemen yang cacat sebagai masalah bisnis, bukan sekadar masalah keamanan. Program-program ini tidak berbeda dalam beberapa hal dengan program kualitas dan keamanan dalam industri lain.
4. **Gunakan perangkat lunak sumber terbuka secara bertanggung jawab.** Ketika perangkat lunak sumber terbuka digunakan, bertanggungjawablah dengan memeriksa paket sumber terbuka, menumbuhkan kontribusi kode kembali ke ketergantungan, dan membantu mempertahankan pengembangan dan pemeliharaan komponen-komponen kritis. Sebagai referensi, Kementerian Ekonomi, Perdagangan, dan Industri Jepang (METI) telah menerbitkan "Koleksi Contoh Kasus Penggunaan Mengenai Metode Manajemen untuk Memanfaatkan OSS dan Memastikan Keamanannya."
5. **Berikan bawaan yang aman bagi para pengembang.** Jadikan rute bawaan selama pengembangan perangkat lunak rute yang aman dengan menyediakan elemen pembangunan yang aman bagi para pengembang. Sebagai contoh, mengingat kelaziman kerentanan injeksi SQL yang menyebabkan bahaya dunia nyata, pastikan bahwa pengembang menggunakan perpustakaan yang terawat dengan baik guna mencegah kelas kerentanan tersebut. Juga dikenal dengan "jalan beraspal" atau "jalan setapak berpenerangan baik", praktik ini menjamin baik kecepatan maupun keamanan, dan mengurangi kesalahan manusia.

[3] NIST SSDF, PO 1.2, Contoh 2: "Tentukan kebijakan yang menentukan persyaratan keamanan bagi organisasi perangkat lunak, dan verifikasikan kepatuhan pada poin-poin kunci dalam SDLC (misalnya, kelas-kelas kelemahan perangkat lunak diverifikasi oleh gerbang, respons terhadap kerentanan yang ditemukan dalam perangkat lunak yang dirilis)."

6. **Menumbuhkan tenaga kerja pengembang perangkat lunak yang memahami keamanan.** Pastikan bahwa pengembang perangkat lunak Anda memahami keamanan dengan melatih mereka mengenai praktik-praktik terbaik pengodean yang aman. Lebih jauh lagi, membantu mentransformasi tenaga kerja lebih luas dengan memperbarui praktik-praktik perekrutan untuk mengevaluasi pengetahuan keamanan dan bekerja sama dengan universitas, perguruan tinggi komunitas, kamp pelatihan, dan edukator lainnya untuk menggabungkan keamanan ke dalam kurikulum ilmu sains komputer dan pengembangan perangkat lunak.
7. **Uji manajemen peristiwa insiden keamanan (SIEM) dan intregasi orkestrasi, automasi, dan respons keamanan (SOAR).** Selain melaksanakan uji lapangan, bekerja sama dengan penyedia SIEM dan SOAR yang populer bersamaan dengan pelanggan terpilih untuk memahami bagaimana tim respons insiden menggunakan log untuk menginvestigasi insiden keamanan yang dicurigai atau aktual. Hanya sedikit pengembang perangkat lunak yang memiliki pengalaman merespons insiden dan mungkin membuat entri log yang tidak membantu responden sebaik yang mereka harapkan. Dengan bekerja dengan teknologi SIEM dan SOAR dan profesional respons insiden nyata, tim pengembangan dapat membuat log yang memberi tahu cerita benar dan lengkap, menghemat waktu dan mengurangi ketidakpastian selama ada insiden.
8. **Sejalan dengan Zero Trust Architecture (ZTA).** Selaraskan panduan pemakaian produk dengan, misalnya, model ZTA NIST dan Zero Trust Maturity Model CISA. Dorong pelanggan untuk menerapkan prinsip-prinsip ini ke dalam lingkungan mereka.

PRAKTIK-PRAKTIK BISNIS PRO-KEAMANAN

1. **Sediakan log tanpa biaya tambahan.** Layanan cloud harus berkomitmen untuk menghasilkan dan menyimpan pencatatan terkait keamanan tanpa dikenai biaya tambahan. Produk-produk lokal juga harus menghasilkan log terkait keamanan tanpa dikenakan biaya tambahan. Selanjutnya, produk harus mencatat peristiwa keamanan secara bawaan karena banyak pelanggan mungkin tidak memahami nilainya hingga setelah insiden tersebut terjadi. Taktik ini memerlukan tinjauan yang menyeluruh terhadap peristiwa keamanan yang seharusnya dicatat untuk memberikan kesadaran status keamanan siber, cara pelanggan mengonfigurasikan pencatatan, berapa lama catatan itu disimpan, cara melindungi integritas dan penyimpanan , dan cara menganalisis catatan itu. Dalam beberapa kasus, tinjauannya mungkin menyarankan perlunya perhitungan ulang arsitektur manajemen log aplikasi agar dapat ditindaklanjuti dan biaya yang dapat diterima oleh produsen. Bekerja sama dengan pakar respons insiden (IR) dapat meningkatkan peluang bahwa log akan berguna untuk investigator di lapangan. Lihat bagian mengenai SIEM.
2. **Eliminasi pajak tersembunyi.** Terbitkan komitmen tidak pernah mengenakan biaya untuk fitur keamanan atau privasi atau integrasi. Sebagai contoh, dalam ruang lingkup identitas dan manajemen akses (IAM) yang lebih luas, terdapat layanan yang disebut sistem masuk tunggal (SSO). Beberapa produsen mengenakan biaya lebih banyak untuk menghubungkan sistem mereka ke layanan SSO (kadang-kadang disebut sebagai penyedia identitas). "Pajak SSO" ini berarti bahwa manajemen identitas dan akses yang baik di luar jangkauan banyak SMO, sehingga menghambat mereka untuk mencapai postur keamanan yang kuat. Beberapa layanan mengenakan biaya lebih banyak untuk mengaktifkan MFA bagi para pengguna. **Keamanan seharusnya tidak boleh dihargai sebagai barang mewah namun harus dianggap sebagai hak pelanggan.** Beberapa produsen berargumentasi bahwa hanya sedikit pelanggan yang meminta fitur ini, dan biaya pemeliharaannya lebih mahal. Argumen-argumen ini mengabaikan fakta bahwa hanya sedikit pelanggan yang akan menelepon untuk mengeluh atau menawar, tidak semua pelanggan benar- benar memahami apa manfaat fitur-fitur ini, dan bahwa semua fitur memerlukan biaya pemeliharaan. Namun, kami tidak melihat banyak produsen mengenakan biaya ekstra untuk ketersediaan atau integritas data. Biaya untuk mendukung atribut utama tersebut dimasukkan ke dalam harga yang dibayar oleh semua pelanggan, sama seperti biaya untuk menyertakan sabuk pengaman, kolom kemudi lipat, dan kantong udara yang menyelamatkan nyawa dalam kecelakaan.
3. **Terapkan standar terbuka.** Implementasikan standar terbuka, terutama seputar jaringan umum dan protokol identitas. Hindari protokol kepemilikan ketika ada standar terbuka.
4. **Sediakan alat peningkatan (upgrade tooling).** Banyak pelanggan enggan mengadopsi versi produk terbaru, termasuk memakai fitur yang lebih baru atau lebih aman seperti koneksi jaringan aman. Produsen perangkat lunak dapat menaikkan adopsi pelanggan terhadap pembaruan terkini dengan memberikan alat untuk membantu mengurangi ketidakpastian dan risiko. Tawarkan lisensi gratis kepada pelanggan untuk menguji peningkatan dan tambalan dalam lingkungan uji sebagai cara untuk memotivasi pelanggan.

PRINSIP 2: Menerapkan Transparansi dan Akuntabilitas yang Radikal

PENJELASAN

Produsen perangkat lunak harus bangga dalam memberikan produk yang aman dan terjamin, serta membedakan diri mereka dari komunitas produsen lainnya berdasarkan kemampuan mereka.

Mari kita atasi kekhawatiran umum tentang transparansi. Ketika para praktisi mendiskusikan transparansi radikal, ada tendensi percakapan itu terhambat karena kekhawatir mereka memberikan "peta jalan kepada para penyerang". Namun, banyak bukti yang menunjukkan bahwa para penyerang akan baik-baik saja tanpa peta jalan semacam itu, dan kekhawatiran tersebut harus dikesampingkan dibandingkan dengan transparansi yang menguntungkan pelanggan langsung, pelanggan tak langsung, rantai pasokan, dan seluruh industri perangkat lunak.

Transparansi membantu industri ini menetapkan konvensi—dengan kata lain, seperti apakah "baik" itu. Hal ini membantu konvensi tersebut berubah seiring dengan waktu sebagai respons terhadap kebutuhan pelanggan, perubahan taktik atau ekonomi pelaku ancaman, atau evolusi teknologi. Transparansi membantu produsen dengan sumber daya yang lebih sedikit untuk belajar dari mereka yang memiliki sumber daya lebih matang dan mampu. Pembicaraan tentang berbagi informasi harus diperluas melampaui indikator ancaman waktu nyata, untuk mencakup elemen-elemen di bawah ini.

Transparansi memaksa dibuatnya keputusan seputar keamanan pada awal proses pengembangan, dan agar menjadi aktivitas berkelanjutan para pemimpin bisnis serta insinyur dan profesional bidang keamanan. Transparansi membangun akuntabilitas ke dalam produk.

Catatan tentang pemilihan kata "radikal" di depan "transparansi". Kini, produsen perangkat lunak jarang menerbitkan informasi terperinci tentang bagaimana mereka mengembangkan dan memelihara perangkat lunak dan cara mereka mematangkan program mereka menggunakan data dari waktu ke waktu. Dalam industri perangkat lunak, hanya sedikit produsen yang menawarkan tur berpemandu tentang bagaimana mereka mendesain perangkat lunaknya. Ada sedikit peluang bagi produsen perangkat lunak untuk melihat bagaimana organisasi sejawat menyusun program SDLC mereka, dan bagaimana program-program itu bertahan di lingkungan pelanggan melawan penyerang nyata. Industri kolektif akan mendapatkan manfaat dari semakin banyaknya pertukaran informasi mengenai topik-topik seperti strategi untuk mengukur dampak kerusakan keamanan dan untuk mengeliminasi kelas-kelas kerentanan. Sebagai hasil dari praktik umum ini, setiap produsen perangkat lunak harus belajar sendiri cara menghadapi sendiri keamanan produk. Mungkin dengan tidak mengenakan pajak barang mewah pada fitur keamanan, dengan demikian keselamatan dan keamanan akan menjadi pusat biaya daripada pusat keuntungan, dan perusahaan akan mendapatkan keuntungan dengan memperingan beban melalui kolaborasi dan transparansi.

Kami ingin fokus pada taktik yang secara nyata akan mempercepat evolusi industri perangkat lunak. Kami tidak dapat lagi membiayai peningkatan oportunis dan bertahap. Jika kita akan mengatasi ancaman yang ditimbulkan oleh musuh cerdas dan adaptif bersama-sama, kita harus transparan hingga ke tingkat yang terasa tidak nyaman saat ini, namun dapat mendorong industri untuk maju. Saat ini, ada produsen yang mewujudkan beberapa prinsip aman berdasarkan desain ini. Sebagaimana yang dikatakan oleh William Gibson, "masa depan sudah di sini, hanya belum terdistribusi dengan merata." **Transparansi radikal akan membantu mendistribusikan informasi tersebut dan lebih menguntungkan para pembela dibandingkan dengan musuh kita.**

Transparansi dapat memberikan manfaat lebih dari sekadar membantu organisasi sejawat untuk menyempurnakan SDLC mereka. Calon pelanggan dan investor dapat mempelajari lebih lanjut tentang investasi dan pengorbanan yang sudah dibuat oleh produsen, serta postur keamanan yang tercipta dari investasi tersebut bagi pelanggan. Produsen yang menerapkan transparansi radikal akan memberikan informasi kepada pelanggan untuk membantu mereka membuat keputusan belanja bukan hanya tentang harga dan fitur, namun juga tentang keamanan.

Sekeras apa pun organisasi bekerja untuk mengamankan rantai pasokan dan SDLC mereka, namun proses pembangunan perusahaan telah disusupi di masa lalu. Menerapkan transparansi radikal harus mengarah pada pengungkapan serangan tersebut kepada publik serta peningkatan yang dilakukan perusahaan untuk mencegah dan mendeteksi serangan masa depan. Bentuk berbagi informasi semacam itu akan membantu organisasi-organisasi lain belajar tanpa harus menderita nasib yang sama.

MENDEMONSTRASIKAN PRINSIP INI

Untuk mendemonstrasikan prinsip ini, produsen perangkat lunak harus mengikuti langkah-langkah berikut ini:

PRAKTIK-PRAKTIK AMAN BERDASARKAN BAWAAN

1. **Terbitkan statistik dan tren keamanan agregat yang relevan.** Contoh topik termasuk penerapan MFA oleh pelanggan dan administrator dan penggunaan protokol lama yang tidak aman.
2. **Terbitkan statistik tambalan (patching).** Perinci berapa persen pelanggan yang menggunakan produk versi terbaru, dan apa yang Anda lakukan agar pembaruan lebih mudah dan andal.
3. **Terbitkan data tentang hak istimewa yang tidak terpakai.** Terbitkan informasi agregat mengenai izin eksesif di seluruh basis pelanggan Anda serta dorongan dan perubahan lain pada produk yang Anda buat untuk mengurangi permukaan serangan pelanggan. Hak istimewa yang tidak terpakai ini mungkin menjadi kandidat bagus untuk peringatan administrator, seperti suara pengingat sabuk pengaman.

PRAKTIK-PRAKTIK PENGEMBANGAN PRODUK AMAN

1. **Menetapkan kontrol keamanan internal.** Banyak perusahaan telah melihat manfaat memindahkan data mereka ke penyedia cloud. Sekarang penyedia cloud menjadi target para penyerang. Penyedia Perangkat Lunak sebagai Layanan (SaaS) harus menerbitkan statistik kontrol internal mereka. Sebagai contoh, penyedia SaaS Mempublikasikan statistik mengenai penerapan autentikasi MFA anti phishing, seperti Fast Identity Online (FIDO) secara internal. Idealnya, mereka harus dapat mengatakan bahwa tidak ada anggota staf yang dapat mengakses data pelanggan atau data sensitif lainnya tanpa autentikasi melalui MFA anti phishing.
2. **Terbitkan model ancaman tingkat tinggi.** Produk aman berdasarkan desain dimulai dengan model ancaman tertulis yang menggambarkan apa dan dari siapa yang ingin dilindungi oleh para kreator. Model ancaman yang efektif diinformasikan dengan melihat cara penyusupan itu terjadi di alam bebas, dan ini harus mencakup baik lingkungan perusahaan maupun pengembangan, dan juga bagaimana produsen perangkat lunak ingin menggunakannya di lingkungan pelanggan.
3. **Terbitkan pembuktian diri SDLC aman yang terperinci.** Produsen yang mengikuti NIST SSDF, atau kerangka kerja serupa lainnya yang secara aktif berusaha menuju siklus hidup pengembangan perangkat lunak yang matang. Menerbitkan pembuktian diri mengenai kontrol mana yang telah diterapkan oleh produsen, dan untuk produk apa, akan mendemonstrasikan komitmen mematuhi praktik-praktik terbaik ini dan memberikan tingkat kepercayaan yang lebih tinggi kepada pelanggannya. Skema sertifikasi lainnya mencakup Metodologi Rantai Pasokan Siber Israel, misalnya.
4. **Terapkan transparansi kerentanan.** Terbitkan komitmen yang akan memastikan bahwa kerentanan produk yang teridentifikasi akan diterbitkan sebagai entri CVE yang benar dan lengkap. Ini benar terutama berlaku di bidang enumerasi kelemahan umum yang mengidentifikasi akar penyebab utama kerentanan. Semakin benar dan lengkap basis data CVE publik, maka semakin besar pula kemampuan industri melacak bagaimana produk menjadi lebih aman, dan kelas-kelas kerentanan mana yang paling umum. Namun, waspadalah terhadap godaan untuk menghitung CVE sebagai metrik negatif, karena angka-angka tersebut juga merupakan tanda komunitas analisis dan pengujian kode yang sehat. Jika produsen mengimplementasikan filosofi aman berdasarkan desain, mungkin saja pada awalnya jumlah CVE mentah mereka akan meningkat karena penemuan yang lebih komprehensif dan remediasi kerentanan dalam kode yang ada. Produsen harus menerbitkan analisis kerentanan masa lalu, termasuk pola dan tindakan-tindakan yang diambil untuk mengatasi seluruh kelas kerentanan. Sebagai contoh, jika persentase besar CVE perusahaan berhubungan dengan pembuatan skrip lintas situs (XSS), mendokumentasikan analisis akar penyebab, respons (seperti berganti ke kerangka kerja templat web yang mencegah XSS), dan hasilnya akan memberi sinyal kepada pelanggan bahwa mereka tidak akan menjadi korban dari suatu kelas kerentanan yang mitigasinya telah dipahami selama puluhan tahun.
5. **Terbitkan Daftar Bahan-Bahan Perangkat Lunak (SBOM).** Produsen harus memiliki kendali atas rantai pasokan mereka. Organisasi-organisasi harus membangun dan menjaga SBOM [2] bagi setiap produk, meminta data dari pemasok mereka, dan menyediakan SBOM bagi pelanggan dan

pengguna hilir. Hal ini akan membantu mendemonstrasikan ketekunan mereka dalam memahami komponen yang mereka gunakan dalam pembuatan produk mereka, kemampuan mereka merespons risiko yang baru teridentifikasi, dan dapat membantu pelanggan memahami cara merespons jika salah satu modul dalam rantai pasokan memiliki kerentanan baru. Sebagai referensi, Kementerian Ekonomi, Perdagangan, dan Industri Jepang (METI) telah menerbitkan "Panduan Pendahuluan Tentang Daftar Bahan-Bahan Perangkat lunak (SBOM) untuk Manajemen Perangkat Lunak." Transparansi harus mencakup firmware dalam perangkat yang disematkan dan data serta model yang digunakan dalam AI/pemelajaran mesin (ML). Selain sekadar membantu dalam keputusan pembelian dan kapabilitas operasional, SBOM berperan penting dalam infrastruktur untuk mendeteksi dan merespons serangan berbahaya terhadap rantai pasokan.

6. **Terbitkan kebijakan pengungkapan kerentanan.** Terbitkan kebijakan pengungkapan kerentanan yang (1) memberi wewenang pengujian terhadap semua produk yang ditawarkan oleh produsen dan persyaratan untuk pengujian tersebut, (2) menyediakan tempat aman secara hukum agar tindakan yang dilakukan konsisten dengan kebijakan tersebut, dan (3) memungkinkan pengungkapan kepada publik tentang kerentanan setelah linimasa yang ditentukan. Produsen harus melakukan analisis akar penyebab kerentanan yang ditemukan dan bertindak untuk menghilangkan seluruh kelas kerentanan semaksimal mungkin. Bacalah Templat Kebijakan Pengungkapan Kerentanan CISA untuk bahasa referensi.

PRAKTIK-PRAKTIK BISNIS PRO-KEAMANAN

1. **Sebutkan secara terbuka sponsor eksekutif senior aman berdasarkan desain.** Dalam banyak organisasi, keamanan (seperti kualitas) didelegasikan kepada tim teknis yang memiliki kemampuan terbatas untuk membuat perubahan struktural untuk memperbaiki keamanan produk secara dramatis. Sebutkan secara terbuka eksekutif bisnis terkemuka untuk mengawasi program aman berdasarkan desain akan mentransformasikan keamanan produk menjadi perhatian bisnis tingkat atas.
2. **Terbitkan peta jalan aman berdasarkan Desain** Produsen harus mendokumentasikan perubahan yang dibuat pada SDLC mereka untuk meningkatkan keamanan pelanggan, termasuk detail tentang laporan uji lapangan, tindakan yang diambil untuk mengeliminasi seluruh kelas kerentanan, dan hal lain yang tercantum dalam prinsip lainnya. Sebagaimana dalam usaha peningkatan kualitas, program-program peningkatan keamanan memiliki fase perencanaan, kontrol dan peningkatan yang berbeda. Dalam semangat untuk menunjukkan bukan menggurui, menerbitkan peta jalan dan detail di balik fase-fase ini akan membangun keyakinan bahwa produk tersebut aman berdasarkan desain. Setelah mencapai kemajuan yang berarti, produsen dapat merincinya dalam laporan transparansi. Dengan begitu tidak hanya mendemonstrasikan komitmen terhadap prinsip aman berdasarkan desain namun dapat menginspirasi pihak lain untuk mengadopsi program serupa dengan menunjukkan bukti yang ada.
3. **Terbitkan peta jalan keamanan memori.** Produsen dapat mengambil langkah-langkah untuk menghilangkan kelas kerentanan terbesar dengan memigrasikan produk yang ada dan membangun produk baru dengan menggunakan bahasa yang aman memori. Sementara hal ini tidak memungkinkan di semua kasus, para produsen dapat mempertimbangkan untuk mengembangkan pembungkus aplikasi dalam bahasa yang aman bagi memori daripada menulis ulang seluruh aplikasi. Ini dapat mencakup bagaimana produsen memperbarui perekrutan, pelatihan, peninjauan kode, dan proses internal lainnya, serta cara mereka membantu komunitas sumber terbuka untuk melakukan hal sama.
4. **Terbitkan hasil.** Saat memperbarui SDLC mereka untuk mewujudkan filosofi aman berdasarkan desain, organisasi-organisasi akan menemukan keuntungan cepat, keuntungan yang lebih intensif sumber daya, dan beberapa kemunduran yang tidak diharapkan. Dengan memperlihatkan kesuksesan dan hambatan internalnya, seluruh industri dapat belajar dari hasilnya.

PRINSIP 3: Memimpin dari Atas

PENJELASAN

Sementara filosofi keseluruhan disebut "aman berdasarkan desain", insentif bagi keamanan pelanggan dimulai jauh sebelum fase desain produk. Mereka mulai dengan tujuan bisnis dan tujuan implisit dan eksplisit dan hasil yang diharapkan. Hanya ketika pemimpin senior menjadikan keamanan prioritas bisnis, membuat insentif internal, dan memupuk budaya di seluruh jajaran untuk menjadikan keamanan sebagai persyaratan desain, barulah mereka akan mencapai hasil terbaik.

Meskipun keahlian bidang teknis bersifat kritis bagi keamanan produk, namun itu bukan hal yang dapat semata-mata diserahkan kepada staf teknis. Ini adalah prioritas bisnis yang harus dimulai dari atas.

Sebagian orang telah bertanya-tanya jika produsen perangkat lunak menerapkan dua prinsip pertama dan memproduksi artefak yang bermakna, apakah prinsip ketiga diperlukan? Bagaimana sebuah perusahaan menetapkan visi, misi, nilai-nilai dan budayanya akan berpengaruh pada produk, dan elemen-elemen itu memiliki komponen berat di atasnya. Kami melihat hal ini terjadi di industri lain yang telah mengalami peningkatan dramatis dalam keamanan dan kualitas. Ahli kualitas yang terkenal J.M. Juran menulis:

> ***Pencapaian kepemimpinan yang berkualitas mengharuskan para manajer tingkat atas untuk secara pribadi mengambil alih manajemen kualitas. Di perusahaan-perusahaan yang telah mencapai kepemimpinan berkualitas, para manajer tingkat atas secara pribadi memandu inisiatif tersebut. Saya tidak mengetahui adanya pengecualian. [3]***

Kami yakin bahwa keamanan merupakan subkategori dari kualitas produk. Ketika keamanan dan kualitas menjadi kewajiban bisnis daripada fungsi teknis yang sepenuhnya diserahkan kepada staf teknis, organisasi akan mampu merespons terhadap kebutuhan keamanan pelanggan mereka lebih cepat dan efisien. Selain itu, menginvestasikan sumber daya yang dibutuhkan untuk memastikan bahwa keamanan perangkat lunak merupakan prioritas bisnis inti dari awal akan mengurangi biaya jangka panjang untuk mengatasi kerusakan perangkat lunak-dan pada akhirnya, menurunkan risiko keamanan nasional.

Dengan cara yang sama tim kepemimpinan telah mengimplementasikan program tanggung jawab sosial perusahaan (CSR), terdapat pula kesadaran yang tumbuh bahwa dewan korporasi, termasuk produsen perangkat lunak, harus mengambil peranan yang lebih aktif dalam memandu program keamanan siber. Istilah tanggung jawab siber perusahaan (CCR) terkadang digunakan untuk menjelaskan gagasan yang muncul ini.

MENDEMONSTRASIKAN PRINSIP INI

Untuk mendemonstrasikan prinsip ini, produsen perangkat lunak seharusnya mengikuti langkah-langkah berikut ini:

1. **Sertakan detail program aman berdasarkan desain dalam laporan keuangan perusahaan.** Jika produsen adalah perusahaan dagang terbuka, tambahkan satu bagian dalam setiap laporan tahunan tentang upaya aman berdasarkan desain. Hal yang sudah biasa bagi laporan keuangan tahunan otomotif untuk menyertakan bagian tentang keselamatan pengemudi dan penumpang, termasuk informasi tentang kualitas terpusat dan terdistribusi serta komite keselamatan. Memperinci program aman berdasarkan desain dalam laporan keuangan akan mendemonstrasikan bahwa organisasi mengaitkan keamanan pelanggan dan hasil keuangan perusahaan dan tidak hanya mengadopsi istilah dalam materi pemasaran hanya karena itu sedang populer.
2. **Berikan laporan berkala kepada dewan direksi Anda.** Laporan Chief information security officer (CISO) kepada dewan perusahaan biasanya mencakup informasi tentang program keamanan saat ini dan yang direncananakan, ancaman, insiden keamanan yang dicurigai dan terkonfirmasi, serta pembaruan lain yang terpusat pada postur keamanan dan kesehatan perusahaan. Selain menerima informasi tentang postur keamanan perusahaan, dewan juga harus meminta informasi tentang keamanan produk dan dampaknya terhadap keamanan pelanggan. Dewan tidak semata-mata bergantung pada CISO, terutama pada anggota lain dalam manajemen perusahaan untuk mengurangi risiko pelanggan.
3. **Berdayakan eksekutif aman berdasarkan desain.** Ada perbedaan signifikan antara satu organisasi yang tim teknisnya memiliki "keterlibatan eksekutif (executive buy-in)", dengan organisasi yang pemimpin bisnisnya secara pribadi mengelola proses peningkatan keamanan pelanggan menggunakan proses bisnis standar. Istilah "executive buy-in" menyiratkan bahwa seseorang harus menjual ide program keselamatan pelanggan bukannya menjadi tujuan bisnis tingkat atas. Eksekutif ini harus diberdayakan untuk mempengaruhi investasi produk agar mencapai keamanan pelanggan.
4. **Ciptakan insentif internal yang bermakna.** Sementara menyadari untuk tidak membuat insentif yang salah, selaraskan sistem penghargaan untuk meningkatkan keamanan pelanggan agar sesuai dengan perilaku I dan hasil yang bernilai lainnya. Dari eksekutif aman berdasarkan desain hingga manajemen produk, pengembangan perangkat lunak, dukungan, penjualan, hukum, dan organisasi-organisasi lain, sertakan juga insentif keamanan pelanggan ke dalam perekrutan, promosi, gaji, bonus, pilihan stok, dan proses umum lainnya dalam menjalankan bisnis. Sebagai contoh, ketika menerapkan kriteria untuk mempromosikan pengembang perangkat lunak, sertakan pertimbangan untuk meningkatkan keamanan produk bersama dengan kriteria lain seperti waktu aktif, kinerja, dan peningkatan fitur.
5. **Buat konsil aman berdasarkan desain.** Dalam beberapa industri, merupakan hal lazim bagi organisasi untuk membentuk konsil kualitas pusat, dan menempatkan perwakilan kualitas di divisi-divisi atau unit bisnis utama. Dengan menyertakan anggota-anggota terpusat dan terdistribusi, kelompok ini bekerja untuk meningkatkan kualitas menurut tujuan tingkat atas saat menerima telemetri dari dalam organisasi. Demikian pula, konsil aman berdasarkan desain akan meningkatkan keamanan terhadap tujuantujuan aman berdasarkan desain di seluruh organisasi.
6. **Ciptakan dan kembangkan konsil pelanggan.** Banyak produsen perangkat lunak memiliki konsil pelanggan yang terdiri dari para pelanggan dari berbagai wilayah, industri dan ukuran. Konsil-konsil ini dapat memberikan banyak informasi tentang kesuksesan dan tantangan pelanggan dalam menerapkan produk perusahaan. Susun agenda konsil dengan topik khusus yang membahas keamanan pelanggan, walaupun topik itu bukan prioritas para pesertanya. Pertimbangkan di mana konsil pelanggan memberikan laporannya dan cara memanfaatkan peserta untuk mendapatkan wawasan tentang keamanan produk saat diterapkan. Sebagai contoh, apakah konsil memiliki bias terhadap tujuan pemasaran dan penjualan, atau manajemen produk? Eksekutif aman berdasarkan desain harus membantu mengarahkan interaksi pelanggan dan harus menghubungkannya dengan elemen lain dalam makalah ini, seperti studi lapangan.

TAKTIK-TAKTIK AMAN BERDASARKAN DESAIN

Kerangka Kerja Pengembangan Perangkat Lunak Aman (Secure Software Development Framework/ SSDF), juga dikenal sebagai Institut Nasional Standar dan Teknologi (NIST) SP 800-218, adalah kumpulan inti praktik pengembangan perangkat lunak aman tingkat tinggi yang dapat diintegrasikan ke dalam setiap tahap siklus hidup pengembangan perangkat lunak (software development lifecycle/ SDLC). Mengikuti praktik ini dapat membantu produser perangkat lunak menjadi lebih efektif dalam menemukan dan menghilangkan kerentanan dalam perangkat lunak yang dirilis, memitigasi potensi dampak dari eksploitasi kerentanan, dan mengatasi akar penyebab kerentanan untuk mencegah pengulangan nantinya.

Organisasi-organisasi penggagas mendorong penggunaan taktik aman berdasarkan desain, termasuk prinsip-prinsip yang mengacu ke praktik SSDF. Produsen-produsen perangkat lunak harus mengembangkan peta jalan tertulis untuk mengadopsi lebih banyak praktik pengembangan perangkat lunak yang aman berdasarkan desain di seluruh portofolio mereka. Berikut ini daftar ilustratif tidak lengkap mengenai praktik-praktik terbaik peta jalan:

- **Bahasa pemrograman aman memori (SSDF PW.6.1.)** Memprioritaskan penggunaan bahasa aman memori di mana pun sebisa mungkin. Organisasi-organisasi penggagas mengakui bahwa mitigasi khusus memori dapat membantu sebagai taktik jangka pendek bagi basis kode versi lama. Contoh termasuk peningkatan bahasa C/C++, mitigasi perangkat keras, pengacakan tata letak ruang alamat (ASLR), integritas aliran-kontrol (CFI), dan fuzzing. Namun demikian, ada konsensus yang tumbuh bahwa adopsi bahasa pemrograman aman memori dapat mengeliminasi kelas kecacatan ini, dan produsen perangkat lunak harus mengeksplorasi cara mengadopsinya. Beberapa contoh bahasa aman memori modern termasuk C#, Rust, Ruby, Java, Go, dan Swift. Bacalah keamanan memori NSA lembar informasi untuk mengetahui lebih lanjut.
- **Fondasi Perangkat Keras Aman.** Menggabungkan fitur-fitur arsitektural yang memungkinkan proteksi memori berdetail tingkat tinggi, seperti yang dijelaskan oleh Capability Hardware Enhanced RISC Instructions (CHERI) yang dapat memperluas Arsitektur Perangkat Instruksi (ISA) perangkat keras konvensional, serta fitur-fitur lain seperti Trusted Platform Module (Modul Platform Terpercaya) dan Hardware Security Module (Modul Keamanan Perangkat Keras). Untuk informasi lebih lanjut kunjungi, halaman web CHERI Universitas Cambridge.
- **Komponen-komponen Perangkat Lunak Aman (SSDF PW 4.1).** Peroleh dan pelihara komponen perangkat lunak yang diamankan dengan baik (misalnya, perpustakaan perangkat lunak, modul, middleware, kerangka kerja,) dari pengusaha yang terverifikasi, sumber terbuka, dan pengembang pihak ketiga lainnya untuk memastikan keamanan yang kokoh dalam produk perangkat lunak pelanggan.
- **Kerangka kerja templat web (SSDF PW.5.1).** Gunakan kerangka kerja templat web yang mengimplementasikan pelarian otomatis masukan pengguna untuk menghindari serangan web seperti skrip lintas situs.
- **Kueri berparameter (SSDF PW 5.1).** Gunakan kueri berparameter daripada menyertakan masukan pengguna dalam kueri, untuk menghindari serangan injeksi SQL.
- **Pengujian keamanan aplikasi statis dan aplikasi dinamis (SAST/DAST)** (SSDF PW.7.2, PW.8.2): Gunakan alat-alat ini untuk menganalisis kode sumber produk dan perilaku aplikasi untuk mendeteksi praktik yang mudah-galat. Alat-alat ini mencakup masalah mulai dari manajemen memori yang tidak tepat hingga konstruksi kueri basis data yang mudah galat (misalnya, masukan pengguna yang tidak lolos menyebabkan injeksi SQL). Alat-alat SAST dan DAST dapat dimasukkan ke dalam proses pengembangan dan secara otomatis berjalan sebagai bagian dari pengembangan perangkat lunak. SAST dan DAST dapat melengkapi tipe pengujian lain, seperti pengujian unit dan pengujian integrasi, guna memastikan produk mematuhi persyaratan keamanan yang diharapkan. Ketika masalah teridentifikasi, produsen harus melakukan analisis akar penyebab untuk mengatasi kerentanan secara sistemik.

- **Tinjauan kode** (SSDF PW.7.1, PW.7.2): Berusaha memastikan bahwa kode yang dikirimkan ke dalam produk telah melalui teknik kontrol kualitas seperti tinjauan sejawat oleh pengembang lain atau "error seeding (penyemaian kesalahan)".
- **Daftar Bahan-Bahan Perangkat Lunak (Software Bill of Materials/SBOM)** (SSDF PS.3.2, PW.4.1): Masukkan pembuatan SBOM[4] untuk memberikan visibilitas ke dalam set perangkat lunak yang digunakan dalam produk.
- **Program pengungkapan kerentanan** (SSDF RV.1.3): Tetapkan program pengungkapan kerentanan agar para peneliti keamanan dapat melaporkan berbagai kerentanan dan menerima perlindungan hukum dalam melakukannya. Sebagai bagian darinya, pemasok harus menerapkan proses untuk menentukan akar penyebab dari kerentanan yang ditemukan. Proses seperti itu harus mencakup penentuan apakah mengadopsi praktik aman berdasarkan desain dalam dokumen ini (atau praktik serupa) akan dapat mencegah kerentanan.
- **Kelengkapan CVE.** Pastikan bahwa CVE yang diterbitkan mencakup akar penyebab atau enumerasi kelemahan umum (CWE) untuk mengaktifkan analisis seluruh industri dari kelemahan desain keamanan perangkat lunak. Walaupun proses memastikan setiap CVE benar dan komplit bisa memakan waktu ekstra, namun memungkinkan entitas yang berbeda menemukan tren industri yang menguntungkan semua produsen dan pelanggan. Untuk informasi lebih lanjut tentang mengelola kerentanan, bacalah artikel CISA Pedoman Kategorisasi Kerentanan Khusus-Pemangku Kepentingan (SSVC).
- **Pertahanan Menyeluruh.** Rancanglah infrastruktur agar penyusupan terhadap satu kontrol keamanan tidak menyebabkan penyusupan terhadap keseluruhan sistem. Sebagai contoh, memastikan hak istimewa penggunadisediakan secara terbatas, dan daftar kontrol akses diterapkan dapat mengurangi dampak terhadap akun yang terganggu. Selain itu, teknik sandboxing perangkat lunak dapat mengarantina kerentanan untuk membatasi penyusupan seluruh aplikasi.
- **Memenuhi Target Kinerja Keamanan Siber (CPG)** Rancang produk-produk yang memenuhi praktik-praktik keamanan dasar. Tujuan Kinerja Keamanan Siber CISA menguraikan tindakan-tindakan keamanan siber dasar fundamental yang harus diimplementasikan oleh organisasi. Selain itu, untuk mengetahui lebih banyak cara memperkuat postur organisasi Anda, simaklah Kerangka Kerja Penilaian Siber Inggris Raya yang mirip dengan CPG CISA. Jika suatu produsen gagal memenuhi CPG— seperti tidak mengharuskan MFA anti phishing bagi semua karyawan— maka produsen tersebut tidak dapat dianggap menghasilkan produk yang aman berdasarkan desain.

Organisasi-organisasi penggagas mengakui bahwa perubahan-perubahan ini merupakan pergeseran yang signifikan dalam sebuah postur organisasi. Dengan demikian, pengenalannya harus diprioritaskan berdasarkan pemodelan ancaman yang disesuaikan, kekritisan, kekompleksan, dan dampak pada bisnis. Praktik-praktik ini dapat diperkenalkan untuk perangkat lunak baru dan diperluas secara bertahap untuk mencakup kasus penggunaan dan produk tambahan. Dalam beberapa kasus, kekritisan dan postur risiko produk tertentu memerlukan jadwal cepat untuk mengadopsi praktik-praktik ini. Dalam kasus lain, praktik dapat diperkenalkan ke dalam basis kode versi lama dan diperbaiki seiring berjalannya waktu.

[4] Beberapa organisasi penggagas mengeksplorasi pendekatan alternatif untuk mendapatkan jaminan keamanan di sekitar rantai pasokan perangkat lunak.

TAKTIK AMAN BERDASARKAN BAWAAN

Selain pengadopsian praktik pengembangan aman berdasarkan desain, organisasi penggagas merekomendasikan produsen perangkat lunak agar memprioritaskan konfigurasi aman berdasarkan bawaan dalam produk mereka. Konfigurasi tersebut harus berusaha memperbarui produk agar sesuai dengan praktik-praktik ini seiring dengan dilakukannya penyegaran. Sebagai contoh:

- **Eliminasi kata sandi bawaan.** Produk-produk tidak boleh beredar dengan kata sandi bawaan yang dibagikan secara universal. Untuk menghapus kata sandi bawaan, organisasi-organisasi penggagas menyarankan agar produk mewajibkan administrator mengatur kata sandi yang kuat saat instalasi dan konfigurasi atau agar produk dikirimkan dengan kata sandi yang unik dan kuat untuk setiap perangkat.
- **Mewajibkan Autentikasi Multi Faktor (MFA) untuk para pengguna yang memiliki hak istimewa.** Kami mengamati bahwa banyak penerapan dalam perusahaan dikelola oleh para administrator yang belum memproteksi akun mereka dengan MFA. Mengingat bahwa para administrator merupakan target bernilai tinggi, produk harus membuat MFA tidak diikutsertakan (opt-out) daripada diikutsertakan (opt-in). Lebih jauhlagi, sistem harus secara teratur meminta administrator untuk mendaftarkan MFA hingga mereka berhasil mengaktifkannya di akun mereka. NCSC Belanda memiliki panduan yang sejalan dengan pedoman CISA, kunjungi Lembar Fakta Autentikasi Sempurna mereka untuk informasi lebih lanjut.
- **Sistem masuk tunggal (SSO).** Aplikasi-aplikasi TI harus mengimplementasikan dukungan sistem masuk tunggal melalui standar terbuka modern. Contohnya termasuk Bahasa Markup Pernyataan Keamanan (SAML) atau OpenID Connect (OIDC). Kemampuan ini harus tersedia secara bawaan tanpa biaya tambahan.
- **Log Aman.** Menyediakan log audit berkualitas tinggi bagi pelanggan tanpa biaya tambahan atau konfigurasi tambahan. Log audit bersifat krusial dalam mendeteksi dan mengeskalasikan potensi insiden keamanan. Mereka juga bersifat krusial selama investigasi insiden keamanan yang dicurigai atau yang sudah dikonfirmasi. Pertimbangkan praktik-praktik terbaik seperti menyediakan integrasi yang mudah dengan sistem informasi keamanan dan manajemen acara dengan akses antarmuka pemrograman aplikasi (API) yang menggunakan waktu universal terkoordinasi (UTC), format zona waktu standar, dan teknik-teknik dokumentasi yang canggih.
- **Profil Otorisasi Perangkat Lunak.** Pemasok perangkat lunak harus memberikan rekomendasi tentang peranan profil yang diotorisasi dan kasus penggunaan tertentu mereka. Produsen harus memunculkan peringatan yang memberitahukan pelanggan tentang peningkatan risiko jika mereka menyimpang dari otorisasi profil yang direkomendasikan. Sebagai contoh, dokter medis dapat melihat semua rekam medis pasien, namun penjadwal medis memiliki akses terbatas terhadap informasi tertentu yang diperlukan untuk menjadwalkan janji temu.
- **Keamanan masa depan daripada kompatibilitas masa lampau.** Seringkali, fitur yang kompatibel dengan versi lama dimasukkan, dan sering diaktifkan dalam produk-produk walaupun menimbulkan risiko bagi keamanan produk. Prioritaskan keamanan daripada kompatibilitas mundur, memberdayakan tim keamanan untuk menghilangkan fitur-fitur tidak aman itu bahkan jika harus melakukan perubahan.
- **Lacak dan perkecil ukuran "panduan pengerasan" (hardening guide).** Perkecil ukuran "panduan pengerasan (hardening guide)" yang disertakan dengan produk dan pastikan bahwa ukuran terus berkurang seiring dengan versi baru perangkat lunak yang dirilis. Integrasikan komponen-komponen "panduan pengerasan" sebagai konfigurasi bawaan pada produk. Organisasi-organisasi penggagas

mengakui bahwa panduan pengerasan yang dipersingkat yang diperoleh dari kemitraan berkelanjutan dengan pelanggan yang ada dan mencakup usaha dari banyak tim produk, termasuk pengalaman pengguna (UX).

- **Pertimbangkan konsekuensi pengalaman pengguna akan penataan keamanan.** Setiap penataan baru menaikkan beban kognitif pada pengguna akhir dan seharusnya dinilai bersamaan dengan manfaat bisnis yang diperolehnya. Idealnya, satu penataan seharusnya tidak ada; alih-alih, penataan paling aman harus dipadukan ke dalam produk secara bawaan. Ketika konfigurasi diperlukan, opsi bawaan harus aman secara luas terhadap ancaman-ancaman umum.

Organisasi penggagas mengakui perubahan-perubahan ini dapat mempengaruhi cara penggunaan perangkat lunak. Oleh karena itu, masukan pelanggan bersifat kritis dalam menyeimbangkan pertimbangan operasional dan keamanan. Kami yakin bahwa pengembangan peta jalan tertulis dan dukungan eksekutif yang memprioritaskan ide-ide ini ke dalam produk yang paling kritis dari organisasi merupakan langkah pertama menuju perubahan ke praktik pengembangan perangkat lunak yang aman. Walaupun masukan pelanggan itu penting, kami telah mengamati kasus-kasus penting di mana para pelanggan tidak ingin atau tidak mampu mengadopsi standar yang lebih baik, seringkali protokol jaringan. Penting bagi produsen untuk menciptakan insentif yang bermakna bagi pelanggan agar tetap mengikuti perkembangan terkini dan tidak membiarkan mereka tetap rentan selamanya.

PANDUAN PENGERASAN (HARDENING GUIDE) VERSUS PANDUAN PELONGGARAN (LOOSENING GUIDE)

Panduan pengerasan dapat disebabkan oleh kurangnya kontrol keamanan produk yang tertanam dalam arsitektur produk sejak awal pengembangan. Akibatnya, panduan pengerasan dapat juga menjadi peta jalan bagi musuh untuk menentukan dan mengeksploitasi fitur-fitur yang tidak aman. Merupakan hal yang umum bagi banyak organisasi untuk tidak menyadari panduan pengerasan, dengan demikian mereka membiarkan penataan konfigurasi perangkatnya dalam postur tidak aman. Sebuah model kebalikannya yang dikenal dengan panduan pelonggaran harus menggantikan panduan pengerasan semacam itu dan menjelaskan perubahan mana yang harus dilakukan oleh para pengguna sementara juga mendaftar risiko-risiko keamanan yang disebabkannya. Panduan ini harus ditulis oleh praktisi keamanan yang dapat menjelaskan pengorbanannya dalam bahasa yang jelas guna menaikkan peluang untuk menerapkannya dengan benar.

Daripada mengembangkan panduan pengerasan yang mendaftar metode-metode untuk mengamankan produk, organisasi-organisasi penggagas menyarankan produsen perangkat lunak agar beralih ke pendekatan aman berdasarkan bawaan dengan menyediakan "panduan pelonggaran". Panduan-panduan ini menjelaskan keputusan risiko bisnis dalam bahasa yang mudah dimengerti, dan dapat meningkatkan kesadaran organisasi akan risiko terhadap intrusi siber berbahaya. Pengorbanan keamanan harus ditetapkan oleh eksekutif senior pelanggan, guna menyeimbangkan keamanan dengan persyaratan bisnis lainnya.

REKOMENDASI BAGI PELANGGAN

Organisasi penggagas menyarankan organisasi untuk meminta pertanggungjawaban dari pemasok produsen perangkat lunak mereka atas keamanan produknya. Sebagai bagian dari hal ini, organisasi penggagas merekomendasikan agar para eksekutif memprioritaskan kepentingan membeli produk yang aman berdasarkan desain dan aman berdasarkan bawaan. Hal ini dapat terwujud melalui penetapan kebijakan yang mengharuskan departemen TI menilai keamanan perangkat lunak sebelum dibeli, serta memberi wewenang kepada departemen TI untuk melakukan penolakan, jika diperlukan. Departemen TI harus diberi wewenang untuk mengembangkan kriteria pembelian yang menekankan pentingnya praktik aman berdasarkan desain dan aman berdasarkan bawaan (baik yang diuraikan dalam dokumen ini dan lainnya yang dikembangkan oleh organisasi). Selain itu, departemen TI harus didukung oleh manajemen eksekutif ketika menerapkan kriteria tersebut dalam membuat keputusan pembelian. Keputusan organisasi untuk menerima risiko terkait dengan produk teknologi tertentu harus didokumentasikan secara formal, dan disetujui oleh seorang eksekutif bisnis senior, dan secara teratur disampaikan ke dewan direksi.

Layanan kunci IT perusahaan yang mendukung postur keamanan organisasi, seperti jaringan perusahaan, identitas perusahaan dan manajemen akses, dan operasi keamanan serta kapabilitas respons, harus dilihat sebagai fungsi bisnis kritis yang didanai agar selaras dengan kepentingannya bagi kesuksesan misi organisasi. Organisasi harus mengembangkan sebuah rencana untuk meningkatkan kapabilitas ini guna memanfaatkan produsen yang menganut praktik aman berdasarkan desain dan aman berdasarkan bawaan.

Jika memungkinkan, organisasi harus berusaha menjalin hubungan strategis dengan pemasok TI utama mereka. Hubungan semacam itu mencakup kepercayaan di berbagai tingkat organisasi dan memberikan cara untuk menyelesaikan masalah dan mengidentifikasi prioritas bersama. Keamanan harus menjadi elemen kritis dalam hubungan semacam itu dan organisasi harus berusaha memperkuat pentingnya praktik aman berdasarkan desain dan aman berdasarkan bawaan baik dalam dimensi hubungan formal (misalnya, kontrak atau perjanjian vendor) maupun informal. Organisasi harus mengharapkan transparansi dari pemasok teknologi mereka mengenai postur pengendalian internal serta peta jalan mereka menuju pengadopsian praktik aman berdasarkan desain dan aman berdasarkan bawaan.

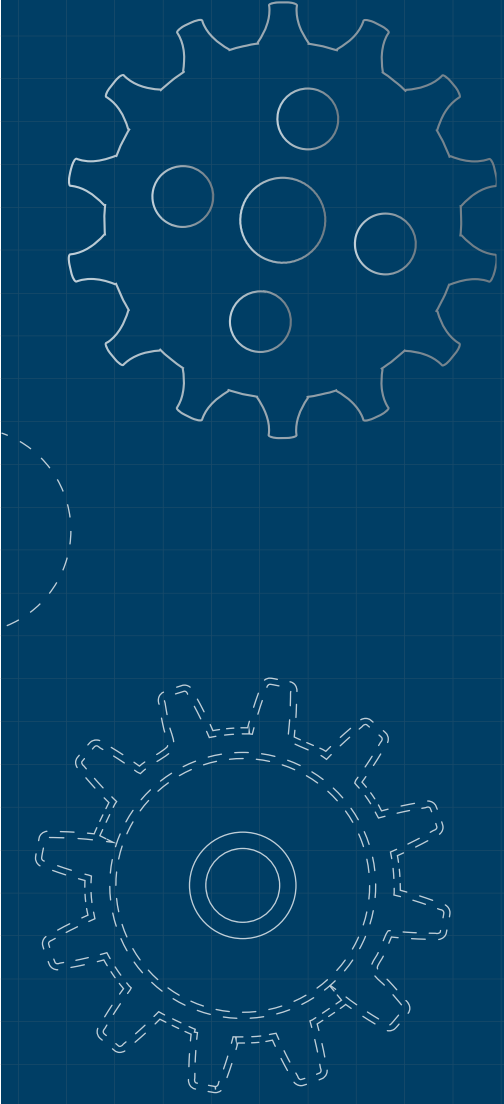
Selain menjadikan aman berdasarkan bawaan sebagai prioritas dalam suatu organisasi, para pemimpin TI harus berkolaborasi dengan rekan-rekan industri mereka untuk memahami produk dan layanan mana yang paling sesuai dengan prinsip-prinsip desain ini. Para pemimpin ini harus mengkoordinasikan permintaan mereka untuk membantu produsen memprioritaskan inisiatif keamanan mendatang. Dengan bekerja sama, pelanggan dapat membantu memberikan masukan yang bermakna bagi produsen dan membuat insentif bagi mereka untuk memprioritaskan keamanan.

Ketika memanfaatkan sistem cloud, organisasi harus memastikan mereka memahami model tanggung jawab bersama dengan pemasok teknologi mereka. Yakni, organisasi harus memiliki kejelasan mengenai tanggung jawab keamanan pemasok daripada tanggung jawab pelanggan saja.

Organisasi harus memprioritaskan penyedia cloud yang transparan mengenai postur keamanan mereka, kontrol internal, dan kemampuan memenuhi kewajiban mereka dalam model tanggung jawab bersama.

PENAFIAN

Informasi dalam laporan ini disediakan "sebagaimana adanya" untuk tujuan yang bersifat informasi saja. CISA dan organisasi-organisasi penggagas tidak mendukung produk atau layanan komersial apa pun, termasuk subjek analisis. Referensi apa pun untuk entitas komersial teretentu atau produk komersial, proses, atau layanan komersial oleh merek layanan, merek dagang, produsen, atau lainnya, bukan merupakan atau menyiratkan dukungan, rekomendasi, atau favoritisme oleh CISA dan organisasi-organisasi penggagas. Dokumen ini merupakan inisiatif bersama dari CISA yang tidak berfungsi secara otomatis sebagai dokumen pengatur.



SUMBER

CISA
- Panduan SBOM CISA
- Tujuan-Tujuan Kinerja Keamanan Siber Lintas Sektor CISA
- Garis Pedoman Mengenai Interoperabilitas Teknologi
- Mempertahankan Diri Dari Serangan Rantai Pasokan Perangkat Lunak CISA dan NIST
- Dampak dari Teknologi Tidak Aman dan Apa Yang Bisa Dilakukan Untuk Mengatasinya | CISA
- Berhenti Mencari Kambing Hitam untuk Keamanan Siber: Mengapa Perusahaan Harus Membangun Keamanan Pada Produk Teknologi (foreignaffairs.com)
- Pedoman Kategorisasi Kerentanan Khusus-Pemangku Kepentingan CISA (SSVC)
- Lembar Fakta MFA Anti Phising CISA
- Pedoman Siber bagi Bisnis Kecil | CISA

NSA
- Lembar Informasi Keamanan Siber NSA tentang Keamanan Memori
- Mengamankan Rantai Pasokan Perangkat Lunak ESF NSA: Praktik Terbaik bagi Pemasok

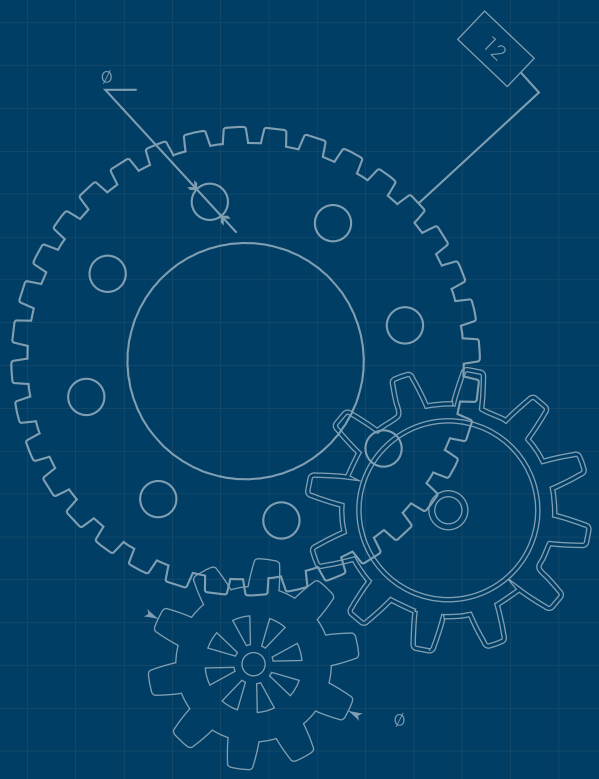
FBI
- Memahami dan Merespons Serangan Rantai Pasokan SolarWinds: Perspektif Federal
- Ancaman Siber - Respons dan Pelaporan
- Strategi Siber FBI

Institut Nasional Standar dan Teknologi (NIST)
- Pedoman Identitas Digital NIST
- Kerangka Kerja Keamanan Siber NIST
- Kerangka Kerja Pengembangan Perangkat Lunak Yang Aman NIST (SSDF)

Pusat Keamanan Siber Australia (ACSC)
- Kode Internet untuk Segala ACSC untuk Pedoman Praktis bagi Produsen

Pusat Keamanan Siber Nasional Kerajaan Inggris Raya (UK)
- Kerangka Kerja Penilaian Siber Inggris Raya
- Panduan Pengembangan dan Pemakaian Aman NCSC UK
- Panduan Manajemen Kerentanan NCSC UK
- Perangkat Pengungkapan Kerentanan NCSC UK
- CHERI Universitas Cambridge
- Sekian dan terima kasih untuk semuanya - NCSC.GOV.UK

Pusat Keamanan Siber Kanada (CCCS)
- Panduan Perlindungan Terhadap Serangan Rantai Pasokan Perangkat Lunak CCCS
- Rantai pasokan siber: Sebuah pendekatan untuk menilai risiko
- Panduan ransomware CONTI Pusat Keamanan Siber Kanada



Kantor Federal untuk Keamanan Informasi Jerman (BSI)

- Ringkasan Perlindungan Dasar BSI (modul CON.8)
- Standar Internasional IEC (International Electrotechnical Commission) 62443, bagian 4-1
- Laporan Situasi Keamanan TI di Jerman, 2022
- Praktik-Praktik Keamanan Aplikasi Web BSI

Pusat Keamanan Siber Nasional Belanda

- Lembar Fakta Autentikasi Sempurna NCSC-NL

Pusat Nasional Kesiapan Insiden dan Strategi untuk Keamanan Siber Jepang (NISC)

- Strategi Keamanan Siber Nasional Jepang

Kementerian Ekonomi, Perdagangan, dan Industri Jepang (METI)

- Panduan Pendahuluan Tentang Daftar Bahan-Bahan Perangkat lunak (SBOM) untuk Manajemen Perangkat Lunak
- Koleksi Contoh Kasus Penggunaan Mengenai Metode Manajemen untuk Memanfaatkan OSS dan Memastikan Keamanannya.

Badan Keamanan Siber Singapura

- Penasihat Teknis tentang Pengembangan API Aman
- Kebijakan Pengungkapan Kerentanan CSA SingCERT
- Daftar Periksa Respons Insiden CSA SingCERT
- Buku Pedoman Respons Insiden CSA SingCERT
- Kerangka Kerja Keamanan berdasarkan Desain CSA
- Daftar Periksa Kerangka Kerja Keamanan berdasarkan Desain CSA
- Pedoman untuk Pemodelan Ancaman Siber CSA
- Skema Pelabelan Keamanan Siber CSA

Lainnya

- Bagaimana Sistem-Sistem Kompleks Gagal
- New Look dalam kegagalan sistem kompleks

REFERENSI

[1] https://csrc.nist.rip/publications/history/ande72.pdf

[2] https://www.cisa.gov/sbom dan referensi SBOM di TR 03183-2 https://www.bsi.bund.de/dok/TR-03183

[3] Juran tentang Kualitas berdasarkan Desain oleh J.M. Juran, 1992.