Edisi Perdana - Mei 2024

ITSECBUZZ

MAJALAH KEAMANAN SIBER

SPIONASE

ZERO-CLICK EXPLOIT

SUPER SPYWARE PEGASUS VS PREDATOR

RISET

TANDA TANGAN DIGITAL DALAM BAHAYA: EKSPLOITASI PADA FILE PDF

KEAMANAN DATA

DATA BOCOR, HARUS BAGAIMANA?

3 FITUR BERBAHAYA ANDROID

ITSEC™
SECURITY DELIVERED

"SECURITY USED TO BE AN INCONVENIENCE SOMETIMES, BUT NOW IT'S A NECESSITY ALL THE TIME"

Martina Navratilova

DAFTAR ISI

EDITORIAL

TENTANG ITSEC ASIA

Joseph Edi Hut Lumban Gaol
President Director PT. ITSEC Asia Tbk.

ITSEC Asia adalah perusahaan keamanan siber lokal yang berkembang secara global. Didirikan pada tahun 2010 sebagai ITSEC Indonesia, kini ITSEC Asia memiliki lebih dari 300 karyawan yang tersebar di empat negara yaitu Indonesia, Singapura, Australia dan Uni Emirate Arab. ITSEC Asia telah menyelesaikan lebih dari 5000 proyek dan memegang berbagai sertifikasi, antara lain ISO 9001 untuk Manajemen Mutu, ISO 27001 untuk Manajemen Keamanan Informasi, verifikasi BSSN, sertifikasi CREST sebagai Penguji Penetrasi, dan sertifikasi CREST sebagai Penyedia Penilaian Kerentanan. Dengan fokus yang kuat pada kepuasan pelanggan, ITSEC telah menjadi nama tepercaya dalam solusi keamanan siber.

PT. ITSEC Asia Tbk merupakan salah satu perusahaan keamanan siber lokal terbesar di Indonesia. ITSEC mempunyai visi menjadi perusahaan keamanan siber terbesar dan tepercaya di Asia Pasifik. Strategi yang akan dijalankan perusahaan dalam bisnisnya pada tahun 2024 ini yaitu dengan melakukan pengembangan solusi dan adopsi teknologi yang sudah ada saat ini, sejalan dengan pilar bisnis perusahaan di bidang konsultasi, solusi teknologi, dan managed security services.

Tim Research & Development ITSEC juga berkelanjutan membuat inovasi-inovasi dalam pembuatan software yang akan memperkuat posisi ITSEC sebagai perusahaan keamanan siber. Pengembangan software atau perangkat lunak terbaru ini nantinya bukan hanya menyasar ke segmen korporasi tetapi juga UKM (Usaha Kecil dan Menengah).

Menurut data dari Kementerian Koordinator Perekonomian, nilai ekonomi digital Indonesia mencapai US$82 miliar pada 2023. Jumlah ini diperkirakan naik hingga US$109 miliar pada 2025. Di sisi lain, laporan terkait Ancaman Digital di Indonesia Semester II 2023 dari Awanpintar.id menemukan total serangan siber di Indonesia mencapai 686 juta. Angka ini melonjak hampir dua kali lipat dibandingkan semester sebelumnya sekitar 347 juta dan berpotensi melonjak di masa mendatang. Itu sebabnya PT. ITSEC Asia menyediakan layanan keamanan siber untuk semua segmen pelaku ekonomi digital.

Artikel ini disadur dari **Bisnis Indonesia**

EDITORIAL

SALAM PERKENALAN

Bismillahirrahmannirrahim,

Puji syukur serta ucapan terima kasih yang tak terhingga kami panjatkan kepada Tuhan Yang Maha Kuasa atas izin-Nya sehingga kami dapat menerbitkan edisi pertama ITSEC BUZZ; majalah elektronik ITSEC Asia.

Di era digital saat ini, memahami prinsip-prinsip keamanan siber dasar sangatlah penting bagi semua orang, mulai dari pelajar, pemilik usaha kecil, hingga masyarakat umum yang sudah sangat terbiasa menggunakan perangkat digital dan koneksi internet.

Misi kami melalui publikasi dalam bentuk majalah elektronik ini adalah untuk meningkatkan kesadaran keamanan informasi diantara masyarakat Indonesia sehingga kami berupaya sebaik mungkin untuk dapat menyajikan ragam informasi terkait keamanan informasi dalam dunia siber yang cenderung rumit kedalam bahasa yang lebih mudah dipahami khususnya bagi masyarakat awam.

Besar harapan kami bahwa artikel-arikel yang tersaji pada edisi pertama dan edisi-edisi berikutnya, majalah elektronik keamanan siber ini dapat memberikan kontribusi pengetahuan sehingga dapat semakin meningkatkan literasi keamanan siber bagi masyarakat Indonesia.

Salam Hangat,

Muhammad Rasyid Sahputra
Editor-in-chief | Head of Research & Development

Editor-in-chief:
M. Rasyid Sahputra

Penulis:
M.Akmal
M.Haekal Al-Ghifari
H. Cika Agustina
A. Indra Prabowo
D.Rahma Hermawan
Z. Ananda

Design & Layout:
Z. Ananda

Proof Reader:
Irra Fachriyanthi
Bambang Susilo

Kontributor:
F.Suprata

Property Gambar:
ITSEC Asia
Freepik.com
AI Generated

INFORMASI

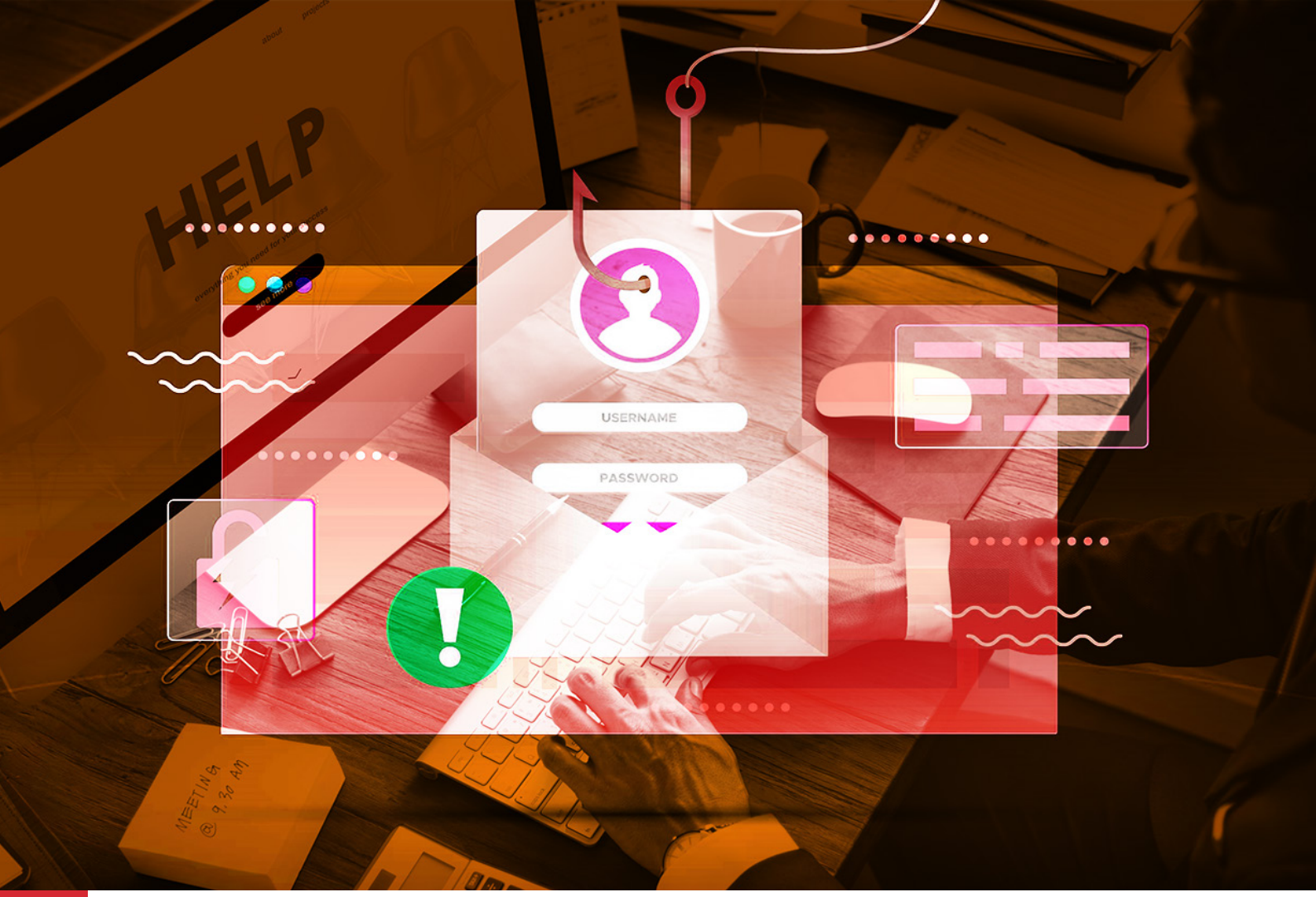
MEMBAHAS TENTANG SOCIAL ENGINEERING

Social Engineering adalah jenis serangan siber yang mengeksploitasi aspek psikologi dan emosional manusia dengan melakukan manipulasi agar mau memberikan informasi sensitif atau menjalankan aktifitas tertentu. Rasa percaya, takut, gembira, ingin tahu, antusias dan serakah adalah perasaan yang paling sering dimainkan di social engineering.

Faktor manusia adalah hal yang paling rentan dan efektif dalam memulai serangan siber. Menurut Purplesec, 98% serangan siber melibatkan faktor social engineering. Begitu si korban terpancing untuk membuka tautan atau file maka pintu akses menuju sistem terbuka lebar.

Phishing, PreTexting, Baiting, Scareware adalah beberapa jenis social engineering yang paling umum dilakukan. Meskipun target utama adalah orang terkenal atau pimpinan perusahaan, seringkali penyerang memindahkan fokus mereka ke karyawan biasa sebagai pintu masuk menuju target yang lebih tinggi, disebabkan karena rendahnya pengetahuan mengenai serangan siber. Itu sebabnya, cara yang paling tepat untuk mencegah social engineering adalah mengedukasi setiap orang, meningkatkan kesadaran akan bahaya dari sebuah serangan siber dan juga cara merespon yang tepat.

> “Total 50% dari social engineering adalah PreTexting, mengaku sebagai orang yang dikenal korban dan memberikan berbagai alasan agar korban mau menyerahkan informasi sensitif yang diinginkan”



Fun Fact

Pria lebih sering terkena serangan phishing daripada wanita

CARA MENCEGAH SOCIAL ENGINEERING

- Jika anda menerima email/pesan yang mencurigakan dan meminta informasi rahasia, pastikan identitas pengirimnya benar.
- Tetap tenang dan hindari larut dalam perasaan setelah membaca isi email. Pastikan maksud pesan yang disampaikan benar.
- Hindari membuka link dan file dari sumber yang tidak tepercaya.
- Pastikan antivirus/antimalware terinstall dan terupdate.
- Selalu mengaktifkan MFA (Multi Factor Authentication).
- Gunakan VPN atau ZeroTrust saat terhubung ke jaringan WiFi publik.
- Minimalisir jejak digital. Hindari postingan yang berisi informasi pribadi.

MALWARE

RAYUAN MALWARE MENYERANG INDUSTRI PERHOTELAN

Penulis: Z.Ananda

Akhir-akhir ini, serangan siber telah melanda banyak pelaku industri perhotelan di Amerika Serikat, termasuk hotel-hotel kecil hingga yang berbintang lima. Umumnya, motif di balik serangan siber pada industri perhotelan adalah pencurian data pelanggan. Namun, ketika korbannya adalah hotel besar dengan omset yang juga besar, seringkali serangan tersebut tidak hanya bertujuan untuk mencuri data tetapi juga untuk mendapatkan uang tebusan.

Pola awal serangan pada hotel-hotel di Amerika Serikat ini memiliki kemiripan. Semuanya menggunakan social engineering, merayu para staff hotel untuk "membukakan" pintu masuk.

Seperti yang terjadi pada MGM Hotel & Casino di Las Vegas pada bulan September 2023, serangan dimulai dari sebuah panggilan telepon ke bagian IT Helpdesk. Penyerang mengaku sebagai seorang pegawai dan meminta melakukan reset password. Dengan melakukan riset menyeluruh melalui media sosial, si penyerang berhasil mengumpulkan informasi yang cukup untuk meyakinkan petugas IT Helpdesk agar memberikan semua permintaannya. Kabarnya, dalam waktu hanya 10 menit, para peretas berhasil mendapatkan akses yang mereka inginkan.

BERPURA-PURA SEBAGAI PELANGGAN

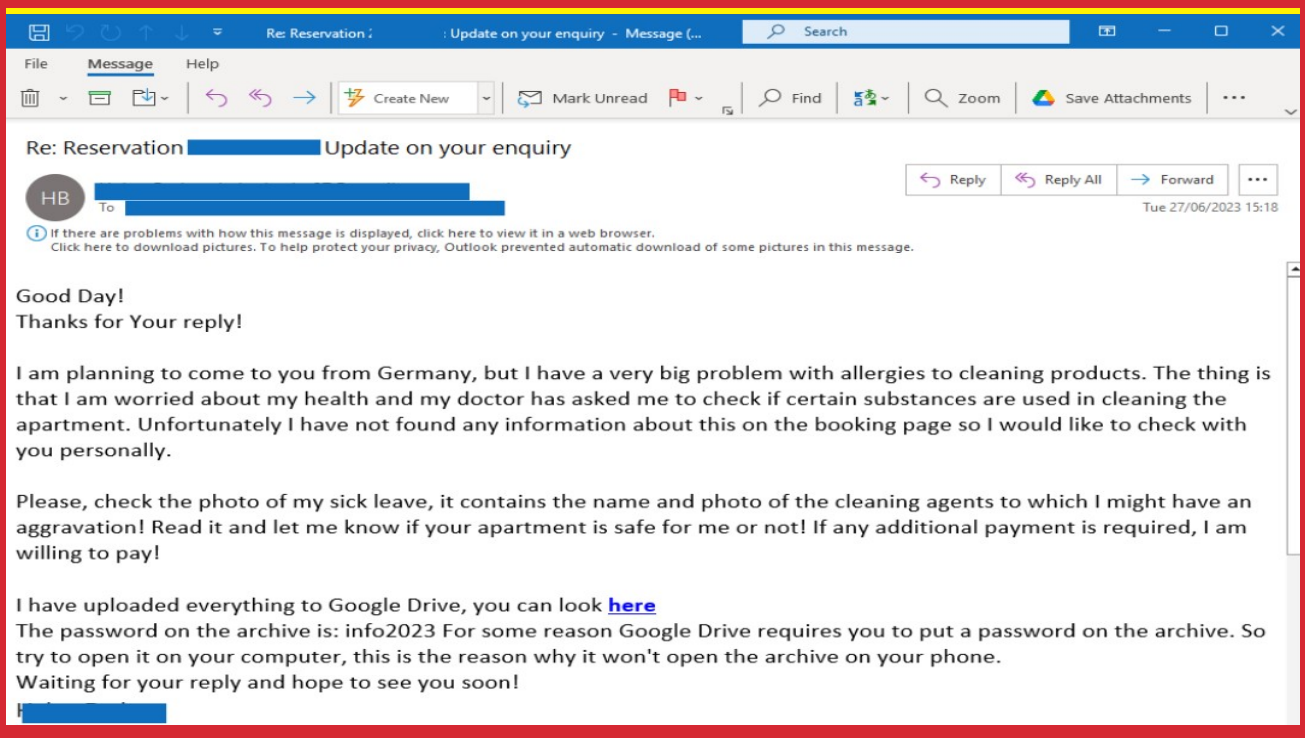
Skenario serangan pada beberapa hotel kelas menengah di Amerika Serikat melibatkan tindakan yang umum terjadi, yaitu melakukan reservasi hotel. Setelah reservasi dilakukan, penyerang mengirimkan email ke bagian reservasi hotel untuk memastikan kebenaran reservasi tersebut. Setelah mendapat balasan dari pihak hotel, penyerang akan merespons dengan menyertakan informasi seolah-olah mereka akan menginap di hotel dengan membawa anak kecil atau orang tua yang memiliki kondisi penyakit.

Bahkan, penyerang meminta sentuhan personal dengan menyatakan keinginan memberikan kejutan untuk anggota keluarga saat mereka melakukan check-in nanti. Esensinya, isi email tersebut dirancang untuk memberikan kesan empati atau urgensi kepada staf hotel.

Kegiatan ini dapat diulangi berulang kali untuk menciptakan "kedekatan" yang semakin terasa.

Begitu sudah terlihat nyaman dan lengah, si penyerang mulai mengirim link yang "katanya" berisi dokumen pribadi seperti rekam medis, surat dokter ataupun file-file lain yang akan membantu selama menginap. Bahkan si penyerang sampai memberikan password kepada staff hotel untuk membuka file-file pribadi tersebut. Ini memberi kesan kalau si penyerang sudah nyaman dan percaya pada staff hotel tersebut dan mengharapkan ada timbal balik rasa percaya.

Apabila link itu diklik oleh petugas hotel, otomatis sebuah infostealer malware akan terinstal secara diam-diam dan siap mencuri data-data yang ada pada sistem manajemen hotel.



SPIONASE

ZERO-CLICK EXPLOIT

Zero-Click adalah sebuah jenis serangan siber tanpa perlu mengarahkan korban untuk melakukan action seperti mengklik tautan atau membuka file.

Zero-click mengeksploitasi kerentanan pada sebuah aplikasi atau sistem operasi yang belum diperbaiki atau bahkan belum diketahui oleh pembuatnya untuk menginstal malware pada perangkat yang dituju. Teknologi zero-click dapat dianggap sangat canggih dan tidak banyak yang mampu membuatnya, sehingga memiliki nilai tinggi di pasar siber.

Itu sebabnya, korban dari serangan zero-click ini biasanya individu yang high profile seperti politisi, aktivis, pemerintah, petinggi perusahaan dan militer. Apalagi di tengah situasi politik yang memanas seperti perang ataupun Pemilu, penggunaan serangan ini untuk tujuan spionase menjadi lebih umum dan meresahkan.

Eksploitasi zero-click sangat umum terjadi pada Android dan iOS karena umum dipakai oleh banyak orang. Hal ini memudahkan penyerang melakukan serangan pada target individu yang diinginkan.

PEGASUS VS PREDATOR

Kalau membicarakan zero click, kita harus membahas tentang dua spyware papan atas buatan Israel yang selalu mengeksploitasi kerentanan ini di setiap serangannya, yaitu Pegasus dan Predator.

Pegasus dan Predator dijual secara resmi khusus kepada pemerintah negara untuk keperluan penegakan hukum, serta memantau pelaku kejahatan dan terorisme demi kepentingan masyarakat.

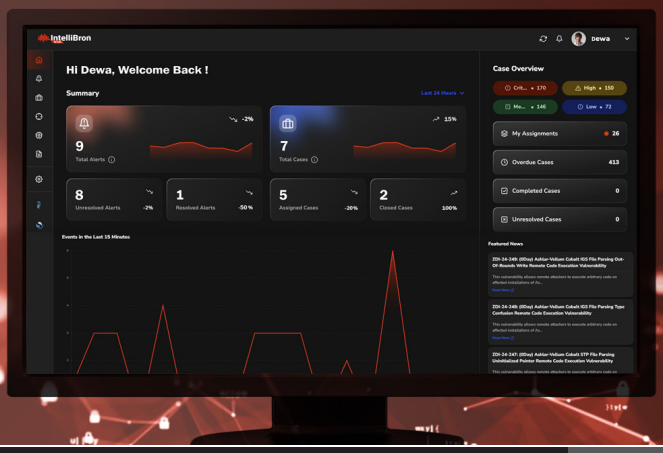
Setelah spyware ini terinstal pada sebuah ponsel, penyerang dapat mengakses kamera, mikrofon, GPS, daftar kontak, password, serta isi pesan dari aplikasi pesan, media sosial, dan email.

PEGASUS

Pegasus, yang dibuat oleh NSO Group dari Israel, pertama kali terdeteksi pada tahun 2016 ketika digunakan terhadap seorang aktivis di UEA (Uni Emirat Arab). Walaupun tujuan utama dari spyware ini adalah untuk penegakan hukum, seringkali disalahgunakan untuk memata-matai figur penting demi kepentingan pribadi ataupun politik. Banyak pihak yang melancarkan protes dan menuntut NSO Group untuk menghentikan operasional Pegasus. Akibatnya, dalam 2-3 tahun terakhir, NSO Group mengurangi aktivitasnya di pasaran.

PREDATOR

Predator adalah spyware yang dikembangkan oleh Cytrox, perusahaan berbasis di Macedonia Utara. Pada tahun 2019, Cytrox diakuisisi oleh Intellexa, milik Tal Dilian, seorang mantan intelijen Israel. Nama Tal sempat mencuat ketika dia memamerkan sebuan mobil van berisi peralatan yang bisa menyadap semua ponsel pada radius 1 kilometer. Fungsi Predator sangat mirip dengan Pegasus, dipasarkan ke pemerintah negara untuk penegakan hukum. Konon, harga spyware ini mencapai $10 Juta.



IntelliBroń Orion

PRODUK & SOLUSI

SOLUSI KEAMANAN SIBER UNTUK SEGMEN USAHA KECIL & MENENGAH

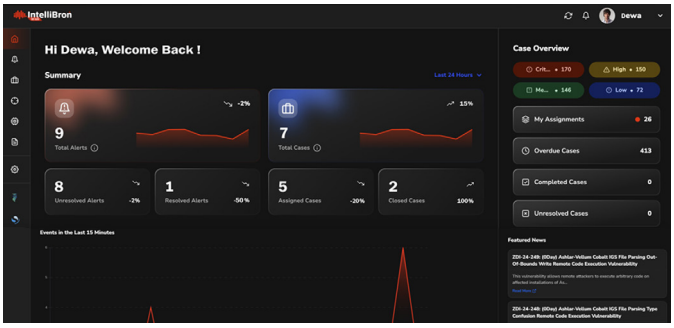
Di era digital ini semakin banyak layanan yang bisa diakses dari mana-mana. Ini menyebabkan area ancaman semakin luas dan dinamis. Jumlah serangan siber di Indonesia pada tahun 2023 naik dua kali lipat dibanding tahun sebelumnya. Kerugian yang dialami bisa bermacam-macam, mulai dari terganggunya operasional, kerugian finansial, tersebarnya informasi rahasia atau tercorengnya reputasi.

Memilih sistem keamanan siber yang tepat memerlukan pertimbangan berbagai faktor seperti aset, infrastruktur, skalabilitas, dan anggaran. Solusi untuk perusahaan besar belum tentu cocok atau memiliki fitur yang berlebih untuk sebuah perusahaan kecil atau menengah. Tantangan lain adalah merekrut tenaga ahli. Ini membuat sistem yang ramah pengguna dan mudah dipelajari menjadi pilihan ideal.

Melihat minimnya pilihan serta tantangan yang dimiliki segmen usaha kecil dan retail, ITSEC Asia meluncurkan IntelliBroń Orion, sistem deteksi dan respon yang dapat memenuhi kebutuhan esensial dalam memproteksi sistem sebuah perusahaan.

Threat Detection

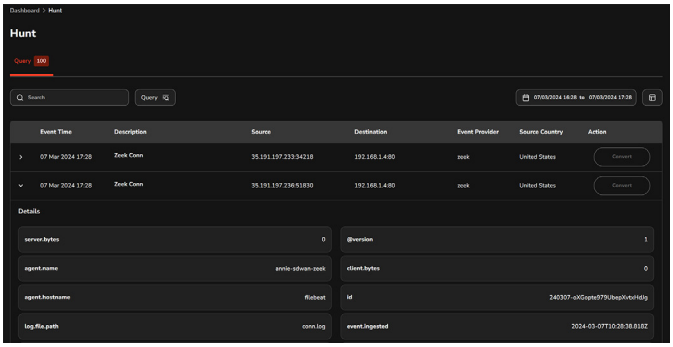
Dengan teknologi berbasis kecerdasan buatan (AI), IntelliBroń Orion dapat mendeteksi aktivitas mencurigakan dengan cepat dan efisien. Dengan bantuan machine learning, sistem ini dapat mempelajari pola serangan untuk mengidentifikasi ancaman dengan lebih akurat.



Dashboard Utama

Threat Hunting

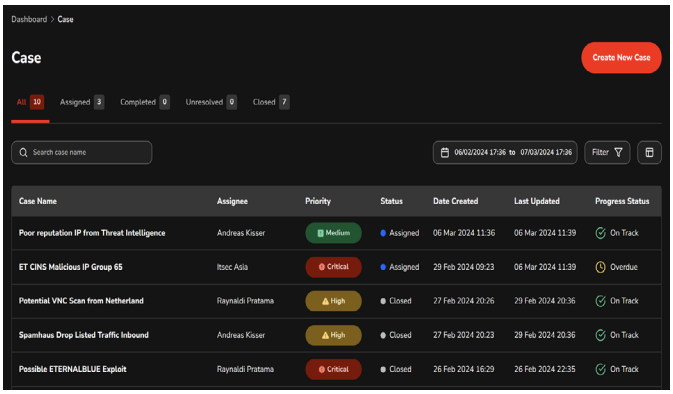
Fitur ini memungkinkan pengguna untuk mengeksplorasi data dan mengidentifikasi ancaman yang masuk ke dalam jaringan tanpa terdeteksi sebelumnya. Dengan kemampuan ini, tim keamanan dapat dengan mudah mencari tanda-tanda aktivitas yang mencurigakan.



Fitur Threat Hunt

Alert dan Ticketing Yang Terkonsolidasi

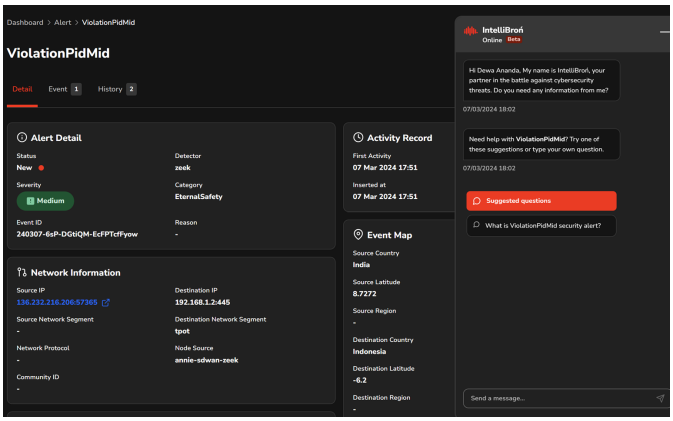
IntelliBroń Orion menyediakan notifikasi ancaman yang dapat disesuaikan dengan kebutuhan, serta fitur ticketing untuk membuat laporan ancaman agar memudahkan tim keamanan merespons dan menindaklanjuti ancaman yang terdeteksi.



Fitur Case Ticketing

Asisten Berbasis AI

Ini merupakan fitur chatbot yang membantu memberikan informasi dan saran dalam menangani kasus keamanan. Dengan fitur ini, pengguna dengan pengetahuan terbatas dalam keamanan siber juga dapat menggunakan IntelliBroń Orion dengan mudah.



Fitur Asisten Berbasis AI

Sensor Yang Praktis

Sensor Rigel, yang merupakan bagian dari IntelliBroń Orion, memiliki desain yang kecil dan mudah diintegrasikan dalam berbagai skala jaringan. Sensor ini bertugas mengumpulkan data dan menganalisis paket yang melewati jaringan untuk mendeteksi ancaman.

Sensor Rigel

ERA BARU PEMETAAN OTOMATIS SURICATA KE MITRE ATT&CK

Penulis: M.Haekal Al Ghifary &
Dicky Rahma Hermawan

Di zaman yang serba digital ini, keamanan jaringan adalah hal yang sangat penting untuk segala jenis organisasi. Diperlukan sistim untuk memonitor dan memproteksi komputer dan data di dalam jaringan organisasi agar tidak dicuri atau dirusak hingga menimbulkan kerugian. Salah satu sistem yang cukup mumpuni untuk mendeteksi serangan siber adalah Suricata, sebuah aplikasi open source yang berfungsi untuk memonitor lalu lintas dan intrusi jaringan.

Untuk meningkatkan efektivitas deteksi terhadap ancaman siber, Suricata kerap dipasangkan dengan MITRE ATT&CK, sebuah framework, menggambarkan taktik dan teknik para penjahat siber.

Dengan framework ini, para profesional keamanan siber dapat memahami dan mengembangkan strategi pertahanan terhadap ancaman siber. Apabila dipasangkan dengan MITRE ATT&CK tim keamanan dapat lebih mudah mengidentifikasi TTP (Taktik, Teknik, dan Prosedur) yang digunakan oleh penyerang, sehingga memungkinkan deteksi yang lebih cepat dan akurat.

Permasalahan muncul ketika memetakan aturan Suricata ke framework milik MITRE ATT&CK. Ada puluhan ribu aturan Suricata yang perlu dipetakan ke taktik dan teknik MITRE ATT&CK, apabila dilakukan secara manual dapat menjadi tantangan karena beberapa alasan.

Pertama, dengan banyaknya aturan Suricata, menyulitkan identifikasi pemetaan yang benar. Kedua, proses pemetaan memerlukan pengetahuan domain yang mendalam serta keahlian dalam keamanan jaringan, yang mungkin tidak dimiliki oleh organisasi. Ketiga, pemetaan manual adalah tugas yang membosankan dan berulang-ulang yang dapat menimbulkan kesalahan karena faktor manusia.

Mengotomatisasi proses pemetaan menggunakan algoritma pembelajaran mesin adalah solusi cocok dalam memecahkan permasalahan. Namun, mengembangkan algoritma pembelajaran mesin yang efektif memerlukan pertimbangan cermat terhadap beberapa faktor, termasuk data pelatihan dan algoritma yang digunakan serta metrik evaluasi untuk menilai performa.

Metodologi

Untuk melatih model, kami menggunakan set aturan milik ET Open, termasuk signature yang dikontribusikan oleh Proofpoint dan hasil riset dari komunitas. Set aturan ini dipilih karena mudah didapat dan bisa dibilang memiliki signature yang cukup banyak, 32.806 signature. Sayangnya baru 2.878 signature yang terlah dipetakan ke taktik dan teknik MITRE ATT&CK mewakili 11 teknik dan taktik. Signature yang tersisa tidak digunakan dalam proses pelatihan.

Tantangan dari data-data ini adalah ketidakseimbangan sebaran. Dua teknik terbanyak diwakili oleh 60% signature sedangkan dua teknik terbawah hanya diwakili oleh 0.65% signature. Dengan kondisi ini, besar kemungkinan dua teknik teratas akan sangat sering terdeteksi dan dua teknik terbawah akan sangat jarang karena minimnya signature.

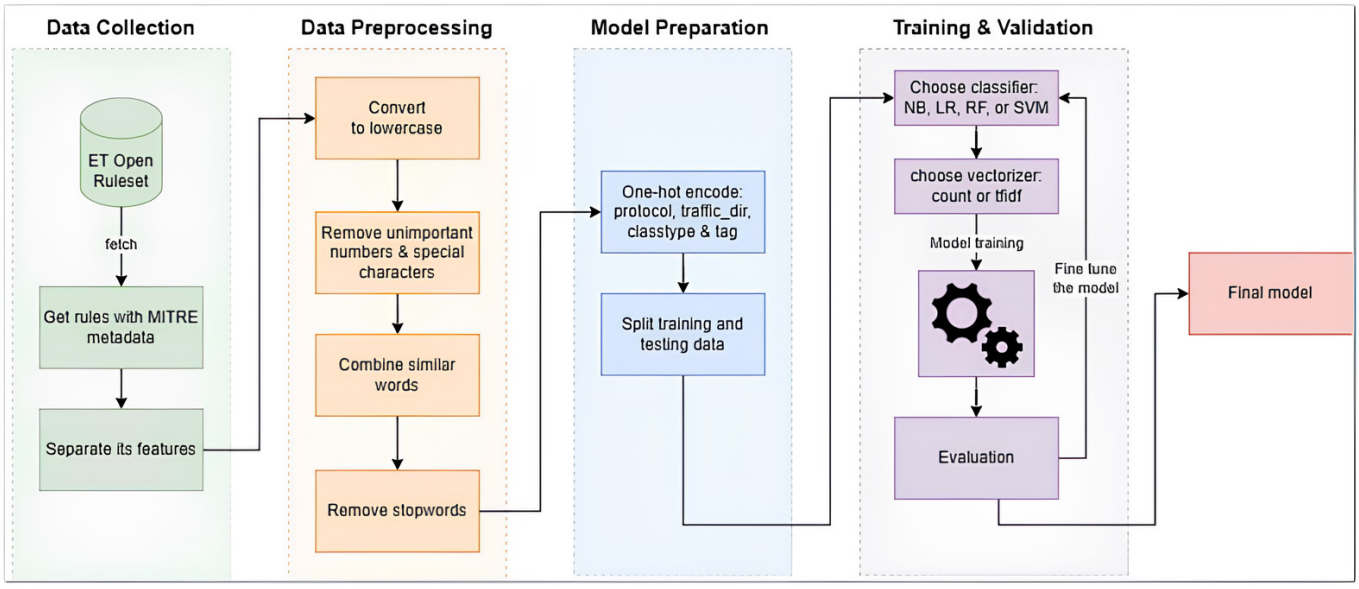
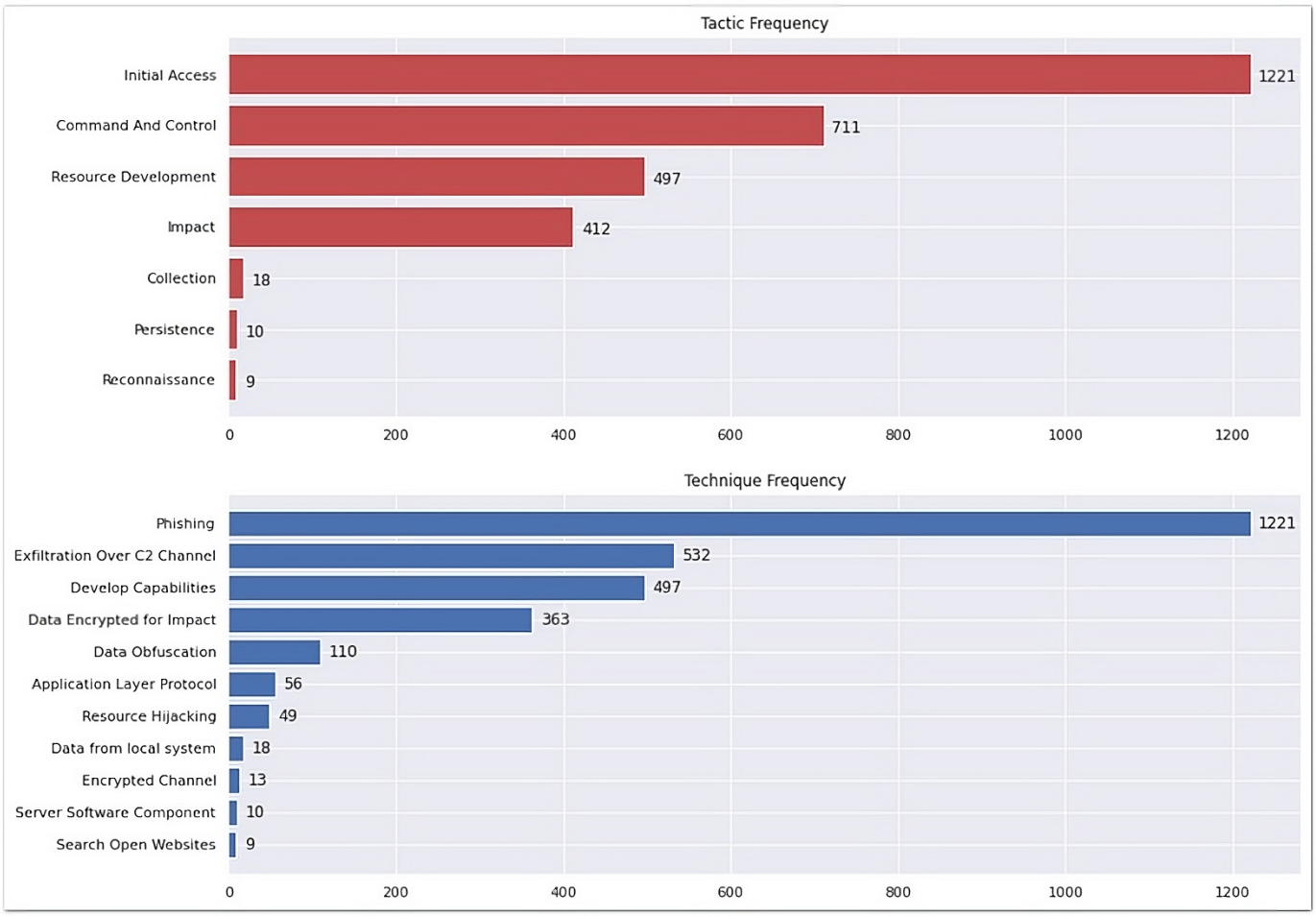


Diagram alur proses metodologi



Sebaran taktik dan teknik pada set aturan milik ET Open

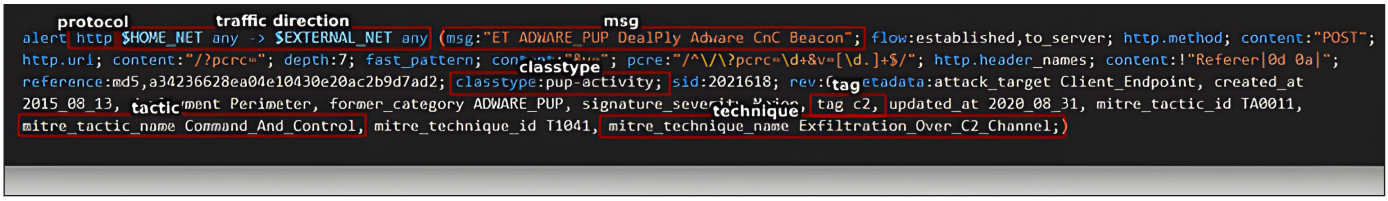
Fitur Pelatihan Model

Berikut adalah fitur yang dipilih dalam melatih model:

- Protokol - Untuk menentukan jenis protokol jaringan yang digunakan oleh jaringan, seperti TCP, UDP, ICMP. Dan juga protokol yang digunakan aplikasi seperti HTTP, FTP dan TLS.
- Arah traffic jaringan - Untuk menentukan arah traffic seperti masuk (ke HOME_NET) atau keluar (ke EXTERNAL_NET), mengidentifikasikan apakah traffic berasal dari penyerang atau korban.
- Msg - Berisi penjelasan singkat mengenai traffic yang memicu eksekusi aturan (rule).
- Classtype - Memberikan informasi mengenai klasifikasi aturan (rule) dan peringatan (alert).
- Tag - label yang diciptakan pengguna untuk mengkategorikan aturan (rule) sesuai kriteria spesifik.

Ada beberapa contoh dimana data tidak memiliki tag sama sekali. Tapi kami membiarkan kondisi tersebut dan memilih algoritma pembelajaran mesin yang dapat menangani situasi ini. Output dari model ini adalah taktik dan teknik MITRE ATT&CK yang sesuai dengan ancaman jaringan.

Untuk data yang memiliki tag, kami mengadopsi beberapa algoritma antara lain Naive Bayes, Random Forest, Logistic Regression, dan Support Vector Machines (SVM), dimana proses pembelajaran ini diawasi dan dievaluasi secara rutin.



Contoh aturan Suricata yang terpetakan ke Mitre ATT&CK

Pra-Pemrosesan Data

Proses awal sebelum pemrosesan data adalah mengubah data mentah menjadi format yang mudah digunakan oleh algoritma pembelajaran mesin. Tujuan dari prapemrosesan data adalah untuk menyiapkan data berkualitas yang dapat digunakan untuk melatih model pembelajaran mesin secara efektif, dengan menghilangkan noise dan inkonsistensi serta mengubah data ke dalam format yang mudah dipahami oleh algoritma.

Pelatihan Model

Setelah pra-pemrosesan data selesai, data dibagi menjadi dua set, set pelatihan dan set pengujian. Set pelatihan, yang mencakup 80% data, digunakan untuk melatih model, sedangkan 20% sisanya digunakan untuk mengevaluasi performa model. Pembagiannya dikelompokkan berdasarkan masing-masing teknik, memastikan bahwa distribusi kelas dalam set pelatihan dan pengujian serupa.

Mengenai pemilihan fitur, untuk fitur 'pesan', kami mempertimbangkan dua metode ekstraksi fitur: CountVectorizer dan TfidfVectorizer.

CountVectorizer menghitung frekuensi setiap kata dalam dokumen dan membuat matriks jumlah kata. TfidfVectorizer adalah singkatan dari Term Frekuensi-Inverse Document Frekuensi Vectorizer, yang juga menghitung frekuensi setiap kata dalam dokumen, namun juga memperhitungkan seberapa sering kata tersebut muncul. Kemudian skor akan diberikan pada se-tiap kata yang mencerminkan relevansinya dengan dokumen, dan skor yang lebih tinggi akan diberikan pada kata-kata yang lebih unik pada dokumen tersebut.

Untuk memastikan ekstraksi fitur yang optimal untuk skenario ini, kami memutuskan untuk mengevaluasi kinerja CountVectorizer dan TfidfVectorizer pada setiap model pengklasifikasi. Selain itu, kami memilih untuk menggunakan one-hot encoding untuk fitur lainnya (protokol, arah traffic, classtype & tag) karena dapat diterapkan ke variabel kategori tanpa urutan atau peringkat.

Metrik Evaluasi Model

Evaluasi kinerja merupakan langkah penting dalam menilai efektivitas model. Meskipun akurasi adalah metrik performa yang paling umum digunakan, akurasi bukanlah ukuran yang disukai untuk pengklasifikasi, terutama saat menangani kumpulan data yang tidak seimbang. Misalkan kita mempunyai pengklasifikasi yang hanya dapat menebak 4 dari 11 teknik teratas dengan benar. Jika kita mengevaluasi kinerja pengklasifikasi ini menggunakan akurasi, kita akan mendapatkan skor: ((1221+711+497+412)/2878) * 100 = 98.71%. Sekilas ini mungkin tampak seperti skor yang bagus, tetapi sebenarnya menyesatkan. Alasannya adalah kumpulan data tersebut sangat condong ke empat kategori teratas, yang memiliki jumlah kejadian jauh lebih tinggi dibandingkan dengan kategori lainnya. Oleh karena itu, meskipun akan mencapai skor akurasi yang tinggi, hal ini tidak akan berguna dalam praktiknya.

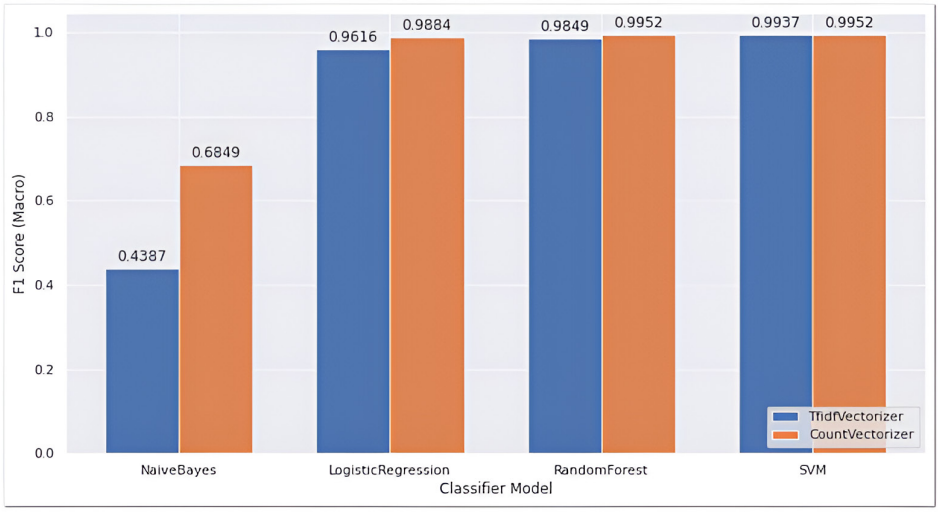
Dalam kasus seperti ini, cara yang lebih baik untuk mengevaluasi performa pengklasifikasi adalah dengan melihat matriks konfusi, yang memberikan informasi mendetail tentang performa model dalam bentuk positif sebenarnya, positif palsu, negatif sebenarnya, dan negatif palsu.

Selain itu, metrik lain seperti presisi, perolehan, dan skor F1 sering digunakan untuk mengevaluasi kinerja pengklasifikasi. Presisi mengukur proporsi positif sebenarnya di antara semua kejadian yang diberi label positif oleh pengklasifikasi.

Skor F1 dirancang untuk memberikan tingkat kepentingan yang sama terhadap presisi dan perolehan dengan menggabungkan keduanya ke dalam satu metrik. Hal ini membutuhkan presisi dan recall yang memiliki nilai tinggi agar skor F1 dapat meningkat. Skor F1 dapat dihitung menggunakan metode berbeda seperti F1-mikro dan F1-makro. F1-micro menghitung skor F1 secara global dengan menghitung total positif benar, negatif palsu, dan positif palsu, sedangkan F1-makro menghitung skor F1 secara independen untuk setiap kelas dan kemudian membuat rata-rata skor di semua kelas. Jadi, jika Anda sama-sama peduli terhadap setiap sampel, disarankan untuk menggunakan skor f1 rata-rata “mikro”; jika Anda sama-sama peduli terhadap setiap kelas, disarankan untuk menggunakan skor f1 rata-rata “makro”.

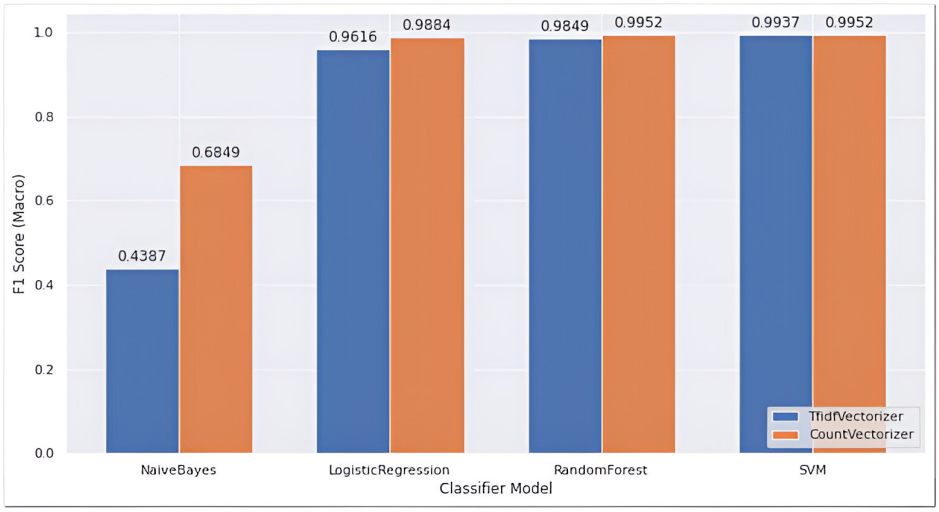
Lalu bagaimana dengan hasilnya?

Hasil dan Analisis



Grafik Model F1-Mikro

Dari grafik, terlihat bahwa CountVectorizer secara umum menghasilkan kinerja yang lebih baik untuk semua model F1-Micro. TfidfVectorizer menunjukkan hasil yang sebanding untuk model Random Forest dan SVM. Model SVM memiliki performa terbaik dari semua model dengan mencapai skor 0,9983 untuk kedua metode ekstraksi fitur. Selain itu, performa tiap model, setelah Naive Bayes, sangat kecil, yang menunjukkan bahwa performa model sangat bergantung pada kualitas fitur masukan.

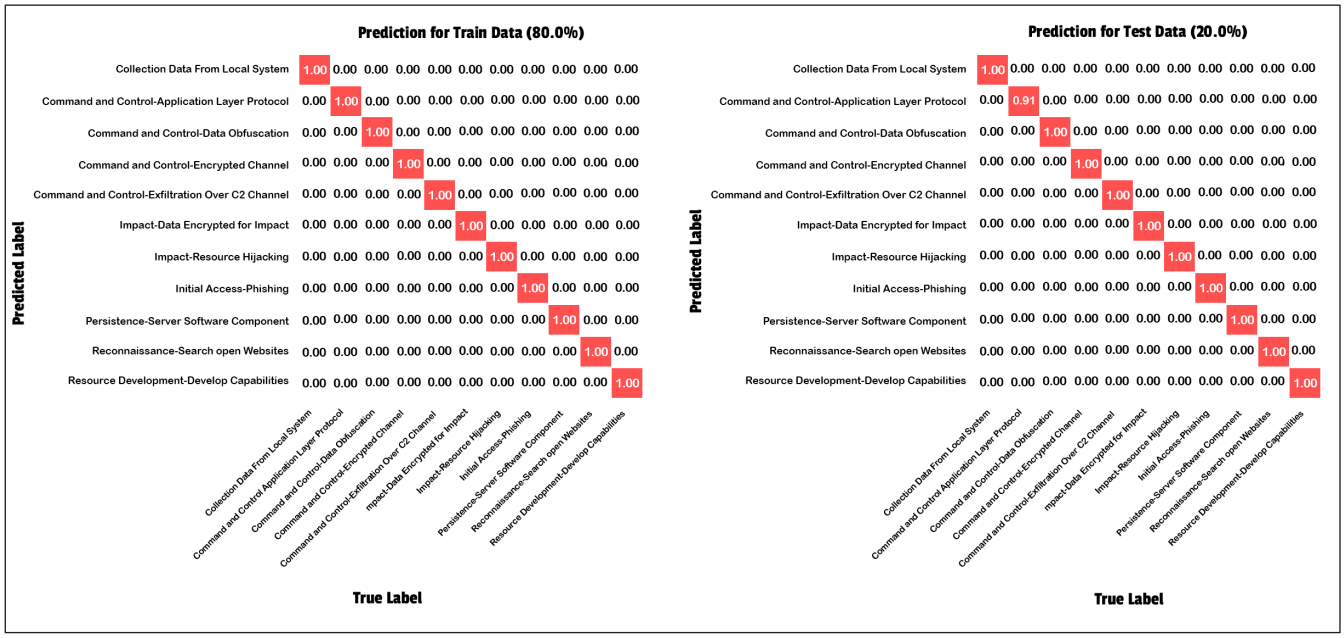


Grafik Model F1-Makro

Untuk F1-Makro, grafik mengikuti tren yang sama dengan skor F1-Micro, dengan CountVectorizer mengungguli TfidfVectorizer. Demikian pula, SVM memiliki skor tertinggi dari semua model untuk kedua metode pemilihan fitur. SVM dengan CountVectorizer mencapai skor tertinggi dengan 0,9952, menjadikannya pengklasifikasi dan pemilihan fitur yang ideal untuk digunakan sebagai model.

Hasil Klasifikasi dalam Matriks Konfusi

Matriks konfusi di bawah memberikan representasi visual dari prediksi akurat dan tidak akurat berdasarkan model yang dipilih.



Matriks Konfusi

Diagram kiri menunjukkan performa model pada data pelatihan, dan yang kanan menunjukkan performa pada data pengujian. Secara keseluruhan, model tersebut tampaknya mampu memprediksi dengan tepat semua data yang dilihatnya selama fase pelatihan, sebagaimana dibuktikan dengan nilai diagonal yang tinggi di bagian kiri matriks. Performa model pada data pengujian, walaupun belum sempurna, bisa dibilang cukup baik.

Melihat matriks konfusi untuk data pengujian, kita dapat melihat bahwa model tersebut berkinerja sangat baik dalam memprediksi sebagian besar kelas. Namun, tampaknya model tersebut masih kesulitan membedakan dengan benar antara dua teknik 'Protokol Layer Aplikasi' dan 'Eksfiltrasi Melalui Channel C2'. Hal ini menunjukkan bahwa penyempurnaan fitur lebih lanjut dan data pelatihan tambahan dapat membantu meningkatkan performa model dalam membedakan kelas-kelas ini.

Keterbatasan

Ada beberapa batasan dari temuan ini yang perlu disebutkan. Pertama, kumpulan data yang digunakan dalam penelitian ini relatif kecil, dan beberapa sampel berukuran sangat kecil, sehingga tidak optimal untuk pengujian, sehingga membatasi kemampuan generalisasi hasil.

Kedua, langkah-langkah pra-pemrosesan yang digunakan cukup kasar dan perlu perbaikan. Perlu diperhatikan bahwa kualitas data bisa dibilang merupakan hal terpenting dalam melatih model pembelajaran mesin. Seperti kata pepatah, "sampah masuk, sampah keluar." Oleh karena itu, penelitian di masa depan harus mempertimbangkan teknik pra-pemrosesan yang lebih canggih untuk memastikan bahwa data berkualitas tinggi dan cocok untuk melatih model pembelajaran mesin.

Ketiga, algoritma yang lebih kuat dapat digunakan dan diuji. Meskipun model yang digunakan dalam penelitian ini memiliki kinerja yang cukup baik, mungkin ada algoritma lain yang dapat mencapai tingkat akurasi yang lebih tinggi. Penelitian di masa depan dapat mengeksplorasi alternatif-alternatif ini untuk mengidentifikasi algoritma yang paling efektif untuk tugas ini.

Keempat, fitur-fitur lain yang lebih sesuai dapat dipertimbangkan sebagai masukan. Ada kemungkinan bahwa beberapa fitur yang relevan tidak disertakan dalam penelitian ini dan fitur lainnya mungkin lebih efektif untuk tugas ini. Oleh karena itu, penelitian di masa depan harus mengeksplorasi rangkaian fitur alternatif untuk menentukan fitur mana yang paling berguna untuk mendeteksi ancaman keamanan jaringan. Secara keseluruhan, keterbatasan dan penelitian di masa depan menyoroti perlunya penelitian lanjutan di bidang ini untuk meningkatkan akurasi dan efektivitas deteksi ancaman keamanan jaringan berbasis pembelajaran mesin

Terakhir, menggabungkan keahlian manusia dan feedback akan bermanfaat untuk membantu model belajar dan beradaptasi terhadap jenis serangan baru. Performa model ini juga bisa dipengaruhi oleh perubahan lanskap ancaman, sehingga pemantauan dan pembaruan secara kontinyu diperlukan untuk menjaga efektivitasnya. Secara keseluruhan, model pembelajaran mesin menunjukkan harapan untuk memetakan aturan Suricata ke MITRE ATT&CK, namun pengembangan dan pengujian harus terus berlanjut untuk mengevaluasi potensi sepenuhnya.

Kesimpulan

Temuan ini menyoroti pentingnya penggunaan pembelajaran mesin dan kerangka kerja seperti MITRE ATT&CK untuk meningkatkan deteksi dan klasifikasi ancaman keamanan siber. Kami berharap temuan ini memberikan wawasan dan manfaat berharga bagi komunitas keamanan informasi. Dengan memetakan aturan Suricata ke taktik dan teknik MITRE ATT&CK, analis keamanan dapat memperoleh pemahaman yang lebih baik tentang serangan yang mereka hadapi dan mengambil langkah proaktif untuk melindungi organisasi mereka. Penggunaan algoritme pembelajaran mesin dapat meningkatkan kemampuan deteksi dan memberikan intelijen ancaman yang lebih akurat dan handal.

KEAMANAN DATA

DATA BOCOR, HARUS BAGAIMANA?

Penulis: Z.Ananda

Apa sih yang dilakukan si pencuri pada data yang telah mereka ambil? Jawabannya, tergantung dari datanya. Yang pasti, banyak pihak yang membutuhkan data tersebut.

Kebocoran data adalah sebuah hal yang makin sering kita dengar beberapa tahun belakangan ini. Ini diakibatkan oleh kelalaian pemilik dalam mengamankan data. Banyak juga yang tidak mengerti kalau data itu sesuatu yang berharga apabila jatuh ke orang yang tepat, entah itu digunakan untuk kejahatan atau memang sebagai data pendukung dalam mengambil keputusan.

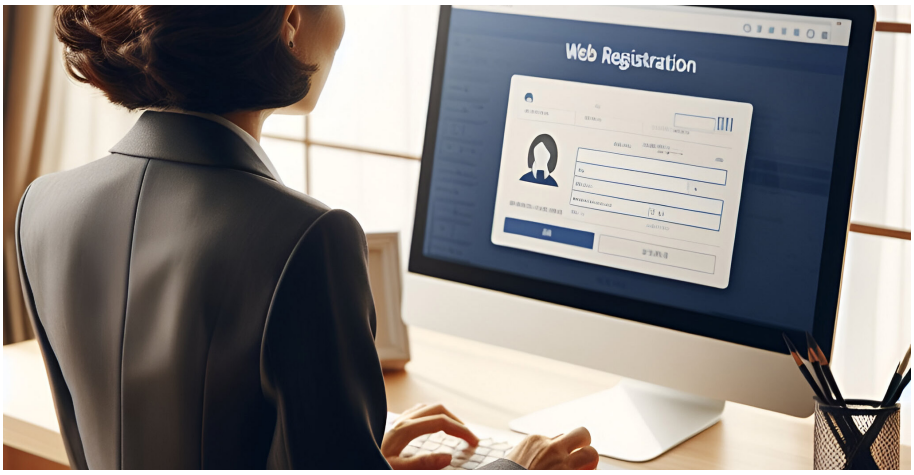
Sayangnya, banyak orang yang tidak tahu, tidak peduli, atau mungkin tidak ada pilihan untuk menyerahkan data dan meninggalkan jejak digital dimana-mana. Jejak yang kalau diurunut satu persatu bisa membuka semua data pribadi kita, hingga data yang kita tidak pernah share.

Data NIK atau KTP

Data ini yang sangat sering kita berikan. Sepertinya sekarang, kalau ada apa-apa pasti minta data tersebut. Kalau sampai data satu KTP bocor, bisa dipakai untuk mencuri benda lain yang atas nama si pemilik KTP tersebut. Bisa juga dipakai untuk membuat KTP palsu lalu menggunakan penipuan mengatasnamakan si pemilik KTP sebenarnya. Apabila hanya NIK yang bocor, kalau memang kepo, si pencuri bisa mendapatkan data kita lainnya hanya dengan melakukan pencarian menggunakan NIK tersebut. Seperti kita tahu, ada website-website resmi yang mengizinkan kita melakukan pencarian berdasarkan NIK yang kita masukkan.

Data Registrasi Online

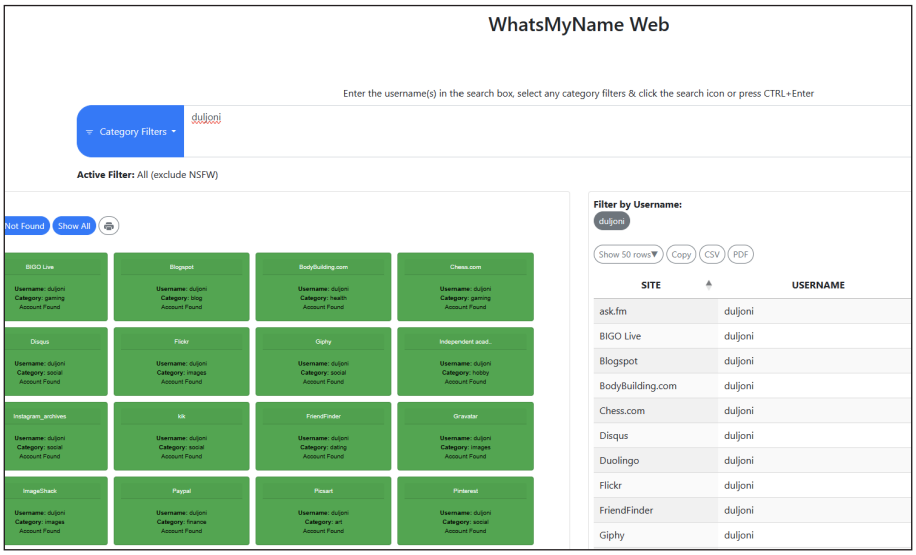
Data yang dimaksud adalah data-data yang kita masukkan ketika melakukan registrasi pada sebuah jasa online seperti e-commerce, webmail, dll. Kalau sebatas data pribadi seperti nama, alamat, nomor telepon atau riwayat order, kemungkinan besar data tersebut akan dijual ke data agregator/broker, pihak yang membutuhkan data untuk profiling. Ada kemungkinan juga data kita dipakai untuk melakukan penipuan mengatasnamakan pemilik data tersebut tapi si pencuri harus mencari data lebih lengkap lagi. Balik lagi, seberapa kepo.



Apabila data kartu kredit atau password yang bocor, dampak nya akan jauh lebih besar. Untuk kartu kredit, sudah banyak merchant yang mewajibkan 2FA dalam melakukan transaksi. Jadi kemungkinan menyalahgunakan kartu debit/kredit akan kecil walaupun masih ada merchant yang bisa melakukan transaksi tanpa OTP. Jadi tetap, apabila data kartu kredit bocor, kita harus meminta pemblokiran kartu agar tidak dipakai oleh pencuri tersebut.

Kesalahan terbesar pemilik akun adalah menggunakan password yang sama untuk setiap akun yang dia miliki. Andai terjadi kebocoran password, pada satu situs maka si pencuri bisa mencari akun-akun lain yang memiliki username yang sama lalu mencoba masuk dan melakukan kejahatan.

Pada website whatsmyname.app, kita bisa mencari jejak digital kita berupa akun yang telah kita (atau orang lain?) buat menggunakan email yang kita miliki atau username yang sering kita gunakan. Ambil contoh akun "duljoni", begitu kita cari, akan keluar list akun di situs-situs populer dengan username "duljoni". Kebayang kalau satu password bocor dan si pemilik akun selalu menggunakan password yang sama maka kemungkinan akun-akun yang lain akan bocor juga.



Hasil Pencarian Whatsmyname.app

Kita sebagai pemilik harus pintar mengamankan data kita. Dimulai dari proses registrasi, kita harus memutuskan apakah **WAJIB** memasukkan data kita sedetail mungkin. Kita harus memilah data apa saja yang memang wajib dan mau kita bagikan. Kalau kita membicarakan keamanan password, pastikan untuk selalu menggunakan password yang berbeda-beda dan juga mengaktifkan 2FA. Coba mengontrol apa yang bisa kita kontrol. Begitu kita menyerahkan data kita ke orang lain, kontrol sudah berpindah tangan. Kita berharap mereka pun juga mengamankan data kita dan juga pelanggan lainnya sebaik mungkin. Yang bisa kita lakukan adalah secara rutin mengganti password lalu mengecek apakah data kita bocor.

3 FITUR BERBAHAYA ANDROID

Android adalah sistem operasi telepon selular dengan jumlah pengguna terbanyak, sekitar 2,5 miliar. Seiring berjalannya waktu, tingkat keamanan sistem operasi ponsel ini semakin meningkat. Hal ini terbukti dari penurunan jumlah serangan siber pada ponsel, yang mencapai sekitar 2 juta serangan pada tahun 2022.

Dapat dikatakan bahwa setelah ponsel dikeluarkan dari kotak dan dinyalakan, pengaturan dari sistem operasi ini sudah aman. Namun, ketika pengguna mulai melakukan penyesuaian terhadap pengaturan telepon genggam, tindakan tersebut justru dapat membuka peluang bagi para hacker untuk memanfaatkan tingkat keamanan yang menurun akibat perubahan setting yang dilakukan.

Pada artikel ini kita akan membahas tiga fitur atau pengaturan pada Android OS yang sangat berbahaya jika dirubah atau diaktifkan oleh pengguna ponsel.

1. Accessibility

Fitur Accessibility ini sangat berguna, terutama bagi pengguna disabilitas, seperti masalah penglihatan. Pengguna dengan disabilitas ini memerlukan aplikasi yang mampu menerima perintah suara dan membacakan teks yang terdapat di layar. Fitur ini akan aktif apabila pengguna memberikan izin kepada aplikasi untuk mengakses dan "melihat" aplikasi lain yang sedang aktif.

Masalah akan muncul apabila pengguna mengizinkan pemakaian fitur ini pada aplikasi yang ternyata sebuah spyware. Dengan memberikan izin pada spyware untuk melihat aplikasi-aplikasi lain yang sedang aktif atau melakukan perintah dengan suara, maka menimbulkan kerugian bagi pengguna ponsel tersebut.

Itu sebabnya kita harus berhati-hati dalam memberikan akses pada aplikasi yang tidak diketahui keamanannya.

2. Rooting (Jailbreak)

Fitur ini adalah fitur yang jarang diaktifkan. Jika aktif, fitur ini memberikan pengguna kemampuan untuk mendapat akses secara menyeluruh pada sistem file di ponsel. Apabila memiliki akses penuh, pengguna bisa melakukan modifikasi apa saja pada ponselnya. Biasanya pengguna melakukan berbagai modifikasi di ponsel untuk mengatur akses jaringan pada aplikasi yang diinginkan, mempercepat kerja chipset atau menghapus aplikasi bawaan yang tidak digunakan.

Bahayanya, jika ponsel sampai terinfeksi malware, maka malware tersebut bisa memiliki akses penuh dan melakukan lebih dari biasanya.

Menurut statistik, di tahun 2017, Indonesia sempat masuk ke dalam daftar 10 besar jumlah pengguna rooting. Untungnya angka itu semakin membaik dan menurut data Kaspersky 2022, Indonesia sudah terlempar dari daftar 10 teratas.

3. Install Unknown Apps

Secara mendasar, Android memiliki toko aplikasi resmi untuk menginstal aplikasi, yaitu Google Play Store. Meskipun demikian, Android memungkinkan pengguna untuk menginstal aplikasi melalui cara lain di luar Play Store, contohnya melalui Samsung Galaxy Store bagi pengguna ponsel merk Samsung serta AppGallery bagi pengguna Huawei. Bahkan pengguna Android dapat menginstal aplikasi jika memiliki file installer, yaitu file .apk. Fitur yang mengizinkan pengguna untuk menginstal aplikasi selain melalui Play Store ini adalah fitur Install Unknown Apps.

Permasalahan utamanya adalah bahwa app store lain cenderung tidak memberikan perhatian yang cukup terhadap aplikasi yang terdaftar di platform mereka, sehingga meningkatkan risiko aplikasi yang terinfeksi malware. Terutama jika pengguna menginstal aplikasi melalui file .apk yang diperoleh dari sumber yang tidak jelas. Fitur Install Unknown Apps ini biasanya sudah aktif secara default. Jadi, pengguna justru disarankan untuk menonaktifkan fitur tersebut.

RISET

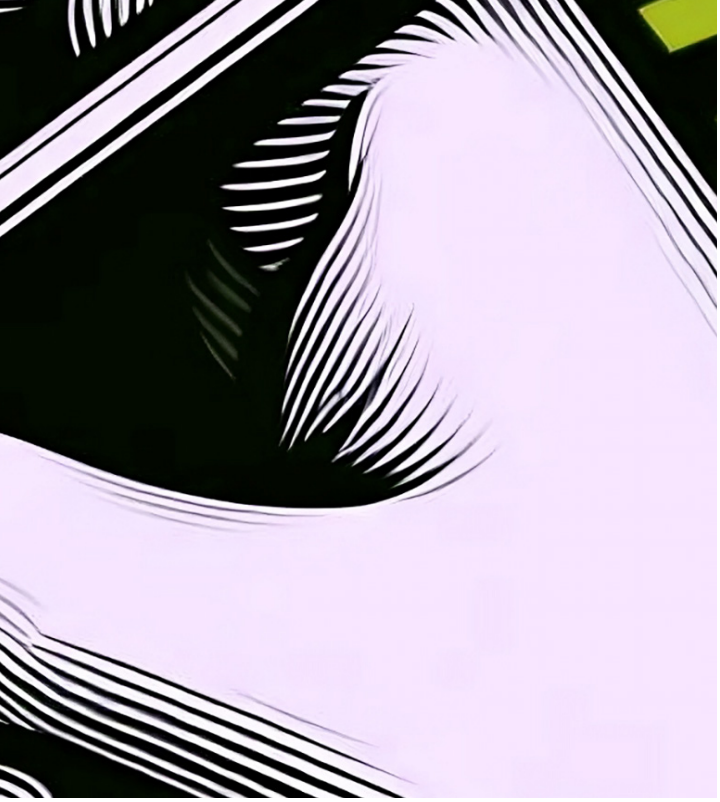
TANDA TANGAN DIGITAL DALAM BAHAYA: EKSPLOITASI PADA FILE PDF

Penulis: M.Rasyid. Sahputra

Sebuah riset yang dilakukan oleh Universitas Ruhr Bochum, Jerman menemukan eksploitasi yang disebut sebagai 'Shadow Attack'. Eksploitasi ini dapat mengubah isi suatu dokumen PDF yang telah ditandatangani digital (digital signature). Ini menjadi sangat menarik di era New Normal saat ini, dimana penggunaan tanda tangan elektronik dan digital semakin meningkat, terutama di Indonesia. Pada tulisan ini, saya akan menjelaskan kerentanan pada tanda tangan digital pada file PDF yang ditemukan para periset tersebut.

Keberhasilan eksploitasi tanda tangan digital memiliki dampak yang sangat signifikan. Aspek legalitas tanda tangan digital diakui secara hukum oleh berbagai pihak di berbagai negara. Oleh karena itu, jika tanda tangan digital berhasil dieksploitasi dan dipalsukan, banyak pihak dapat dirugikan.

Perlu diingat, tanda tangan digital berbeda dengan tanda tangan elektronik yang berupa tanda tangan yang dilakukan dengan pena digital atau hasil scan dari tanda tangan basah yang ditempelkan pada sebuah dokumen digital. Tanda tangan digital tidak memiliki bentuk visual seperti tanda tangan konvensional, melainkan berupa sertifikat yang mengidentifikasi pemilik tanda tangan dan kunci digital untuk membuka dan memverifikasi dokumen yang telah ditandatangani.



Fun Fact

Di Indonesia, saat artikel ini ditulis, hanya ada 10 perusahaan yang diakui pemerintah dalam menerbitkan sertifikat tanda tangan digital. Tiap perusahaan memiliki proses sendiri dalam memvalidasi sebuah tanda tangan.

Kunci digital adalah bagian dari PKI (Public Key Infrastructure), yang berfungsi untuk mengenkripsi dokumen dan menerbitkan dua jenis kunci. Private key digunakan untuk menandatangani dan mengenkripsi dokumen, sementara public key digunakan oleh penerima dokumen untuk memastikan validitas dokumen melalui proses dekripsi.

Contoh di dunia nyata:

- Doni hendak mengirim dokumen rahasia ke Lea.
- Doni menggunakan private key dan menandatangani dokumen tersebut.
- Doni mengirim dokumen tersebut ke Lea.
- Lea menerima dokumen tersebut dan menggunakan public key Doni untuk memverifikasi tanda tangan.
- Apabila dokumen berhasil dibuka dan tanda tangan tervalidasi, Lea bisa yakin bahwa dokumen itu memang benar dari Doni.

Mungkin pembaca bertanya, bagaimana Lea bisa mendapatkan public key milik Doni. Nah, ada beberapa cara yang dapat dilakukan, tergantung proses pertukaran dokumen dilakukan:

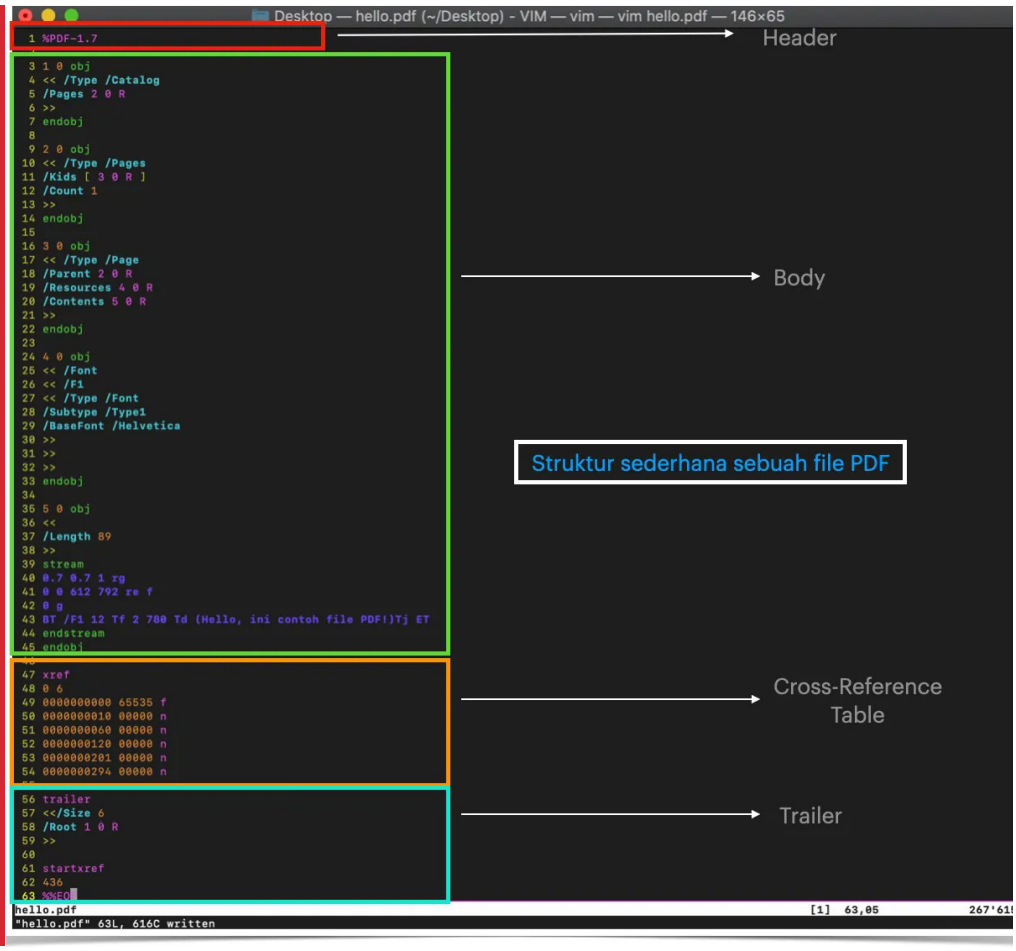
1. Melalui email yang support OpenPGP melalui opsi pencarian public key.
2. Distribusi manual, dikirim lansung oleh Doni ke Lea.
3. Menggunakan CA (Certificate Authority) yang menerbitkan sertifikat digital dan memastikan kunci digital valid dan bisa dipakai oleh Lea.

Tanda Tangan Digital Pada PDF

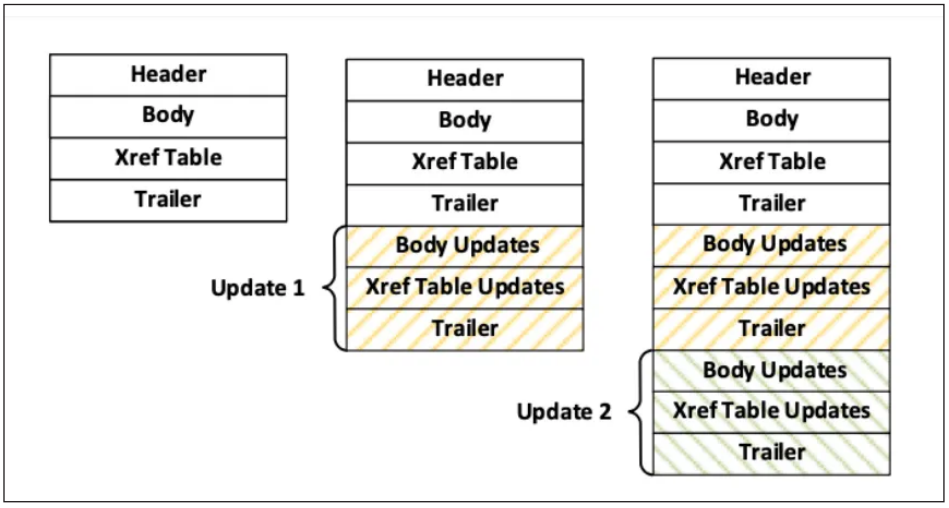
Salah satu jenis dokumen yang dapat ditempelkan tanda tangan digital adalah file PDF (Portable Document Format). File PDF memiliki struktur sederhana yang terbagi menjadi empat bagian: header, body, cross-reference table (Xref), dan trailer. Bagian header berisi informasi tentang file PDF, sementara bagian body berisi semua konten seperti isi, halaman, font, dan elemen lainnya.

Isi dari file PDF dapat mencakup berbagai objek seperti gambar, elemen multimedia, dan tanda tangan digital. Fungsi Xref adalah sebagai penunjuk lokasi objek-objek tersebut. Ini memudahkan pembaca menemukan objek untuk menemukan obyek dengan cepat tanpa harus membaca seluruh isi file. Sementara itu, bagian trailer pada PDF berisi informasi tentang file, seperti checksum dan ukuran file.

Struktur sederhana sebuah file PDF, terdiri dari Header, Body, Cross-Ref Table dan Trailer



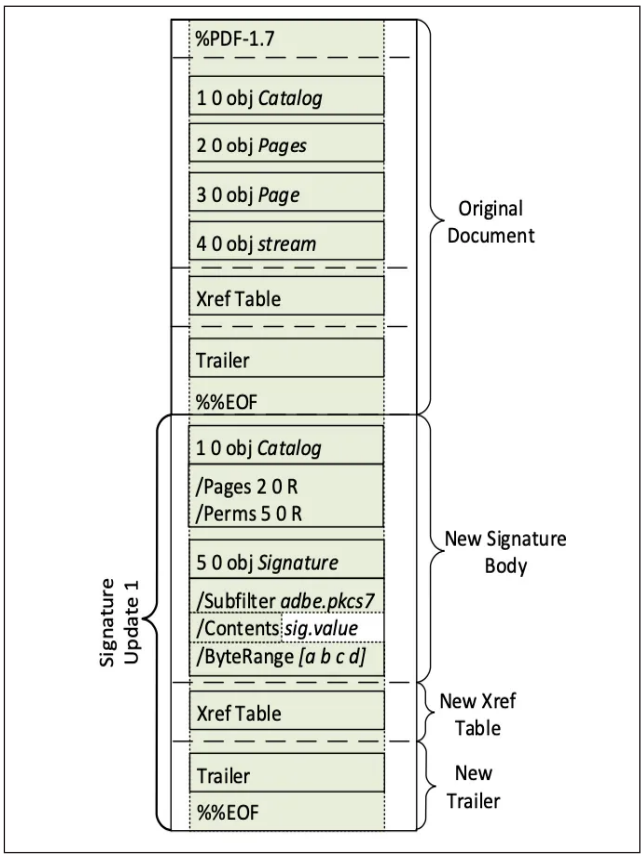
PDF mengadopsi sistem incremental update dimana, setiap ada update atau penambahan obyek, tidak mengubah struktur awal dari file, tetapi ditambahkan pada bagian akhir dengan nama xxx Updates.



Incremental Update pada file PDF

Pada gambar diatas, tabel 1 mengambarkan struktur awal file pdf. Namun, ketika ada penambahan objek atau konten baru, objek atau konten tersebut akan diletakkan di struktur tambahan pada bagian akhir dari struktur awal (tabel 2). Obyek atau konten yang tidak dipakai lagi tidak akan dihapus dari file tapi akan ditandai dengan flag. Jika terjadi perubahan lagi, maka akan ada penambahan struktur update tanpa menghapus struktur update sebelumnya (tabel 3)

Penambahan tanda tangan digital juga menggunakan fitur incremental update di mana tanda tangan akan ditambah setelah isi file dibuat. Gambar selanjutnya bisa menjelaskan bagaimana struktur file berubah setelah adanya penambahan tanda tangan digital pada sebuah file PDF.



Ilustrasi penambahan tanda tangan digital pada file PDF

Pada gambar di samping, terdapat penambahan bagian Catalog baru yang di dalamnya didefinisikan objek baru (disebut Signature). Di dalam objek Signature, terdapat bagian content yang akan diisi oleh tanda tangan digital dan informasi lain, seperti byte range.

Aplikasi pembaca file PDF memvalidasi keabsahan tanda tangan digital dengan metode berbeda-beda karena PDF tidak mendefinisikan cara validasi secara spesifik sehingga para developer aplikasi PDF reader menggunakan metode ataupun library sesuai logic implementasi mereka masing-masing.

Eksploitasi File PDF

Kembali ke awal artikel mengenai Shadow Attack, dimana penyerang memanfaatkan kerentanan pada file PDF. Ide dasar dari shadow attack adalah menciptakan dua konten berbeda pada satu file PDF. Konten yang awalnya bisa dilihat oleh penanda tangan akan diubah menjadi satu konten lain yang dilihat oleh penerima dokumen. Perubahan konten ini pun ternyata tidak bisa sembarangan. Merubah dengan cara mengganti kata-kata ternyata bisa terdeteksi sehingga ketika dokumen dibuka, validasi tanda tangan digital akan error.

Lalu perubahan apa yang bisa lolos?

Sembunyi

Cara ini dipakai untuk menyembunyikan overlay object seperti foto. Salah satu caranya adalah dengan mengubah tipe objek pada Xref dari gambar menjadi metadata, yang pada file PDF tidak ditampilkan ketika file dibuka.

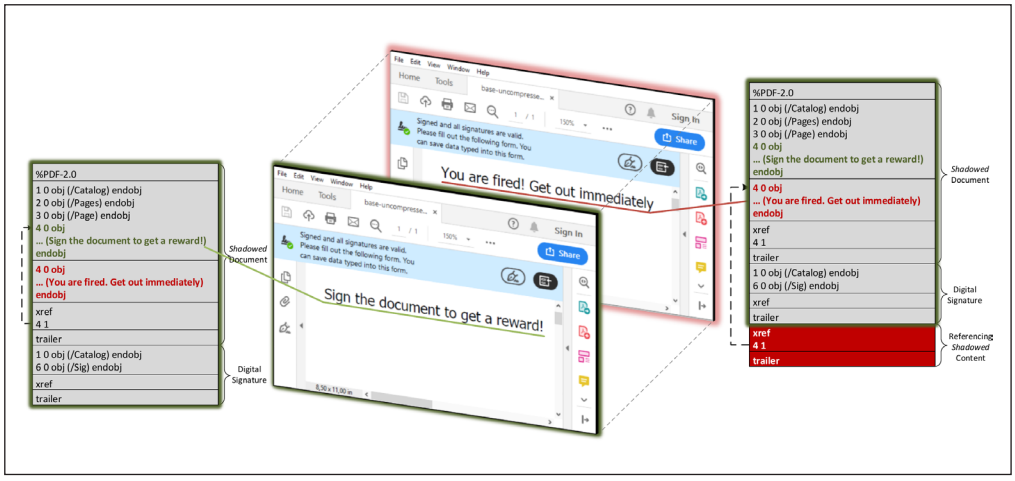
Ganti

Seperti penjelasan di atas, mengubah kata, gambar, atau objek lain pada sebuah file PDF yang sudah memiliki tanda tangan digital akan menyebabkan eror pada file ketika dibuka. Tapi ada juga yang tidak terdeteksi, misalnya mengganti jenis font. Mengubah font Arial menjadi Calibri tentu tidak menjadi masalah. Namun, jika font diubah menjadi font yang telah dimodifikasi, misalnya mengubah karakter yang seharusnya angka 3, tapi di font itu menjadi angka 5, ini tidak akan terdeteksi.

Sembunyi & Ganti

Pada cara ini, penyerang biasanya sudah mempersiapkan dokumen sejak awal. Caranya dengan membuat satu dokumen yang memiliki konten yang disembunyikan. Setelah konten diberi tanda tangan digital, konten yang ditampilkan adalah konten yang awalnya disembunyikan.

Ilustrasi di bawah menunjukkan dua objek dengan ID yang sama. Salah satu objek ditampilkan, sementara objek lain disembunyikan. Setelah dokumen diberi tanda tangan digital, isi Xref diubah dengan menukar referensi objek ID ke objek yang tadinya disembunyikan. Setelah referensi bertukar, berubah pula konten yang ditampilkan.



Dari ketiga cara eksploitasi di atas, mana yang paling sering berhasil? Jawabannya tergantung dari aplikasi PDF reader korban. Seperti yang telah dijelaskan sebelumnya, setiap developer aplikasi PDF reader memiliki cara masing-masing dalam membaca file PDF. Agar bisa berhasil, penyerang harus tahu aplikasi apa yang digunakan oleh korban sebelum menentukan tipe eksploitasi yang dipakai. Bukan tidak mungkin, eksploitasi ini sudah ada solusinya pada update aplikasi terbaru, maka sudah menjadi tugas kita untuk selalu update aplikasi.

PENERAPAN DEVSECOPS DALAM PENGEMBANGAN INTELLIBRON
(Part 1)

Penulis: M. Akmal

Saat ini, produk dan layanan TI telah menjadi komoditas yang tak bisa dipisahkan dari manusia baik itu untuk hiburan, produktivitas ataupun pendidikan. Tapi apapun jenisnya, setiap produk atau layanan memerlukan waktu yang panjang pada proses pembuatan serta memformulasikan metode pengembangan yang tepat dan efektif.

Fase pembuatan IntelliBroń oleh divisi R&D ITSEC Asia dimulai pada akhir 2022. Proses ini dimulai dengan ideasi, penyusunan visi, dan lanjut ke tahap implementasi. Seiring berjalannya waktu, jumlah anggota tim juga terus berkembang, sehingga memerlukan metode yang efektif demi kelancaran pengembangan produk.

Dalam pengembangan IntelliBroń, metode yang digunakan adalah DevSecOps. Penerapan metode ini bukan hal yang bisa dilakukan dengan sekedar copy-paste karena setiap produk memiliki kebutuhan yang berbeda-beda.

Ada banyak faktor yang membuat setiap pengembangan produk berbeda-beda, seperti teknologi yang digunakan, model bisnis, anggota tim, dan lingkungan operasional. Oleh karena itu, diperlukan pendekatan DevSecOps yang paling sesuai kebutuhan.

Pada tulisan ini, kami ingin berbagi pengalaman dalam mengimplementasikan DevSecOps. Semoga ini bisa memberikan wawasan baru bagi yang ingin mengubah metode pengembangan produk dari tradisional menjadi DevOps atau DevSecOps.

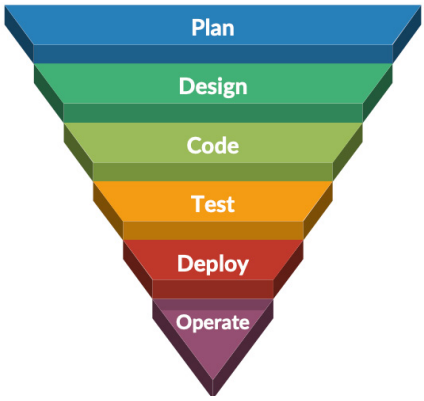
Sebelum dimulai, ada baiknya untuk menjelaskan atau mengingatkan kembali beberapa metodologi yang umum dipakai dalam pembuatan aplikasi.

Metodologi Waterfall

Seperti namanya, proses dalam metodologi ini dilakukan secara berurutan dari perencanaan hingga operasi. Selama fase perencanaan (plan), kemampuan produk atau layanan dalam mencapai tujuan tertentu dinilai berdasarkan berbagai kriteria. Setelah itu, kita masuk masuk ke fase desain (design), di mana produk atau layanan mulai dirancang. Kemudian, proses dilanjut ke fase kode (code), di mana proses penulisan code dimulai, diikuti dengan beberapa proses pengujian selama di fase uji (test) untuk memastikan bahwa kode berfungsi sesuai kebutuhan.

Jika semua code lulus uji, kita masuk ke tahap implementasi (deployment), sehingga produk dan layanan yang telah dibuat bisa digunakan oleh pengguna. Ini disebut fase operasi (operate).

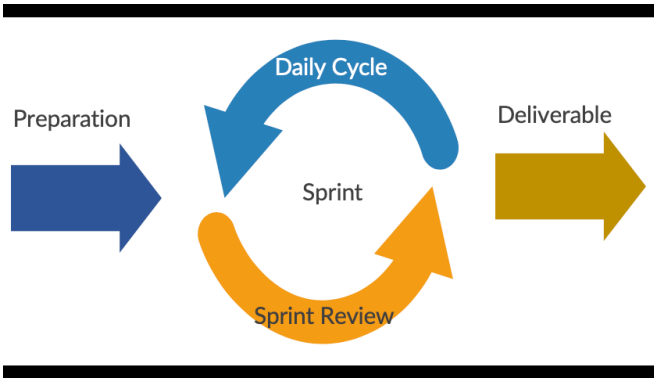
Namun, meskipun metode ini menerapkan proses yang berurut, ketika ada perubahan kebutuhan saat pengembangan, termasuk perubahan bisnis, proses ini harus diulang dari awal. Kondisi ini membuat proses pengembangan menjadi lebih lama sehingga biaya yang dibutuhkan juga lebih tinggi. Oleh karena itu, kami berusaha mengadopsi metodologi yang lebih baru dan lebih baik, yaitu metodologi agile.



Metodologi Waterfall

Metodologi Agile

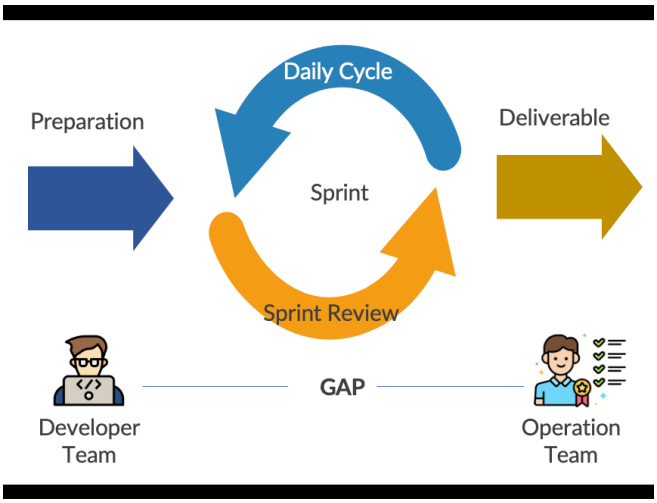
Metodologi agile berjalan dengan pendekatan back-and-forth melalui pembuatan prototipe. Pendekatan ini memungkinkan developers menunjukkan produk secara langsung ke pengguna, sehingga feedback bisa diterima lebih cepat. Jika feedback yang didapat tidak memenuhi kebutuhan pengguna, developers bisa memperbaiki prototipe dengan cepat.



Metodologi Agile

Namun, karena fleksibilitasnya, produk atau layanan sering kali hanya diuji oleh developers. Tim operasional tidak serta merta mengetahui permasalahan yang ada di produk tersebut.

Hal ini menunjukkan adanya kesenjangan komunikasi antara tim developer dan operasional



Masalah Metodologi Agile

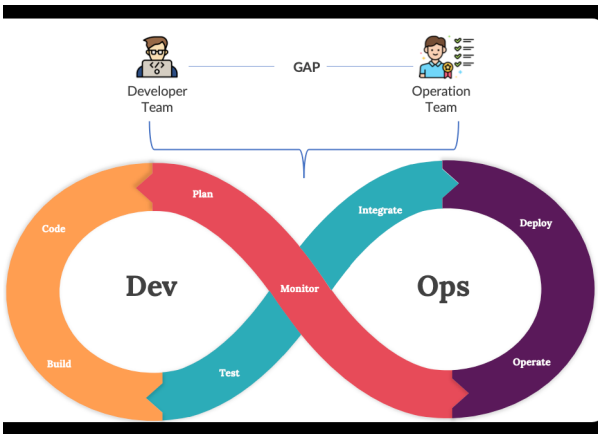
Metodologi DevOps

Metode DevOps diciptakan untuk menjembatani celah antara tim programmer dan operasional, meningkatkan efektivitas metode Agile. Pendekatan DevOps ke arah kolaborasi dan proses yang kontinyu menghancurkan pemisah antara kedua tim dan mengedepankan kerjasama tim. Tentunya setiap metode selalu ada kelemahan, dan pada artikel ini kami memfokuskan kelemahan pada aspek keamanan.

Pada metode DevOps, pengujian keamanan pada produk yang dibuat dilakukan sebelum proses deployment. Apabila ditemukan kerentanan, maka produk tersebut dikembalikan kepada tim yang bertanggung jawab.

Tentunya situasi ini memberikan dilemma pada organisasi. Pada satu sisi akan sangat beresiko apabila merilis produk yang tidak lolos uji keamanan

tapi di sisi lain ada tekanan untuk merilis tepat waktu. Untuk menghindari dilemma seperti ini, diperlukan metode baru yang lebih baik.

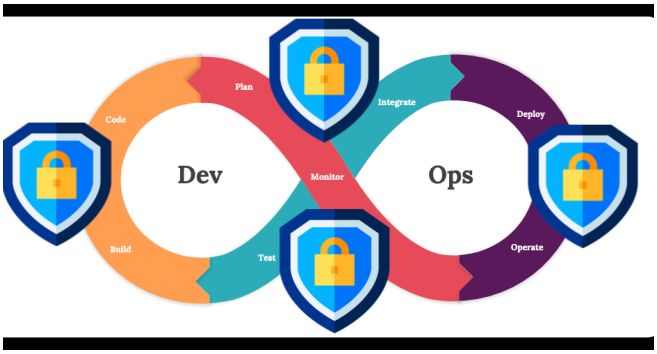


Metodologi DevOps

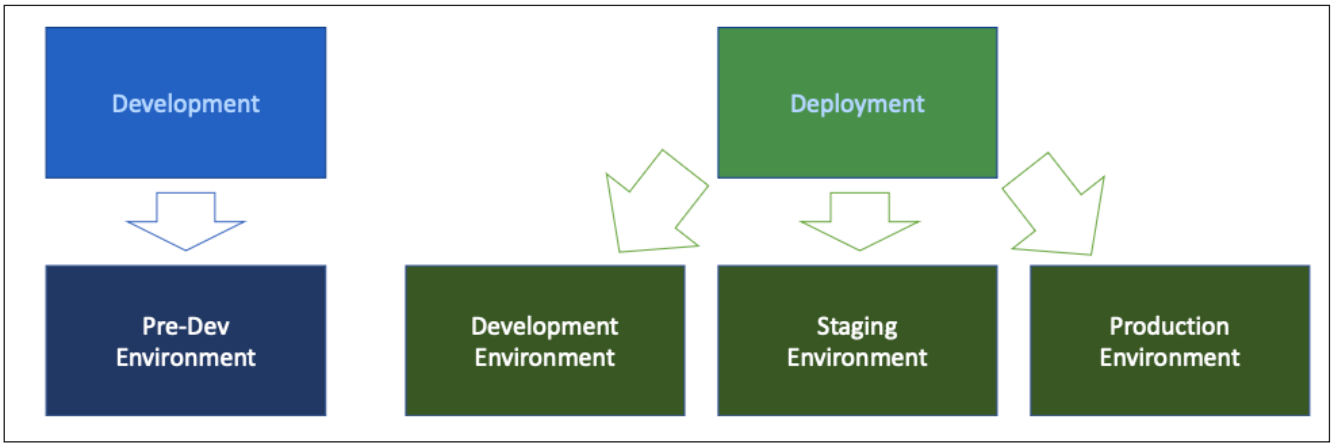
Metodologi DevSecOps

DevSecOps memperkenalkan security sejak awal proses pengembangan produk. Metode ini menekankan bahwa security merupakan tanggung jawab semua pihak yang terlibat.

Dengan pendekatan seperti ini, kolaborasi antara programmers, IT security, dan tim operasional, menjadi lebih baik, memungkinkan kerentanan dapat segera terdeteksi di setiap proses, sehingga produk atau layanan yang dikembangkan menjadi lebih aman.



Metodologi DevSecOps



Segregasi Area

Segregasi Area

Dalam pembuatan IntelliBroń, ada dua proses utama yang dilibatkan, yaitu pengembangan dan perilisan. Proses Pengembangan dilakukan oleh tim yang terdiri dari analis, desainer, programmer, dan tester. Semua bekerja di lingkungan pre-dev, menyusun perencanaan, desain, code, dan test. Area ini lebih fokus memastikan pemrograman sudah dilakukan dengan benar dan aman. Sementara itu, proses perilisan dilakukan oleh tim yang terdiri dari programmer lead, tim DevOps, tim infrastruktur, release manager, tester, pengguna, change manager, dan pihak manajemen.

Di sini, ada tiga area yang terlibat: Development, Staging dan Production. Area Development memastikan hasil pre-dev berjalan dengan baik di server.

Pada area Staging, pengetesan dilakukan pada server yang isi dan konfigurasinya sangat mirip dengan server produksi, memastikan semua fungsi berjalan dengan baik. Sementara itu, area Production adalah area yang dipakai oleh pengguna sehari-hari.

Segregasi lingkungan ini memang sesuai standar keamanan IT, seperti ISO/IEC 27001:2022, untuk memastikan produk dikembangkan sesuai proses yang benar dan memenuhi kebutuhan dari segi fungsionalitas, reliabilitas, availabilitas, resiliensi dan keamanan.

Pada edisi majalah ITSec Buzz selanjutnya, kami akan melanjutkan penjelasan proses penerapan DevSecOps dimulai pada proses Development.

PERJUANGAN MEMBONGKAR MALWARE BARU PADA IPHONE

Pada awal tahun 2023, sistem SIEM milik Kaspersky mendeteksi adanya aktifitas jaringan yang mencurigakan pada kantor pusat mereka di Moscow. Setelah diselidiki, ternyata telah terjadi serangan siber yang menargetkan iPhone dan iPad milik beberapa karyawan mereka. Dengan segera mereka mengidentifikasi gadget-gadget yang terinfeksi dan menginisiasi proses forensik digital untuk mencari dan mempelajari malware yang dipakai pada serangan tersebut.

Kaspersky menulis perjalanan mereka dalam menginvestigasi serangan siber ini yang ternyata tidaklah mudah dan sebentar. Perjalanan membongkar misteri malware baru ini mereka namakan Operation Triangulation. Pada tulisan ini, kami merangkum langkah-langkah yang telah mereka lakukan untuk "menangkap" aplikasi yang telah berhasil menyusup ke perangkat-perangkat milik karyawan mereka.

Penemuan Awal

Setelah tim Kaspersky melakukan investigasi, sejumlah fakta menarik terungkap:

1. Tidak lama sebelum melakukan aktivitas jaringan yang mencurigakan, perangkat tersebut terhubung ke server iMessage untuk menerima pesan dan file lampiran.
2. Setelah download lampiran berukuran beberapa kilobytes, perangkat terhubung ke server backuprabbit.com selama kurang dari satu menit.
3. Selanjutnya, perangkat terkoneksi dengan salah satu dari server berikut untuk sesi yang lebih lama:
   - cloudsponcer.co
   - snoeweanalytics.co
   - topographyupdates.com
   - unlimitedteacup.com
   - virtuallaughing.com
4. Saat perangkat di-reboot, semua aktivitas mencurigakan berhenti.

Investigasi Isi Perangkat

Langkah berikutnya adalah menyelidiki isi perangkat. Agar tidak menarik perhatian, tim Kaspersky menggunakan backup iTunes untuk menduplikasi perangkat tersebut. Menariknya, tidak ada indikasi eksploitasi atau instalasi malware. Namun, penemuan mengejutkan terjadi saat menganalisis rekaman aktivitas jaringan.

```
2022-09-13 10:04:11.890351Z Datausage
IMTransferAgent/com.apple.datausage.messages (Bundle ID:
com.apple.datausage.messages, ID: 127) WIFI IN: 0.0, WIFI OUT:
0.0 - WWAN IN: 76281896.0, WWAN OUT: 100956502.0
2022-09-13 10:04:54.000000Z Manifest
Library/SMS/Attachments/65/05 - MediaDomain
2022-09-13 10:05:14.744570Z Datausage BackupAgent (Bundle ID: ,
ID: 710) WIFI IN: 0.0, WIFI OUT: 0.0 - WWAN IN: 734459.0, WWAN
OUT: 287912.0
```

Rekam Aktifitas Jaringan (foto: Kaspersky)

Dalam rekaman tersebut, terungkap data yang menunjukkan aktivitas jaringan yang melibatkan proses yang disebut "BackupAgent". Meskipun proses ini sebenarnya normal dalam iOS, tapi proses ini seharusnya tidak menggunakan jaringan untuk menjalankan tugasnya.

Sebelum proses ini dimulai, ada proses IMTransferAgent yang berfungsi untuk download lampiran dari iMessage. Namun, keanehan muncul karena tidak ada bukti adanya file baru di dalam folder lampiran, meskipun catatan waktu menunjukkan perubahan dalam isi folder tersebut.

Berbagai upaya dilakukan untuk menangkap file lampiran, mulai dari memblokir pesan iMessage hingga mencoba mencegat komunikasi HTTPS menggunakan server mitmproxy, namun semuanya tidak berhasil. Hingga suatu hari, server tersebut berhasil menangkap paket komunikasi dengan server C2 milik hacker.

Server C2 adalah server pusat milik penjahat siber yang berfungsi untuk menjalin komunikasi dengan perangkat yang telah terinfeksi.

Setelah dilakukan analisis lebih lanjut, ternyata komunikasi ini berasal dari proses JS Validator, berfungsi mengumpulkan informasi tentang browser yang digunakan oleh korban. Sayangnya, isi komunikasi tersebut tidak dapat didekripsi karena JS Validator menggunakan algoritma enkripsi berbasis kriptografi kunci-publik. Proses ini melibatkan:

1. Pembuatan sepasang kunci (privat dan publik).
2. Berbagi kunci yang terbentuk dari kombinasi kunci privat JS Validator dan kunci publik server C2. Kunci ini digunakan untuk enkripsi dan dekripsi.

```
var o = function (t) {
          function u(t) {
            this.D = nacl.box.keyPair();
            this.L = nacl.box.before(t, this.D.secretKey);
          }
          u.prototype.encrypt = function (t, nonce) {
            return nacl.box.after(t, nonce, this.L);
          };
          u.prototype.decrypt = function (ciphertext, nonce) {
            return nacl.box.open.after(ciphertext, nonce, this.L);
          };
          u.prototype.N = function () {
            return nacl.randomBytes(nacl.box.nonceLength);
          };
          u.prototype.k = function () {
            return this.D.publicKey;
          };
          return new u(t);
        }(h.values.R)
```

Proses JS Validator (foto: Kaspersky)

Diperlihatkan dalam gambar di atas, JS Validator membuat sepasang kunci dengan memanggil metode **nacl.box.keyPair().** Untuk mendekripsi alur komunikasi, tim Kaspersky memutuskan untuk menginterupsi proses pembuatan pasangan kunci ini. Mereka membuat add-on pada server mitmproxy yang mencari panggilan metode **nacl.box.keyPair()**. Lalu, menggantinya dengan metode lain, **nacl.box.keyPair.fromSecretKey()**, yang menginisialisasi pasangan kunci dari kunci privat yang telah ditentukan, bukan diciptakan secara acak.

```
logging.info("plain")
server_public = self.extract_webkit_key(flow.response.content)
flow.response.content =
flow.response.content.replace(b"nacl.box.keyPair()",
b"nacl.box.keyPair.fromSecretKey(new
Uint8Array([
```

Interupsi proses pembuatan pasangan kunci (foto: Kaspersky)

Dengan langkah ini, skema enkripsi JS Validator menjadi terganggu, memungkinkan untuk mendekripsi semua komunikasi yang terjadi.

Investigasi Isi Pesan dan Eksplorasi iMessage

Setelah berhasil membuka dan menyelidiki isi pesan antara iPhone korban dan server C2 hacker, yang ternyata berupa payload, ditemukan bahwa hacker tersebut mengeksploitasi kerentanan pada WebKit dan iOS kernel untuk menjalankan validator binary di iPhone korban. Salah satu tugas validator binary adalah menjalankan perintah SQL pada database SMS.db untuk menghapus jejak pesan iMessage.

```
SELECT guid FROM attachment WHERE uti == "com.apple.watchface" AND
LENGTH(transfer_name) > 32 AND INSTR(transfer_name, CHAR(0x2013)) == 9;
```

Perintah SQL pada SMS.db (foto: Kaspersky)

Dengan membaca kode 'uti == "com.apple.watchface"', dapat diketahui bahwa file lampiran yang dikirim melalui pesan iMessage adalah file watchface yang biasa digunakan pada smartwatch milik Apple.

Untuk mendapatkan file lampiran tersebut, tim Kaspersky mempelajari cara iMessage mengirim file lampiran. Berikut caranya:

1. Pengirim membuat kunci AES secara acak dan mengenkripsi file lampiran.
2. File lampiran yang telah dienkripsi dikirim ke server iCloud.
3. Tautan iCloud yang disambung ke file lampiran serta kunci AES dikirim ke penerima pesan, dan keduanya dienkripsi menggunakan kunci-publik RSA milik perangkat.

Langkah selanjutnya untuk mendapatkan file lampiran adalah dengan mendapatkan isi tautan dan kunci AES untuk mendekripsi. Meskipun mendapatkan tautan iCloud terbilang mudah dengan server mitmproxy yang dapat menghentikan alur komunikasi dengan server iCloud, namun untuk mendapatkan kunci AES, tantangannya jauh lebih besar. Hal ini dikarenakan kunci AES dikirim melalui protokol iMessage yang tidak dapat dicegah oleh server mitmproxy.

Oleh karena itu, tim Kaspersky harus menemukan cara untuk mencegah berhasilnya proses download, eksekusi, serta penghapusan file lampiran dan kunci AES dari database SMS.db. Akhirnya, mereka memutuskan untuk memodifikasi alur komunikasi antara server C2 dan iPhone korban, sehingga tautan iCloud tidak dapat di-download. Setelah itu, mereka mengekstrak kunci AES melalui backup iTunes. Dengan langkah-langkah ini, mereka berhasil membuka kunci dan mendekripsi file lampiran.

Penemuan Malware

Setelah berhasil memperoleh file lampiran, langkah berikutnya adalah mendapatkan komponen lanjutan yang akan di-download saat file lampiran tersebut dijalankan. Validator binary bertanggung jawab melakukan hal ini dengan menggunakan kombinasi kunci RSA dan AES untuk berkomunikasi dengan server C2. Penggunaan RSA sekali lagi membuat proses dekripsi menjadi rumit.

Di tahap ini, validator binary melakukan beberapa langkah:

1. Membuat kunci AES secara acak.
2. Mengenkripsi kunci AES dengan menggunakan kunci-publik RSA server C2, yang terdapat dalam konfigurasi validator.
3. Mengenkripsi pesan yang dikirim ke server C2 dengan menggunakan kunci AES.
4. Menerima respons dari server C2 setelah pesan terkirim.
5. Mendekripsi respons menggunakan kunci AES yang sama.

Potongan kode di bawah ini memberikan gambaran bagaimana validator binary mengenkripsi seluruh data selama berkomunikasi dengan server C2.

```
E0 03 18 AA                     MOV             X0, X24
2E 1A 00 94                     BL              serialize_plist
A0 04 00 B4                     CBZ             X0, loc_100006CD4
E1 03 17 AA                     MOV             X1, X23
39 1A 00 94                     BL              encryptData
```

Proses enkripsi data (foto: Kaspersky)

Kode pertama menjalankan fungsi `serialize plist" untuk menyiapkan data yang akan dikirim. Setelah itu, register X0 menunjuk lokasi data yang akan dikirim.

Fungsi berikutnya yang dipanggil adalah encryptData. Fungsi ini memiliki dua argumen: pointer ke data yang dienkripsi dilewatkan ke register X0, sementara register X1 berisi plist dengan data konfigurasi, termasuk parameter enkripsi. Setelah fungsi ini dieksekusi, register X0 berisi pointer ke ciphertext.

Seperti langkah sebelumnya, diperlukan cara untuk mengubah proses enkripsi ini agar data dari iPhone korban dapat dicegat. Oleh karena itu, tim Kaspersky memutuskan untuk mengganti panggilan ke fungsi encryptData dengan instruksi NOP.

Dengan cara ini, isi register X0 tidak ditimpa oleh pointer ke data yang terenkripsi, membuat validator mengirim data dalam bentuk teks murni ke server mitmproxy. Setelah data tiba di server mitmproxy, simulasi pengenkripsian data dilakukan menggunakan kunci enkripsi yang telah disiapkan. Dengan demikian, mereka memiliki kunci untuk mendekripsi respons yang dikirim oleh server C2 dan mengekstrak isi dari komponen tambahan untuk dipelajari.

Penutup

Sungguh panjang perjalanan tim Kaspersky untuk mendapatkan malware dan file pendukungnya. Mereka berhasil mengatasi tantangan dengan menerapkan serangkaian langkah forensik digital yang cermat dan inovatif, mulai dari penemuan awal hingga pemecahan kode enkripsi yang rumit. Kasus ini juga menyoroti kompleksitas dan daya gedor serangan siber modern yang melibatkan eksploitasi kerentanan pada perangkat lunak, manipulasi komunikasi jaringan terenkripsi, hingga pemodifikasian alur komunikasi untuk mendapatkan akses tak terbatas. Perjalanan mereka belum selesai. Masih banyak temuan-temuan lainnya yang mereka bagikan secara rutin dalam bentuk tulisan. Tentunya akan kami rangkum di artikel berikutnya.

Dengan mengekspos teknik dan taktik yang digunakan oleh penyerang, artikel ini tidak hanya memberikan wawasan mendalam kepada komunitas keamanan digital tetapi juga menjadi peringatan bagi semua pihak untuk meningkatkan upaya dalam melindungi keamanan perangkat dan data pribadi dari ancaman siber yang semakin canggih.

ANALISA BINARY MENGGUNAKAN RetDec DAN SYNOPSYS Code DX

Penulis: M.Rasyid. Sahputra

RetDec adalah sebuah aplikasi yang dapat mengonversi kode binary (bahasa mesin) menjadi kode bahasa pemrograman (decompile), seperti C atau Python, yang dapat dimengerti manusia. Kode binary ini dihasilkan dari kode pemrograman yang ditulis oleh programmer agar bisa dieksekusi oleh system operasi. RetDec dapat mengonversi berbagai macam file binary dengan arsitektur yang berbeda-beda (32bit dan 64bit) tanpa memerlukan source-code aslinya. RetDec sangat berguna untuk menganalisa binary pada proses reverse engineering, analisa malware, deteksi kerentanan, dan lainnya.

Artikel ini akan menunjukkan proses analisa keamanan sebuah aplikasi dengan cara mengonversi (decompile) aplikasi tersebut menggunakan RetDec, lalu menganalisa dan melakukan tes keamanan menggunakan Synopsis Code Dx., sebuah aplikasi yang cukup sering digunakan untuk melakukan tes keamanan.

Cara Kerja RetDec

Untuk tes ini kami menggunakan kode pemrograman sederhana. Kode pada gambar di samping akan dikonversi dengan GCC pada mesin MacOS. Jika kode binary dieksekusi, kalimat "Hello World" akan muncul di layar.

```c
#include <stdio.h>

int main(void) {
  printf("Hello World\n");
}
```

```
┌─[MacBook-16] as cyberheb in ~/Codespaces/Git/Github.com/hello-world/c on (main)xxx
└─(/˚Д˚)/ gcc C.c -o hello-world

┌─[MacBook-16] as cyberheb in ~/Codespaces/Git/Github.com/hello-world/c on (main)xxx
└─(/˚Д˚)/ ./hello-world
Hello World
```

Tampilan compile dengan GCC dan eksekusi

RetDec memiliki beberapa subaplikasi (tools) dengan fungsi yang berbeda-beda. Di sini, kami menggunakan RetDec-fileinfo, sebuah aplikasi command-line yang digunakan untuk menganalisa file binary dan menampilkan informasi format, arsitektur dan karakteristik file. Hasil RetDec-fileinfo di atas menunjukkan format file Mach-O, 64-bit dan arsitektur ARM (Prosesor M1 milik Apple). Ini adalah informasi dari file binary yang telah dikonversi.

```
retdec@15abc056b0e9:/apps$ retdec-fileinfo hello-world
Input file               : hello-world
CRC32                    : 73ed0267
MD5                      : fea806139ec317550d4ffe0322e2f94a
SHA256                   : 7168753fc6b16c4a8ef4f3f56336a90536004f36eb6cded32ffad7a90111e4a9
File format              : Mach-O
File class               : 64-bit
File type                : Executable file
Architecture             : ARM (little endian, 64-bit mode)
Endianness               : Little endian
Entry point address      : 0x100003f7c
Entry point offset       : 0x3f7c
Entry point section name : __text
Entry point section index: 0
Bytes on entry point     : fd7bbfa9fd0300910000009000a03e91040000940000852fd7bc1a8c0035fd6100000b0100240f900021fd648656c6c6f20
Warning: Unknown compiler or packer.
```

Output RetDec-fileinfo

Setelah proses tersebut, RetDec-decompiler dijalankan untuk mengembalikan file binary menjadi kode bahasa pemrograman, menghasilkan output yang terdiri dari beberapa file.

```
retdec@15abc056b0e9:/apps$ retdec-decompiler hello-world
Running phase: Unpacking ( 0.06s )
Running phase: Initialization ( 0.06s )
Running phase: Providers initialization ( 0.07s )
Running phase: Input binary to LLVM IR decoding ( 0.10s )
Running phase: LLVM ( 0.10s )
Running phase: x86 address spaces optimization ( 0.10s )
Running phase: x87 fpu register analysis ( 0.10s )
Running phase: Main function identification optimization ( 0.10s )
Running phase: Libgcc idioms optimization ( 0.10s )
Running phase: LLVM instruction optimization ( 0.10s )
Running phase: Conditional branch optimization ( 0.10s )
Running phase: Syscalls optimization ( 0.10s )
Running phase: Stack optimization ( 0.10s )
Running phase: Constants optimization ( 0.10s )
Running phase: Function parameters and returns optimization ( 0.10s )
Running phase: LLVM instruction optimization using RDA ( 0.10s )
Running phase: LLVM instruction optimization ( 0.10s )
Running phase: Simple types recovery optimization ( 0.10s )
Running phase: Disassembly generation ( 0.10s )
Running phase: Assembly mapping instruction removal ( 0.11s )
Running phase: C++ class hierarchy optimization ( 0.11s )
Running phase: Selected functions optimization ( 0.11s )
Running phase: Unreachable functions optimization ( 0.11s )
Running phase: LLVM instruction optimization ( 0.11s )
Running phase: Make all registers local ( 0.11s )
```

Proses decompile

```
-rwxr-xr-x 1 retdec retdec  33432 Feb 20 09:11  hello-world
-rw-r--r-- 1 retdec retdec   1707 Feb 20 09:22  hello-world.dsm
-rw-r--r-- 1 retdec retdec    498 Feb 20 09:22  hello-world.ll
-rw-r--r-- 1 retdec retdec   1376 Feb 20 09:22  hello-world.bc
-rw-r--r-- 1 retdec retdec 258521 Feb 20 09:22  hello-world.config.json
-rw-r--r-- 1 retdec retdec    517 Feb 20 09:22  hello-world.c
```

Hasil decompile

File-file yang dihasilkan proses decompile masing-masing mewakili beberapa bahasa pemrograman. Contohnya, file .dsm (disassembly) berisi kode pemrograman dengan bahasa assembly, sementara file dengan akhir .c berisi kode pemrograman dengan bahasa C.

```
retdec@15abc056b0e9:/apps$ cat hello-world.c
//
// This file was generated by the Retargetable Decompiler
// Website: https://retdec.com
//

#include <stdint.h>

// ------- Dynamically Linked Functions Without Header --------

int32_t _printf(char * a1, ...);

// ------------------------ Functions -------------------------

// Address range: 0x100003f7c - 0x100003f9c
int main(int argc, char ** argv) {
    // 0x100003f7c
    _printf("Hello World\n");
    return 0;
}

// --------------------- Meta-Information ---------------------

// Detected functions: 1
```

Output Bahasa Pemrograman C

Walaupun kode yang dihasilkan sangat berbeda dengan yang dibuat di awal, tapi pada dasarnya, dari pandangan mesin/komputer, logika dan output yang dihasilkan adalah sama.

Binary ke SAST

Di beberapa kasus, terutama pada industri keamanan teknologi informasi, perlu dilakukan analisa dari file binary untuk mengetahui kerentanannya, baik itu file binary yang “baik” ataupun yang “bermasalah” seperti malware.

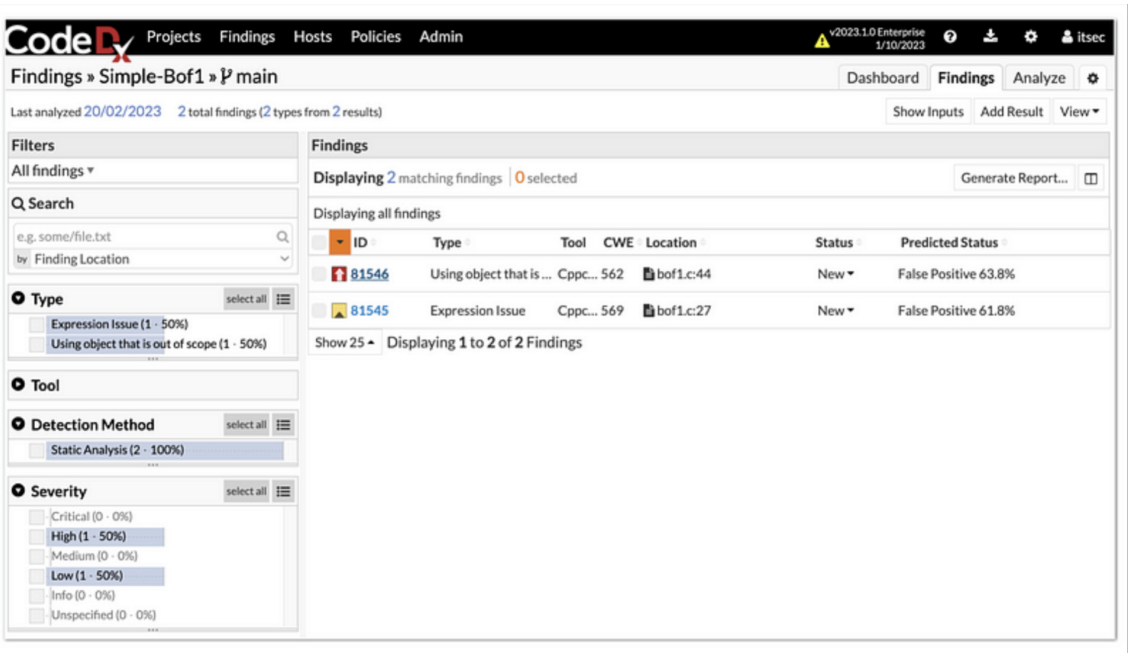
```
retdec@89dd5e6f8410:/apps$ cat bof1.c
//
// This file was generated by the Retargetable Decompiler
// Website: https://retdec.com
//

#include <stdint.h>

// ------- Dynamically Linked Functions Without Header --------

int64_t ___sprintf_chk(int64_t * a1, int64_t a2, int64_t a3, char * a4);
int64_t ___stack_chk_fail(int64_t a1);
int64_t ___strncat_chk(char * a1, int64_t a2, int64_t a3, int64_t a4);
int64_t * _memcpy(int64_t * a1, int64_t * a2, int32_t a3);
int32_t _printf(char * a1, ...);
int32_t _strlen(char * a1);

// ------------------------ Functions -------------------------

// Address range: 0x100003dfc - 0x100003f30
int main(int argc, char ** argv) {
    int64_t v1 = 0x100000000 * argc;
    int64_t v2; // bp-1173, 0x100003dfc
    _memcpy(&v2, (int64_t *)"Welcome to the argument echoing program\n", 41);
    char v3 = 0; // bp-1132, 0x100003e48
    _printf((char *)&v2);
    int64_t v4 = (int64_t)"Welcome to the argument echoing program\n"; // 0x100003e58
    if (v1 != 0) {
        // 0x100003e60
        int64_t v5; // bp-132, 0x100003dfc
        v4 = &v5;
        int64_t v6 = v1;
        ___sprintf_chk(&v5, 0, 100, "argument %d is %s\n");
        v6 -= 0x100000000;
        ___strncat_chk(&v3, v4, 999, 1000);
        int32_t v7 = _strlen(&v3); // 0x100003ed4
        while (v6 != 0) {
            // 0x100003e60
            ___sprintf_chk(&v5, 0, 100, "argument %d is %s\n");
            v6 -= 0x100000000;
            ___strncat_chk(&v3, v4, 999 - (int64_t)v7, 1000);
            v7 = _strlen(&v3);
        }
    }
    int32_t v8 = _printf("%s", (char *)v4); // 0x100003ef8
    int64_t v9 = *(int64_t *)*(int64_t *)0x100004010; // 0x100003f08
    if (v9 != *(int64_t *)*(int64_t *)0x100004010) {
        // 0x100003f18
        ___stack_chk_fail((int64_t)v8);
    }
    // 0x100003f1c
    return 0;
}

// --------------------- Meta-Information ---------------------

// Detected functions: 1
```

Untuk melakukan analisa ini, ada beberapa aplikasi dan metode yang bisa digunakan. Di sini, analisa dilakukan dengan menggunakan Synopsis Code Dx.

Synopsis Code-Dx adalah sistem manajemen resiko yang mengotomatisasi dan mempercepat pendeteksian serta perbaikan kerentanan pada source-code aplikasi.

Sebagai contoh, kami telah menyiapkan sebuah program buffer-overflow sederhana lalu melakukan proses seperti yang dilakukan sebelumnya, yaitu mengcompile menjadi file binary lalu decompile kembali ke kode pemrograman menggunakan RetDec.

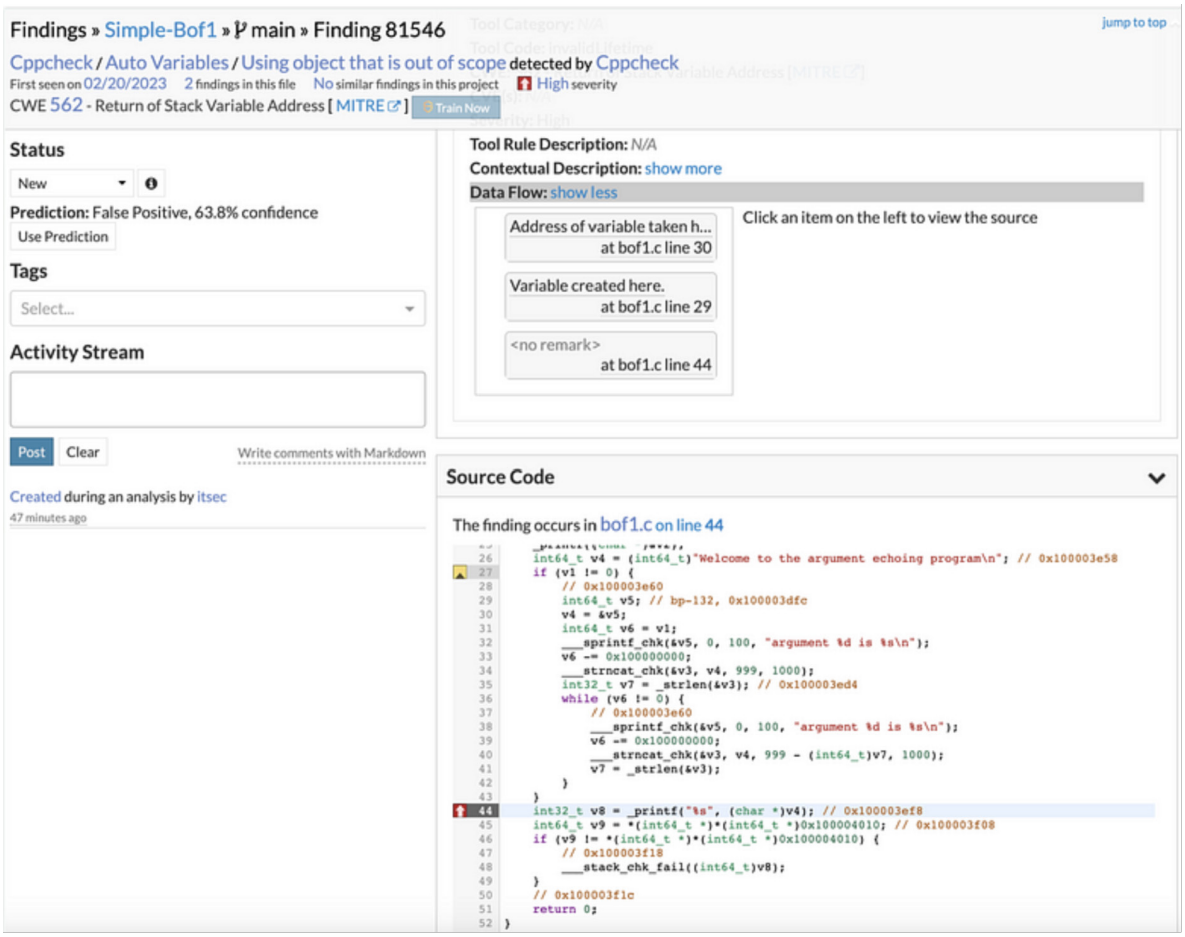
Setelah proses decompile dari Ret-Dec selesai dilakukan, hasil outputnya kami gunakan untuk pengetesan kerentanan menggunakan aplikasi Synopsis Code-Dx.



Upload hasil decompile ke Synopsis Code Dx

Hasil pengetesan menunjukkan bahwa Code-Dx dapat mendeteksi bahasa pemrograman yang digunakan, yaitu C. Dan ketika dilakukan analisa statis pada kode pemrograman, hasil menunjukkan adanya satu kerentanan yang terdeteksi, dimana kode yang rentan itu berada dan juga cara mencapai kerentanan tersebut melalui Data Flow.

Dari hasil tes ini menunjukkan gunanya RetDec dalam menganalisa file binary. Apabila terintegrasi otomatis dengan DevSecOps akan menjadi fitur yang sangat menarik.



Hasil analisa Synopsis Code Dx

Dari hasil analisa diatas, Code-Dx tidak hanya menunjukkan lokasi dari kode yang rentan tapi juga menunjukkan alur data dari permulaan hingga mencapai kode tersebut.

Demonstrasi sederhana yang kami rangkum di artikel ini menunjukkan integrasi aplikasi RetDec dan aplikasi SAST seperti Synopsis Code-DX sangat bisa diandalkan dalam proses menganalisa keamanan sebuah program. Output dekompilasi dari RetDec dapat dipakai dalam proses pendeteksian serta menganalisa kerentanan sebuah aplikasi. Bisa dibayangkan akan sangat mempermudah kalau proses ini diotomatisasi ketika terintegrasi ke DevSecOps dalam pengembangan aplikasi.

QUIZ

Yuk kita tes pengentahuan IT Security kamu.

---

1 Dari pilihan dibawah, manakah yang menjadi tanda-tanda email phishing?

A. Banyak typo dan kesalahan tata kata.
B. Ditujukan ke penerima umum, tidak menyapa nama.
C. Meminta informasi pribadi dan kredensial.
D. Semuanya benar.

2 Apa kepanjangan dari IDS?

A. Internal Defense System.
B. Intrusion Detection System.
C. Identity Detection Service.
D. Internet Denial Service.

3 Metode kriptografi mana yang melibatkan penggunaan dua kunci – satu publik dan satu pribadi?
A. Enkripsi simetris
B. Enkripsi asimetris
C. Hash Function
D. Stenography

4 Jenis Malware: Jenis malware apa yang dirancang untuk mereplikasi dirinya sendiri dan menyebar ke komputer lain?
A. Virus
B. Trojan
C. Spyware
D. Adware

5 Apa tujuan utama penerapan sistem Security Information and Event Management (SIEM)?

A. Untuk mengelola update dan patch perangkat lunak.
B. Untuk memberikan perlindungan firewall dan antivirus.
C. Untuk mengumpulkan, menganalisis, dan melaporkan data keamanan dari berbagai sumber.
D. Untuk mendeteksi serangan siber.

ITSEC™

SECURITY DELIVERED

PT. ITSEC Asia
Noble House, Level 11
jakarta, Indonesia 12950

+62 (21) 29783050

contact@itsecasia.com

itsec.asia