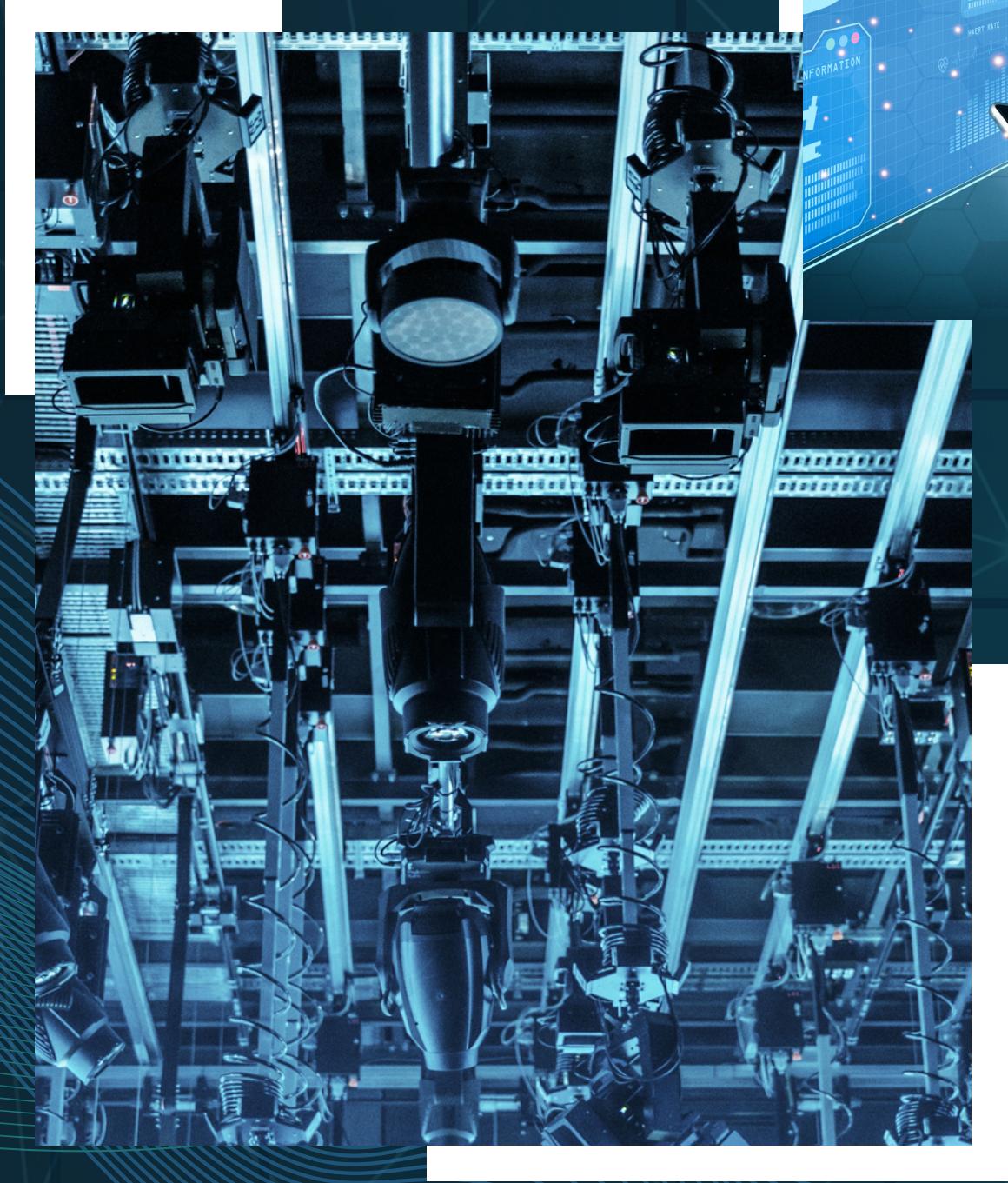




CODE REVIEW WITH A TOOL

PRESENTED BY: DOCUMENT MANAGEMENT TEAM



MANUAL CODE REVIEW (TANPA TOOLS)

Manual code review dalam konteks keamanan perangkat lunak adalah proses pemeriksaan kode sumber secara manual oleh seorang atau sekelompok pengembang, tanpa keterlibatan alat atau perangkat lunak otomatis. Tujuannya adalah untuk menemukan dan memperbaiki potensi kerentanan keamanan, kelemahan logika, atau masalah lain dalam kode yang dapat dieksplorasi oleh penyerang.

METODE UNTUK MELAKUKAN MANUAL CODE REVIEW



Threat Modeling: Membuat model ancaman untuk memahami potensi ancaman yang mungkin dihadapi oleh aplikasi.

Code Walkthroughs with Security Focus: Memeriksa kode dengan fokus pada aspek keamanan seperti sanitasi input, validasi, dan penghindaran serangan seperti SQL injection dan XSS.

Security Checklist Reviews: Menggunakan checklist keamanan untuk memastikan aspek-aspek keamanan telah diperiksa dengan teliti.

Peer Code Review with Security Emphasis: Melakukan review kode oleh rekan sejawat dengan fokus pada keamanan.

Security Testing in Code Review: Melakukan uji penetrasi sederhana untuk memastikan bahwa aplikasi memiliki lapisan pertahanan yang memadai.

Documentation and Remediation: Mendokumentasikan temuan keamanan dan melakukan perbaikan sesuai rekomendasi.

Learning from Past Vulnerabilities: Belajar dari kerentanan keamanan yang ditemukan di masa lalu untuk mencegah pengulangan kesalahan.



Teknologi masa depan atau teknologi yang akan dikembangkan merupakan konsekuensi dari inovasi teknis (dalam hal ini ilmu teknik). Prediksi masa depan sering kali melebih-lebihkan laju perkembangan teknologi dalam jangka pendek.

MENDORONG TUMBUHNYA INDUSTRI KREATIF

DI MASA DEPAN TEKNOLOGI BERKEMBANG SANGAT PESAT



3. JELASKAN APA YANG DIMAKSUD DENGAN STATIC ANALYSIS TOOL?



Static analysis tools adalah perangkat lunak yang digunakan untuk menganalisis kode program tanpa menjalankannya secara aktif. Mereka memeriksa kode sumber untuk menemukan berbagai jenis masalah potensial atau kerentanan, seperti bug, kelemahan keamanan, atau pelanggaran aturan gaya kode. Analisis statis dilakukan dengan memeriksa struktur dan konten kode program tanpa memperhatikan jalannya eksekusi program

4

BEBERAPA TOOLS YANG DAPAT DIGUNAKAN UNTUK MELAKUKAN STATIC ANALYSIS TOOL



Coverity



SonarQube



Fortify



Ounce Labs



Checkmarx



PVS-Studio

CODE REVIEW WITH TOOLS

5

Code Review with Tools: Meningkatkan Keamanan Perangkat Lunak

Code review with tools adalah proses meninjau kode sumber perangkat lunak dengan bantuan alat bantu. Alat bantu ini dapat membantu mendeteksi berbagai kelemahan keamanan dan meningkatkan kualitas kode secara keseluruhan.

MANFAAT CODE REVIEW WITH TOOLS

Meningkatkan
Keamanan

1

Meningkatkan
Kualitas Kode

2

Meningkatkan
Produktivitas

3

6 JELASKAN TOOLS APA SAJA YANG DAPAT DILAKUKAN UNTUK MELAKUKAN CODE REVIEW?



Github



Bitbucket



Gitlab



Gerrit

BINARY ANALYSIS: MENGANALISIS KEAMANAN PERANGKAT LUNAK PADA TINGKAT BINER

7

Binary analysis adalah proses menganalisis program perangkat lunak pada tingkat biner, yaitu pada level instruksi mesin yang dipahami oleh komputer. Berbeda dengan code review yang meninjau kode sumber, binary analysis meneliti program setelah dikompilasi menjadi kode biner.

MANFAAT BINARY ANALYSIS

Meningkatkan
Keamanan

1

Memverifikasi
Integritas

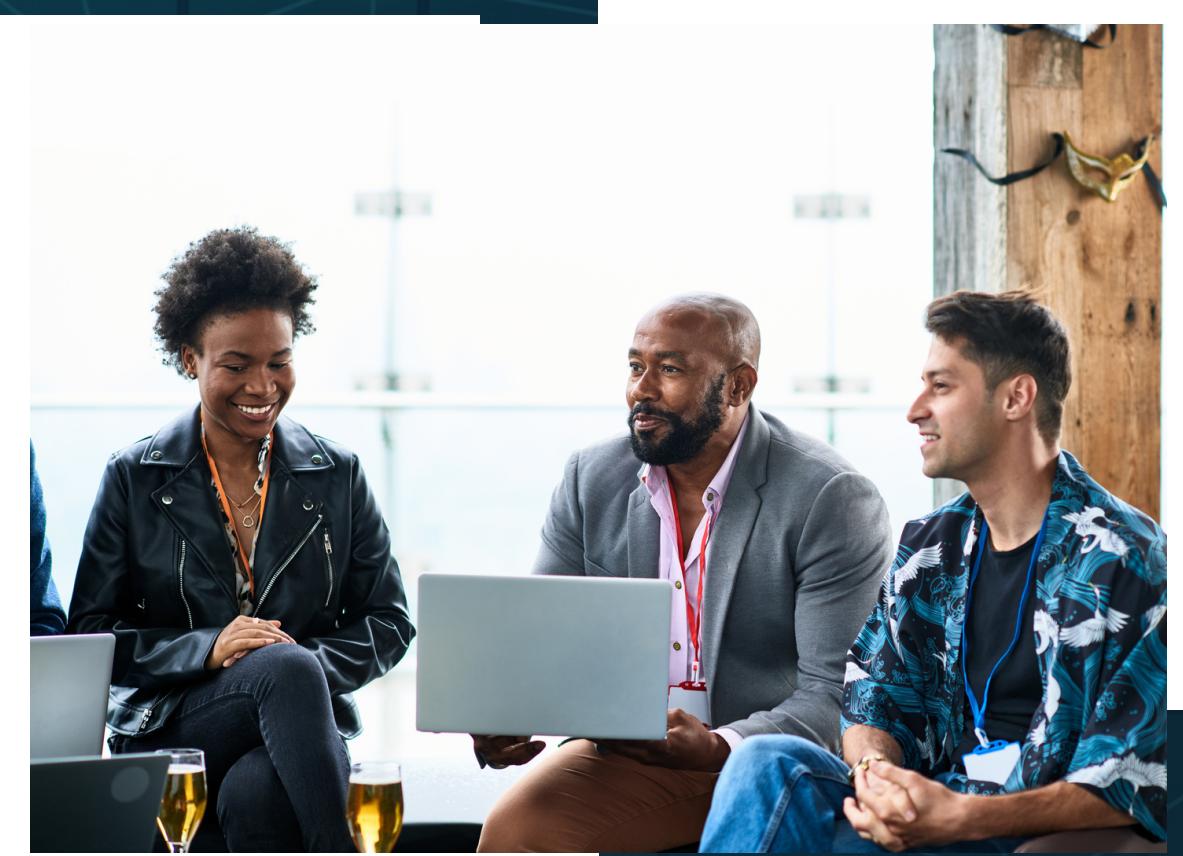
2

Melakukan
Troubleshooting

3



- Meningkatkan Kode
 - Konsistensi Kode
 - Penghematan Waktu
 - Peningkatan Keamanan
 - Pelacakan Perubahan
 - Kolaborasi Tim
- Kualitas





JELASKAN BAGAIMANA MELAKUKAN CODE REVIEW WITH TOOLS 9



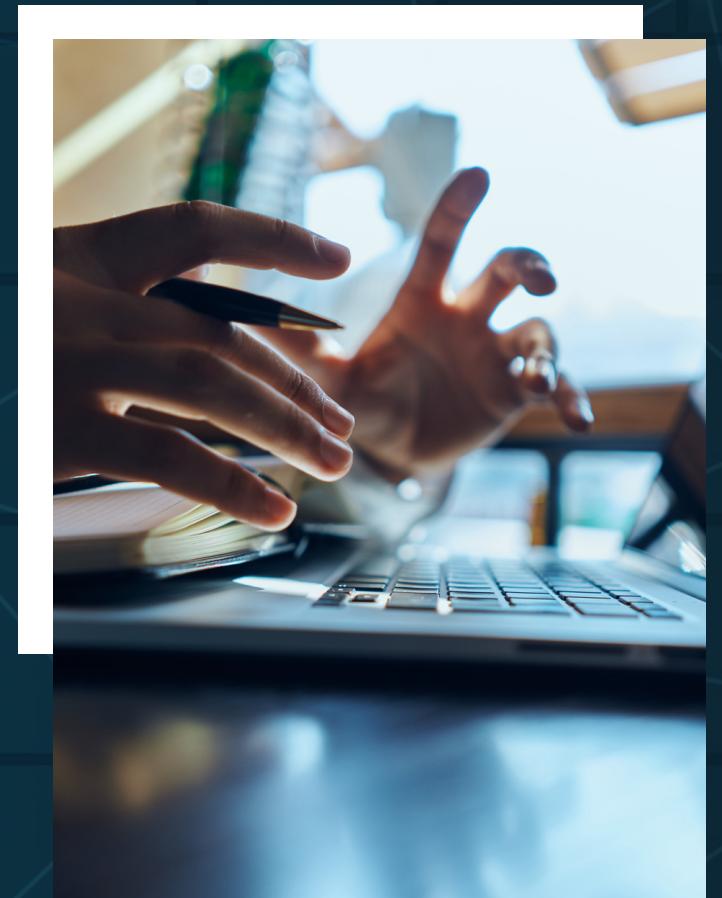
TAKTIK PEMASARAN YANG EFEKTIF

Untuk membantu klien meningkatkan infrastruktur TI mereka dan memudahkan mereka dengan Layanan terbaik



PENGELOLAAN KEUANGAN YANG BAIK

Untuk membantu klien meningkatkan infrastruktur TI mereka dan memudahkan mereka dengan Layanan terbaik



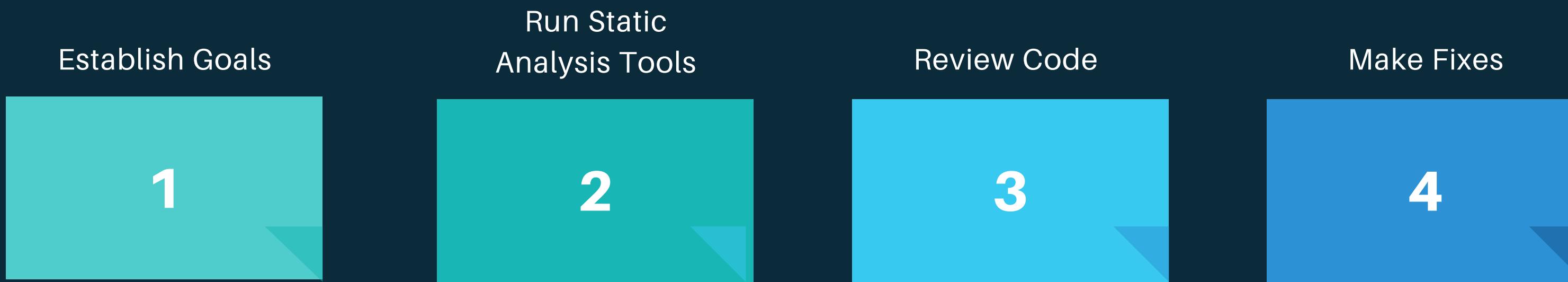
MELAKUKAN CODE REVIEW DENGAN MENGGUNAKAN ALAT BANTU ANALISIS STATIS

7-Step Code Review with tools Analys



THE CODE REVIEW CYCLE

4-Step Code Review Cycle



ESTABLISH GOALS

Langkah pertama dalam siklus code review adalah menetapkan tujuan untuk revisi kode. Ini mungkin mencakup memastikan kepatuhan dengan standar kode, meningkatkan keamanan, atau meningkatkan kualitas dan kinerja. Alat bantu untuk manajemen proyek seperti Jira, Trello, atau Asana dapat digunakan untuk menetapkan dan melacak tujuan ini. Selain itu, platform code review seperti GitHub, GitLab, atau Bitbucket juga menyediakan fitur untuk menambahkan catatan atau checklist yang terkait dengan tujuan-tujuan ini di setiap permintaan penggabungan.

RUN STATIC ANALYSIS TOOLS

Sebagian besar platform pengembangan modern menyertakan integrasi dengan alat analisis statis kode, seperti ESLint untuk JavaScript, pylint untuk Python, atau SonarQube untuk berbagai bahasa pemrograman. Alat-alat ini secara otomatis menganalisis kode untuk menemukan potensi bug, pelanggaran gaya, atau masalah keamanan. Integrasi ini dapat dikonfigurasi untuk berjalan secara otomatis setiap kali ada pembaruan kode atau sebagai bagian dari alur kerja code review.

REVIEW CODE

Proses utama code review melibatkan pemeriksaan manual oleh sesama pengembang. Platform code review seperti yang telah disebutkan sebelumnya (GitHub, GitLab, Bitbucket) menyediakan antarmuka yang memudahkan untuk melihat perubahan kode, memberikan komentar, dan memulai diskusi. Fitur-fitur seperti penyorotan sintaksis, pembanding perubahan, dan integrasi dengan sistem kontrol versi membuatnya mudah bagi reviewer untuk melihat perbedaan antara kode lama dan baru.

MAKE FIXES

Setelah code review selesai, pemilik kode (pengembang yang mengajukan permintaan penggabungan) melakukan perbaikan sesuai dengan komentar dan saran yang diberikan oleh reviewer. Proses ini sering dijalankan secara iteratif, dengan pemilik kode membuat perubahan, kemudian meminta review ulang, dan begitu seterusnya hingga perubahan dinyatakan memenuhi standar yang ditetapkan.

TERIMA KASIH

DI MASA DEPAN TEKNOLOGI BERKEMBANG SANGAT PESAT